VIDEO CLOCK TIME RECONITION BASED ON TEMPORAL PERIODIC PATTERN CHANGE OF THE DIGIT CHARACTERS

Yiqun LI, Kongwah WAN, Xin YAN, Xinguo YU, Changsheng XU

Institute for Infocomm Research
21 Heng Mui Keng Terrace, Singapore 119613
{yqli, kongwah, yanxin, xinguo, xucs}@i2r.a-star.edu.sg

ABSTRACT

A novel Temporal Neighboring Pattern Similarity (TNPS) measure is proposed to recognize the time of a digital clock overlay. TNPS detects the presence of a clock overlay by monitoring the periodic changes of the clock digit, and infers the clock time by its natural transition cycle. Compared to traditional methods such as OCR, this method is faster and more reliable because it converts a pattern recognition problem to a pattern change detection problem. Experiments show the recognition result is promising and accurate. One of the applications for this method is to detect the start time of soccer game for our real time live soccer video highlights and event alerts.

1. INTRODUCTION

In sports video analysis, accurate event detection and scene segmentation is a very challenging research topic. It is very difficult to find unique features to identify certain events and segment video scenes because of the wide variation of the video. Researchers have worked on these based on the video structure analysis [1] and multimodal features analysis [2]. Domain objects detection in specific sports has also been explored [3]. However, accurate event boundary detection is still very challenging. For example, in our live broadcast soccer video highlights and event alerts project, we need to know the game start time in order to synchronize the game time with the live web-casting text. Because there are many variations for the game start scene in different soccer games, the audio feature of a whistle for the game start is also not unique and easy to detect, using scene detection and audio features to detect this start event is not reliable. In another prior work, Gu et al [4] used adaptive field color model and temporal pattern to do play segmentation. However, no work has been found on the accurate game start time detection.

By our observation, a video clock overlay is used in many sports games such as soccer and basketball to indicate the game lapsed time. Figure 1 shows a few samples of that in some soccer games such as UEFA, EPL and FIFA. Normally the name of the two teams, their scores are also shown on the overlay. In most of the time, the overlay starts to appear when the game starts. Therefore, recognize the clock time can help us to locate the game start time.

Figure 1. Video clocks in different soccer games.

The clock is actually four digit characters showing the number of minutes and seconds lapsed for the soccer game. Naturally, OCR is the first idea coming to our mind to recognize the digit characters of the video clock. However, the challenges are:

1. The video images are decoded from the MPEG-1 video. As shown in Figure 1, their quality is very bad. The clock digit characters are small and noisy. Their color, font and size are different in different videos.
2. There are other characters including digits around the clock. Thus it is difficult to locate the clock digit characters.
3. We have to process the video in real-time, the algorithm must be fast enough to meet the requirement of our real-time application.

From literature [5-7], we know that although OCR is a mature technology and there are many products in the market, the accuracy of OCR depends a lot on the quality of the image and its application domain. In our application, the noisy and small size digit characters will degrade the OCR performance. Furthermore, requiring training of the OCR algorithms is also a drawback. Therefore, using traditional OCR method to recognize the digit characters is not a good choice in our application. This challenges us to propose a simple but reliable method to solve the problem.

Compared to a static two dimensional image, video has a third dimension which is the time. Making use of the third dimensional information may help us to recognize the clock time. As we know, the digit characters on the video clock change periodically along time. This is a special feature compared to other video text characters. By making use of this property, we can locate the clock digit characters. The temporal change of the digit characters also follows certain regulation. This enables us to recognize the clock time by inference.

2. PERIODIC PATTERN CHANGE OF THE DIGIT CHARACTERS ON THE VIDEO CLOCK

Since the digit characters on the video clock change periodically, the image pattern of each digit character will change periodically. As shown in Figure 2, the SECOND digit character changes from 0, 1, 2, … to 9 every second. Namely, for every 25 or 29.97 frames depending on the frame rate of PAL or NTSC system video, the SECOND digit will change once. In the same way, the TEN-SECOND digit changes once every 10 seconds, while the MINUTE digit every one minute and the TEN-SECOND digit every 10 minutes. For the change relationship among different digit characters, at the moment when the SECOND digit becomes "0", the TEN-SECOND digit will increase. When the MINUTE digit changes, both the TEN-SECOND and SECOND digit will be "0". And when the TEN-MINUTE digit changes, all the MINUTE, TEN-SECOND and SECOND digit are "0". Based on above regulation, we can infer the clock time when we first detect a pattern change of one of the digit characters.

