A generative learning method for low-resolution character recognition

Hiroyuki Ishida

Abstract

A generative learning method for low-resolution character recognition

Hiroyuki Ishida

In this thesis, a training method for the low-resolution character recognition task is proposed. It is named "Generative learning method," since the training images are generated artificially from an original image. The generative learning method is applied to camera-based character recognition and traffic sign recognition.

Recognition technologies using digital cameras have gained considerable interest in recent years. Provided that these technologies come into practical use, there could be useful applications for camera-equipped devices. However, even with the improvements of digital cameras, the quality of captured images is still insufficient for the recognition in many practical cases.

This work focuses on the training of image degradation characteristics. Recognition methods presented in this thesis are based on the generative learning method in which training images are artificially generated. Conventional approaches used camera-captured images as training data, which required exhaustive collection of the sample images by actual capturing. The proposed generative learning method, instead, allows to obtain these training images based on a small set of actual images. Since the training images need to be generated on the basis of actual degradation characteristics, the estimation step of the degradation characteristics is introduced. This framework is applied to three applications — character recognition, text recognition, and traffic sign recognition.

In the camera-based character recognition application, optical blur and the vibration of hand-held cameras seriously affect the recognition accuracy. The proposed method copes with the optical blur and the motion blur by generating degraded training images. They are generated using a PSF (Point Spread Function) that preserves the actual blur characteristics.

In the text recognition application, segmentation of characters is an unavoidable problem. It is difficult especially in the case where the image resolution is low. The proposed method copes with this problem by introducing a segmentation model to the generative learning method. In addition to characters, patterns of spaces between two adjacent characters are generated and used for accurate segmentation of the characters.

In the traffic sign recognition application, various factors influence the captured

traffic sign images. In the proposed framework, degradation parameters are defined for the simulation of these degradation factors. The distribution of the degradation parameters are estimated from actual images because it can produce appropriate parameters for generating the training images.

Results obtained here have proven that the proposed generative learning method is effective for the applications suffering from various image degradations.

Contents

Chapter 1

Introduction

1.1 Overview

Image recognition technology using digital cameras has gained attention in recent years. Provided that this technology should come to practical use, there could be many potential applications for supporting man-machine interaction. For example, we can use portable cameras as an input device of documents. Also, on-vehicle cameras can recognize road traffic signs automatically. Unlike the conventional scanner-based recognition systems, a camera-based system can be used in less constrained environments. However, even with the improvements of digital imaging devices, the quality of images captured by these cameras are still insufficient for accurate recognition in many practical cases. Image degradations such as blur and reduction of resolution are major problems in recognizing objects in the captured images. In order to solve these problems, it is necessary to seek for approaches to cope with the degradations.

In this thesis, recognition methods for degraded images are proposed. The proposed methods are based on the generative learning method [1] that generates degraded training images sorely from original templates. Historically, such a generation-based approach has been applied to the scanner-based handwriting character recognition application [2, 3]. In contrast to these works, this work focuses on the camera-based recognition of degraded images. A framework for generating the degraded training images is established and applied to the following three applications.

- Recognition of blurred characters captured by portable digital cameras
- Recognition of character-strings captured by portable digital cameras

- Recognition of low-quality traffic signs captured by car-mounted cameras

Camera-based character recognition has gained attention with the growing use of camera-equipped cellular phones [4]. Many studies carried out on scanner-based character recognition such as [5]–[8] are not suitable for camera-based recognition because there are problems peculiar to camera-captured images; the images undergo reduction of resolution and various types of blurring. The generative learning method is employed to solve these problems. In order to cope with the degradation, training images are generated using point spread functions (PSFs). The proposed recognition method takes full advantage of the generative learning method. It estimates the blur parameters also in the recognition step. Characters are identified from their images and the estimated parameters.

Character-string recognition also is a challenging task in the domain of camera-based document analysis. It involves several difficult problems which do not occur in the single character recognition task. Segmentation of characters is one of the important problems, since the adjacent characters tend to touch each other in low-resolution images. The segmentation task is considered to be identical to the recognition of segmentation boundaries in the given character-string image. Accordingly, in the proposed method, character-strings are recognized by features from the characters and also from the inter-character spaces, where the inter-character space consists of the gray-scale features between two adjacent characters. The generative learning method is employed for collecting the training images of the individual characters and the inter-character spaces.

