Photo time-stamp detection and recognition

Xiangrong Chen
Dept. of Statistics, UCLA, USA
xrchen@stat.ucla.edu
Hong-Jiang Zhang
Microsoft Research Asia, Beijing China
hjzhang@microsoft.com

Abstract

This paper addresses the problem of photo time-stamp detection and recognition, which is to determine the time a photo was taken. Since the time-stamp is exposed together with the scene, the main challenge comes from complex background and saturation of the time-stamp. An MAP (Maximum a posterior probability) based template matching method is proposed to detect and recognize the time-stamp simultaneously from a photo. A key step of this method is skeleton-matching algorithm for individual digit, which supports partial matching and deals with complex background and saturation.

Practically, to avoid exhaustive search, a saliency map is obtained by applying a set of morphologic operators on a photo. Proposals can be generated from the saliency map and the proposed method calculates the probability for each proposal. An application with strong prior knowledge is also given in which the proposed method achieves high accuracy in almost real time.

1. Introduction

Time label of a photo, i.e., when the photo is taken, plays a key role in photo access and retrieval. Although photos from digital camera have time label with the file, tons of traditional photos digitalized by scanner remain unlabelled. Some of the unlabelled photos, however, have chance to be automatically assigned a time label by time-stamps located usually on the corner of the photo.

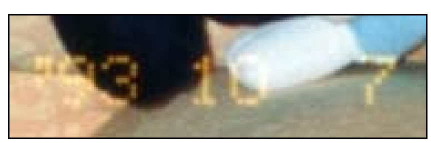
Generally, the time-stamp is exposed together with the scene, which indicates an addition imaging process. The imaging process brings two main difficulties, as listed below, to the traditional problem of text detection and recognition.

- Complex background. The photo can be taken in all kinds of scenes. Because of the addition imaging process, the color and texture of the scene affect the appearance of the time-stamp, as shown in Figure 1(a).
- Saturation. When the background is too bright, the time-stamp will be saturated and hard to be recognized, as shown in Figure 1 (b).

Related studies include Video OCR [1-4], traffic signs detection, such as vehicle license plates [5, 6], street signs [7] etc., OCR from WWW images [8], text from natural scenes [9,10], etc. Although all of them tried to detect and recognize text, different tasks have different concerns, thus adopt different methodologies. For example, Video OCR deals with complex background and small fonts using multi-frame integration and sub-pixel interpolation. The former process can't be applied to a single image (or photo) and the latter is not necessary for time-stamp recognition. Unlike photo time-stamp, vehicle license plates and street signs have frames around them, providing additional evidence for their presence. Besides, none of the above methods addresses the problem of saturation, which is one of the main concerns in this study.

In this paper, an MAP (Maximum a posterior probability) based template matching method is proposed to deal with the two difficulties. As a key step of this method, a skeleton-matching algorithm for individual digit is developed. The algorithm employs intensity comparison between each stroke skeleton of the font and its local background. It supports partial template matching and results likelihood between image patch and the candidate digit.

Section 2 briefly introduces different types of the time-stamp. Proposed method is developed in section 3. Section 4 shows the experiment results in both a general purposed scenario and a specific application with strong prior knowledge. Finally a short conclusion presents followed by a glance of future work.



(a) Complex background

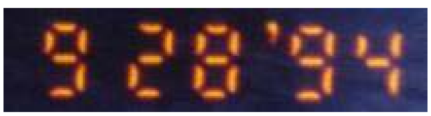
Proceedings of the Seventh International Conference on Document Analysis and Recognition (ICDAR 2003)
0-7695-1960-1/03 $17.00 © 2003 **IEEE**

(b) Saturation

Figure 1. Challenges

2. Different type of time-stamps

There are several different styles of time-stamp. They differ from the font, size, color and date type. In details, the font could be composed of short-line or dot. The digit size could be slightly different according to different camera. The color could be red, orange, yellow or green. The date type could be Year-Month-Day, Month-Day-Year or Day-Month-Year, and the distances between digits are also different. Figure 2 shows two types of time-stamp with different font, date type and distances between digits.



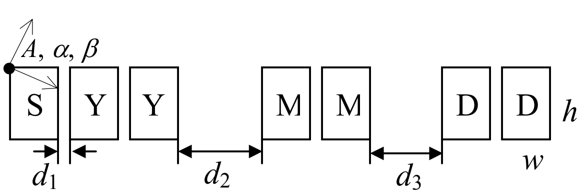
(a) Short line font with Month-Day-Year type

(b) Dot matrix font with Year-Month-Day type

Figure 2. Different type of time-stamps

3. Proposed method

Proposed method designs two levels of templates to represent the time-stamp. The upper level is format template and the lower level is font template. An MAP based template matching method uses the two templates to detect and recognize the time-stamp simultaneously.



S: year sign, Y: year digit, M: month digit, D: day digit, d_1, d_2, d_3: distances between two digits, w, h: width and height of the font, A: coordinates of the upper-left corner, α: scale factor, β: rotation angle

Figure 3. A format template for time-stamp

3.1. Format template

A format template of time-stamp consists of the font size, order of year, month and day, and distances between every two adjacent digits. Figure 3 shows a format template with Year-Month-Day type.

