Reading Both Single and Multiple Digital Video Clocks Using Context-Aware Pixel Periodicity and Deep Learning

Xinguo Yu, Central China Normal University, Wuhan, China
Wu Song, Central China Normal University, Wuhan, China
Xiaopan Lyu, Central China Normal University, Wuhan, China
Bin He, Central China Normal University, Wuhan, China
Nan Ye, University of Queensland, Brisbane, Australia

ABSTRACT

This article presents an algorithm for reading both single and multiple digital video clocks by using a context-aware pixel periodicity method and a deep learning technique. Reading digital video clocks in real time is a very challenging problem. The first challenge is the clock digit localization. The existing pixel periodicity is not applicable to localizing multiple second-digit places. This article proposes a context-aware pixel periodicity method to identify the second-pixels of each clock. The second challenge is clock-digit recognition. For this task, the algorithms based a domain knowledge and deep learning technique is proposed to recognize clock digits. The proposed algorithm is better than the existing best one in two aspects. The first one is that it can read not only single digit video clock but also multiple digit video clocks. The other is that it requires a short length of a video clip. The experimental results show that the proposed algorithm can achieve 100% of accuracy in both localization and recognition for both single and multiple clocks.

KEYWORDS

Clock Digit Localization, Clock Digit Recognition, Context-Aware Pixel Periodicity, Deep Learning, Digit-Sequence

INTRODUCTION

Reading digital video clocks, also called time recognition, is an application-oriented research problem because clock time is the critical information of multiple applications in video analysis, video surveillance, panorama video production, and video indexing and retrieval (Bu, Sun, Ding, Miao, & Yang, 2008; Covavisaruch & Saengpanit, 2004; Li, Wan, Yan, Yu, & Xu, 2006; Li, Xu, Wan, Yan, & Yu, 2006; Xu, Wang, Wan, Li, & Duan, 2006; Yin, Hua, & Zhang, 2002; Yu, 2012; Yu & Ding, 2015; Yu, Li, & San Lee, 2008; Yu, Li, & Leong, 2009; Yu, Cheng, Wu, & Song, 2016; Yu, Ding, Zeng, & Leong, 2015; Yu, Lyu, Xiang, & Leong, 2017). Reading digital video clocks, especially reading multiple digital video clocks of a video, is a very challenging special case of reading text from overlaid video object, because reading digital video clock has multiple extra difficulties such as multiple asynchrony clocks, low resolution, and tight processing time. In fact, reading general scene text still is an open research problem(Anthimopoulos, Gatos, & Pratikakis, 2013; Epshtein,

DOI: 10.4018/IJDCF.2020040102

Ofek, & Wexler, 2010; Ghanei & Faez, 2015, 2016; Jaderberg, Simonyan, Vedaldi, & Zisserman, 2016; Lee, Lee, Lee, Yuille, & Koch, 2011; Lyu, Song, & Cai, 2005; Mishra, Alahari, & Jawahar, 2012; Neumann & Matas, 2012, 2013, 2015; Pan, Hou, & Liu, 2011; Shi, Wang, Xiao, Zhang, & Gao, 2013; Shi, Wang, Xiao, Gao, & Hu, 2014; Shivakumara, Phan, & Tan, 2011; Wang, Babenko, & Belongie, 2011; Wang, Wu, Coates, & Ng, 2012; Weinman, Learned-Miller, & Hanson, 2009; Zhong, Jin, Zhang, & Feng, 2016; Zhu & Zanibbi, 2016).

The clock time plays a critical role in video semantics analysis. The time on clocks often indicates the game time or event time in sports and video surveillance (Xu et al., 2006; Zhong et al., 2016; Zhu & Zanibbi, 2016). This paper considers the common case in which digital video clocks have been superimposed on video. While current videos already can have a text channel to store the encoded clock or/and timestamp information, this paper proposed algorithm does not need to use these encoded clocks or timestamps (Bu et al., 2008; Covavisaruch & Saengpanit, 2004). Thus, the proposed algorithm has a wider application range. More importantly, it can avoid the harm from the malicious modification to the encoded timestamp stored in text channel.

A lot of sports and surveillance videos have superimposed digital video clocks or/and timestamps for various reasons — such as to show game-related time or to show the time of the recording. For example, video clocks in a soccer video indicate game time lapsed at a frame, whereas reversely-running game clocks in basketball videos indicate the remaining game time at a frame and reversely-running shot clocks indicate the longest remaining time of the current ball possession. Examples of single and multiple digital video clocks in soccer and basketball videos are shown in Figure 1. In surveillance videos, superimposing digital video clocks or timestamps into videos (Yu et al., 2016) is one method guard against malicious tampering of the encoded timestamp information stored in video text channel. Hence, there is a need for algorithms for reading the superimposed digital video clock, independently of the clock or timestamp encoded in video text channel.

In sports (soccer, basketball) video analysis, the clock time is not only the key element of the video metadata in video indexing and annotation, but it is also the best reference for synchronization of multimodal contents extracted from videos, such as videos, gamelogs, and related reports on the same event (Lee et al., 2011; Li, Wan, et al., 2006). For example, by synchronizing the clock time (game time) with the event time provided by gamelog, soccer events can be detected more accurately at a low computing cost (Yu et al., 2009). Sequentially the events detected in a very short time can be used to provide live event alert service in sports video analysis, as reported in (Yu et al., 2009). In basketball videos, the superimposed video clock runs when the game is playing and it stops when the game pauses. Hence the information of both game and shot clocks can be used in segmenting play and break sections (Yu & Ding, 2015). In panorama video production, videos from multiple cameras may have multiple clocks so that producing high quality panorama video needs to remove clocks from individual video and to plant a clock into panorama video based on the results of reading clocks. In surveillance video, clocks indicate real time of taking videos. Hence, clock time recognition helps find time chains of person-activity from surveillance videos of city surveillance systems. In summary, reading multiple digital video clocks is a valuable and active research problem with important applications in video analysis, video processing, video summary, and video surveillance.

The problem of reading digital video clocks can be divided into two sub-problems: clock-digit localization and clock-digit recognition. The first sub-problem is a special case of the character localization problem. However, text localization, which is a main step of character localization, is a difficult problem too and the existing algorithms cannot have a satisfactory performance in term of the industrial criteria, especially for localizing digits due to that it is hard to differentiate some digits (Anthimopoulos et al., 2013; Epshtein et al., 2010; Ghanei & Faez, 2015, 2016; Jaderberg et al., 2016; Lee et al., 2011; Lyu et al., 2005; Mishra et al., 2012; Neumann & Matas, 2012, 2013, 2015; Pan et al., 2011; Sermanet, Kavukcuoglu, & LeCun, 2009; Shi et al., 2013, 2014; Shivakumara et al., 2011; Wang et al., 2011; Wang et al., 2012; Weinman et al., 2009; Yi & Tian, 2011, 2012; Zhong et al., 2016; Zhu & Zanibbi, 2016). Thus, researchers change to design custom algorithms for localizing clock digits (Bu et al., 2008; Covavisaruch & Saengpanit, 2004; Li, Wan, et al., 2006; Li, Xu, et al., 2006; Yin et al., 2002; Yu, 2012; Yu & Ding, 2015; Yu et al., 2008, 2009, 2016, 2015).

Reading both single and multiple digital video clocks is even more difficult than some scene text recognition problems because the digital video clocks in sports and surveillance videos pose several extra difficulties with respect to digit localization and clock-digit recognition:

- Asynchronized multiple digital video clocks may appear in the same video;
- Digits of the video clock are in very low resolution. The dimension of a digit region is in the range of only 4×7 to 7×12 pixels in MPEG-1 and MPEG-2 videos, respectively;
- Digital video clocks vary in color, size, font, and format for different videos.

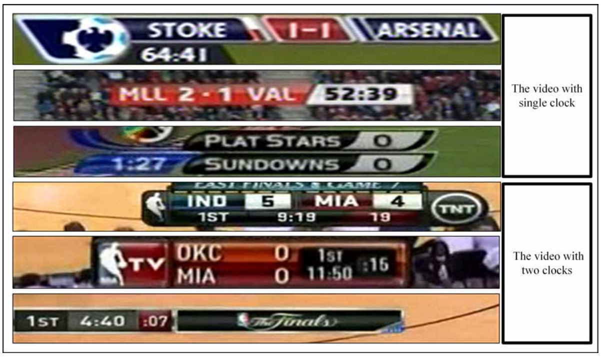
The sample video clocks in Figure 1 illustrate some of these challenges. The experimental results in this paper will show that existing OCR and deep learning algorithms applied to individual video frames cannot reliably read digital video clocks under the required conditions. Researchers have designed custom algorithms for solving problem of reading digital video clocks (Bu et al., 2008; Covavisaruch & Saengpanit, 2004; Li, Wan, et al., 2006; Li, Xu, et al., 2006; Yin et al., 2002; Yu, 2012; Yu & Ding, 2015; Yu et al., 2008, 2009, 2016, 2015).