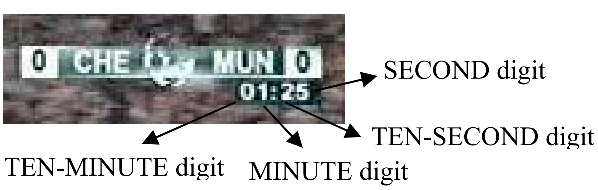


Figure 2. The digit characters on the video clock.

3. TEMPORAL NEIGHBORING PATTERN SIMILARITY (TNPS) SEQUENCE

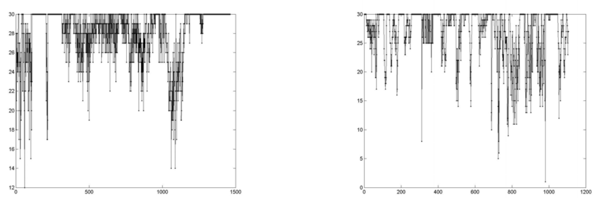
At time t, provided the frame sequence number is n, the digit character image can be represented by $f_n(x, y)$, where $(x, y) \in I$. I is the image region of the digit character. After binarization, $f_n(x, y)$ becomes $B_n(x, y)$.

$$B_n(x, y) = \begin{cases} 1; & \text{if } f_n(x, y) \geq \text{threshold} \\ 0; & \text{if } f_n(x, y) < \text{threshold} \end{cases}$$

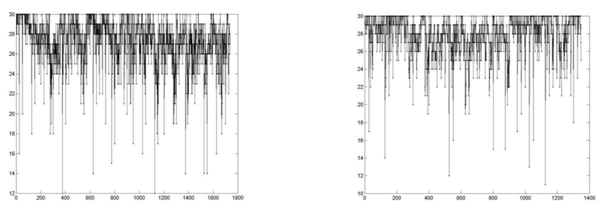
The TNPS is the similarity measure of the digit character image in two consecutive frames. The time sequence of this TNPS is,

$$S(n) = \sum_{(x, y) \in I} B_{n-1}(x, y) \otimes B_n(x, y) \qquad (1)$$

Where n is the frame sequence number. If the digit character image of frame n-1 is the same as that of frame n, ideally S(n) would get maximum value, which is the digit character image size. However, in actual situation, the images in the video are different even if they are showing the same digit character. We need to compare the difference between the S(n) when the digit doesn't change and when it changes. We have conducted many experiments on recorded soccer videos and below are two of them.



(a) TNPS in the region without video clock.



(b) TNPS in the region with SECOND digit.

Figure 3. TNPS difference between image regions with and without video clock digit character.

Figure 3 shows two of our experimental observations of the TNPS in the region with or without the clock's SECOND digit character. The horizontal axis is the decoded image frame sequence number. The vertical axis is S(n), which is calculated according to equation (1). Figure 3(a) shows the S(n) in a region without video clock. We can see S(n) is a random distributed sequence without any regular pattern. While Figure 3(b) is the S(n) in the region of the clock's SECOND digit character, S(n) has a periodic minimum value every 25 frames. Except the periodic minimum values, other values are close to the maximum value of 30, the variation of these values is small. Upon verifying against the video images, we found the periodic minimum values do happen at the time when the SECOND digit character changes. During the period of when the SECOND digit doesn't change, S(n) does vary but normally it is smaller than when the digit character changes. These experiments prove that in actual situation, the minimum values of S(n) in every cycle of one second video do happen at the moment when the SECOND digit character changes.

4. DETECTING THE PRESENCE OF THE VIDEO CLOCK BY THE "SECOND" DIGIT TNPS

From the decoded video image, we first locate the static overlay region. From this region we segment all the ROIs for characters. The TNPS is calculated for each ROIs and monitored to see whether a ROI is for the SECOND digit character. After the SECOND digit character is located and segmented, the other digit characters are also located and segmented referring to the SECOND digit character position. We will not discuss the detail segmentation process here due the limited pages of this paper but focus on the recognition process based on correct segmentation result.

