Mosaicing-by-recognition: a technique for video-based text recognition

Hiromitsu Miyazaki, Seiichi Uchida, and Hiroaki Sakoe
Graduate School of Information Science and Electrical Engineering,
Kyushu University, 6-10-1 Hakozaki, Higashi-ku, Fukuoka-shi, 812-8581 Japan

Abstract

In this paper, a mosaicing-by-recognition technique is proposed where video mosaicing and text recognition are simultaneously and collaboratively optimized in a one-step manner. Specifically, multiple frames capturing a long text line are optimally concatenated with a guide of the text recognition framework. In this optimization process, rotation, scaling, vertical shift, and speed fluctuation, which often appear in video frames captured by hand-held cameras, are compensated. The optimization is performed by a DP-based algorithm. The results of experiments to evaluate not only the accuracy of text recognition but also that of video mosaicing indicates that the proposed technique is practical and can provide reasonable results in most cases.

1. Introduction

Text recognition in video frames (Fig. 1) has been investigated [1] as an alternative to text recognition in a still camera image, because of the following merits:

- By mosaicing consecutive frames, i.e., by matching and concatenating the frames, it is possible to recognize longer texts.
- By integrating the frames, it is also possible to improve the quality of the text image (e.g., super-resolution, noise removal).

Previous attempts to recognize texts in video sequences assumes a two-step manner that consecutive frames are firstly concatenated to create a large (and often elongated) image using some video mosaicing technique (e.g., [2]) and then the text in the mosaic image is recognized by using a usual OCR technique. Thus, the failure at the first step (probably the most difficult step) will be fatal.

In this paper, a *mosaicing-by-recognition* technique is proposed where video mosaicing and text recognition are simultaneously and collaboratively performed in a one-step manner. Specifically, multiple frames capturing a long text line are optimally matched and concatenated with a guide of the text recognition framework. The optimization is performed by a DP-based algorithm and can compensates rotation, scaling, vertical shift, and speed fluctuation, which often appear in texts captured by hand-held cameras.

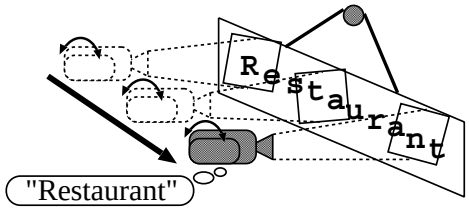

Figure 1. Text recognition in video frames captured by a hand-held camera.

In the followings, we will firstly discuss a *simple case* that video frames undergo only speed fluctuation and then discuss a *general case* that video frames undergo not only speed fluctuation but also rotation, scaling, and vertical shift. The mosaicing-by-recognition problem on the simple case is easy to understand because it is reduced to a well-known *segmentation-by-recognition* problem of continuous speeches [3] and texts [4]. The mosaicing-by-recognition problem on the general case is derived as an extension of the simple case.

2 Mosaicing-by-recognition

2.1 Outline

Consider video frames recorded by moving a hand-held camera from left to right and a long text line is captured in those frames. Also assume that the motion of the camera is slow and each character of the text is included in multiple frames. Major distortions appeared those frames are speed fluctuation, rotation, scaling and vertical shift. The left side of Fig. 3 shows those distortions.

For simplification, a *slit* (shown in Fig. 2(a)) is utilized here, which is a central part of the frame and has 1 pixel width and H pixel height. Since its width is 1 pixel, a slit generally does not contain two or more characters. The right side of Fig. 3 shows images created by concatenating those slits.

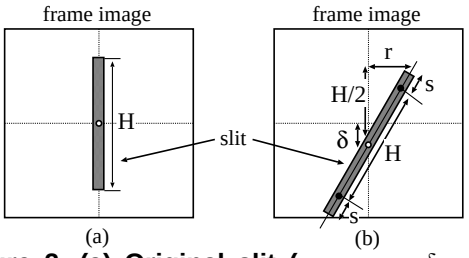

Figure 2. (a) Original slit ($r = s = \delta = 0$). (b) Controlled slit.

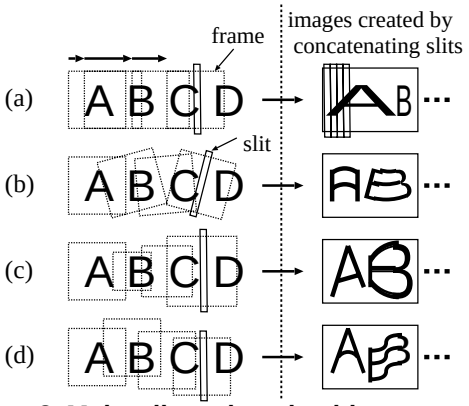

Figure 3. Major distortions in video sequence obtained by a hand-held camera. (a) Speed fluctuation. (b) Rotation. (c) Scaling. (d) Vertical shift.