Some methods take an image processing approach to localize clocks (Bu et al., 2008; Covavisaruch & Saengpanit, 2004). The main idea of these methods is to use a series of image processing operations to only keep some pixels that belong to clocks (Bu et al., 2008; Covavisaruch & Saengpanit, 2004). It was reported that the performance of these methods can only achieve 85.6% of accuracies for timestamp detection and that they are also time consuming. The procedures in (Li, Wan, et al., 2006) and (Li, Xu, et al., 2006) for localizing clock digits mainly use image processing techniques too. Hence, they are tedious and not robust. For clock-digit localization (Yu, 2012; Yu et al., 2015) proposed a pixel periodicity method. This method has a very good performance for localizing the second-digit place if only a single digital clock is in a video, but it is not applicable to localization of multiple second-digit places because multiple clocks probably have asynchrony periodicities in digit transit, i.e. they change their digits on second-digit places at different frames.

The second sub-problem of reading digital video clocks is clock-digit recognition, a special case of OCR (Optical Character Recognition) problem since the time on digital video clock consists of digits. Again, researchers developed the custom algorithms for recognizing clock digits.

The first algorithm for reading digital video clocks is based on the idea of detecting transit frames (Li, Wan, et al., 2006; Li, Xu, et al., 2006). Transit frames with respect to a digit place are frames on which the considered digit place transits its digit. That means that the algorithm makes

Figure 1. The samples of various digit video clock boards. The bottom three among the above 6 clock boards are samples that have two clocks.



use of the domain knowledge that the clock-digits change according to the rules of clock: the second-digit cycles through 0 to 9 and then it causes a transit in the ten-second-digit, and so on. Based on this, the method proposed in (Li, Wan, et al., 2006) and (Li, Xu, et al., 2006) first detects the transit frame for the ten-second-digit place of a clock. The algorithm can then infer the number represented by the second-digit (which is reset to 0 after the transit frame). The detection of the transit frame is solved by monitoring the changes of the ten-second-digit over time and can be modeled as finding a local extrema for the correlation function of frame transits. This method involves the use of time-consuming image processing techniques such as clock board detection, local color analysis, character candidate extraction, CCA (connected component analysis), and morphology. One obvious demerit of this method is that it requires a longer clip (maybe several tens of seconds) as input to detect transit frames of ten-second-digit.

In contrast, the algorithm proposed in (Yu et al., 2008) and (Yu et al., 2015) is based on the idea of recognizing the periodic transits of the second-digit sequence as follows: since the second-digit goes through the periodic transit from 0 to 9, once the transit frames of the second-digit are known, the digit image instances of from 0 to 9 can be collected and they are in the periodic increasing order if they placed in appearance order in the video. The algorithm then can recognize this periodic digit sequence, called as digit-sequence recognition. For this method to work, the clock-digits must be first localized and extracted before the digit-sequence recognition. One demerit of this method cannot work for reading multiple digit video clocks because the multiple clocks may have asynchronized transit frames. Another demerit of this method is that the digit-sequence recognition requires that the length of input clip is at least 8 second long to become very robust.

This paper develops an algorithm that can read not only single digital video clock but also multiple digital video clocks. For second place localization, a context-aware pixel periodicity method (CPP) is proposed to strengthen the pixel periodicity method (PPM) presented in (Yu, 2012) and (Yu et al., 2015). CPP can directly identify the pixels belong to second places without detecting the transit frames of second places. This solves the difficulty that multiple digital video clocks have different transit frames. This method captures the facts that some pixels in second-digit place of a running clock will change their gray values every second, some other pixels keep relative constancy for several seconds, and they are mutual support by being neighbors. The CPP method can reliably identify the pixels belong to second place without knowing transit frames so that it can directly and simultaneously localizes multiple second-digit places.

For clock-digit recognition, this paper proposes customized algorithms based on domain knowledge and deep learning technique to recognize digit-sequence and repeated digits. First a digit-sequence deep learning algorithm is developed to recognize digits on the second places of multiple clocks and to find the transit frames. Then a digit-repeated deep learning algorithm is developed to recognize the digits on other digit places.

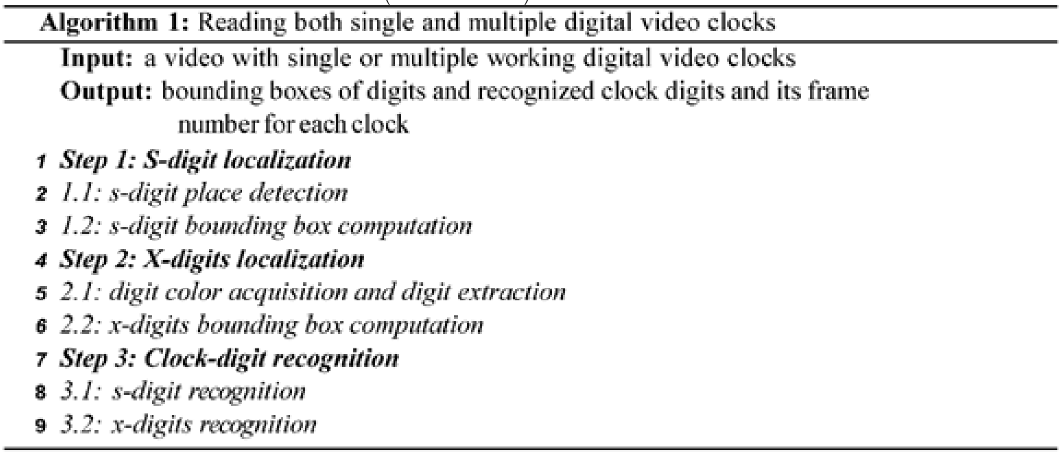
The rest of the paper is organized as follows. The second section gives an overview of the proposed algorithm. The third section presents the technical details of the proposed algorithm for reading single and multiple digital video clocks. Experimental results are presented in the fourth section. The fifth section concludes the paper.

PROBLEM DEFINITION AND ALGORITHM OVERVIEW

This section first formalizes the problem of reading digital video clocks and then gives an overview of the proposed algorithm. A video may have several clocks. For example, a basketball video may have game clock and shot clock. The number of digits on a digital video clock can vary from 3 to 14, but the core task is to read the four clock-digits representing a second, ten seconds, a minute, and ten minutes, denoted as s-digit, ts-digit, m-digit, and tm-digit in the rest of this paper, respectively. A clock may only have 3 digits to represent time when it needs a single digit to represent minutes, but for simplicity of presentation a clock is assumed to have four clock digits in the rest of the paper. With this convention, the problem of reading digital video clocks is stated as follows.

Problem of Reading Digital Video Clocks: Let $B_i^k = \left(r_i^k, c_i^k, w_i^k, h_i^k\right)$ for $i = 1$ to 4 and $k = 1$ to M be the bounding boxes of s-digit, ts-digit, m-digit, and tm-digit of k th working digital video clock, respectively. The problem of reading multiple digital video clocks is to:

1. Localize boxes $B_i^k = \left(r_i^k, c_i^k, w_i^k, h_i^k\right)$ for $i = 1$ to 4 and $k = 1$ to M;
2. Recognize digits in $B_i^k = \left(r_i^k, c_i^k, w_i^k, h_i^k\right)$ for $i = 1$ to 4 and $k = 1$ to M.



```
Algorithm 1: Reading both single and multiple digital video clocks
  Input: a video with single or multiple working digital video clocks
  Output: bounding boxes of digits and recognized clock digits and its frame
          number for each clock
Step 1: S-digit localization
1.1: s-digit place detection
1.2: s-digit bounding box computation
Step 2: X-digits localization
2.1: digit color acquisition and digit extraction
2.2: x-digits bounding box computation
Step 3: Clock-digit recognition
3.1: s-digit recognition
3.2: x-digits recognition
```

An algorithm for reading both single and multiple digital video clocks is depicted in Algorithm 1. The input of the algorithm is a video that has single or multiple working digital video clocks and the output is both the frame number of the first frame in which the time is recognized and the time of video clock in this frame for each clock. The objective of the algorithm is to recognize the time of digital video clock as early as possible in term of video time with a low cost of computing. Suppose that the video clock starts to appear in the i th frame and the algorithm recognizes the time in the $\left(i + \Delta i\right)$ th frame, where Δi wants to be as small as possible and the computing time is as short as possible.

