Mosaicing-by-recognition for video-based text recognition

Seiichi Uchida [a,*] Hiromitsu Miyazaki [a] Hiroaki Sakoe [a]

[a] *Faculty of Information Science and Electrical Engineering, Kyushu University, 744 Motooka, Nishi-ku, Fukuoka-shi, 819–0395 Japan*

Abstract

Text recognition captured in multiple frames by a hand-held video camera is a challenging task because it is possible to capture and recognize a longer line of text while improving the quality of the text image by utilizing the redundancy of the overlapping areas between the frames. For this task, the video frames should be registered, i.e., mosaiced, after compensating for their distortions due to camera shakes. In this paper, a mosaicing-by-recognition technique is proposed where the problems of video mosaicing and text recognition are formulated as a unified optimization problem and solved by a dynamic programming-based optimization algorithm simultaneously and collaboratively. Experimental results indicate that, even if the frames undergo various distortions such as rotation, scaling, translation, and nonlinear speed fluctuation of camera movement, the proposed technique provides fine mosaic image by accurate distortion estimation (around 90% of perfect estimation) and character recognition accuracy (over 95%).

Key words: video-based text recognition, mosaicing

1 Introduction

Text recognition for a still image captured by a camera has been investigated from the early 1990s. As listed in [1], there are many difficult problems in this task. Previous research efforts, however, have succeeded in developing practical technologies such as commercial cellular phones which can recognize e-mail addresses, URLs, single words, and so on.

Video-based text recognition (**Fig. 1**) also has been investigated as an alternative and challenging task because of its potential to overcome the limitations

* Corresponding author. **e-mail**: uchida@is.kyushu-u.ac.jp (S. Uchida); **tel**: +81-92-802-3586; **fax**: +81-92-802-3600

To be published in Pattern Recognition, vol. 41, no. 4, pp. 1230-1240, 2008.

of the still image. That is, it is possible (i) to recognize a longer line of text by capturing them in multiple frames and (ii) to improve image quality by utilizing the redundancy of the overlapping area between consecutive frames (e.g., super-resolution, noise removal, and close-up). Video-based text recognition will have many applications. This is because video-based text recognition will turn a video camera into a high-accuracy hand-held OCR without limitation on text length. For example, it will be possible to recognize and translate a lengthy food name on a foreign menu.

Video-based text recognition is generally comprised of two processes: video mosaicing and text recognition. Video mosaicing [2] is the technique to register (align) the multiple video frames while compensating their geometric distortions due to camera shakes, such as rotation, vertical shift, scaling, and speed fluctuation [1] (**Fig. 1**). As reviewed in the next section, in previous attempts at video-based text recognition, those two processes are performed in a two-step manner; that is, an entire text image is firstly created by mosaicing and then the text image is subjected to some ordinary text recognition approach.

In this paper, we introduce a *mosaicing-by-recognition* technique, which employs a one-step manner instead of the two-step manner of the previous attempts. Specifically, the proposed technique solves the video mosaicing problem and the text recognition problem as a unified optimization problem and thus provides video mosaicing and text recognition results simultaneously. The optimal solution of the unified problem can be provided efficiently by a dynamic programming (DP)-based algorithm. We can expect a synergy between video mosaicing and text recognition by this one-step organization. In fact, as discussed later and shown through experimental results, video mosaicing is stabilized by text recognition and vise versa.

The remainder of this paper is organized as follows: First, Section 2 provides a brief review on the previous attempts at video-based text recognition. After outlining our task in Section 3, a mosaicing-by-recognition problem for a simple case is formulated and its DP algorithm is provided in Section 4. Then in Section 5, the problem and the algorithm are extended to a general case where more kinds of geometric distortions are assumed than the simple case. Experimental results for qualitative and quantitative evaluation of the proposed algorithm are presented in Section 6. Finally, conclusions are drawn and future work is listed in Section 7.

[1] Although video mosaicing can compensate several types of occlusions [3], we does not assume occlusion here; this is because, for example, texts are generally printed on flat areas and thus it is not necessary to deal with self-occlusion [3].

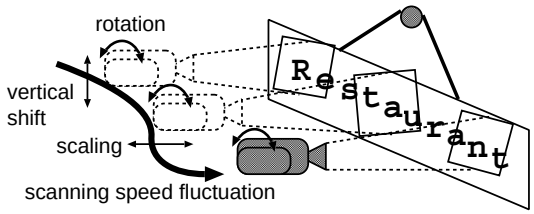

Fig. 1. Recognition of text captured in a video frame sequence.

2 Related work

As noted above, previous attempts at video-based text recognition are generally organized in a two-step manner where video frames are firstly mosaiced, i.e., registered with each other to create a large image and then the created text image, called mosaic image, is subjected to a text recognizer. Zandifar et al. [4] have developed a video-based OCR system organized in the two-step manner. Sato et al. [5] have proposed a video mosaicing technique which can estimate extrinsic camera pose parameters and applied it to document images. Nakano et al. [6] have proposed a special device where a video camera is attached to a mouse. The trajectory of mouse movement on a document sheet guides the estimation of mosaicing parameters. Various ideas of document mosaicing techniques [7–10] which assume a few (say, 2∼10) still images covering an entire document sheet will be useful for video mosaicing.

We also can find video mosaicing techniques where large redundancy in the video frames is utilized for better recognition performance. Super-resolution in the image (gray-scale) domain [5,11–13] is typical one. Super-resolution in the (directional) feature domain [14] is a promising attempt. If we can expect a good segmentation scheme for isolating characters, the redundancy can be utilized in a recognition step like [15].

The proposed technique is the first attempt to perform video mosaicing and text recognition in a one-step manner; that is, the proposed technique performs video mosaicing and text recognition simultaneously and collaboratively. The advantage of this one-step organization is higher mosaicing and recognition accuracies. The two-step organization of the previous techniques may fail at the first step since video mosaicing by minimizing the registration error between consecutive frames (hereafter called the interframe matching cost) often fails due to various noises and the ambiguity in the correspondence between those frames. In contrast, in the proposed one-step organization, video mosaicing is stabilized by minimizing the interframe matching cost together with another complementary matching cost prepared for text recognition (called the intraframe matching cost). Equally, text recognition is stabilized by the interframe matching cost. The advantage of the one-step organization will be demonstrated through the experimental result provided in Section 6.