2.2 DP algorithm for simple case

In this section, a mosaicing-by-recognition algorithm for the simple case is provided, where only the fluctuation of scanning speed is considered. Other distortions will be considered in the next section.

On the simple case, the problem is reduced to the well-known segmentation-by-recognition problem of a character sequence. The text contained in the frames can be treated as a deformed character sequence in the image created by concatenating the slits of all T frames (shown in the right side of Fig. 3). Thus, the text in the image can be recognized and partitioned by solving the optimal path problem on the search space indexed by t and (c, j), where c ($c = 1, \ldots, C$) is the character category and j ($j = 1, \ldots, J_c$) is the index for the row of the reference pattern image of the category c (Fig. 4). It is also well-known that this problem can be solved effectively by a DP algorithm.

Figure 5 shows the DP algorithm on the simple case, where $d_t(c, j)$ is the matching cost between the slit of the tth frame and the jth column of reference pattern of category c. The value $g_t(c, j)$ is the minimum cost accumulated along with the optimal path to the point indexed by t, c and j in the DP-search space. The speed fluctuation is compensated by controlling j' in the DP recursion of Step 11.

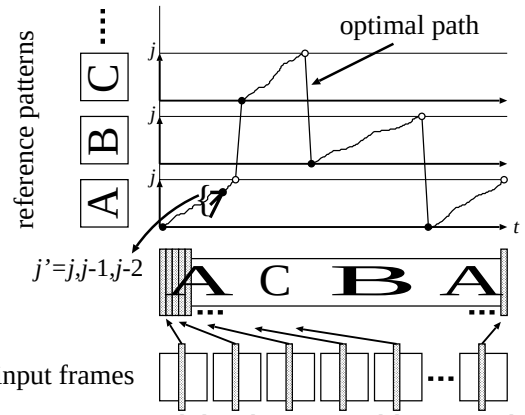

Figure 4. Mosaicing-by-recognition on the simple case that video frames undergo only speed fluctuation.

/* Initialization */
1 **for** $c := 1$ **to** C **do begin**
2 $g_1(c, 1) := d_1(c, 1)$
3 **for** $j := 2$ **to** J_c **do**
4 $g_1(c, j) := \infty$
5 **end**
6 $D_1 := \infty$
/* DP Recursion */
7 **for** $t := 2$ **to** T **do begin**
8 **for** $c := 1$ **to** C **do begin**
9 $g_t(c, 1) := d_t(c, 1) + \min\{g_{t-1}(c, 1), D_{t-1}\}$
10 **for** $j := 2$ **to** J_c **do**
11 $g_t(c, j) := d_t(c, j) + \min_{j' \in \{j, j-1, j-2\}} g_{t-1}(c, j')$
12 **end**
13 $D_t := \min_{c' \in C} g_t(c', J_{c'})$
14 **end**

Figure 5. The DP algorithm for mosaicing-by-recognition on the simple case. Several steps for backtracking operation is omitted.

Specifically, as shown in Fig. 6, $j' = j - 2$ is selected when the scanning speed is 2pixel/frame and $j' = j$ is selected when it is 0pixel/frame. The computational complexity of the algorithm is $O(TCJ)$, where J is the average of J_c $(c = 1, \ldots, C)$.

The result of character recognition is obtained by backtracking the optimal (c, j)-sequences (illustrated as an optimal path in Fig. 4) after performing the DP algorithm. An optimal mosaic image is also obtained by backtracking as shown in the Section 2.4. Thus, the mosaicing of video frames is optimized simultaneously with the text recognition, and therefore we call the above procedure *mosaicing-by-recognition*.

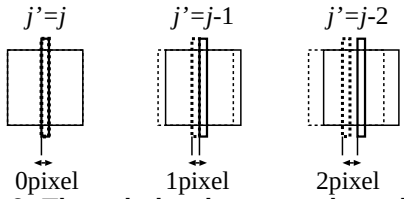

Figure 6. The relation between the selection of j' and scanning speed.