The proposed algorithm for reading multiple digital video clocks possesses three main phases: s-digit localization, x-digit localization, and clock-digit recognition, and they are described as follows:

- **S-digit localization:** This is the first critical step of the proposed algorithm, which finds the bounding boxes of s-digits. In the literature (Li, Wan, et al., 2006) and (Li, Xu, et al., 2006), the bounding box of an s-digit is obtained by a tedious image processing procedure which conducts character candidate extraction, connected component analysis, and character candidate monitoring. Such a procedure is time consuming and error-prone. Unlike this approach, the method proposed in our previous papers takes the video analysis approach to localize s-digit (Yu, 2012; Yu et al., 2015). Particularly this method uses the grey value change information of s-digit pixels to compute the bounding box of s-digit. On the pixel value change of s-digit a piece of knowledge is discovered, called pixel periodicity by using the fact that some pixels in s-digit region will change their values when s-digit changes its digit and their values keep relatively constant between two transit frames. Besides this periodicity, this paper discovers and uses other pieces of knowledge, which is that some pixels in s-digit region are relatively constant because they are the background pixel, although it belongs to second place. In addition, two types of pixels are mutual support by being neighbors because they are within the same s-digit place. Based on these observed facts, a context-aware pixel periodicity method (shorted as CPP) is formed. A set of functions are defined to capture the knowledge and thus the s-digit localization problem is converted into the problem of computing these functions. Three actions are designed to obtain the s-digit bounding box. The first action is to obtain the set of the s-digit pixel candidates leveraging on the context-aware pixel periodicity method. The second one is to find the clusters of s-digit pixel candidates as the approximate regions of s-digits. Third one is to use a further local image analysis process in a small area to get the precise s-digit bounding;

- **X-digits localization:** To localize x-digits, the digit color and the background color need to be known first so that they can be used to extract digits on the other three-digit places. Since the s-digit bounding box has been localized by the s-digit localization method, a set of instances can be collected from an s-digit of the given video. Thus, the digit color and digit background colors can be found by analyzing the set of digit instances and identified fore- and back-ground s-digit pixels. Using the two acquired colors, all clock digits of a clock can be extracted. At the same time, all digit pixels can be converted into white and all background pixels can be converted into black. Thus, the color difference of various digital video clocks is eliminated for the following steps of the algorithm. As you may know, the four clock-digits are in a horizontal line in the same size and that the gaps between two neighbor digits are the same, except that a semicolon may be added between the m-digit and ts-digit. A digit location equation system with two unknowns is built to localize other digit places by using the above-observed facts. And a Hough-like search procedure is proposed to find the solution of this system. This completes the sub-problem of the clock-digit localization;
- **Clock-digit recognition:** Assume that the frame rate *R* is integer in this paper because the related formula through slight changes can be applied to the case that *R* is float. Generally speaking, the s-digit transit frames can be found by defining a function based on pixel-value change with the localized s-digit bounding box (Yu, 2012; Yu et al., 2015). However, to calculate this function reliably the input clip is at least 11 second long. This paper proposes a new method that can find the transit frames and recognize s-digit at the same time. This method goes to recognize and compare 3-digit sequences, which are the digit sequence formed from the digits on s-digit place on frame b . $R+b$, $2R+b$ for b .from 1 to R . Then a digit-sequence deep learning algorithm is designed to recognize these 3-digit sequences and to inference transit frames and digits on s-digit places. By adopting partition strategy, this algorithm only needs to recognize $\lceil \ln R \rceil$ .times of 3-digit sequences. All transit frames of x-digits are known after s-digit transit frames are known. Based on this fact a digit-repeated deep learning algorithm is proposed to recognize the repeated digits extracted from x-digit places.

READING DIGITAL VIDEO CLOCKS

This section presents the technical details of the three steps forming the proposed algorithm for reading digital video clocks in the following three sub-sections, respectively.

S-Digit Localization

This section presents procedure for finding the bounding boxes of clock-second (s-digit) places, leveraging on the proposed context-aware pixel periodicity method (CPP), which uses not only the periodicity of s-digit pixels but also the context information of the s-digit pixels. This new method captures the facts that some pixels in s-digit region will significantly change their gray values when s-digit transits its digit, some other pixels in s-digit region or on s-digit bounding boxes have relatively constant gray values, and these pixels from the same s-digit are neighbors. The pixel periodicity captures that the observation that some pixels in s-digit region of a working video clock change their grey values every second. Figure 2 gives the illustration of this pixel periodicity on the s-digit place. The notations and concepts are defined first, and then the formula for computing the s-digit bounding box are presented.

Let F_i be the considered frame. Then F_{i-R}, F_{i-R+1}, ..., F_{i-1} and F_i, F_{i+1}, ..., F_{i+R-1} are the R frames in the preceding second and the succeeding second, respectively. Let $c\left(k,p\right)$ be the grey value of pixel p in frame F_k. Then the authors have following definitions.

Definition 1: (Constant Pixel) Let F_k for $k=1$ to L to be L frames including at least 3 second consecutive frames. Pixel p is called as a constant pixel if it meets the following condition:

1. $\left|c\left(k,p\right)-C_1\right|<\beta_1$ for $k=i$ to L, where $C_1=\frac{1}{L}\sum_{k=1}^{L}c\left(k,p\right)$, where β_1 is a threshold.

Definition 2: Pixel p is called a periodicity pixel at frame i if it simultaneously meets the following three conditions:

1. $\left|c\left(k,p\right)-C_2\right|<\beta_2$ for $k=i-R+1$ to $i-1$, where $C_2=\frac{1}{R}\sum_{k=i-R}^{i-1}c\left(k,p\right)$;
2. $\left|c\left(k,p\right)-C_3\right|<\beta_2$ for $k=i$ to $i+R-1$, where $C_3=\frac{1}{R}\sum_{k=i}^{i+R-1}c\left(k,p\right)$;
3. $\left|C_2-C_3\right|>\beta_3$ where β_2 and β_3 are two thresholds.

Definition 3: (Periodicity Pixel) Pixel p is called a $\left(O,k\right)$ periodicity pixel if it meets the following conditions:

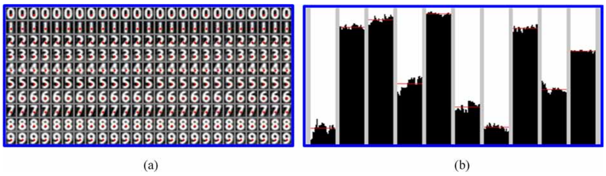
1. It is a second periodicity pixel at i_1, i_2, ..., i_k frames;
2. $abs(\left(i_u-i_v\right)$ mod $R)<2$ for $1\le v<u\le k$.

Definition 4: (S-Digit Pixel Candidate) Pixel p is called an s-digit pixel candidate if it is $\left(O,k\right)$ periodicity pixel for $k\ge 2$ and it is less than β_4 pixels away from a constant pixel or another s-digit pixel candidate.

Definition 5: (S-Digit Region) A cluster of s-digit pixel candidate is considered as an s-digit place if its cardinality is larger than β_5.

Once an s-digit place is found the bounding box of this s-digit place can be obtained through a local analysis.

Figure 2. The second-pixel periodicity illustration of a sample pixel in s-digit region in 10 seconds. (a) The red dots are the positions of a monitored pixel in s-digit region through 10 seconds (the resolution of each instance is 11 × 14 and the coordinate of the red dots is at piexl (6,9)). (b) The graph of the grey value of the monitored pixel in 10 seconds. Each short horizontal line indicates the average of the corresponding second of the gray values of the monitored pixel.



X-Digits Localization

This section presents the procedure that localizes the x-digits of a clock using a Hough-like procedure which considers that the bounding box of s-digit has acquired in the preceding section as its input. This procedure does color learning and color-based digit extraction.