Senda et al. [16] have proposed a text recognition technique organized in a different two-step manner; in their technique, text recognition is firstly performed at each frame independently. Then the recognition results at all frames are combined as the recognition result of the entire text. While this is another reasonable strategy, it has no function to estimate geometric distortion due to camera shake. Actually, in [16], it is assumed that each text line is captured within a horizontal window which is specified to exclude rotation and vertical translation. The proposed technique estimates distortion during the one-step optimization and therefore requires no window.

The disadvantage of the proposed technique will be its computational complexity, as detailed in 5. There, however, are many complexity reduction strategies which can be applied to our DP-based algorithm. Beam search, or pruning, is a popular one. In this paper, those strategies will not be employed for observing the basic performance of the proposed technique.

3 The task

Our task is the recognition of a text captured fragmentarily in video frames by mosaicing the frames while removing their distortions. We make the following assumptions on solving this task (**Fig. 1**):

- A hand-held video camera moves from left to right to capture a single line of text fragmentarily on video frames.
- Every character in the line of text is captured in multiple frames.
- Each frame undergoes geometric distortion due to camera shake.
- The speed of the camera movement fluctuates.
- A reference pattern, which is a standard character image (often called prototype or template), is already prepared and stored for each category.

The assumptions on the singleness of the text line will be reasonable because the resolution of the hand-held camera is often low and users are required to capture a single line of text by putting the camera closer to the text. In addition, we can expect some segmentation technique for the extraction of a single text line. We also can expect a more reliable extraction by using a horizontal window, which has already been employed in the OCR screen of commercial cellular phones (e.g., [16]).

As the geometric distortions by camera shake, we will tackle the following major geometric distortions: rotation, scaling, vertical shift. Scanning speed fluctuation appears as the changes in the degree of the overlap between consecutive frames. For example, if the speed is slowed down, the overlapping area becomes larger (and thus the same character will be captured in many

Fig. 2. Video frame sequence capturing a line of text. (a) Simple case. (b) General case.

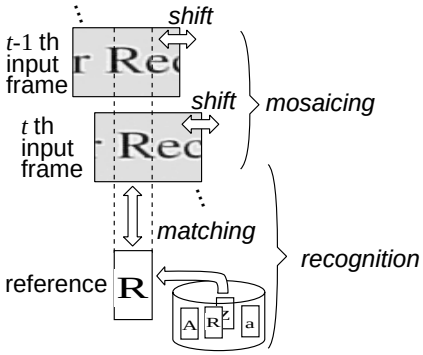

Fig. 3. Basic idea of mosaicing-by-recognition (simple case).

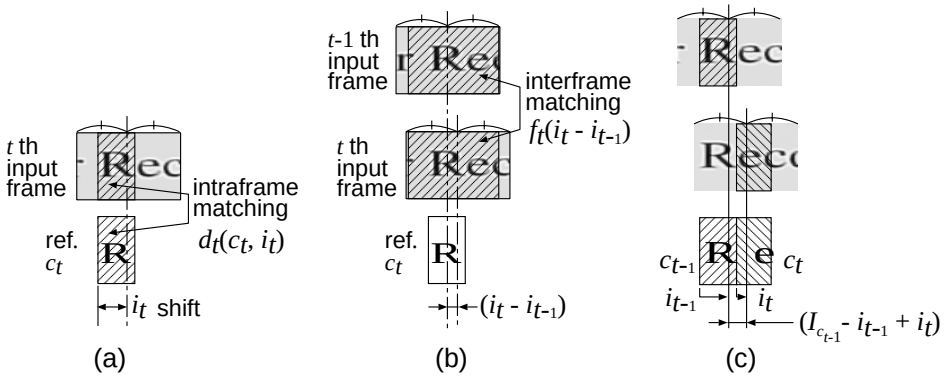

Fig. 4. (a) Horizontal shift i_t. (b) If $c_t = c_{t-1}$, the difference $(i_t - i_{t-1})$ represents the horizontal displacement between the tth and $(t-1)$th frames. (c) If $c_t \neq c_{t-1}$, $(I_{c_{t-1}} - i_{t-1} + i_t)$ represents the horizontal displacement.

frames).

4 Mosaicing-by-recognition for simple case

In this section, we consider the *simple case* where video frames undergo only the speed fluctuation. **Fig. 2** (a) shows a video frame sequence of the simple case. This simplification is quite useful to grasp the basic idea of mosaicing-by-recognition.

In the proposed technique, character recognition, that is, the estimation of the category of a character within a frame is based on template matching between the frame and every reference pattern. Since the position of the character is unknown in the frame, the position also should be estimated. It is important to note that the category and the position are not estimated within the frame; this is because consecutive frames may include the same character and thus there is strong dependency between the frames. Consequently, as described below, the estimation result is obtained after solving an optimization problem where the relation between the consecutive frames is considered carefully.

4.1 The basic idea

Fig. 3 illustrates mosaicing-by-recognition in the simple case. As shown in this figure, there is a reference pattern which matches the character pattern at the horizontal center of the tth input frame by an appropriate horizontal shift. This fact provides the basic idea of mosaicing-by-recognition as follows:

- The category of the matched reference pattern is considered as the character recognition result of the tth input frame.
- If both of the tth and $(t-1)$th input frames are appropriately shifted to match the same reference pattern, the horizontal displacement (i.e., camera movement) between those frames is expressed by the difference of their shifts. This means that their registration, i.e., mosaicing can be done by using the shifts.

Consequently, recognition and mosaicing can be done simultaneously by optimizing c_t and i_t $(t = 1, \ldots, T)$ where c_t and i_t denote the matched category and shift of the tth input frame, respectively, and T is the number of input frames. In other words, the optimized category sequence $c_1, \ldots, c_T$ represents the recognition result and the optimized shift sequence $i_1 \ \ldots, i_T$ can built the mosaic image . In the following sections, the optimization problem of $\{c_t, i_t\}$ is firstly formulated and then its solution by a DP-based algorithm is described.

As shown in **Fig. 4** (a), the shift i_t is defined as the displacement between the horizontal center of the tth input frame and the left side of the reference pattern of the category c_t. As shown in **Fig. 4** (b), if $c_t = c_{t-1}$ (that is, the tth and $(t-1)$th frames match the same reference pattern), the horizontal displacement between the tth and $(t-1)$th input frames is represented by $(i_t - i_{t-1})$.