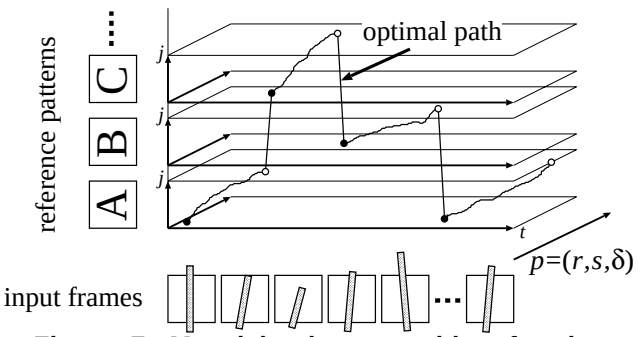

Figure 7. Mosaicing-by-recognition for the general case that video frames undergo not only speed fluctuation but also rotation, scaling, and vertical shift.

2.3 DP algorithm for general case

We can derive a DP algorithm for the general case, where not only the speed fluctuation but also the other distortions are considered, by extending the algorithm for the simple case. The main idea of the extension is to control (i.e., rotate, scale, and vertical shift) the slit according to the distortions. The optimal control, however, is not known in advance. Thus, the optimal control parameters are searched for in the DP framework. Specifically, the problem becomes an optimal path problem in the search space indexed by t and (p, c, j), where $p = (r, s, \delta)$ is the parameter set and r, s, and δ are the parameters of rotation, scaling, and vertical shift, respectively (Fig. 7). The definition of those control parameters are shown in Fig. 2(b).

Figure 8 shows DP algorithm on the general case. In the DP recursion of Step 14, the smoothness of the distortion is assumed by constraining the parameters of consecutive frames (p and p') by

$$\begin{aligned}\mathrm{pre}(p) = \{(r', s', \delta') \mid\ & r-1 \le r' \le r+1, \\ & s-1 \le s' \le s+1,\ \delta-1 \le \delta' \le \delta+1\}\end{aligned}$$

The computational complexity of the algorithm is $O(TCJRS\Delta)$, where R, S and Δ are the ranges of r, s, and δ, respectively. Similar to the simple case, the result of character recognition is obtained by backtracking the optimal path after performing the DP algorithm.

/* Initialization */
1 **for all** $p \in \{(r, s, \delta)\}$ **do begin**
2 **for** $c := 1$ **to** C **do begin**
3 $g_1(p, c, 1) := d_1(p, c, 1)$
4 **for** $j := 2$ **to** J_c **do**
5 $g_1(p, c, j) := \infty$
6 **end**
7 $D_1(p) := \infty$
8 **end**
/* DP Recursion */
9 **for** $t := 2$ **to** T **do begin**
10 **for all** $p \in \{(r, s, \delta)\}$ **do begin**
11 **for** $c := 1$ **to** C **do begin**
12 $g_t(p, c, 1) := d_t(p, c, 1) + \min_{p' \in \mathrm{pre}(p)} \{g_{t-1}(p', c, 1), D_{t-1}(p')\}$
13 **for** $j := 2$ **to** J_c **do**
14 $g_t(p, c, j) := d_t(p, c, j) + \min_{\substack{p' \in \mathrm{pre}(p) \\ j' \in \{j, j-1, j-2\}}} g_{t-1}(p', c, j')$
15 **end**
16 $D_t(p) := \min_{c' \in C} g_t(p, c', J_{c'})$
17 **end**
18 **end**

Figure 8. The DP algorithm for the general case.

2.4 Mosaicing

Although conventional video mosaicing techniques require several corresponding points among consecutive frames, the proposed technique does not. On the simple case, the mosaic image can be obtained by placing the tth frame with a 0$\sim$ 2 pixel horizontal shift according to the relation between j' and j, which can be obtained by the backtracking operation for the optimal path. (See also Fig. 6.)

On the general case, the mosaic image can be obtained by a similar procedure. The only difference is a dewarping operation of the controlled slit of each frame is necessary in advance to placing it with a 0$\sim$ 2 pixel horizontal shift. The dewarping can be done by using the optimal parameter p at frame t, which can be obtained by the backtracking operation.

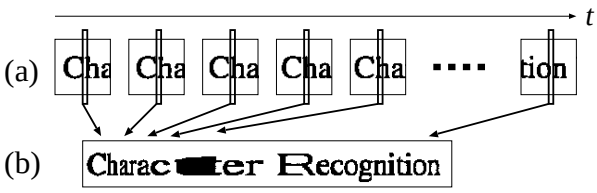

Figure 9. (a) Example of video frames and (b) image created by concatenating their slits.