Given a photo I, the posterior probability of a format template X fitting to I is

$$P(X|I;\theta) = \frac{p(I|X;\theta)P(X;\theta)}{p(I)}$$

in which θ is a set of parameters of the matching format template. For the template shown in Figure 3, $\theta = \{d_1, d_2, d_3, w, h\}$. In the rest of the paper, θ is omitted for clarity. Note coordinates A, scale factor α and rotation angle β are random variables which construct the spatial searching space. Assume I follows uniform distribution. The optimal fit X' to I is

$$X' = \arg\max_{X\in\Omega}(p(I|X)P(X)) \qquad (1)$$

where $P(X)$ is the prior probability of the mathcing template. It contains the prior knowledge of the spatial searching space $\{A\}\times\{\alpha\}\times\{\beta\}$, date type and the date represented by $\{T\}=\{\text{date type}\}\times\{\text{YYMMDD}\}$.

$$p(X) = p(A)p(\alpha)p(\beta)p(T) \qquad (2)$$

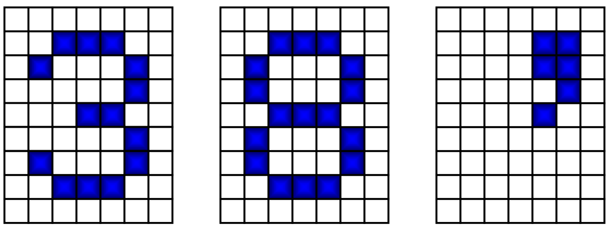
In equation (1), $p(I|X)$ is the likelihood of the photo I given the template X. Once X is given, $p(I|X)$ is only involved in a small part of the photo, i.e. where the template locates. In our algorithm,

$$p(I \mid X) = \prod_k p(I_k \mid D_k) \qquad (3)$$

where I_k is the small image patch in photo I corresponding to the kth individual digits in the template. D_k is the individual digit template, which will be constructed in section 3.2. Here α and β are assumed to follow Gaussian distribution.

3.2. Font template and skeleton matching

To calculate the likelihood $p(I_k|D_k)$ in equation (3), a skeleton-matching algorithm is developed for the lower level font template. As stated in the introduction, the imaging process often affects the presence of the time-stamp. When the scene is dark, the time-stamp will dilate. When the scene is bright, the time-stamp will shrink. Thus in the font template, only center point of each stroke is kept. Figure 4 shows some examples of the skeleton templates for individual digits of the dot matrix font. A "year sign", which indicates the year in the time-stamp, is also included in the templates, as shown in Figure 4 (c). The center of each blue square is the stroke skeleton point of the digit, while the center of each white square is the background skeleton point. The value of each skeleton point is set to be the value of center pixel in the corresponding square.



(a) '3' (b) '8' (c) year sign

Figure 4. Template examples of dot matrix font

Using the above skeleton template, the skeleton matching algorithm can calculate the likelihood $p(I_k|D_k)$. Suppose a digit template contains N stroke skeleton points whose pixel values are represented by s_i, $i = 1,\ldots,N$. Each stroke skeleton point has a background set B_i consists of pixel values of its 4-neighborhood background skeleton points. Let

$$v_i = \frac{|\{b_m : b_m \in B_i, b_m < s_i\}|}{|B_i|},$$

which can be viewed as a probability of a candidate point to be a stroke skeleton point. Then

$$p(I_k \mid D_k) = \sum_{i=1}^{N} \alpha_i * v_i \ ,$$

where

$$\alpha_i \geq 0, \qquad \sum_{i=1}^{N} \alpha_i = 1$$

are weights for each stroke skeleton point. They measure the contribution of each stroke skeleton point for identifying the digit. An off-line learning process is performed to obtain α_i from all the font templates.

4. Experiment

The proposed method is fit into two experimental scenarios – a general case and a specific application. Table 1 gives the results for both cases on a 180 photos data set. Each photo has dimension 1000×690.

Table 1. Experiment results

	General	Specific
Average time(s)	10.0	0.3
Fail to detect	0	3
False recognition	2	5

4.1. General case

Although it is possible to use all the templates to do exhaustive search, that is to assume all the prior probabilities in equation (2) are assumed to be uniform distribution, it is time-consuming and impractical. To reduce the searching space, some bottom-up techniques are employed. In details, a saliency map M is obtained by applying a set of well-known morphologic operators on the photo.

$$M = I - close\{open(I)\}$$

Proposals can be generated from the saliency map and the proposed method calculates the probability for each proposal. Figure 5 shows a saliency map of the photo patch Figure 1(a). The middle column in Table 1 shows its performance.