Digit Color Acquisition and Conversion

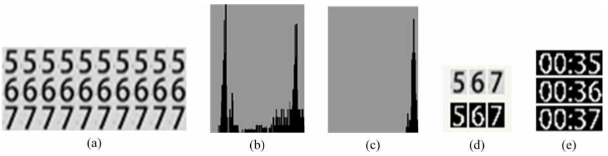
It is presented the procedure for acquiring the digit color and the digit background color for the digit instances of a given video. At first, all the digit instances on an s-digit place from a clip of *v* seconds, forming a set $S = \{s_i : i = 1 \text{ to } v{*}R\}$. Figure 3 (a) shows all the s-digit instances of a 3-second long clip. Figure 3 (c) is the histogram of these digit instances, called the instance histogram. The histogram of these digit instances consists of two portions: one corresponding to the digit color and the other corresponding to the background color. The constant pixels near the boundary of the bounding box can form a single-peak histogram as shown in Figure 3 (b). Thus, the digit histogram can be obtained by subtracting the background color histogram from the instance histogram, as illustrated in Figure 3 (c). Then the digit color Gaussian and the background color Gaussian can be found from the digit and the background color histogram. Next the two color Gaussians are used to identify the pixels on the digit. At the same time each digit pixel is changed into 255 (white) and the one of each digit background pixel into 0 (black) as Figure 3 (d) and (e) shows. Based on the above discussion and description a procedure, called as Procedure I (for the concise of the presentation this procedure is not presented here), is formed to extract digits and to convert digit instances for digital video clock.

X-Digits Bounding Box Computation

This section aims to determine the bounding boxes of x-digits after the s-digit bounding box is acquired and color conversion has done. Let $B = \left(r, c, w, h\right)$ denote the s-digit bounding box found in the preceding section. Let $B_i = \left(r_i, c_i, w_i, h_i\right)$ for $i = 1$ to 4 denote the boxes of s-digit, ts-digit, m-digit, and tm-digit in a digital video clock, respectively. For the convenience, B or B_1 is alternatively used to represent the box of s-digit. Due to the four boxes having the same dimension and also the distance between B_1 and B_2 being the same as the distance between B_3 and B_4 an equation system for $B_i = \left(r_i, c_i, w_i, h_i\right)$ for $i = 1$ to 4 forms as follows:

$$\Theta : \begin{cases} \left(r_2, c_2, w_2, h_2\right) & = & \left(r, c - d_1, w, h\right) \\ \left(r_3, c_3, w_3, h_3\right) & = & \left(r, c - d_1 - d_2, w, h\right) \\ \left(r_4, c_4, w_4, h_4\right) & = & \left(r, c - 2d_1 - d_2, w, h\right) \end{cases} \tag{1}$$

Figure 3. The digit instances, their histograms, and the samples of the digit extraction and conversion. The digit instances of s-digit are shown in (a) from a s-digit place of a 3 second clip; the histogram of all instances in (a) is given in (b); the histogram of the constancy pixels is given in (c); three s-digit instances and their converted version are given in (d) and a color converted clock is given in (e).



The locations of the four clock-digits are determined after a Hough-like procedure is used to determine d_1 and d_2. The Hough space is $H = \left\{ \left(d_1, d_2 \right) : \eta_1^1 \le d_1 \le \eta_2^1 \,\&\, \eta_1^2 \le d_1 \le \eta_2^2 \right\}$ where $\eta_1^1, \eta_2^1, \eta_1^2, \eta_2^2$ are constant integers.

Let l_i be the vertical middle line of B_i and $d\left(p, l_i\right)$ is the distance from p to l_i. A potential energy function is defined as follows:

$$E\left(p\right) = \begin{cases} \left[d\left(p, l_i\right)\right]^2 & if\, p\, is\, a\, digit\, pixel \\ 0 & if\, p\, is\, not\, a\, digit\, pixel \end{cases} \tag{2}$$

Thus, the measure function of each cell of *H* can be defined as follows:

$$\Lambda\left(d_1, d_2\right) = \sum_{i=2}^{4} \sum_{\beta \in B_i} E\left(p\right) \tag{3}$$

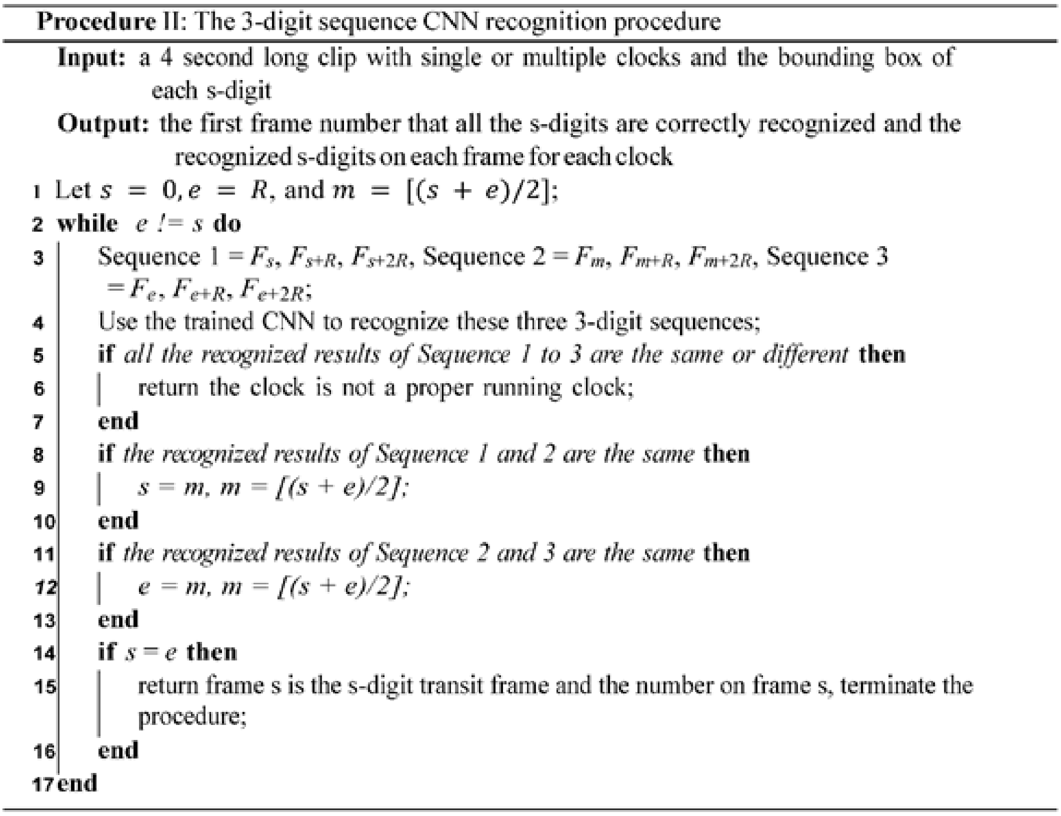
For all pairs of $\left(d_1, d_2\right)$ in H $\Lambda\left(d_1, d_2\right)$.is computed. And $\left(d_1, d_2\right)$ corresponding to the minimum of $\Lambda\left(d_1, d_2\right)$ is the solution of the equation system.

Clock-Digit Recognition

Digit-sequence recognition method used in (Yu et al., 2008) and (Yu et al., 2015) is a robust method that can recognizes s-digit when both the bounding box of s-digit and s-digit transit frames are known. However, the digit-sequence method requires the input clip is at least 8 second long to be robust. To reduce the requirement to the length of the input clip, this paper proposes a 3-digit sequence deep learning procedure to find the s-digit transit frames and to recognize s-digits simultaneously. We use the trained CNN[1].

The bounding box of s-digit of each clock is known by using the s-digit localization procedure presented in the preceding sections. In fact, frames from $t + k{*}R + 1$ to $t + \left(k + 1\right){*}R$ have the same s-digit if frame t is s-digit transit frame because the s-digit transit frames are the frames that s-digit transits its digit. Thus, the s-digit in the frames $t + k{*}R + 1$ to $t + \left(k + 1\right){*}R$ is number k if the s-digit in the frames from t to $t + R$ is “0”. In other words, the s-digits in the frames from t to $t + v{*}R$ form a digit periodic increasing sequence according to the clock knowledge, supposed that the input clip is v second long $(v < 10)$.