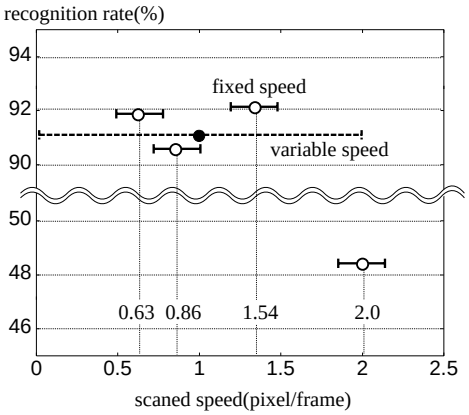

Figure 10. Recognition rate for the simple case.

3. Experimental results

3.1. Performance evaluation on simple case

3.1.1. Data preparation. We prepared 20 text lines printed on white A4-sized papers. Since each text line contains about 50 characters (of capital/small English alphabets and digits), about 1000 characters were prepared in total. Each character was printed in the same Times-Roman font. The character height ($\sim H$) in the frame was about 40 pixels.

Each text line was captured in multiple frames by moving a video camera. A special equipment with a variable speed motor was used for moving the camera horizontally. Thus, we could fix or fluctuate the speed of camera movement, while excluding rotation, scaling, and vertical shift.

Figure 9 (a) shows video frames captured under variable speed and (b) shows the image created by concatenating their slits. This image (b) indicates that scanning speed became very low around "t" in the word "Character".

3.1.2. Recognition experiment. The video frames acquired by the above procedure were subjected to the DP algorithm of Fig. 5. Figure 10 shows the character recognition rates at several scanning speed conditions. When scanning speed was (nearly) fixed at a value under 2 pixel/frame, recognition rates exceeded 90%. Even when scanning speed

Figure 11. (a) Mosaic image and (b) recognition result of the video frames of Fig. 9.

was varied between 0 and 2 pixel/frame, the recognition rate remained around 90%. Those facts show that the proposed technique can compensate nonlinear fluctuation of scanning speed successfully. Although the attained rate itself is not high compared to recent OCR's rates, it seems satisfiable one because we used neither a sophisticated pixel feature nor a word lexicon. Note that when scanning speed exceeds 2 pixel/frame, the recognition performance was degraded drastically. This will because the maximum scanning speed allowed by the restriction between j and j' (Fig. 6) was 2 pixel/frame. The use of a wider slit will be a possible remedy to relax this limitation.

Figure 11 (a) and (b) show the mosaic image and the recognition result of the frames of Fig. 9 (i.e., the frames captured with a variable speed).

Most of misrecognitions observed in the experimental result were segmentation errors such that "m" is misrecognized as "r" and "n". The misrecognitions of this type are often found in the results of segmentation-by-recognition techniques. A well-known remedy for this problem is the use of a word lexicon.

3.2. Performance evaluation on general case

3.2.1. Data preparation. Video frames of the general case were artificially synthesized by rotating, scaling, and vertically shifting each frame of the speed-fluctuated (0$\sim$ 2 pixel/frame) video sequence of 3.1.1. On the synthesis, the maximum amplitude of distortions were limited so that the distortions can be theoretically compensated by $\{(s, r, \delta) \mid |s| \leq k, |r| \leq k, |\delta| \leq k\}$, where k was fixed at 1, 2, 3, or 4 (pixels). (This means $k = R = S = \Delta$.)

3.2.2. Recognition experiment. Figure 12 shows character recognition rates under (i) a single distortion or (ii) all of the three distortions. This result shows the robustness of the proposed technique to scaling and vertical shift. In contrast, recognition rates were degraded when large rotations were added. This will be because of the error on matching rotated low resolution patterns.

Figure 13 shows a result of the general case. While most part of the mosaic image (c) is well created, the part with misrecognition shows some degradation. For example, the last character "o" is deformed to be close to "v" by abusing the flexibility on controlling slits. Thus, this misrecognition ("o"$\rightarrow$ "v") is caused by so-called *over-fitting*, which

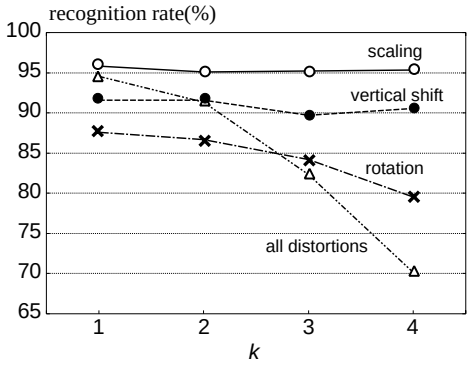

Figure 12. Recognition rate for the general case. The horizontal axis represents the amplitude of distortions, k (pixels).