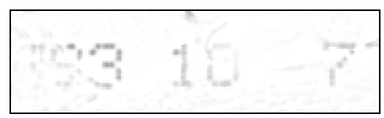


Figure 5. Saliency map of Figure 1 (a)

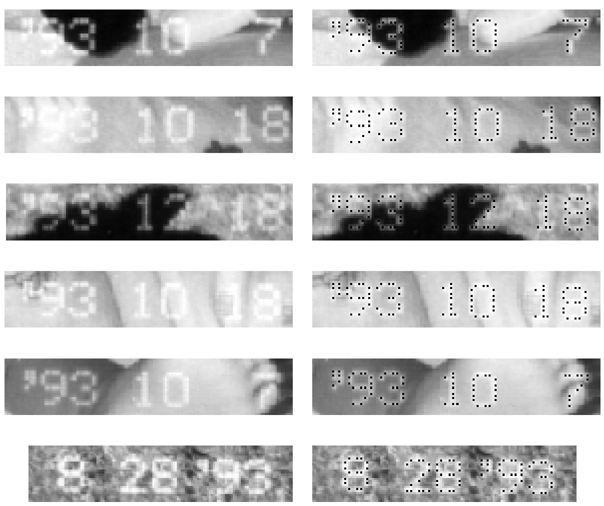
4.2. A specific application

One of the applications of the proposed method is called "Smart photo scanner". When a user put a photo in the scanner, this application can automatically determine the photo location, size, orientation according to the photo edge and the time it was taken according to the time-stamp. In this application, some prior knowledge can be used to further improve the efficiency of the method.

- For a full-size photo (the photo is not discarded), the size of the time-stamp relative to the photo can be estimated. It indicates that the scale factor α is either fixed or follows Gaussian distribution with very small variance.
- For a full-size photo, which is upright according to the photo edge, the time-stamp locates on the corners and aligns along the longer dimension of the photo. This prior put strong restrictions to the location A and rotation angle β.

- Time-stamps from photos with successive time when they were input the system are assumed to have similar, if not the same, location, font and format, and to represent successive date. Thus the probability of the date close to the detected date increases.

After applying the above priors, the searching space is greatly reduced. It takes only 0.3 second to detect and recognize the time stamp from a 1000×690 photo. In Table 1, the three fail-to-detect photos are due to the discard of the original photos. Figure 6 shows some results of the proposed method.



Left column: image patches cropped from photos, right column: detected time-stamps (black dots) overlapped on the image patches

Figure 6. Some results

5. Conclusion and future work

This paper proposes an MAP (Maximum a posterior probability) based template matching method to detect and recognize the time-stamp from a photo. The method can overcome the difficulties of complex background and saturation caused by the imaging process. Two levels of templates are designed for the method to achieve the detection and recognition simultaneously. A skeleton-matching algorithm for individual digit is developed on the font template. Experimental results show that in a specific application, the method can be implemented in real time with very high accuracy.

More work needs to be done on general cases. For example, more meaningful saliency map is required to reduce the proposals and speed up the searching process. A more general font template which characterizes not the fixed points, but topological structure of the strokes, is also desired to avoid large number of font templates.

6. References

[1] T. Sato, T. Kanade, “Video OCR: Indexing digital news libraries by recognition of superimposed caption”, ICCV Workshop on Image and Video retrieval, 1998.

[2] H. Li, D. Doermann, and O. Kia, “Automatic text detection and tracking in digital video”, *IEEE Trans. on Image Processing*, Vol. 9, No. 1, 2000, pp. 147-156.

[3] Y. Zhong, H. Zhang, and A.K. Jain, “Automatic Caption Extraction of Digital Videos”, *Proc. ICIP’99*, Kobe, 1999, pp. 24-27.

[4] X. Chen and H.J. Zhang, Text Area Detection From Video Frames, in *The Second IEEE Pacific-Rim Conference on Multimedia*, Beijing, China, Oct. 24-26, 2001, pp. 222-228.

[5] Y. Cui and Q. Huang, “Character extraction of license plates from video," *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition*, 1997, pp. 502-507.

[6] W.-S. Hwang, "Car License Plate Detection and Recognition System", *http://www.cse.msu.edu/ ~hwangwey/License_plates.html*

[7] A.L. Yuille, D. Snow, and M. Nitzberg, ``Signfinder'', *In Proceedings of the International Conference of Computer Vision.* Bombay, India, January 1998.

[8] J. Zhou, D. Lopresti, and Z, Lei, “OCR forWorld Wide Web images," *In Proceedings of SPIE Document Recognition IV*, vol. 3027, 1997, pp. 58-66.

[9] V. Wu, R. Manmatha, and E. M. Riseman, “Finding text in images," *20th International ACM SIGIR Conference on Research and Development in Information Retrieval*, Philadelphia, 1997, pp. 3-12.

[10] G. Jiang and J. Yang, “An Adaptive Algorithm for Text Detection from Natural Scenes”, in *IEEE Computer Society Conference on Computer Vision and Pattern Recognition* (*CVPR’01*), Hawaii, 2001.

Proceedings of the Seventh International Conference on Document Analysis and Recognition (ICDAR 2003)
0-7695-1960-1/03 $17.00 © 2003 **IEEE**