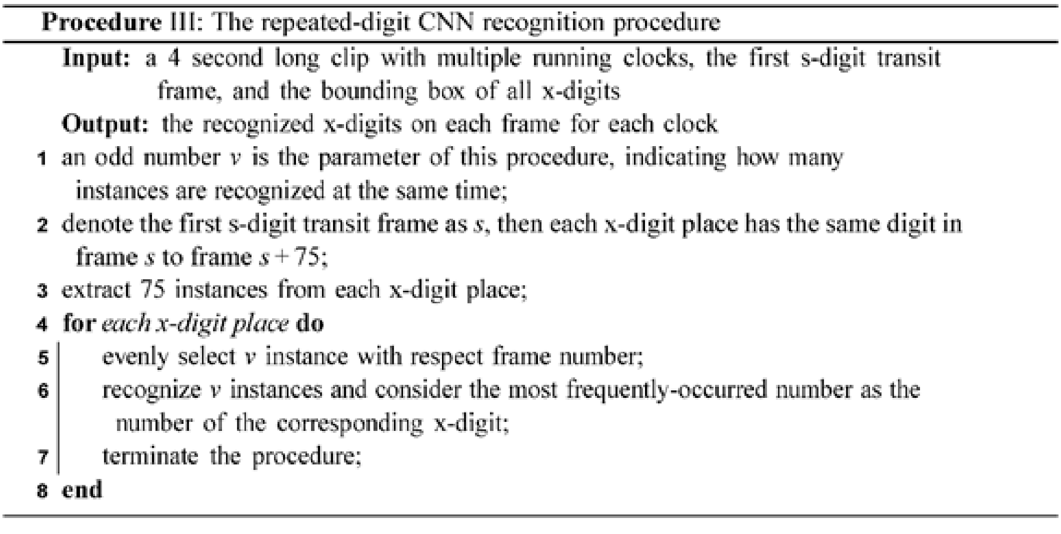
Based on these facts, this paper proposes a 3-digit sequence CNN recognition procedure for finding s-digit transit frames and recognizing s-digits simultaneously is formed as follows, denoted as Procedure II.



Procedure II: The 3-digit sequence CNN recognition procedure

```
Input: a 4 second long clip with single or multiple clocks and the bounding box of
       each s-digit
Output: the first frame number that all the s-digits are correctly recognized and the
        recognized s-digits on each frame for each clock
Let $s = 0, e = R$, and $m = [(s + e)/2]$;
while $e != s$ do
    Sequence 1 = $F_s, F_{s+R}, F_{s+2R}$, Sequence 2 = $F_m, F_{m+R}, F_{m+2R}$, Sequence 3
     = $F_e, F_{e+R}, F_{e+2R}$;
    Use the trained CNN to recognize these three 3-digit sequences;
    if all the recognized results of Sequence 1 to 3 are the same or different then
        return the clock is not a proper running clock;
    end
    if the recognized results of Sequence 1 and 2 are the same then
        $s = m, m = [(s + e)/2]$;
    end
    if the recognized results of Sequence 2 and 3 are the same then
        $e = m, m = [(s + e)/2]$;
    end
    if $s = e$ then
        return frame s is the s-digit transit frame and the number on frame s, terminate the
        procedure;
    end
end
```

Once s-digit transit frames are known, all the transit frames for all x-digits are known. Considering the transit frame between 0 and 24, the authors can take at least 75 frames with the same digit for any x-digit from a 4 second long clip (Notice that the video in this paper is 25 frames per second). Hence an odd number of frames from these 75 frames can be selected to recognize an x-digit in Procedure III.



Procedure III: The repeated-digit CNN recognition procedure

```
Input: a 4 second long clip with multiple running clocks, the first s-digit transit
       frame, and the bounding box of all x-digits
Output: the recognized x-digits on each frame for each clock
an odd number $v$ is the parameter of this procedure, indicating how many
  instances are recognized at the same time;
denote the first s-digit transit frame as $s$, then each x-digit place has the same digit in
  frame $s$ to frame $s + 75$;
extract 75 instances from each x-digit place;
for each x-digit place do
    evenly select $v$ instance with respect frame number;
    recognize $v$ instances and consider the most frequently-occurred number as the
      number of the corresponding x-digit;
    terminate the procedure;
end
```

Procedure II and III together can recognize all digits of all clocks. In other word, clock-digit recognition step is finished.

EXPERIMENTAL RESULTS

The algorithm for reading both single and multiple digital video clocks presented in this paper is implemented in C++. A dataset is built to evaluate the proposed algorithm. This dataset has 1500 MPEG-2 video clips consisting of three 500 clips and each clip is 15 second long. Each of the first 500 clips contains a single clock, each of the second 500 clips contains two clocks, and each of the third 500 clips contains four clocks. The first 500 clips are edited from broadcast soccer, basketball videos, and surveillance videos. The second 500 clips are edited from broadcast basketball videos. Each of the third 500 clips is generated by a program from two different two basketball clips, which produces a clip with four clocks by copying the two clocks from a frame of a clip and pasting the copied two clocks on the corresponding frame of the other clip. The produced clip is accepted when the four clocks are at the different locations. All the clocks in the dataset work from the beginning to the end of the clip. These clips vary in digit color, digit background color, size, and font.

Performance of Clock Digits Localization

(Yu, 2012) and (Yu et al., 2015) presented a pixel periodicity method (shorted as PPM) to localize the second digit of digital video clock. This method can localize the second place with 100% of accuracy, but it requires that the length of input clips is at least 11 second long. And it is not applicable to localize multiple second places because it needs to detect the transit frames first. This paper proposes a context-aware pixel periodicity method (shorted as CPP) that is enhanced from PPM. The experiments are conducted to compare the accuracies of PPM and CPP for localizing single and multiple s-digits. And the results are given in Table 1.

In Table 1, “Single Clock” means 500 clips that each clip carries a single digital video clock, “Two Clocks” means 500 clips that each clip carries two digital video clocks, and “Four Clocks” means 500 clips that each clip carries four digital video clocks and they are tested in batch; “X” means that the method is not applicable for the corresponding cases. Table 1 shows that the second digit periodicity method (PPM) can achieve a 100% of accuracy for single s-digit localization until the length of input clips reaches 11 second and that CPP can localize not only single but also multiple s-digits if the require length of clips reach 5 seconds.

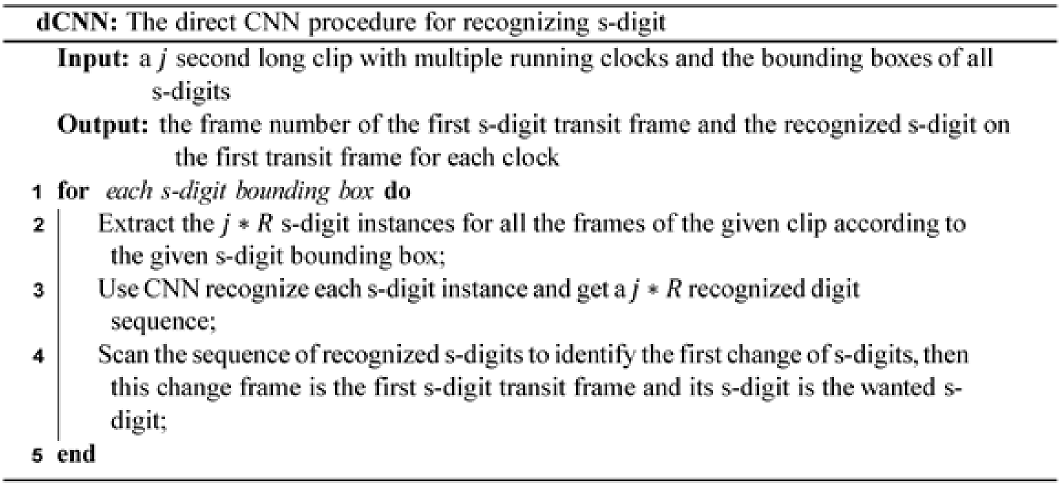
Performance Comparison on Clock Digit Recognition

The authors first present two procedures, called as dCNN and CollectiveCNN. Then the experiments are conducted to compare the performances of s-digit recognition of four different methods, dSequence

Table 1. Comparison on accuracies of PPM and CPP for localizing s-digit on 500 clips with single digit clock, 500 clips with two digit clocks, and 500 clips with four digit clocks

Length in Second of Clip			4	5	6	7	8	9	10	11
Accuracy in %	Single Clock	PPM	0	0	3	72	81	82	91	100
		CPP	86.4	100	100	100	100	100	100	100
	Two Clocks	PPM	X	X	X	X	X	X	X	X
		CPP	80.3	100	100	100	100	100	100	100
	Four Clocks	PPM	X	X	X	X	X	X	X	X
		CPP	70	100	100	100	100	100	100	100

(digit-sequence procedure presented in (Yu et al., 2015)), Procedure II, dCNN, and CollectiveCNN. It is a widely-taken approach to use deep learning technique to recognize the localized character. Hence the authors take dCNN and CollectiveCNN as baseline procedures to evaluate the proposed procedures. Procedure dCNN is prepared by directly using CNN to recognize s-digit and find s-digit transit frames.