It should be noted that the difference $(i_t - i_{t-1})$ is meaningless if $c_t \neq c_{t-1}$, that is, if the center character is altered during the transition from $(t-1)$ to t. This is because the shifts i_t and i_{t-1} are measured for different reference patterns. Thus, in this case, the difference $(I_{c_{t-1}} - i_{t-1} + i_t)$ should be used

instead of $(i_t - i_{t-1})$, where I_{c_t} is the width of the reference pattern of the category c_t. **Fig. 4** (c) shows the difference $(I_{c_{t-1}} - i_{t-1} + i_t)$ in the case of $c_t \neq c_{t-1}$. In the following discussion, this case is not described explicitly for notational simplicity.

4.2 Intraframe matching cost

The matching between the reference pattern of category c_t and the tth input frame is evaluated by the *intraframe matching cost* $d_t(c_t, i_t)$, which is simply defined as the L^2-distance of their matched area. **Fig. 4** (a) illustrates the intraframe matching cost, where the hatched area is the matched area to be evaluated. The value of $d_t(c_t, i_t)$ is normalized by the number of the pixels in the matched area. Therefore, it ranges from 0 to 255 if each frame is an 8-bit gray-scale image. This normalization is necessary to make $d_t(c_t, i_t)$ insensitive to the size of the reference pattern.

A weakness of the intraframe matching cost is its sensitivity against distortions of input character patterns such that complex background, lighting condition, font shape, etc. Under those distortions, i_t and c_t are not evaluated satisfactorily only by the intraframe matching cost. Thus, as described below, another complementary matching cost is introduced to overcome the weakness.

4.3 Interframe matching cost

We introduce another matching cost, *interframe matching cost* $f_t(i_t - i_{t-1})$, which is defined as the L^2-distance of the overlapping area between the tth and $(t-1)$th frames under the horizontal displacement of $(i_t - i_{t-1})$ pixels. The hatched area of **Fig. 4** (b) is the overlapping area to evaluate the interframe matching cost. Since the interframe matching cost is independent of reference patterns, its minimization will help to provide the correct horizontal displacement even under complex background and other distortions of input frames.

Like $d_t(c_t, i_t)$, the value of $f_t(i_t - i_{t-1})$ is normalized by the number of the pixels in the overlapping area. Therefore it ranges from 0 to 255. Without this normalization, smaller overlapping is preferred and spurious repetition of the same character is often produced in the resulting mosaic image.

4.4 Constraint

The following constraint is imposed on $(i_t - i_{t-1})$, i.e., the horizontal displacement between two consecutive frames;

$$0 \leq (i_t - i_{t-1}) \leq Q. \tag{1}$$

The non-negativity of $(i_t - i_{t-1})$ comes from the assumption of left-to-right camera movement. The constant Q specifies the allowable maximum speed of the camera movement, since $(i_t - i_{t-1})$ is proportional to the speed of camera movement. For example, if the frame rate of the video camera is 30 frame/s, the allowable maximum speed is $30Q$ pixel/s.

4.5 The optimization problem

According to the above discussion, we can formulate the mosaicing-by-recognition problem as follows:

$$\left.\begin{array}{l} \text{minimize:} \\ \quad J = \alpha d_1(c_1, i_1) + \displaystyle\sum_{t=2}^{T} \left(\alpha d_t(c_t, i_t) + (1-\alpha) f_t(i_t - i_{t-1})\right), \\ \text{with respect to:} \quad \{(c_t, i_t) \mid t = 1, \ldots, T\}, \\ \text{subject to:} \quad 0 \leq (i_t - i_{t-1}) \leq Q,\ 1 \leq i_t \leq I_{c_t},\ 1 \leq c_t \leq C, \end{array}\right\} \tag{2}$$

where C is the number of categories and α $(0 \leq \alpha \leq 1)$ is a weight constant to balance the intraframe matching cost against the interframe matching cost. The value of α will be determined through a preliminary experiment. Note again that when $c_t \neq c_{t-1}$, the difference $(I_{c_{t-1}} - i_{t-1} + i_t)$ should be used instead of $(i_t - i_{t-1})$ for both of $f_t(i_t - i_{t-1})$ and the constraint (1).

4.6 DP algorithm

For solving the minimization problem (2), now we consider a function $g_t(c_t, i_t)$ defined as

$$g_t(c_t, i_t)$$

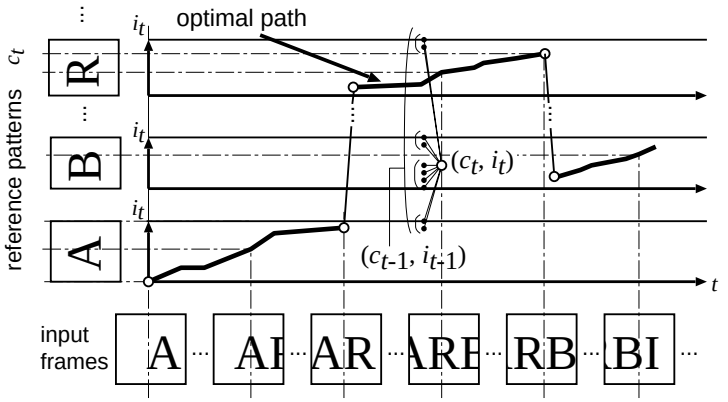

Fig. 5. Mosaicing-by-recognition (simple case) as an optimal path problem.

$$= \min_{\substack{\{c_k,i_k|k=1,\ldots,t-1\} \\ 0\le(i_k-i_{k-1})\le Q}} \left[\alpha d_1(c_1,i_1) + \sum_{k=2}^{t} \left(\alpha d_k(c_k,i_k) + (1-\alpha) f_k(i_k - i_{k-1})\right)\right]. \tag{3}$$

Clearly, the minimum of J is represented as

$$\min J = \min_{c_T, i_T} g_T(c_T, i_T). \tag{4}$$

According to the Markovian property, (3) can be rewritten as

$$\begin{aligned} & g_t(c_t, i_t) \\ &= \min_{\substack{c_{t-1},i_{t-1} \\ 0\le(i_t-i_{t-1})\le Q}} \left[g_{t-1}(c_{t-1},i_{t-1}) + \alpha d_t(c_t,i_t) + (1-\alpha) f_t(i_t - i_{t-1})\right] \\ &= \alpha d_t(c_t,i_t) + \min_{\substack{c_{t-1},i_{t-1} \\ 0\le(i_t-i_{t-1})\le Q}} \left[g_{t-1}(c_{t-1},i_{t-1}) + (1-\alpha) f_t(i_t - i_{t-1})\right]. \end{aligned} \tag{5}$$

This equation is a so-called *DP recursion* and represents the recursive procedure for obtaining the optimal solution. The optimal solution of (2) will be obtained by starting from $g_1(c_1,i_1) = \alpha d_1(c_1,i_1)$, then calculating (5) for all possible (c_t, i_t) from $t = 2$ to T, and finally using (4).