Traffic sign recognition has a potential application for supporting drivers. Many works that have been studied on this research domain were mainly directed toward relatively high-resolution images. In practical cases, however, images captured by car-mounted cameras are in low-resolution. The traffic sign symbols, which give the most important information on the signs, are difficult to be classified due to the degradations. As a solution for this problem, the generation-based approach is applied. Like in the other applications, training images should follow actual degradation factors. The proposed method introduces generation parameters to simulate the degradation factors. In the training step, the parameter distribution is estimated from real samples and used for the generation of the training images.

The common idea of these methods in this thesis is training of degradation characteristics via generated images. It is a reasonable way of preparing the training images because all of them are generated from predefined degradation models. Besides, it eliminates the exhaustive collection of training images. In addition to this

generation framework, recognition methods using the generated training images are proposed and tested in the latter half of this thesis.

1.2 Related works

In this section, works related to this thesis are outlined. The objective of this work is clarified here.

1.2.1 Recognition of low-quality characters

Although character recognition has been studied over several decades, a drastic solution for seriously degraded characters is not found yet [9]. Recently, problems on image degradations have come into a major issue, since camera captured images suffer from this undesired effect [10].

Image restoration is one of the schemes to cope with the degradations [11]. In [12], Hobby proposed a super-resolution method for small characters, and in [13], Li et al. used multiple images to create super-resolution image. Mancas-Thillou et al. used a Teager filter to de-blur low-resolution text images [14].

Point spread function (PSF) is often used to remove optical blur. The compound method proposed by Tsunashima et al. [15] is a simple but effective method to obtain the optical blur PSF. It allows to estimate the PSF by simply averaging multiple captured images. Another method by Fujimoto et al. [16] uses captured images with multiple levels of blur intensities for the estimation of the PSF. Hashimoto and Saito proposed a method for removing shift variant blur [17].

Another form of degradation which should be removed is motion blur. For removing the motion blur, the identification of blur parameters is required [18, 19]. Ben-Ezra et al. proposed a method for removing the motion blur using a PSF [20].

In practical applications, however, restoring an image is not always effective for the character recognition because small characters are difficult to de-blur. This thesis presents a recognition method not by restoration but by the generative learning method. The recognition method is presented in Chapter 3.

1.2.2 Character-string recognition

A problem in camera-based text recognition is the segmentation of characters. The characters in camera-captured images tend to be small, which makes the segmentation task difficult. These characters should be segmented properly for the accurate

recognition of character strings. As has been discussed in many studies [21], however, neither recognition nor segmentation of low-quality character-string images can be done independently. Proper segmentation should be performed for proper recognition, and then again proper recognition should be performed for proper segmentation. One solution is recognition-based segmentation, in which tasks of segmentation and recognition are jointly performed.

In recognition-based segmentation methods [22, 23], hypothesis graph is employed to search an optimal segmentation result. The hypothesis graph is efficiently constructed from recognition results of the individual characters in a string. Meanwhile, the problem of the hypothesis graph is that the segmentation accuracy decreases sharply with a reduction of the image resolution. This is because the boundary of characters tends to be positioned wrongly. To cope with this problem, some local features should be used for the segmentation. A sophisticated method developed by Sun et al. [24] combines several features for segmentation and recognition. In the conventional methods, however, features such as inter-character spaces have not been used effectively. The inter-character space is a region between two adjacent characters. A recognition method using the inter-character spaces is proposed in Chapter 4.

1.2.3 Traffic sign recognition

Technologies for supporting drivers with car-mounted cameras have gained considerable industrial interests in recent years. Many studies have been conducted on pedestrian detection [25], traffic signal recognition [26, 27], road marking recognition [28], vision-based self-position estimation [29, 30], and raindrop detection for the automatic control of wipers [31]. Traffic sign recognition is another important task. If such a recognition system comes into practice, it could support drivers by informing them of the current speed limit, for instance. Also, it could be applicable for periodically updating a road map database [32] used for navigation systems. Two main issues in the traffic sign recognition are detection and classification. Various attempts have been carried out on the detection of traffic signs: edge detection mask [33], hierarchical template [34], shape information [35], and color information [36]. There are methods proposed specifically for circular sign detection [37, 38]. Works [39] and [40] present methods for shape classification. On the other hand, relatively few studies have been conducted on the category classification of extracted signs. Furthermore, most of them are mainly oriented toward high-quality images. In [41], results from high-quality images are preferentially used for avoiding degradations.