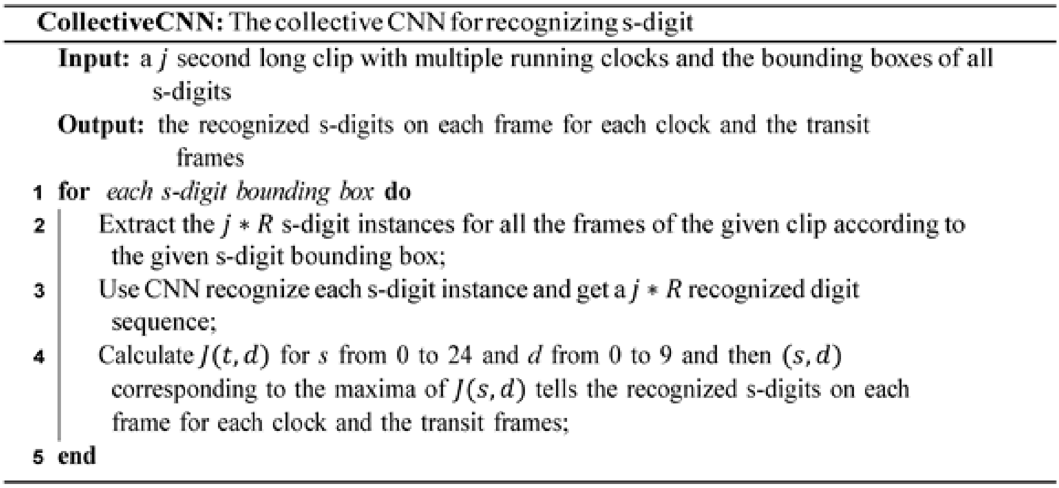
dCNN: The direct CNN procedure for recognizing s-digit

```
Input: a j second long clip with multiple running clocks and the bounding boxes of all
       s-digits
Output: the frame number of the first s-digit transit frame and the recognized s-digit on
        the first transit frame for each clock
for each s-digit bounding box do
    Extract the $j * R$ s-digit instances for all the frames of the given clip according to
      the given s-digit bounding box;
    Use CNN recognize each s-digit instance and get a $j * R$ recognized digit
      sequence;
    Scan the sequence of recognized s-digits to identify the first change of s-digits, then
      this change frame is the first s-digit transit frame and its s-digit is the wanted s-
      digit;
end
```

To present the collective CNN procedure, denoted as CollectiveCNN, the authors first define $J(t,d)$.

Definition 6: Let $s_1, s_2, \ldots s_{j*R}$ be $j*R$ s-digit instances for a given bounding box from j second long clip. Let r_i be the recognized digit by apply CNN on s_i. The digit series should be $v_1, v_2, \ldots, v_{j*R}$ if t is the first transit frame and that d is the digit on s_1. With these notations, $J(t,d)$ is defined as below:

$$J(t,d) = \sum_{i=1}^{j*R} (r_i \equiv v_i) \tag{4}$$



CollectiveCNN: The collective CNN for recognizing s-digit

```
Input: a j second long clip with multiple running clocks and the bounding boxes of all
       s-digits
Output: the recognized s-digits on each frame for each clock and the transit
        frames
for each s-digit bounding box do
    Extract the $j * R$ s-digit instances for all the frames of the given clip according to
      the given s-digit bounding box;
    Use CNN recognize each s-digit instance and get a $j * R$ recognized digit
      sequence;
    Calculate $J(t, d)$ for $s$ from 0 to 24 and $d$ from 0 to 9 and then $(s, d)$
      corresponding to the maxima of $J(s, d)$ tells the recognized s-digits on each
      frame for each clock and the transit frames;
end
```

Table 2 gives the experimental results of s-digit recognition by using four different pro- cedures, dSequence, dCNN, CollectiveCNN, and Procedure II. In Table 2, "Single Clock", "Two Clocks", "Four Clocks", and "X" have the same meaning as in Table 1. Table 2 presents accuracy of recognizing s-digit by using dSequence, dCNN, CollectiveCNN, and Procedure II proposed in this paper. There are three conclusions from Table 2. The first one is that dSequence can be used to recognize s-digits of single digital clocks but not multiple s-digits. The second one is that dCNN cannot get 100% of accuracy. Third one is that both Collective and Procedure II can recognize s-digits of not only single digital clocks, but also multiple digital clocks and it can achieve a 100% of accuracy when the input video is longer than 4 second.

In Table 3, "v (procedure parameter)" is the parameter of Procedure III, indicating how many instances are used in recognition process; "Single Clock", "Two Clocks", and "Four Clocks" have the same meaning as in Table 1. Table 3 shows that CNN cannot get a 100% of accuracy if it used to recognize single instance. In contrast, Table 3 shows that the proposed Procedure III can achieve a 100% of accuracy if v is equal or larger than 5, i.e. $v \geq 5$ for not only single digital clock but also multiple digital clocks.

Comparison on Computational Time Costs

The experiments are conducted to compare the computational time costs and their variants of clock digit localization and clock digit recognition by using the proposed algorithm and the relevant various procedures. In the experiment, each of three 500 clips are split into five 100-clip groups, respectively.

In Table 4, μ and σ are the means and the variances of the computation times of finishing a task for a batch of 100 clips. From the table, the authors have two conclusions. The first one is that each step of our algorithm is very fast and the whole algorithm finishes within 4.9 seconds. The other one is that for s-digit recognition the proposed Procedure II is faster than CollectiveCNN and dCNN by 5 to 6 times.

From Table 2 and Table 3 the authors have the following conclusion. The proposed Procedure II is faster than CollectiveCNN by 5 times, though both Procedure II and CollectiveCNN can achieve a 100% of accuracy for recognizing s-digits for single and multiple digital clocks when the length of the input clips reaches 4 seconds.

Table 2. Accuracy comparison of recognizing s-digit by using dSequence, dCNN, CollectiveCNN, and Procedure II on 500 clips with single clock, 500 clips with two clocks, and 500 clips with four clocks

Length in Second of Clip		2	3	4	5	6	7	8	9	10
Single Clock	dSequence	X	0	0	3	72	81	82	92	100
	dCNN	95.7	95.7	95.7	95.7	95.7	95.7	95.7	95.7	95.7
	CollectiveCNN	99.9	100	100	100	100	100	100	100	100
	Procedure II	X	X	100	100	100	100	100	100	100
Two Clocks	dSequence	X	X	X	X	X	X	X	X	X
	dCNN	92.5	92.5	92.5	92.5	92.5	92.5	92.5	92.5	92.5
	CollectiveCNN	99.8	100	100	100	100	100	100	100	100
	Procedure II	X	X	100	100	100	100	100	100	100
Four Clocks	dSequence	X	X	X	X	X	X	X	X	X
	dCNN	91.6	91.6	91.6	91.6	91.6	91.6	91.6	91.6	91.6
	CollectiveCNN	99.8	100	100	100	100	100	100	100	100
	Procedure II	X	X	100	100	100	100	100	100	100

Table 3. Accuracy change of recognizing x-digits by using Procedure III against the number of repeated frames on 500 clips with single clock, 500 clips with two clocks, and 500 clips with four clocks

v (Algorithm Parameter)	1	3	5	7	9	11
Single Clock	99.5	100	100	100	100	100
Two Clocks	99.4	99.9	100	100	100	100
Four Clocks	99.2	99.5	100	100	100	100

Table 4. Comparison on computational time costs of clock digit localization and clock digit recognition by using various procedures on 500 clips with single clock, 500 clips with two clocks, and 500 clips with four clocks