This DP algorithm can be considered as an optimal path problem as illustrated in **Fig. 5**. The optimal path in this figure represents the sequence $(c_1,i_1),\ldots,(c_t,i_t),\ldots,(c_T,i_T)$ given after the backtracking operation described below. The local slope of the path represents the horizontal displacement between consecutive frames, $(i_t - i_{t-1})$ and thus it is constrained by (1). The big jump in the path indicates the frame where the center character is altered, i.e., $c_t \neq c_{t-1}$. Note that the range of the big jump is constrained by

$0 \leq (I_{c_{t-1}} - i_{t-1} + i_t) \leq Q$, which is the special case of (1).

The backtracking operation for obtaining the optimal (c_t, i_t)-sequence starts from $(c_T, i_T) = \mathrm{argmin}_{c_T, i_T}\, g_T(c_T, i_T)$, recursively finds (c_{t-1}, i_{t-1}) which gives the minimum in (5), and arrives (c_1, i_1) at last. The resulting sequence $(c_1, i_1), \ldots, (c_t, i_t), \ldots, (c_T, i_T)$ is the optimal solution of (2) and provides the text recognition result from the transition of $c_1, \ldots, c_T$. Simultaneously, a mosaic image is also obtained by using $i_1, \ldots, i_T$ as the horizontal displacement of consecutive frames. On creating the mosaic image, we should determine the pixel value of the overlapping area between two consecutive frames. In the following experiment, a simple averaging was employed.

The mosaicing-by-recognition algorithm for the simple case is closely related to segmentation-by-recognition, or recognition-based segmentation [17,18], which is a well-known technique for recognizing a single line of text in a still image. Appendix A details this relation.

5 Mosaicing-by-recognition for general case

In this section, we describe the mosaicing-by-recognition algorithm for the general case, where not only the speed fluctuation but also the other distortions, i.e., rotation, scaling, and vertical shift, are considered. **Fig. 2** (b) shows a video frame sequence of the general case. The main extension from the simple case is the geometrical transformation of the input frame for compensating for those distortions. As shown in **Fig. 6** where a video frame is represented by a rectangle, the tth video frame is geometrically transformed by three parameters, r_t, s_t, and v_t, which represent rotation, scaling, and vertical shift, respectively.

It is assumed that the parameters r_t, s_t, and v_t are integers representing the number of pixels and bounded by $\pm K$, that is, $-K \preceq \boldsymbol{p}_t = (r_t, s_t, v_t) \preceq K$, where the notation $K \preceq \boldsymbol{p}_t$ denotes that all the elements of the vector $\boldsymbol{p}_t$ are larger than or equal to the scalar K. **Fig. 7** shows frames which undergo distortions controlled by r_t, s_t, or v_t .

The mosaicing-by-recognition problem for the general case is formulated by extending the simple case such that the parameters $\{\boldsymbol{p}_t\}$ are optimized to-

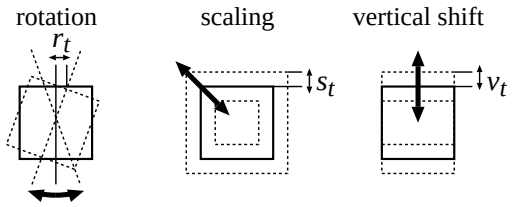

Fig. 6. Transformation of video frame in order to compensate for distortions.

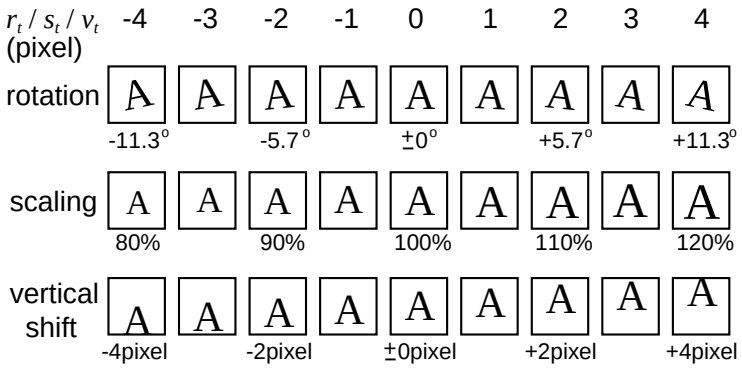

Fig. 7. Frames which undergo rotation, scaling, or vertical shift.

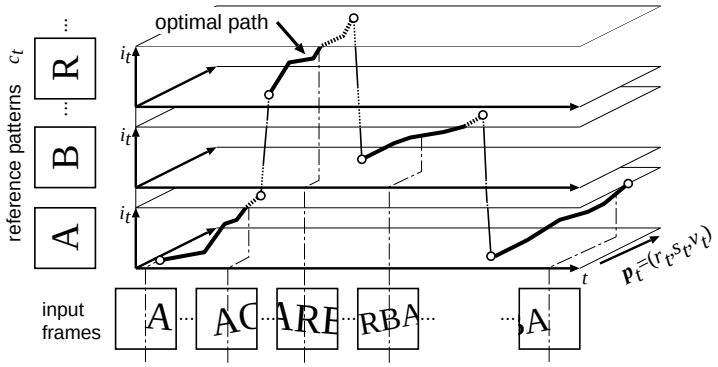

Fig. 8. Mosaicing-by-recognition (general case) as an optimal path problem.

gether with $\{(c_t, i_t)\}$, that is,

$$
\left.
\begin{aligned}
&\text{minimize:} \\
&\quad J = \alpha d_1(c_1, i_1, \boldsymbol{p}_1) \\
&\qquad + \textstyle\sum_{t=2}^{T} \left(\alpha d_t(c_t, i_t, \boldsymbol{p}_t) + (1-\alpha) f_t(i_t - i_{t-1}, \boldsymbol{p}_t, \boldsymbol{p}_{t-1})\right), \\
&\text{with respect to:} \quad \{(c_t, i_t, \boldsymbol{p}_t) \mid t = 1, \ldots, T\}, \\
&\text{subject to:} \\
&\quad 0 \le (i_t - i_{t-1}) \le Q,\ 1 \le i_t \le I_{c_t},\ 1 \le c_t \le C, \\
&\quad -1 \preceq (\boldsymbol{p}_t - \boldsymbol{p}_{t-1}) \preceq 1,\ -K_{\max} \preceq \boldsymbol{p}_t \preceq K_{\max},
\end{aligned}
\right\} \qquad (6)
$$

where $K_{\max}$ represents the degree of the maximum distortion which can be compensated by the algorithm. Thus, $K_{\max}$ should be set at a value larger than the expected value of K for compensating any distortion. The constraint on $(\boldsymbol{p}_t - \boldsymbol{p}_{t-1})$ is used to ensure that $\boldsymbol{p}_{t-1}$ and $\boldsymbol{p}_t$ are similar. This constraint relies on the fact that the distortions between two consecutive frames will vary smoothly.