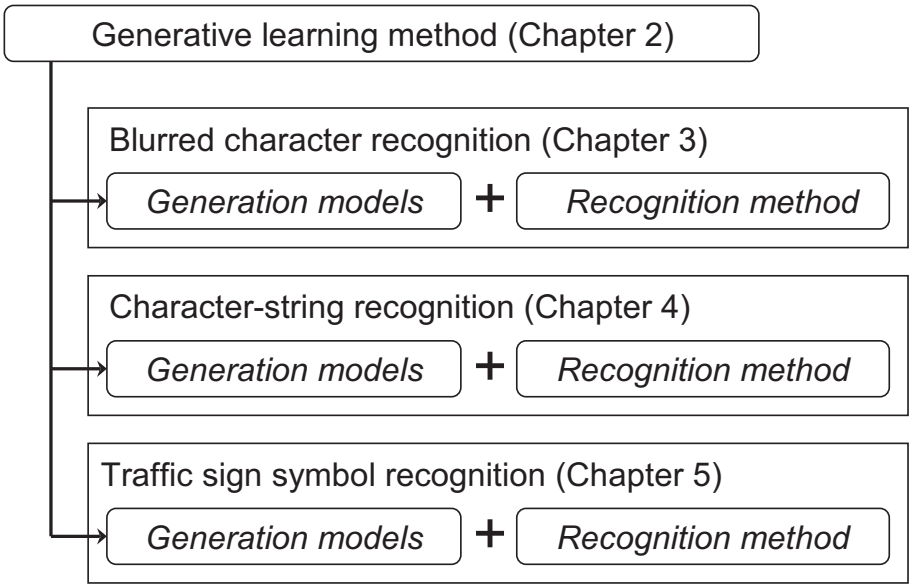


Figure 1.1: Relationships among methods proposed in this thesis. Recognition methods in Chapters 3–5 are based on the generative learning method presented in Chapter 2.

A method specializing in speed sign classification [42] copes with the rotation of traffic sign symbols. However, few studies have focused on the various degradations appearing in camera-captured images. This thesis focuses on the classification of variously degraded traffic sign symbols in Chapter 5.

1.3 Organization of this thesis

This thesis is organized as illustrated in Fig. 1.1.

Chapter 2 outlines a generic framework of the generative learning method. Specifically, generation models based on PSFs are described. Multiple types of PSFs are introduced; an optical blur PSF and a motion blur PSF are used for the generation. Experiments were performed to verify the effectiveness of the PSF-based models.

Chapter 3 describes the application of the generative learning method to camera-based character recognition. Since blurring is the main problem in camera-based recognition systems, a strategy for the effective use of the motion blur is introduced. The recognition method using the eigenspace method [43] is proposed, where the

motion blur parameters are used for the classification in an eigenspace.

Chapter 4 describes the application of the generative learning method to camera-based character-string recognition. For this task, a segmentation model is introduced to the generative learning method. It is used for the simulation of characters appearing in character-string images. The proposed recognition method uses the features of inter-character spaces for improving the recognition accuracy. Training images of the inter-character spaces also are generated from character images.

The framework of the generative learning method can be useful for the classification of traffic sign symbols. Chapter 5 describes another application of the generative learning method to traffic sign symbol recognition. For this task, various generation models are defined corresponding to the actual degradation and deformation models. Together with these models, an optimization algorithm for learning parameter distributions is introduced.

Finally, Chapter 6 concludes this thesis.

Chapter 2

Framework of generative learning method

2.1 Overview

Digital video cameras and camera-equipped cellular phones have come into widespread use in recent years. Recognition technologies using such digital equipments are especially of practical concern. However, the images captured by these cameras tend to be low-resolution and blurred, which has a serious effect on the recognition accuracy. The generative learning method is developed to solve the problem of degradation. It generates artificially degraded pattens, and allows to make classifiers trained by them. Traditionally, this generation-based approach has often been used for learning distorted characters in handwritten character recognition [2, 3]. In this thesis, the generative learning method is applied to camera-based recognition systems.

This chapter describes a framework of the generative learning method for camera-captured images. The effectiveness of the generative learning method is investigated in Section 2.5 through preliminary experiments.