Task		s-Digit Localization		Other Digits Localization		s-Digit Recognition						Other Digits Recognition		Total Time	
Method		**CSP**		**Procedure I**		**Procedure II**		**CollectiveCNN**		**dCNN**		**Procedure III**			
v-Type	**#total**	μ	σ	μ	σ	μ	σ	μ	σ	μ	σ	μ	σ	μ	σ
Single clock	1st-100	0.198	0.027	0.004	0.001	0.661	0.011	4.352	0.401	4.266	0.468	1.002	0.010	1.865	0.269
	2nd-100	0.221	0.043	0.004	0.003	0.697	0.022	5.025	0.123	4.717	0.234	1.225	0.020	2.147	0.449
	3rd-100	0.219	0.039	0.003	0.001	0.698	0.034	5.042	0.504	4.698	0.249	1.169	0.030	2.089	0.474
	4th-100	0.233	0.041	0.004	0.002	0.701	0.036	5.045	0.322	4.702	0.253	1.228	0.032	2.166	0.458
	5th-100	0.212	0.038	0.004	0.002	0.695	0.028	5.039	0.421	4.689	0.241	1.095	0.025	2.006	0.443
Two clocks	1st-100	0.237	0.024	0.004	0.001	1.419	0.085	12.008	1.123	11.896	0.988	1.556	0.076	3.216	0.666
	2nd-100	0.242	0.026	0.003	0.001	1.422	0.087	11.557	1.491	11.326	1.023	1.563	0.077	3.227	0.722
	3rd-100	0.227	0.029	0.003	0.001	1.655	0.078	11.323	1.616	11.036	1.265	1.677	0.069	3.562	0.858
	4th-100	0.235	0.025	0.003	0.001	1.511	0.082	11.962	1.115	11.656	1.323	1.670	0.073	3.419	0.865
	5th-100	0.231	0.025	0.004	0.001	1.476	0.085	12.001	1.365	11.921	1.236	1.670	0.076	3.381	0.788
Four clocks	1st-100	0.223	0.034	0.004	0.002	2.903	0.115	19.125	1.536	14.597	1.362	1.556	0.083	4.686	0.853
	2nd-100	0.225	0.029	0.005	0.001	2.789	0.099	21.223	1.798	15.633	1.712	1.563	0.067	4.582	0.925
	3rd-100	0.214	0.032	0.005	0.001	2.762	0.083	18.365	1.885	15.051	1.613	1.677	0.092	4.658	0.736
	4th-100	0.244	0.031	0.004	0.002	2.831	0.095	19.995	1.815	15.412	1.214	1.670	0.081	4.749	1.062
	5th-100	0.225	0.027	0.005	0.001	2.981	0.102	22.107	2.062	15.887	1.512	1.670	0.096	4.881	1.125

CONCLUSION

This paper has presented an algorithm that can read not only single digital clock but also multiple digital video clocks at a low computational time cost requiring a short length of input clips. The algorithm is easy to implement because it converts the digit localization procedure into computing several functions, replaced a very tedious and error-prone proce- dure in the traditional algorithms. These functions properly consist of an implementation of the context-aware pixel periodicity method, which captures the facts that some second- pixels change their grey value secondly, some other second-pixels keep constancy for several seconds, and they are neighbors. Another important contribution is that a 3-digit-sequence CNN procedure is the proposed, which achieves a 100% of accuracy in recognizing s-digit and finding transit frames of s-digits. Experimental results show that our new algorithm has a much better performance in terms of accuracy and the length of input clip compared with the existing methods for reading single digital clock (Bu et al., 2008; Covavisaruch & Saengpanit, 2004; Li, Wan, et al., 2006; Li, Xu, et al., 2006; Yu, 2012; Yu & Ding, 2015; Yu et al., 2008, 2009, 2016, 2015) and that it achieved an excellent performance in reading multiple digital video clocks.

This article has three technique contributions. The first one that it proposes a context-aware pixel periodicity method for localizing second places of single and multiple digital video clocks, which evolved from the pixel periodicity method presented in (Yu, 2012) and (Yu et al., 2015). This method can directly and simultaneously identify second places of multiple clocks at a fast way, has solved the difficulty of detecting transit frames of multiple s-digit places (Li, Wan, et al., 2006; Li, Xu, et al., 2006; Yu, 2012; Yu et al., 2015). A second technical contribution is a new s-digit recognition procedure that is based on domain knowledge and deep learning technique. This procedure can accurately recognize s-digits and find s-digit transit frames. The third one is that it proposes a new procedure, called Procedure II, which is based on the domain knowledge and deep learning technique. This procedure can accurately recognize all x-digits at a low computational time cost.

For the concise the presented algorithm has three assumptions that can be removed with minor changes. The first assumption is that it works on the recorded video. To remove this assumption a step can be added to collect the frames of the real-time video for the required seconds. The second assumption is that the clock is a normal clock not a count-down one as in the basketball. This can be removed by reading the video reversely. The third assumption is that a clock has four digits. This assumption can be removed by adding a step to detect how many digits are there after clock digit extraction.

The five thresholds, denoted as β_1 to β_5 are used during localizing s-digit places. Luckily it is easy to give their values because these thresholds have strong physics meaning. The first threshold β_1 relates with the stability of gray values of background pixels within s-digit place; the second threshold β_2 relates with the intra-second stability of gray values of both foreground and background pixels within s-digit place; the third threshold β_3 relates with the contrast between the average of grey values of a foreground s-digit pixel and the average of grey values of a background s-digit pixel; the fourth threshold β_4 is the distance between two s-digit pixels; the fifth threshold β_5 is the minimum number of s-digit candidates to confirm it is a s-digit place. The notations, definitions, and their values of these five thresholds in the proposed algorithm are given in Table 5.

The authors are considering the application of the new method to solve other related problems, such as digital timestamp removal. The authors will also apply algorithms to video semantics analysis for video event detection, video indexing, and video retrieval. The authors plan to apply the algorithm into the people-activity event detection for city surveillance systems because clock time recognition helps find the time chain of person-activity from surveillance videos of city surveillance systems. In addition, the authors are developing a library of reading multiple digital video clocks for public use.

Table 5. Notations, definitions, and their values of the five thresholds used in the proposed algorithm in this paper

Notation	Definition	Value	Where
	It is a threshold on the differences between grey values of background s-digit pixels and their average within seconds. It is based on the fact that s-digit pixel is relatively constant in color if it is background pixel for a period of time.	13	Def 1
	It is a threshold on the differences between grey-values of s-digit pixels and their average within a second. It is based on the fact that s-digit pixel is intra-second stability in color.	15	Def 2
	It is a threshold on the contrast between the average of grey values of a foreground s-digit pixel and the average of grey values of a background s-digit pixel.	27	Def 2
	It is a threshold on the distances between two s-digit pixel candidates.	3	Def 4
β_5	It is the lower bound of the cardinality of s-digit pixel candidates cluster that a cluster can be confirmed to be an s-digit place.	5	Def 5

ACKNOWLEDGMENT

This study is funded by National Natural Science Foundation of China (No. 61877026), Fundamental Research Funds for the Central Universities (No. CCNU17QN005), and China Postdoctoral Science Foundation (No. 2019M652678).

REFERENCES

Anthimopoulos, M., Gatos, B., & Pratikakis, I. (2013). Detection of artificial and scene text in images and video frames. *Pattern Analysis & Applications*, *16*(3), 431–446. doi:10.1007/s10044-011-0237-7

Bu, F., Sun, L.-F., Ding, X.-F., Miao, Y.-J., & Yang, S.-Q. (2008). Detect and recognize clock time in sports video. In Proceedings of the Pacific-rim conference on multimedia (pp. 306–316). Academic Press. doi:10.1007/978-3-540-89796-5_32

Covavisaruch, N., & Saengpanit, C. (2004, June). *Time Stamp Detection and Recognition in Video Frames* (pp. 173–178). CISST.

Epshtein, B., Ofek, E., & Wexler, Y. (2010). *Detecting text in natural scenes with stroke width transform. In Proceedings of the 2010 IEEE conference on computer vision and pattern recognition (cvpr)* (pp. 2963–2970). IEEE. doi:10.1109/CVPR.2010.5540041

Ghanei, S., & Faez, K. (2015). Robust localization of texts in real-world images. *International Journal of Pattern Recognition and Artificial Intelligence*, *29*(7), 1555012. doi:10.1142/S0218001415550125

Ghanei, S., & Faez, K. (2016). A robust approach for scene text localization using rule-based confidence map and grouping. *International Journal of Pattern Recognition and Artificial Intelligence*.

Jaderberg, M., Simonyan, K., Vedaldi, A., & Zisserman, A. (2016). Reading text in the wild with convolutional neural networks. *International Journal of Computer Vision*, *116*(1), 1–20. doi:10.1007/s11263-015-0823-z

Lee, J. J., Lee, P. H., Lee, S. W., Yuille, A., & Koch, C. (2011, September). Adaboost for text detection in natural scene. In Proceedings of the 2011 International Conference on Document Analysis and Recognition (pp. 429-434). IEEE. doi:10.1109/ICDAR.2011.93

Li, Y., Wan, K., Yan, X., Yu, X., & Xu, C. (2006). *Video clock time recognition based on temporal periodic pattern change of the digit characters. In Proceedings of the 2006 IEEE international conference on acoustics speech and signal processing proceedings* (Vol. 2, pp. II–II). IEEE.

Li, Y., Xu, C., Wan, K. W., Yan, X., & Yu, X. (2006). Reliable video clock time recognition. In *Proceedings of the 18th international conference on pattern recognition (icpr'06)* (Vol. 4, pp. 128–131). Academic Press.