The function $d_t(c_t, i_t, \boldsymbol{p}_t)$ is the intraframe matching cost after transforming the tth frame by $\boldsymbol{p}_t = (r_t, s_t, v_t)$. That is, the function $d_t(c_t, i_t, \boldsymbol{p}_t)$ evaluates the matching between the reference c_t and the tth frame distorted by the rotation r_t, the scaling s_t, and the vertical shift v_t. Similarly, the function $f_t(i_t - i_{t-1}, \boldsymbol{p}_t, \boldsymbol{p}_{t-1})$ is the interframe matching cost after transforming the tth and the $(t-1)$th frames by $\boldsymbol{p}_t$ and $\boldsymbol{p}_{t-1}$, respectively.

Like the simple case, we can derive a DP algorithm to solve the optimization problem of $\boldsymbol{p}_t$ together with (c_t, i_t). The derived algorithm can be illustrated as **Fig. 8** where the optimal path represents the optimal sequence of $(c_t, i_t, \boldsymbol{p}_t)$. The mosaic image can be obtained by firstly transforming the tth frame by the parameter $\boldsymbol{p}_t$ and then placing it next to the (transformed) $(t-1)$th frame with a horizontal displacement of $(i_t - i_{t-1})$ pixels.

The computational complexity of the DP algorithm for the general case is proportional to the size of the search space illustrated in **Fig. 8** and thus $O(TCIK_{\max}^3)$, where I is the average width of the reference patterns. For the reduction of the complexity, we can introduce beam search, which has often been employed in DP-based algorithms and is based on a pruning operation to eliminate unpromising search paths at every t. Specifically, if $g_t(c_t, i_t, \boldsymbol{p}_t)$ exceeds a threshold, no path emerges from $(c_t, i_t, \boldsymbol{p}_t)$ (that is, $(c_t, i_t, \boldsymbol{p}_t)$ is excluded from the predecessor set of $(c_{t+1}, i_{t+1}, \boldsymbol{p}_{t+1})$).

6 Experimental results

6.1 Data preparation

For performance evaluation, 20 text lines were printed on A4-sized papers with three different backgrounds of **Fig. 9**. Each text line contains about 50 characters (of capital/small English alphabets and digits) and thus about 1000 characters were prepared for each background (that is, about 3000 characters in total). All the characters were printed in the same Times-Roman font.

The complex backgrounds of Figs. 8(b) and (c) were (diagonal parallel) straight hatching and cross hatching, respectively, and created by a drawing software.

Fig. 9. Type of background. (a) Plain. (b) Complex (straight-hatching). (c) Complex (cross-hatching).

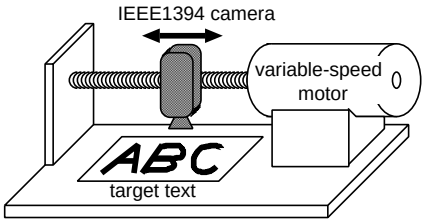

Fig. 10. Equipment for moving camera.

Both of those backgrounds are generally unwelcome to mosaicing techniques because their repetitive property will produce ambiguity on determining the translation parameter. Those backgrounds, of course, will also degrade recognition accuracy.

The video frames of the simple case were firstly prepared by capturing each text line in multiple frames by moving a video camera of 30 frame/s. Since lighting condition was not strictly controlled and the video camera was not a special one, slight but visible background noise was introduced into the video frames. Each frame, which was an RGB image originally, was then converted into an 8-bit gray-scale image by a very simple strategy (i.e., (R+G+B)/3). Special equipment with a variable speed motor (**Fig. 10**) was used for moving the camera horizontally along a line of text, while excluding rotation, scaling, and vertical shift. In the experiment, the camera speed was varied manually and randomly from 0 to 3 pixel/frame (i.e., from 0 to 90 pixel/s) by controlling the motor. The frame size was 60×60 pixels and the character width was about $10 \sim 25$ pixels. Thus, each character was captured in at least $(60 - 25)/3 \sim 11$ frames. Note that the results obtained under higher speeds ($\sim$ 5 pixel/frame) were similar to the following results and thus omitted in this paper.

The video frames of the general case were then prepared by rotating, scaling and vertically shifting the above video frames of the simple case artificially. Under a certain value of K, the value of $\boldsymbol{p}_t$ was determined randomly while satisfying the constraint $\boldsymbol{p}_{t-1} \in \Psi(\boldsymbol{p}_t)$ and $|r_t| \leq K$, $|s_t| \leq K$, and $|v_t| \leq K$ (that is, rotations, scaling, and vertical shift will appear at the same time in each frame). Thus, the ground-truth of $\boldsymbol{p}_t = (r_t, s_t, v_t)$ and K was available in

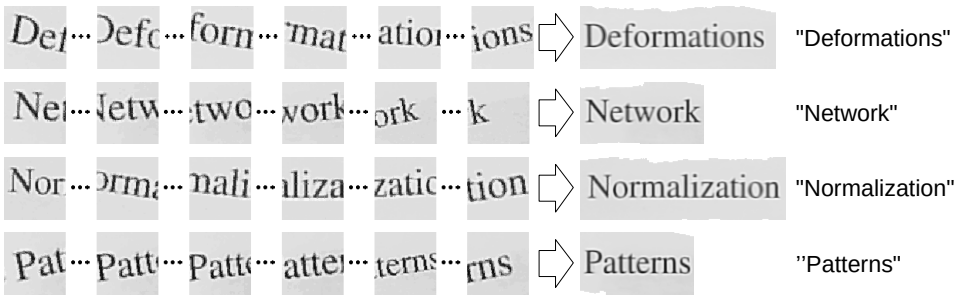

Fig. 11. Mosaicing and recognition results under plain background.