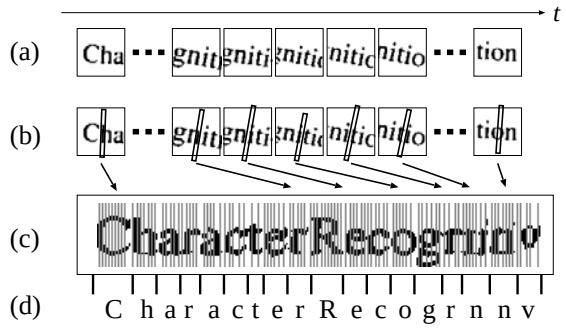

Figure 13. (a) Frames which undergo various distortions. The original text is "Character Recognition". (b) The optimally controlled slits. (c) Mosaic image. (d) Recognition result.

often degrades the performance of elastic matching-based character recognition (e.g., [5]). The use of some sophisticated pixel feature and a word lexicon will be still helpful to reduce such misrecognitions due to overfitting.

4. Conclusion and future work

A mosaicing-by-recognition technique was proposed for recognizing texts in multiple video frames and mosaicing those frames. Those two procedures, i.e., recognition and mosaicing, are simultaneously and collaboratively optimized in a one-step manner by a DP-based algorithm. Experimental results showed that the proposed technique can attain about 90% character recognition rate even when rotation, scaling, vertical shift, and speed fluctuation appear in the frames.

Future work will focus on the following points:

- *Using a wider slit*: In this paper, the width of the slit was fixed at 1 pixel for simplifying the problem. This means, however, that most of information contained in each frame is disregarded. If a wider slit is used and the area commonly contained by consecutive frames are utilized (like [6, 7, 8]), reliability of recognition and mosaicing results will be improved. In addition, scanning speeds over 2 pixel/frame will be allowed. Note that we should compensate the projective distortion within the wider slit.
- *Using lexicon*: The proposed technique often produces misrecognitions by erroneous segmentations (e.g., "m"→ "r" and "n") and overfitting (e.g., "o" → "v"). Like the other text recognizer based on segmentation-by-recognition framework, the use of lexicon will be helpful to exclude such misrecognitions.
- *Using sophisticated pixel feature*: In the experiment conducted in this paper, only naive pixel feature, i.e., intensity feature, was used. Since this feature is very weak to geometrical distortions, sophisticated pixel features should be used for improving the matching between a slit and a reference pattern.
- *Reducing computational complexity*: The proposed algorithm, especially in the general case, requires huge computations. For example, a PC (pentium IV) required 2.79 seconds/character even at $k = 1$. Beam search techniques (cost-based pruning and lexicon-based pruning) will be effective to reduce the computational complexity.

Acknowledgment: This work was supported in part by the Research Grant of The Okawa Foundation and MEXT in Japan (Grant No. 17700198).

References

[1] D. Doermann, J. Liang and H. Li, "Progress in Camera-Based Document Image Analysis," Proc. ICDAR, pp. 606–616, 2003.

[2] A. Zappala, A. Gee, M. Taylor, "Document mosaicing," Image and Vision Computing, vol. 17, no. 8, pp. 585–595, 1999.

[3] H. Sakoe, H. Fujii, K. Yoshida, and M. Watari, "A high-speed DP-matching algorithm based on frame synchronization, beam search and vector quantization," Systems and Computers in Japan, vol. 20, no. 11, pp. 33-45, 1989.

[4] R. Plamondon and S. N. Srihari, "On-Line and Off-Line Handwriting Recognition : A Comprehensive Survey," IEEE Trans. Pat. Anal. Mach. Intell., vol. 22, no. 1, pp. 63–84, Jan. 2000.

[5] S. Uchida and H. Sakoe, "Eigen-deformations for elastic matching based handwritten character recognition," Pattern Recognition, vol. 36, no. 9, pp. 2031–2040, 2003.

[6] T. Sato, S. Ikeda, M. Kanbara, A. Iketani, N. Nakajima, N. Yokoya, and K. Yamada, "High-resolution video mosaicing for documents and photos by estimating camera motion," Proc. SPIE Electronic Imaging, vol. 5299, 2004.

[7] H. Li and D. Doermann, "Text Enhancement in Digital Video Using Multiple Frame Intergration," Proc. ACM Multimedia, pp. 19–22, 1999.

[8] J. Kosai, K. Kato, and K. Yamamoto "Recognition of low resolution character by a moving camera," Proc. 5th Int. Conf. Quality Control by Artificial Vision (QACV'99), pp. 203-208, 1999.