To recognize the video clock time, we first detect its presence based on the presence of periodic SECOND digit character image pattern change. By monitoring the TNPS in the region of the SECOND digit character, as described in section 3, we know whether it is a video clock or not. Our algorithm calculates the minimum value of TNPS every second (25 frames of video image if the video is in PAL system format). If the video frame locations of the accumulated M consecutive minimum values have equal period of one second, we conclude the video clock is present. Figure 4 shows M cycles of the SECOND digit character changes just after the video clock starts to appear. The digit character change positions are indicated as N_0, N_1, …, N_M, at which the TNPS sequence have minimum values in its respective one second period. The periods between these positions are equal and their periods are one second. Before the N_0 position, we cannot see any periodic minimum value, which means that the video clock hasn't appeared yet.

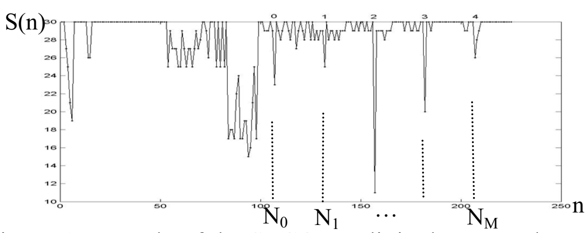


Figure 4. M cycle of the SECOND digit character changes.

According to Figure 3, the TNPS sequence has smaller variation except at the time of digit character changes. We can verify whether the pattern is a running clock digit or other fast changing patterns by calculating the variance of S(n) as equation (2) for all values except at the positions of N_0, N_1, …, N_M. If the variance is small enough, it means that the image pattern doesn't change except on the above periodic positions. For other fast changing patterns, it may happen to have periodic minimum values, but the value of E should be larger.

$$E = \sum_{\substack{n=N_0+1 \\ n\neq N_1,\ldots,N_{M-1}}}^{N_M-1} (S(n)-\bar{S})^2 \qquad (2)$$

Where

$$\bar{S} = \frac{1}{N_M - N_0 - M} \sum_{\substack{n=N_0+1 \\ n\neq N_1,\ldots,N_{M-1}}}^{N_M-1} S(n)$$

5. DETECTING THE FRAME POSITION OF THE MINUTE DIGIT PATTERN CHANGE AND INFER THE CLOCK TIME

After the presence of the video clock is detected by using the method described in section 4, we start to monitor the pattern change of the MINUTE digit character. When we first detect a pattern change, as we already mentioned in section 2, we infer the clock is showing "01:00" at that moment, namely, the soccer game has started for one minute. We can also monitor the pattern change of the TEN-SECOND digit character instead of the MINUTE digit character. In this case the clock will show "00:10" when a pattern change is detected, and we get the clock time sooner than we use the MINUTE digit character. In our application, we use the MINUTE digit character because the video clock may start to appear later more than 10 seconds after the game has started. For example, the clock may show "00:32" when it first appears and it has already passed "00:10". In this case the inference of the clock time will be wrong. In our statistics, most of the cases the video clock starts to appear before it reach one minute. Therefore, detecting the pattern change of the MINUTE digit character is more reliable for our application although we get the result a bit later. Fortunately, a few minutes delay is not a problem for our application.

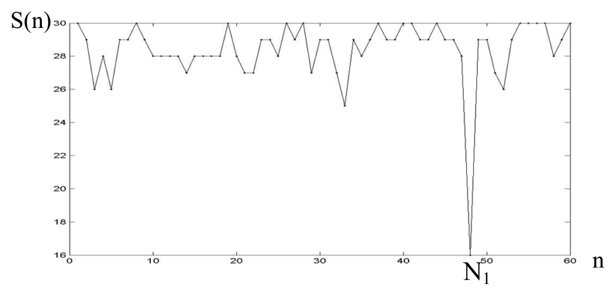


Figure 5. One minute of TNPS sequence.

To detect the video frame position when the MINUTE digit character pattern change, we look for the minimum value of TNPS in one minute of the video images, starting from when we detected the presence of the video clock. Here we assume that the video clock appears at least M seconds before it reaches one minute. Thus we can get the result latest M seconds before the clock reaches 2 minutes. At the moment when the minimum value of TNPS happened, the clock was showing “01:00”. Figure 5 shows one minute of TNPS sequence starting from just after the presence of the clock is confirmed. We can see at frame N_1 there is a minimum value, if this is the first time that the MINUTE digit character changes, the digit change from “0” to “1” at that time. Since when the MINUTE digit change, the SECOND and TEN-SECOND digits must be “0”, the clock time is inferred as “01:00”. From that time, the frame sequence number is synchronized with the clock time. The clock time can be calculated via the frame sequence number N as below,

$$t = (N-N1)/F + 60$$

where F is the frame rate. The unit of t is second.

