COUNTING OF VIDEO CLIP REPETITIONS USING A MODIFIED BMH ALGORITHM: PRELIMINARY RESULTS

Silvio Jamil Ferzoli Guimarães, Renata Kelly Rodrigues Coelho and Anne Torres

Audio-Visual Image Processing Laboratory (VIPLab)
Information Systems / Institute of Computing / PUC Minas
sjamil@pucminas.br, tonely@pop.com.br, anne_torres@uol.com.br

ABSTRACT

In this work, we cope with the problem of identifying the number of repetitions of a specific video clip in a target video clip. Generally, the methods that deal with this problem can be subdivided into methods that use: (i) video signatures afterward the step of temporal video segmentation; and (ii) string matching algorithms afterward transformation of the video frame content into a feature values. Here, we propose a modification of the fastest exact string matching algorithm, the Boyer-Moore-Horspool, to count video clip repetitions. We also present some experiments to validate our approach, mainly if we are interested in found identical video clips according to spatial and temporal features.

1. INTRODUCTION

Traditionally, visual information has been stored analogically and indexed manually. Nowadays, due to the improvements on digitalization and compression technologies, database systems are used to store images and videos, together with their meta-data and associated taxonomy. Unfortunately, these systems are still very costly. Thus, the need of efficient systems to process and index the information is unquestionable, mainly if we are interested in information retrieval. The most important problems are associated with: the exponential increase of the Internet, and consequently, the increase in the number of duplicated documents; and distribution across of communication channels, like TV, of thousands of hours of streaming broadcast media. According to [1, 2], broadcast monitoring for the purpose of market analysis is an application which has arisen in the domain of television, more specifically, identify the number of repetitions of a specific video clip in a target video clip into two groups.

Here, we cope with the problem of identifying the number of repetitions of a specific video clip in a target video clip. The main difficulties to identify video repetitions correspond to: (i) definition of the dissimilarity measures of video clips; (ii) processing time of the algorithms due to the huge amount of information to analyse; (iii) insertion of intensional distortions; and (iv) different frame rates. Generally, the methods that deal with this problem can be subdivided into: (i) computation of video signatures afterward the step of temporal video segmentation, like the methods described in [2, 3, 4]; and (ii) utilization of string matching algorithms afterward transformation of the video frame content into a feature values, like the methods described in [5, 6]. When video signatures are used, we need to apply methods for temporal video segmentation [7]. Even if the temporal video segmentation is a well study problem, it represents another problem that can be considered in the repetition analysis. When string matching algorithms are used, the developed methods take into account the efficiency of the these algorithms when compared to image/video algorithms to solve the video repetition problem. In [5, 6] were used the longest common substring algorithm to deal with problem, however it requires a $O(mn)$ space and time cost, in which m and n represent the size of the query and target video clips, respectively.

In this work, we propose the utilization of a modified version of the fastest algorithm of exact string matching, the Boyer-Moore-Horspool (BMH) [8, 9], to deal with the problem of video clip repetition in which we transform the video frame content into a feature value (e.g. mean, entropy, histogram intersection with an ideal frame). Our proposed method is fast and presents a recall of 90%, when applied to videos with the same frame rate. This paper is organized as follows. In Sec. 2, we give some basic definitions and we define the video clip repetition problem. In Sec. 3, we propose a methodology to identify the number of video clip repetitions using a modified BMH algorithm. In Sec. 4, we discuss about the experiments and the setting of algorithm parameters. Finally, in Sec. 5, we give some conclusions and future works.

2. PROBLEM DEFINITION

Let $\mathbb{A} \subset \mathbb{Z}^2$, $\mathbb{A} = \{0, ..., \mathrm{H}-1\} \times \{0, ..., \mathrm{W}-1\}$, where H and W are the width and height of each frame, respectively.

Definition 2.1 (Frame) *A frame f_t is a function from $\mathbb{A}$ to $\mathbb{Z}$, where for each spatial position (x, y) in $\mathbb{A}$, $f_t(x, y)$ represents*

Thanks to CNPq agency and PUC Minas University for funding.

the grayscale value at pixel location (x, y).