Lyu, M. R., Song, J., & Cai, M. (2005). A comprehensive method for multilingual video text detection, localization, and extraction. *IEEE Transactions on Circuits and Systems for Video Technology*, *15*(2), 243–255. doi:10.1109/TCSVT.2004.841653

Mishra, A., Alahari, K., & Jawahar, C. (2012). *Top-down and bottom-up cues for scene text recognition. In Proceedings of the 2012 IEEE conference on computer vision and pattern recognition (CVPR)* (pp. 2687–2694). IEEE. doi:10.1109/CVPR.2012.6247990

Neumann, L., & Matas, J. (2012). *Real-time scene text localization and recognition. In Proceedings of the 2012 IEEE conference on computer vision and pattern recognition (CVPR)* (pp. 3538–3545). IEEE.

Neumann, L., & Matas, J. (2013). Scene text localization and recognition with oriented stroke detection. In *Proceedings of the IEEE international conference on computer vision* (pp. 97– 104). IEEE. doi:10.1109/ICCV.2013.19

Neumann, L., & Matas, J. (2015). Efficient scene text localization and recognition with local character refinement. In *Proceedings of the 2015 13th international conference on document analysis and recognition (ICDAR)* (pp. 746–750). Academic Press. doi:10.1109/ICDAR.2015.7333861

Pan, Y.-F., Hou, X., & Liu, C.-L. (2011). A hybrid approach to detect and localize texts in natural scene images. *IEEE Transactions on Image Processing*, *20*(3), 800–813. doi:10.1109/TIP.2010.2070803 PMID:20813645

Sermanet, P., Kavukcuoglu, K., & LeCun, Y. (2009). Eblearn: Open-source energy-based learning in c++. In *Proceedings of the 2009 21st IEEE international conference on tools with artificial intelligence* (pp. 693–697). IEEE.

Shi, C., Wang, C., Xiao, B., Gao, S., & Hu, J. (2014). End-to-end scene text recognition using tree-structured models. *Pattern Recognition*, *47*(9), 2853–2866. doi:10.1016/j.patcog.2014.03.023

Shi, C., Wang, C., Xiao, B., Zhang, Y., & Gao, S. (2013). Scene text detection using graph model built upon maximally stable extremal regions. *Pattern Recognition Letters*, *34*(2), 107–116. doi:10.1016/j.patrec.2012.09.019

Shivakumara, P., Phan, T. Q., & Tan, C. L. (2011). A laplacian approach to multi-oriented text detection in video. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, *33*(2), 412–419. doi:10.1109/TPAMI.2010.166 PMID:20733217

Wang, K., Babenko, B., & Belongie, S. (2011). End-to-end scene text recognition. In Proceedings of the 2011 international conference on computer vision (pp. 1457–1464). Academic Press. doi:10.1109/ICCV.2011.6126402

Wang, T., Wu, D. J., Coates, A., & Ng, A. Y. (2012). End-to-end text recognition with convolutional neural networks. In *Proceedings of the 2012 21st international conference on pattern recognition (ICPR)* (pp. 3304–3308). Academic Press.

Weinman, J. J., Learned-Miller, E., & Hanson, A. R. (2009). Scene text recognition using similarity and a lexicon with sparse belief propagation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, *31*(10), 1733–1746. doi:10.1109/TPAMI.2009.38 PMID:19696446

Xu, C., Wang, J., Wan, K., Li, Y., & Duan, L. (2006). Live sports event detection based on broadcast video and web-casting text. In *Proceedings of the 14th ACM international conference on multimedia* (pp. 221–230). ACM. doi:10.1145/1180639.1180699

Yi, C., & Tian, Y. (2011). Text string detection from natural scenes by structure-based partition and grouping. *IEEE Transactions on Image Processing*, *20*(9), 2594–2605. doi:10.1109/TIP.2011.2126586 PMID:21411405

Yi, C., & Tian, Y. (2012). Localizing text in scene images by boundary clustering, stroke segmentation, and string fragment classification. *IEEE Transactions on Image Processing*, *21*(9), 4256–4268. doi:10.1109/TIP.2012.2199327 PMID:22614647

Yin, P., Hua, X.-S., & Zhang, H.-J. (2002). Automatic time stamp extraction system for home videos. In Proceedings of the IEEE international symposium on circuits and systems ISCAS 2002 (Vol. 2, pp. II–73). IEEE.

Yu, X. (2012). Localization and extraction of the four clock-digits using the knowledge of the digital video clock. In *Proceedings of the 2012 21st international conference on pattern recognition (ICPR)* (pp. 1217–1220). Academic Press.

Yu, X., Cheng, J., Wu, S., & Song, W. (2016). A framework of timestamp replantation for panorama video surveillance. *Multimedia Tools and Applications*, *75*(17), 10357–10381. doi:10.1007/s11042-015-3051-1

Yu, X., & Ding, W. (2015). Game suspension boundary detection by reading two clocks for broadcast basketball video. In *Proceedings of the 7th international conference on internet multimedia computing and service* (p. 45). Academic Press. doi:10.1145/2808492.2808537

Yu, X., Ding, W., Zeng, Z., & Leong, H. W. (2015). Reading digital video clocks. *International Journal of Pattern Recognition and Artificial Intelligence*, *29*(4). doi:10.1142/S021800141555006X

Yu, X., Li, L., & Leong, H. W. (2009). Interactive broadcast services for live soccer video based on instant semantics acquisition. *Journal of Visual Communication and Image Representation*, *20*(2), 117–130. doi:10.1016/j.jvcir.2008.12.004

Yu, X., Li, Y., & San Lee, W. (2008). Robust time recognition of video clock based on digit transition detection and digit-sequence recognition. In *Proceedings of the 19th international conference on pattern recognition ICPR 2008* (pp. 1–4). Academic Press. doi:10.1109/ICPR.2008.4761379

Yu, X., Lyu, X., Xiang, L., & Leong, H. W. (2017). Reading Two Digital Video Clocks for Broadcast Basketball Videos. In *Proceedings of the Pacific Rim Conference on Multimedia* (pp. 457-466). Academic Press.

Zhong, Z., Jin, L., Zhang, S., & Feng, Z. (2016). Deeptext: A unified framework for text proposal generation and text detection in natural images.

Zhu, S., & Zanibbi, R. (2016). A text detection system for natural scenes with convolutional feature learning and cascaded classification. In *Proceedings of the IEEE conference on computer vision and pattern recognition* (pp. 625–632). IEEE. doi:10.1109/CVPR.2016.74

ENDNOTES

[1] https://github.com/xiaopanlyu/clock_digit_recognition

Xinguo Yu is with the National Engineering Research Center for E-Learning (NERCEL), Central China Normal University, Wuhan, China. He received his B.Sc. degree in Mathematics from Wuhan University of Technology, his M.Eng degree from Huazhong University of Science and Technology, another M.Eng. degree from Nanyang Technological University, Singapore and his Ph.D. in Computer Science from the National University of Singapore. He is a member of IEEE and ACM. His research focuses mainly on educational intelligent technology, multimedia analysis, computer vision, machine learning, artificial intelligence, and e-learning.

Wu Song is a Ph.D. student at the National Engineering Research Center for E-Learning (NERCEL), Central China Normal University. He received his B.S. degree in Technology of Computer Application from National University of Defense Technology, Changsha, China in 1998 and his M.S. degree in Technology of Computer Application from Yunnan University, Kunming, China in 2004. His research interests include video processing, data mining and machine learning.

Xiaopan Lyu is a Ph.D. student at the National Engineering Research Center for E-Learning, Central China Normal University. He received his B.S. degree in Network Engineering from Hubei Polytechnic University and his M.S. degree in Technology of Computer Application from Central China Normal University. His research interests include educational intelligent technology, deep learning, video understanding, and e-learning.

Bin He obtained his B.S. Degree at South Central University for Nationalities in China in 2004, and completed his M.S. and Ph.D. with Prof. Kaiping Wei and Prof. Zongkai Yang at Central China Normal University in China in July 2008 and 2013 respectively. He joined Prof. Xinguo Yu's group at Central China Normal University as an Assistant Research Fellow in August 2016. His current research is focused on the development of machine solving and educational robot techniques for intelligent tutoring application.

Nan Ye received double first class honours degree in Computer Science and Applied Mathematics from National University in 2008, his PhD from National University of Singapore in 2013. He was a Vice Chancellor Researcher Fellow at Queensland University of Technology from 2015 to 2018. He is currently a lecturer in University of Queensland. His research interests include planning under uncertainty, probabilistic graphical models, numerical optimisation, active learning, and learning theory.