2.2 Generation steps of training images

In general, training images ought to be collected from images taken in the real world. Such a collection-based approach may be the most straight-forward approach to obtain a set of training samples. In many practical cases, however, camera-based collection of a sufficient amount of training images is unrealistic. Let us consider collecting the training images for many categories. In the character recognition

task, for instance, the number of categories tends to be large, and at the same time, printed text may even contain various types of fonts. This diversity of characters makes the collection difficult. Moreover, various conditions that cause respective distortions in captured images should be taken into account. Altogether, collecting training images in various conditions is still difficult in practice.

In contrast, the generative learning method eliminates the exhaustive collection of training images by capturing. All the training images are generated artificially from a smaller number of original images. However, if such artificial generation is performed regardless to realistic models, this method might not be sufficient as a "training" method; it is important to simulate the actual degradation systems. Thus, models are initially defined corresponding to the actual degradation factors. The proposed training method consists of two main parts: (1) estimation of actual degradation systems and (2) generation of training images based on the estimated degradation models. Details of each part are described below.

2.2.1 Modeling of degradation characteristics

Degradation characteristics are needed to be modelled before working on the generation. They can be optical blur, motion blur, segmentation errors, and so on. For each of these models, parameters to control the degree of degradations are defined. Let $\boldsymbol{p}$ be a vector containing parameters from all the models, a training image is generated from the original image using a parameter vector $\boldsymbol{p}$, and then a set of training images is obtained by applying a set of different parameter vectors. These parameters are applicable to all categories, therefore training images for all categories can be obtained. Figure 2.1 illustrates an example of the degradation model, where the parameter σ controls the standard deviation of the Gaussian blur function.

2.2.2 Estimation of degradation characteristics

Once the degradation model is defined, it becomes possible to generate a wide variety of training images. However, they are parameters that actually determine the properties of the generated samples. This is why parameter estimation is necessary for the reproduction of actual degradation characteristics. In the example of the blur model in Fig. 2.1, it is important to estimate the value of σ in general cases.

If the blur function cannot be assumed to be a Gaussian, then σ should be replaced by a point spread function (PSF). This PSF is used in the case where the degradation characteristic of a camera is unknown. The generative learning method

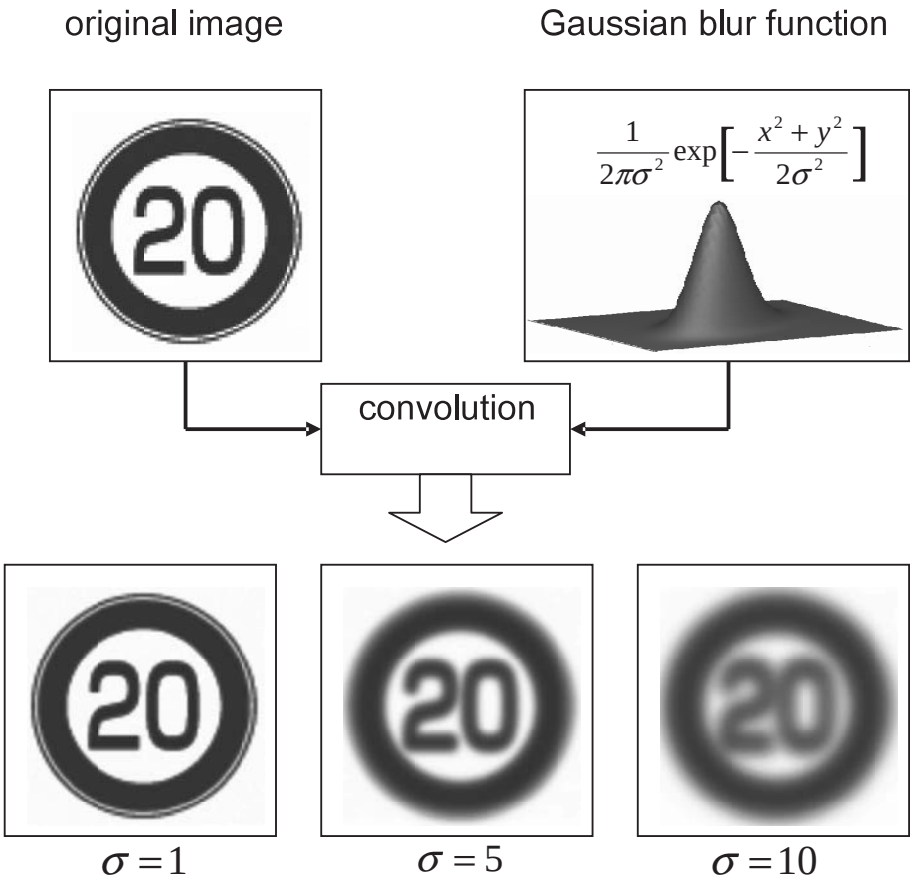


Figure 2.1: The Gaussian blur model. The level of blurring is controlled by a parameter σ.

proposed in this thesis basically simulates the blur characteristics by means of the PSF.