Definition 2.2 (Video) *A video* V_n*, in domain* $2\mathbb{D} + t$*, can be seen as a sequence of frames* f_t*. It can be described by*

$$\mathrm{V}_n = (f_t)_{t\in[0,n-1]} \tag{1}$$

where n *is the number of frames contained in the video.*

Definition 2.3 (Histogram) *Consider a partition of* $\mathbb{Z}$ *in L intervals* $I_0 \cdots I_{L-1}$ *called bins. A histogram* H_{f_t} *of an image* f_t *is given by*

$$H_{f_t}(i) = \#\{(x, y)|f_t(x, y) \in I_i\} \tag{2}$$

where $\#$ *denotes the cardinality of a set* X.

According to [10], the problem of video clip repetition, also called video matching detection, determines if a identical copy for a given video clip appears in the target video, and if so at what locations and the number of repetitions.

Definition 2.4 (Frame similarity) *Let* f_t *and* f_p *two video frames at location* t *and* p*, respectively. Two frames are similar if a similarity measure* $\mathcal{D}(f_t, f_p)$ *between them is smaller than a specified threshold. The frame similarity is defined as*

$$FS(f_t, f_p, \delta) = \begin{cases} 1, & \text{if } \mathcal{D}(f_t, f_p) \leq \delta \\ 0, & \text{otherwise} \end{cases} \tag{3}$$

Definition 2.5 (Video clip repetition (VCR)) *Let* V_m^q *and* V_n^t *be a query video clip with m frames and a target video with n frames, respectively. The video clip repetition (VCR) problem corresponds to identification of a video clip* V_m^q *that belongs to a video clip* V_n^t *if there is a subvideo in* V_n^t *that matches with* V_m^q *according to a frame similarity. Thus, this problem can be defined by*

$$VCR(\mathrm{V}_m^q, \mathrm{V}_n^t) = \{p \mid (\sum_{i=0}^{m-1}(FS(\mathrm{V}_m^q(t), \mathrm{V}_n^t(p+i), \delta))) = m\} \tag{4}$$

where $FS(A, B, \delta)$ *is the frame similarity between the frames A and B considering a error* δ *and* $\forall\, p \in [0, n]$.

The naive algorithm (described in Alg. 1) can be used to solve the video copy problem, however this algorithm is time expensive. In this work, we cope with the VCR problem using a modification of the fastest algorithm to exact string matching, the Boyer-Moore-Horspool algorithm [8, 9], so-called BMH. The idea of BMH algorithm is maximize the length of the shift considering the text character corresponding to the rightmost pattern character, and not for the text character where the mismatch occurred. The preprocessing phase of the BMH algorithm requires $O(m+|\Sigma|)$ time, where m is the size of the pattern and $|\Sigma|$ is the size of the alphabet (defined in Sec. 3), and the $O(|\Sigma|)$ space requirements. Finally, the searching phase requires $O(mn)$ time in the worse case and in average case requires $O(n/|\Sigma|)$, where n is the size of the text.

Algorithm 1 Naive algorithm

Require: Video sequences
Target video clip (V_n^t)
Query video clip (V_m^q)
Threshold value (δ)
Ensure: Frames of the video sequence in any domain
{m = size of the query video clip}
{n = size of the target video clip}
1: **for** $(i = 0; i < n; i++)$ **do**
2: j = 0; count = 0;
3: **while** ($j < m$ && (FS($\mathrm{V}_m^q(j), \mathrm{V}_n^t(i), \delta$))) **do**
4: j++;
5: **end while**
6: **if** j=m **then**
7: "A video clip was found at position i"
8: count = count + 1
9: **end if**
10: **end for**
11: **return** count

3. METHODOLOGY FOR COUNTING OF VIDEO CLIP REPETITION

The aim of this work is to verify if a video clip is contained into another one. However, the naive algorithm to solve this problem is time expensive, and some of the proposed solutions either space expensive or depends on the video segmentation, as before described. Our method propose a modification of the BMH algorithm to cope with video frame content. The proposed method reduces the processing time with respect to traditional video clip repetition methods and also does not consider the temporal video segmentation. To follow, we present the modified BMH algorithm that will be used to deal the VCR problem.