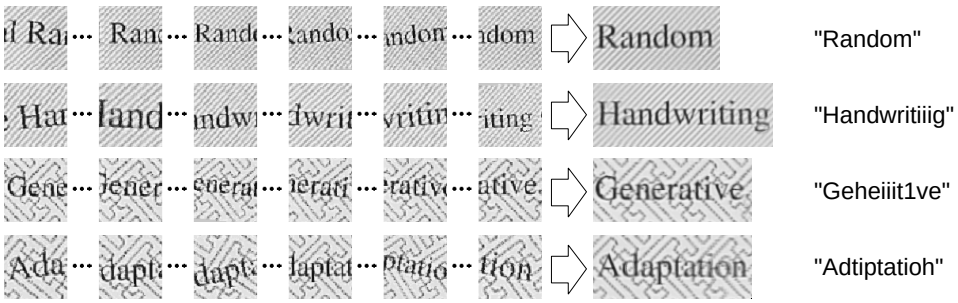

Fig. 12. Mosaicing and recognition results under complex backgrounds.

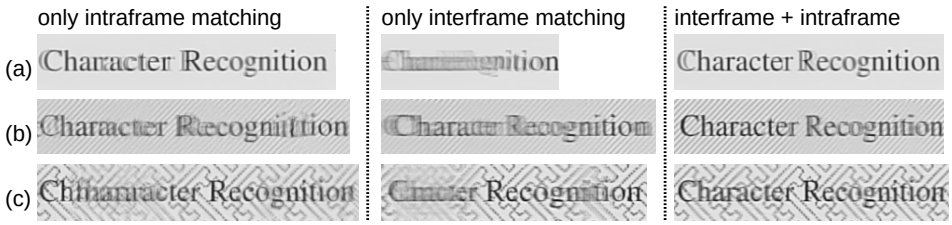

Fig. 13. Mosaicing results by different objective functions. (a) Plain background. (b) Complex background (straight-hatching). (c) Complex background (cross-hatching).

the following experiment.

In the following experiment, only the video frames and the DP-algorithm of the general case were used. The maximum displacement Q was fixed at 3 pixel/frame because the maximum camera speed was 3 pixel/frame. (Preliminary experiments showed that the recognition/mosaicing results were not degraded if Q was fixed at a value larger than 3.) The weight α was fixed at 0.7 by a preliminary experiment unless otherwise mentioned.

Fig. 11 shows video frames which underwent speed fluctuation, rotation, scaling, and vertical-shift and their mosaic images given by the proposed technique. Both of the distortion parameters $K_{\max}$ and K were fixed at 4. Each frame underwent non-trivial distortions because the distortion at $K_{\max} = 4$ corresponds to $\pm 11.3°$ rotation and 80% or 120% scaling as shown in **Fig. 7**. The background of the frames was the plain one. The mosaic images were very clean and contain no artifact (e.g., blur and ghost) and therefore reveal that the proposed DP algorithm could find the correct displacements $\{i_t\}$ and control parameters $\{\boldsymbol{p}_t\}$ as its solution. **Fig. 11** also shows that correct recognition results were obtained for all of the four examples.

Fig. 12 shows the results on the complex backgrounds. The mosaic images were still natural. As shown later, this robustness against the clutter by the complex backgrounds was brought by the interframe matching cost. The recognition results were sometimes erroneous. The complex backgrounds sometimes prevent the intraframe matching cost from selecting the correct category even when the correct displacement, i.e., the correct mosaicing image, is obtained. A possible remedy for eliminating this interference will be the modification of the intraframe matching cost to be invariant to the background. For example, a cost based on chamfer matching [19] will be promising.

Usefulness of combining the two matching costs is confirmed by the mosaic images in **Fig. 13**. This figure shows the mosaic images provided under different matching costs. The left column shows results using only the intraframe matching cost (i.e., $\alpha = 1$ in (6)), the center column shows results using only the interframe matching cost (i.e., $\alpha = 0$), and the right column shows results using both matching costs ($\alpha = 0.7$). For the plain background, the intraframe matching cost alone provided a reasonable mosaic image while slight ghosts around "C" and "R" are observed. For the complex backgrounds, however, it provided mosaic images with heavy blur and ghost due to the erroneous estimation of i_t. The interframe matching cost alone also provided poor results due to noises and, especially, the ambiguity in the correspondence between the consecutive frames. In fact, the mosaic image under the plain background, which is the most ambiguous case, is degraded most seriously. In contrast, when the two matching costs were used, the best results were obtained under any background.

Fig. 13 reveals the weakness of the previous techniques organized in the two-step manner. Generally, the mosaicing process performed in their first step relies on the minimization of the interframe matching cost and the recognition process in their second step should deal with heavily blurred text images like the images at the middle column of **Fig. 13**.

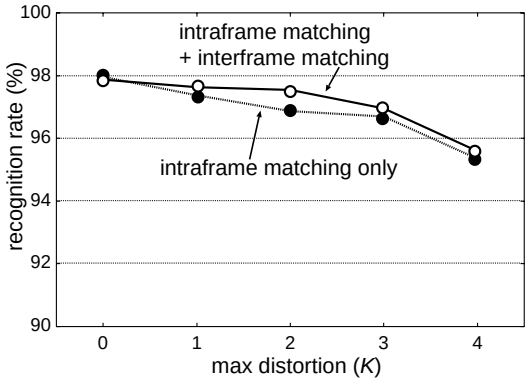

Fig. 14. Recognition accuracy under plain background.

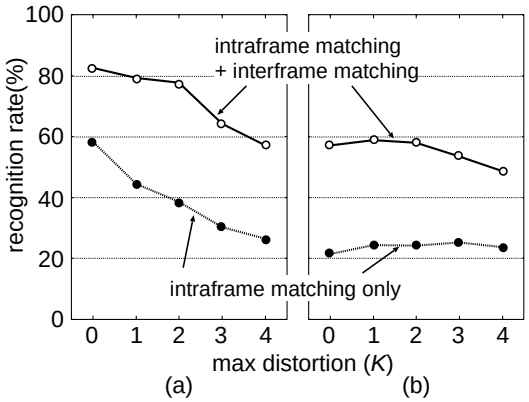

Fig. 15. Recognition accuracy under complex backgrounds. (a) Complex background (straight-hatching). (b) Complex background (cross-hatching).

6.3 Quantitative analysis

6.3.1 Recognition accuracy

Fig. 14 shows the recognition rate as a function of the maximum amplitude of the distortions, K, under the plain background. (Note that the constant $K_{\max}$ was fixed at 4 regardless of K.) The result shows that the proposed technique could attain recognition rates over 95% despite of various distortions. Considering that a naive gray-level feature and the simple L^2-distance were used in d_t and f_t, those rates will be acceptable one.