6. EXPERIMENTAL RESULT

In our project, we use Visual C++ to implement the algorithm. The recorded soccer videos are in MPEG-1 format. An open source MPEG decoder is used to decode the video to image frames in the size of 352x288 while the video is playing back. At the same time the algorithm is taking the decoded image for processing. It calculates the TNPS using the previous image and the current image. The clock time recognition results are shown in the flowing table.

Table 1. Video clock time recognition result.

Game Name	UEFA 2005	Euro 2004	FIFA 2002	over all
games tested	8	4	6	18
out of condition	2	0	1	3
correct recognition	6	4	5	15
recognition rate without conditions	75%	100%	100%	83%
recognition rate with conditions	100%	100%	100%	100%

We have totally tested 18 games of UEFA Champion League 2005, Euro 2004 and FIFA 2002. There are 2 games in UEFA 2005 which the clock digit cannot be segmented because the video clock overlay is semi-transparent thus the digit character is merged with the video background. There is one game in the FIFA 2002 which the clock appears 3 minutes later after the game started. For all other 15 games which meet our conditions, the video clock time is recognized correctly. Here our conditions are:

1. The clock digit characters are segmented correctly.
2. The clock appears at least M seconds before it reaches one minute.
3. The SECOND digit character changes every one second.

In our experiments, we choose M=5. Namely, if we detect 5 SECOND digit character pattern changes, we consider the digital clock is showing on the video.

For the speed of the algorithm, we run our program in a Dell Optiplex P4 3.4Ghz desktop PC with 1GB memory. After segmentation, we do TNPS calculation and analysis for all ROIs in every single frame of image decoded from MPEG-1 video in real time. The processing can be done within 1/25 seconds thus the video can be played back in its normal frame rate of 25 frames per second. Therefore, the real time processing requirement is met.

We have also considered cases that the SECOND digit of the clock is not accurate, which means that sometimes the SECOND digit changes at the 24’th or 26’th frame rather than the 25’th frame exactly in the PAL video. In the case when the error is not large, our algorithm can still work by allowing an error range of the period. But if the SECOND digit error is too large, allowing a wide range of period may cause the clock presence detection error. Thus the inference of clock time will also be wrong. In this case we may use the TEN-SECOND instead of the SECOND digit character to detect the presence of the video clock.

7. CONCLUSION AND FUTURE WORK

A novel method based on the temporal periodic image pattern change is proposed to recognize the clock time showing on video. The recognition result is quite accurate and fast since we recognize the digit characters by detecting the pattern change and inference rather than recognizing the pattern. We have introduced a new way of video text recognition by making use of the temporal feature of these texts. This concept may be applied to other application area of video processing. Future work will be done for cases which the conditions are not met.

8. REFERENCES

[1] L. Xie, et.al, “Structure Analysis of Soccer Video with Domain Knowledge and Hidden Markov Models”, Pattern Recognition Letters, Vol 25, Issue 7, May 2004, pp 767-775.

[2] Kongwah WAN, Changsheng XU, “Efficient Multimodal Features for Automatic Soccer Highlight Generation”, Proc of ICPR, pp. 973-976, 2004.

[3] Kongwah Wan, Xin Yan, Xinguo Yu, Changsheng Xu, “Real-Time Goal-Mouth Detection In MPEG Soccer Video”, Proc of ACM Multimedia, pp. 311-314, 2003.

[4] Liexian Gu, Xiaoqing Ding, and Xian-Sheng Hua, “Online Play Segmentation for Broadcasted American Football TV Programs”, Proc of PCM, LNCS 3331, pp. 57-64, 2004.

[5] Mori, M., “Video text recognition using feature compensation as category-dependent feature extraction”, Proc of ICDAR, pp. 645-649, 2003.

[6] Jun Sun, Yushinobu Hotta, Yutaka Katsuyama, Satoshi Naoi, “Low resolution character recognition by dual eigenspace and synthetic degraded patterns, Proc of the 1st ACM workshop on Hardcopy document processing”, pp. 15-22, 2004.

[7] Rainer Lienhart, “Automatic Text Recognition for Video Indexing”, Proc of ACM Multimedia, pp. 11-20, 1997.