Definition 3.1 (Alphabet) *Let* Σ *be the symbol alphabet,* f_v *be a feature value in the range* $R = [0, S_{f_v}]$*, where* S_{f_v} *represents the maximum feature value, and depends on the feature value considered.*

For example, if the feature value is the mean of the grayscale pixel values, the $S_{f_v} = 255$ and the range will be $R = [0, 255]$. The idea of the string matching algorithm is to find the first occurrence (or all occurrences) of a string B in a string A, and as described before, the main features of the BMH algorithm are the right-to-left scan over the pattern and the distance to move based on rightmost character of the pattern [8], that will be preserved in our algorithm. The main differences between the traditional BMH algorithm and the modified BMH algorithm corresponds to the lines 10 and 17 of the algorithm described in Alg. 2. In the proposed modification, small differences between two correspondent frames are permitted and controlled by a threshold, represented in our algorithm by the variable δ.

Another very important modification is associated with

Algorithm 2 Modified BMH algorithm

Require: Video sequences
Feature values (T) of the target video clip (V_n^t)
Feature values (Q) of the query video clip (V_m^q)
Threshold value (δ)
Require: Sequence of feature value.
Ensure: Feature value in the range [0,S].
{tam = size of alphabet }
{m = size of the query video clip}
{n = size of the target video clip}
{d = containing the length of the shift}

```
1:  for (every k in tam) do
2:     d[k] = m+1;
3:  end for
4:  for (every k in tam) do
5:     d[p[k]]= m-k;
6:  end for
7:  count = 0;
8:  while (i ≤ n) do
9:     k = i; j = m;
10:    while (j > 0 && (FS(V_m^q(j), V_n^t(i), δ))) do
11:       k–; j–;
12:    end while
13:    if j=0 then
14:       "A video clip was found at position k"
15:       count = count + 1;
16:    end if
17:    i += ⋀_{z=−δ}^{δ} d[t[i] + z]
18: end while
19: return count
```

the shift after a mismatch. As we permit small feature value differences, it is necessary to cope with this characteristic, thus in line 17 of our modified BMH algorithm, we consider the smaller distance to move the query pattern to the next alignment verification. Even if the worst case of the modified BMH is $O(mn)$, the average case requires $O(n/(|\Sigma| + \delta))$.

Note that the feature value is represented by only one value in a specific range, and for that, it is not possible to consider a feature vector to represent the frame content.

4. EXPERIMENTS

4.1. Video database

In Table 1, we describe the information of the video corpora. The real digitized video was directly captured of a brazilian cable TV channel and the edited video sequence was obtained by downloading of different video commercial followed by a edition of these videos using cut transition.

The experiments was subdivided into two groups (illustrated in Table 2): (i) utilization of a query video clip directly extracted of the real digitized video; and (ii) utilization of a video commercial that was used to edit a video sequence.

Video sequence	Time	Frame rate
Real digitized video	2h 52m 31s	30 fps
Edited video sequence	1h 36m 44s	30 fps
Total	4h 29m 15s	-

Table 1. Video target corpora

Video sequence	Time	rate	videos	occurrences
Same source	1m 33s	30 fps	7	16
Different source	13m 32s	30 fps	29	113
Total	15m 05s	-	36	129

Table 2. Video query corpora

4.2. Setting of parameters

To realize the experiments, it is necessary to define some parameters of the algorithm:
Feature value - this parameter represents the frame feature content, and due to this his choice influences the quality of the results. Amongst the image discriminators, we considered the mean of the luminance, the entropy of the histogram and the histogram intersection. The two first does not efficiently represent the image content. Even if the histogram intersection used by Swain and Ballard [11, 7] does not represent spatial information, it is considered one of the most important similarity measure between two images. In this work, we need to transform the video frame content into one value to index a vector position that is used by the BMH algorithm. To compute the feature value based on histogram intersection, we consider an ideal image (in which, all pixel values present the uniform distribution) and the frame to index.
Threshold value - unfortunately, all video frames have digitalization problems, and due to this it is necessary consider a relaxation variable to permit small differences between the correspondent frames of the target video and the query video. According to the empirical analysis, a good threshold value δ is equal to 20% and the maximum value is equal to 30%. For values greater than 30%, it appears false identification.