Misrecognitions were mainly due to the following reasons: (i) noises at capturing, (ii) similar characters ("l" $\leftrightarrow$ "1" and "G" $\leftrightarrow$ "C"), (iii) over-segmentation ("H" $\leftrightarrow$ "I I" and "M" $\leftrightarrow$ "v 1"), and (iv) over-fitting ("S" $\leftrightarrow$ "s" and "T" $\leftrightarrow$ "7"). Over-segmentation is a typical misrecognition by segmentation-by-recognition -based algorithms, whereas over-fitting is a typical misrecognition by elastic matching-based algorithms [20]. The proposed algorithm has close relation to those algorithms and thus inherits their misrecognition tendencies. A well-

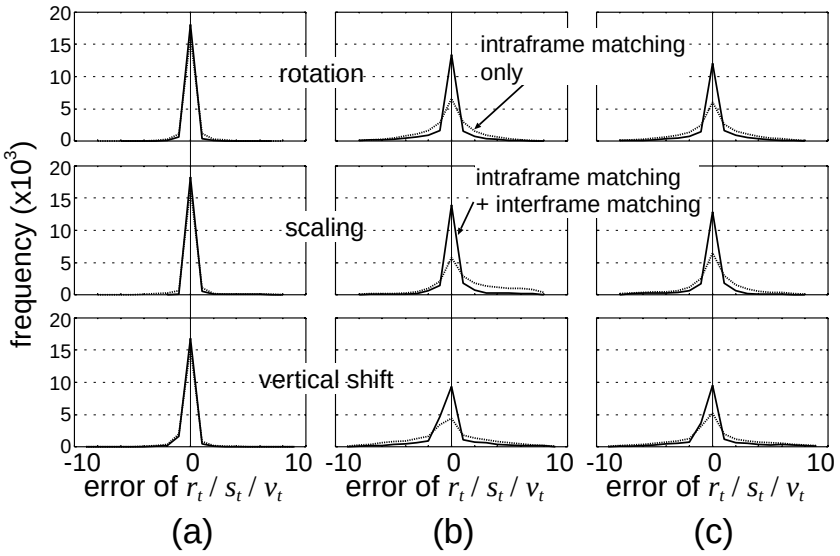

Fig. 16. Histogram of errors of mosaicing parameters. (a) Plain background. (b) Complex background (straight-hatching). (c) Complex background (cross-hatching).

known remedy for over-segmentation is the use of a word lexicon. For over-fitting, the use of some sophisticated pixel feature (e.g., directional feature, background feature, crossing feature, localized moment feature, etc.) will be a possible remedy.

Figs. 15 (a) and (b) show the recognition rates under the straight-hatching and cross-hatching backgrounds, respectively. This result reveals that the recognition rates under the complex backgrounds were improved drastically by incorporating the interframe matching cost into the intraframe matching cost, whereas the recognition rate under the plain background was improved slightly, as shown in **Fig. 14**. This drastic improvement can be explained by the fact that the incorporation of the interframe matching could provide accurate mosaic images under the complex background as shown in **Fig. 13** (b) and (c).

6.3.2 Mosaicing accuracy

As noted in 6.1, the ground-truth of $\boldsymbol{p}_t$ was given and therefore the accuracy of mosaicing results could be evaluated qualitatively by observing the error of the estimated mosaicing parameter $\boldsymbol{p}_t$. **Fig. 16** shows the histograms of the errors of each distortion at $K_{\max} = K = 4$. The error was defined as the simple difference; for example, the error of rotation was defined as the difference $\hat{r}_t - r_t$ where $\hat{r}_t$ is the true parameter value. The frequency in the histogram represents the number of frames having a certain error.

Fig. 16 (a) demonstrates that the mosaicing parameters were estimated very accurately at most frames under the plain background. When both costs were used, the parameters of rotation, scaling, and vertical shift were perfectly estimated at 93.3%, 94.6%, and 86.9% of all the frames, respectively. If errors of ± 1 were allowed to r_t, s_t, and v_t, those rates were improved to 98.3%, 98.1%, 97.6%, respectively. These accuracies were acceptable, considering (i) the contamination of the frames by noise in capturing them via the camera (and digitization noise in applying the artificial distortions to them), (ii) the naive gray-scale feature and L^2 matching distance, and (iii) the nonlinear fluctuation of camera speed.

As shown in **Fig. 16** (b) and (c), the accuracy was degraded under the complex backgrounds. Especially, when the interframe matching cost was not used, serious degradation was observed. It follows that the intraframe matching cost alone was insufficient not only for mosaicing but also for text recognition under complex backgrounds.

7 Conclusion and future work

A mosaicing-by-recognition technique was proposed for recognizing a single line of text captured by hand-held video camera. In the proposed technique, two processes, i.e., text recognition and video mosaicing, are performed simultaneously and collaboratively in a one-step manner by a DP-based optimization algorithm. Experimental results showed that the proposed technique could recognize over 95% characters printed on a plain background despite of rotation, scaling, vertical shift, and speed fluctuation appeared in the frames. The experimental results on complex backgrounds emphasized the necessity of two complementary matching costs, called interframe matching cost and intraframe matching cost, for video mosaicing and text recognition.

Future work will focus on the following points:

- *Sophisticated pixel feature and matching cost*: In this paper, only naive pixel feature, i.e., gray-level feature, and simple L^2 matching cost were employed. Since the gray-level feature is very weak to noises, illumination condition, and complex background, it is necessary to employ more sophisticated pixel features, such as directional feature, background feature, crossing feature, and localized moment feature. Similarly, it is necessary to use matching costs which can compensate for the weakness of the pixel feature.
- *Further analysis on the effect of backgrounds*: In this paper, three background textures have been examined to show the robustness of the proposed technique against the complex backgrounds. Further quantitative and qualitative analyses to clarify the relation between the "busyness" of the back-

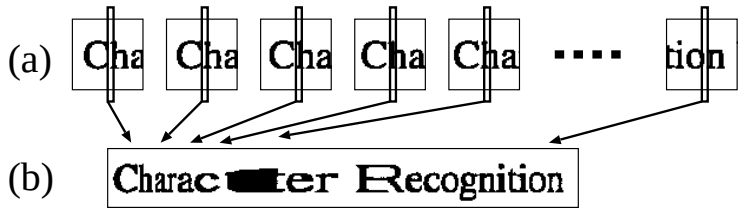

Fig. A.1. (a) A video frame sequence and (b) the still image provided by assuming that the width of each frame is one pixel.

ground and the performance will help the design of the sophisticated pixel feature and the matching cost.