4.3. Analysis of the results

We denote by #Occurrences the number of query video occurrences, by #Video clip identified the number of query video properly identified. The recall α is defined by

$$\alpha = \frac{\#\text{Video clip identified}}{\#\text{Occurrences}} \tag{5}$$

Observe that different algorithms that use string matching produces the same results according to the recall measure, however the main difference between these algorithms correspond to the processing time. As the BMH algorithm is the best exact string matching, theoretically the modified BMH algorithm proposed here can also be considered the best one. If $\delta = 0$, then computational cost of modified BMH algorithm

Type of video sequence	occurrences	identified	α
Real	16	15	93,75%
Edited	113	102	90,27%
Total	129	117	90,70%

Table 3. Results of the video clip repetition identification

is exactly the same of the traditional BMH algorithm. In our experiments, we did not fix the threshold δ to solve the VCR problem for all query video clips. For us, it is interesting to study the real identification even if the setting of parameters, more specifically the threshold, are not the same. The threshold parameter is in range $[0, 30\%]$. In Table 3 we show the results of our experiments.

Due to the setting of threshold parameter in the range $[0, 30\%]$, no false occurrence was identified, and for that the results illustrated in Table 3 represents a precision measure of 100%. An interesting observation of our video corpora corresponds to the quality of the videos. Initially, the original video commercials present different frame rate and resolution, afterward the video edition, all videos are transformed into 30 fps (frame per second) with 320×240 frame size. We believe that this transformation is the main reason of the low precision of the edited video sequence $(90, 27\%)$.

4.4. Theoretical comparative analysis

Our work can be directly compared to the methods developed by Adjeroh et alli [5] and Kim et alli [6]. While these works consider edit distance to compute the similarity between videos in which insertion, remotion and substitution are permitted, our method works very well when the frame rate is constant and no operation edition is applied. Also, the method proposed here is theorically faster than methods described in [5, 6] thanks to the BMH algorithm

5. CONCLUSIONS

In this work, we proposed the utilization of a modified version of the fastest algorithm for exact string matching, the Boyer-Moore-Horspool (BMH), to deal with the problem of video clip repetition in which we transform the video frame content into a feature value (e.g. mean, entropy, histogram intersection with an ideal frame). According to preliminary experiments, we believe that our method can be improved by utilization of new feature value transformations and/or application of the algorithms for different feature values followed by a merge of the results. The recall of our method is approximately 90%. Some video clips are not detected due to problems of digitalization, and to cope with this problem, approximate string matching algorithms can be used.

6. ACKNOWLEDGMENTS

The authors are grateful to PUC MINAS (Pontificia Universidade Católica de Minas Gerais) and CT-Info/MCT/CNPq (Project 506604/2004-7) for the financial support of this work.

7. REFERENCES

[1] Nicholas Diakopoulos and Stephan Volmer, "Temporally tolerant video matching," in *Proc. of the ACM SIGIR Workshop on Multimedia Information Retrieval*, Toronto, Canada, August 2003.

[2] J.M. Gauch and A. Shivadas, "Identification of new commercials using repeated video sequence detection," in *International Conference on Image Processing*, 2005, pp. III: 1252–1255.

[3] A. Joly, C. Frelicot, and O. Buisson, "Content-based video copy detection in large databases: A local fingerprints statistical similarity search approach," in *International Conference on Image Processing*, 2005, pp. I: 505–508.

[4] Xavier Naturel and Patrick Gros, "A fast shot matching strategy for detecting duplicate sequences in a television stream," in *Proceedings of the 2nd ACM SIGMOD International Workshop on Computer Vision meets DataBases*, 2005.

[5] Donald A. Adjeroh, M. C. Lee, and Irwin King, "A distance measure for video sequences," *Computer Vision and Image Understanding: CVIU*, vol. 75, no. 1–2, pp. 25–45, / 1999.

[6] Young tae Kim and Tat-Seng Chua, "Retrieval of news video using video sequence matching.," in *MMM*, 2005, pp. 68–75.

[7] Alberto Del Bimbo, *Visual information retrieval*, Morgan Kaufmann Publishers Inc., San Francisco, CA, USA, 1999.

[8] R. Nigel Horspool, "Practical fast searching in strings.," *Softw., Pract. Exper.*, vol. 10, no. 6, pp. 501–506, 1980.

[9] Gonzalo Navarro, "A guided tour to approximate string matching," *ACM Comput. Surv.*, vol. 33, no. 1, pp. 31–88, 2001.

[10] Changick Kim and Bhaskaran Vasudev, "Spatiotemporal sequence matching for efficient video copy detection.," *IEEE Trans. Circuits Syst. Video Techn.*, vol. 15, no. 1, pp. 127–132, 2005.

[11] D.H. Ballard M.J. Swain, "Color indexing," *International journal of computer vision*, vol. 7, no. 1, pp. 11–32, 1991.