- *Lexicon*: The proposed technique often produces misrecognitions by over-segmentation (e.g., "m"→ "r" and "n"). Like other text recognizers based on the segmentation-by-recognition framework, the use of lexicon will be helpful to tackle with the over-segmentation [17,18].
- *The improvement of mosaic image quality*: In this paper, the overlapping area between frames was simply averaged on creating a mosaic image. The quality of the mosaic image will be improved by super-resolution techniques [5,11–14].
- *Perspective distortion*: Precisely speaking, we should tackle with perspective distortion in addition to rotation, scaling, and translation. One possible strategy is the use of camera pose parameters instead of the transformation parameters, (r_t, s_t, v_t).
- *Reduction of computational complexity*: In this paper, we have adhered to the globally optimal solution by the DP algorithm at the cost of computational complexity, since we wanted to grasp the basic performance of the proposed technique. As noted in 5, we can introduce beam search into the DP algorithm for obtaining an approximate solution by far less complexity. A more drastic reduction can be attained by using various greedy/heuristic search strategies instead of DP.

A The relationship to segmentation-by-recognition

The proposed mosaicing-by-recognition algorithm has been inspired by the segmentation-by-recognition algorithm, which is a classical but still up-to-date strategy for the recognition of cursive words/sentences on a still image [17,18]. The principle of the segmentation-by-recognition algorithm is the simultaneous optimization of segmentation and recognition; specifically, the optimization is done by matching every reference pattern to every possible segment and searching for the continuous segment sequence which provides the minimum total matching cost. Generally, those matching and search processes are

performed simultaneously and efficiently by DP (or its stochastic extension called HMM).

The close relation between the two algorithms can be observed by assuming that the width of each video frame is one pixel in the simple case Under this assumption, we can create a still text image by concatenating those frames of one-pixel width. **Fig. A.1** shows an example of the created still text image. According to the speed fluctuation, the component characters undergo nonlinear geometric distortions in their horizontal direction.

Now, we can consider a DP-based mosaicing-by-recognition algorithm for the still image by replacing t of **Fig. 5** by the horizontal coordinate of the still image. That is, the frame index t corresponds to the tth column of the still image. This algorithm is equivalent to the segmentation-by-recognition algorithm. Thus, we can conclude that the conventional segmentation-by-recognition algorithm is a special case of the proposed mosaicing-by-recognition algorithm.

Acknowledgment: This work was supported in part by the Research Grant of The Okawa Foundation and the Research Grant (No.17700198) of The Ministry of Education, Culture, Sports, Science and Technology in Japan.

References

[1] J. Liang, D. Doermann, and H. Li, "Camera-based analysis of text and documents: a survey," Int. J. Doc. Anal. Recog., vol. 7, no. 2-3, pp. 84-104, 2005.

[2] M. Irani and P. Anandan, "Video indexing based on mosaic representation," Proc. IEEE, vol. 86, no. 5, pp. 905–921, 1998.

[3] C. Toklu, T. Erdem, and A. M. Tekalp, "Two-dimensional mesh-based mosaic representation for manipulation of video objects with occlusion," IEEE Trans. Image Proc., vol. 9, no. 9, pp. 1617–1630, 2000.

[4] A. Zandifar, R. Duraiswami, A. Chahine, and L. Davis, "A video based interface to textual information for the visually impaired," Proc. 4th Int. Conf. Multimodal Interfaces, pp. 325–330, 2002.

[5] 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.

[6] T. Nakano, A. Kashitani, and A. Kaneyoshi, "Scanning a document with a small camera attached to a mouse," Proc. 4th IEEE Workshop on Applications of Comput. Vis., pp. 63-68, 1998.

[7] A. P. Whichello and H. Yan, "Document image mosaicing," Proc. Int. Conf. Pattern Recog., vol. 2 of 2, pp. 1081–1083, 1998.

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

[9] M. Mirmehdi, P. Clark, and J. Lam, "A non-contact method of capturing low-resolution text for OCR," Pattern Anal. Appl., vol. 6, no. 1, pp. 12–21, 2003.

[10] G. H. Kumar, P. Shivakumara, D. S. Guru, and P. Nagabhushan, "Document image mosaicing: a novel approach," Sādhanā, vol. 29, Part. 3, pp. 329–341, 2004.

[11] H. Li and D. Doermann, "Text enhancement in digital video using multiple frame integration," Proc. ACM Multimedia, pp. 19–22, 1999.

[12] D. Capel and A. Zisserman, "Super-resolution enhancement of text image sequences," Proc. Int. Conf. Pattern Recog., vol. 1 of 4, pp. 600–605, 2000.

[13] C. Mancas-Thillou and M. Mirmehdi, "Super-resolution text using the Teager filter," Proc. 1st Int. Workshop Camera-Based Doc. Anal. Recog., pp. 10–16, 2005.

[14] 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, pp. 203-208, 1999.

[15] H. Ishida, S. Yanadume, T. Takahashi, I. Ide, Y. Mekada, and H. Murase, "Recognition of low-resolution character by a generative learning method," Proc. 1st Int. Workshop Camera-Based Doc. Anal. Recog., pp. 45–51, 2005.

[16] S. Senda, K. Nishiyama, T. Asahi, and K. Yamada, "Camera-typing interface for ubiquitous information services," Proc. 2nd Conf. on Pervasive Computing and Communications, pp. 366–370, 2004.

[17] R. G. Casey and E. Lecolinet, "A survey of methods and strategies in character segmentation," IEEE Trans. Pattern Anal. Mach. Intell., vol. 18, no. 7, pp. 690–706, 1996.

[18] A. Vinciarelli, "A survey on off-line cursive word recognition," Pattern Recognit., vol. 35, no. 7, pp. 1433–1446, 2002.

[19] H. G. Barrow, J. T. Tenenbaum, R. C. Bolles, and H. C. Wolf, "Parametric correspondence and chamfer matching: two new techniques for image matching," Proc. 5th Int. Joint Conf. Artificial Intelligence,pp. 659–663, 1977.

[20] S. Uchida and H. Sakoe, "A survey of elastic matching techniques for handwritten character recognition," IEICE Trans. on Information & Systems, vol. E88-D, no. 8, pp. 1781–1790, 2005.