2.3 Application of the generative learning method

In this thesis, the generative learning method is applied to (1) camera-based character recognition, (2) camera-based character-string recognition, and (3) traffic sign symbol recognition. Though all these applications have unique problems that could be worked on via different approaches, there is a common problem that the targets are low-quality images caused by various degradations. The generation scheme

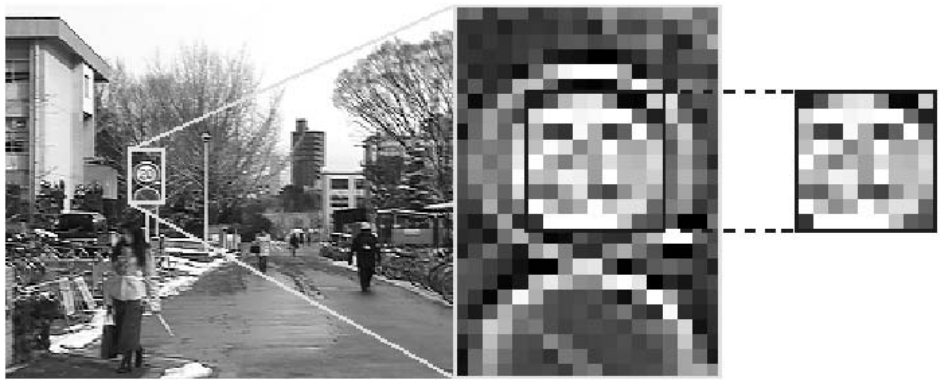


Figure 2.2: A symbol extracted from an actual image taken by a digital video camera.

introduced in this chapter is applicable to all the camera-based recognition tasks.

2.4 Generation by PSF

As discussed in 2.2.1, degradation models for camera-based recognition systems are defined. The following three degradation factors are modeled.

- Optical blur (Fig. 2.3 (b))
- Motion blur due to the movement of the camera (Fig. 2.3 (c))
- Reduction of the image size by sampling (Fig. 2.3 (d))

These factors shown above are characterized by means of the following functions and a parameter.

- Optical blur PSF
- Motion blur PSF
- Resolution parameter

This section describes how a degraded image is generated from an original image using the PSFs. The flow of the process is presented in Fig. 2.4. The generative

(a) (b) (c) (d)

Figure 2.3: Degradation of a character: (a) original image without any degradation, (b) optically blurred image, (c) motion-blurred image, (d) degraded low-resolution image.

learning method is composed of the estimation step of the optical blur PSF and the generation step of the training images. The optical blur PSF is detailed in 2.4.1. The motion blur PSF is detailed in 2.4.2. The resolution transformation model is detailed in 2.4.3. The generation step of the training images is described in 2.4.4.

2.4.1 Optical blur PSF

Generally, identification of blurred characters is a hard task. Image restoration is one approach to improve the recognition accuracy. However, de-blurring of the characters is another hard task. Instead of that, the proposed method generates templates with which the optical blur PSF is convoluted. This optical blur PSF is considered as an averaged blurring filter. We can consider that it preserves the blur characteristics peculiar to the camera.

The estimation of the PSF is accomplished by the compound method [15]. It is simple but has an ability to estimate the optical blur PSF just by averaging multiple captured images. The process of the compound method is as follows.

First, a binary image for the PSF estimation is required. It is printed on a paper as illustrated in Fig. 2.5. An image including texts is desirable for the PSF estimation, because it contains various directions of strokes and edges.

Next, a degraded version of the image is captured onto the computer again. We need to capture it by the camera which is used also for the recognition. The original image is aligned to the captured image by size regularization and translation of the location, since the size of these images needs to be the same. To make use of the compound method, a sufficient number of images are required. It is reasonable to use sequential frames in the captured video stream.

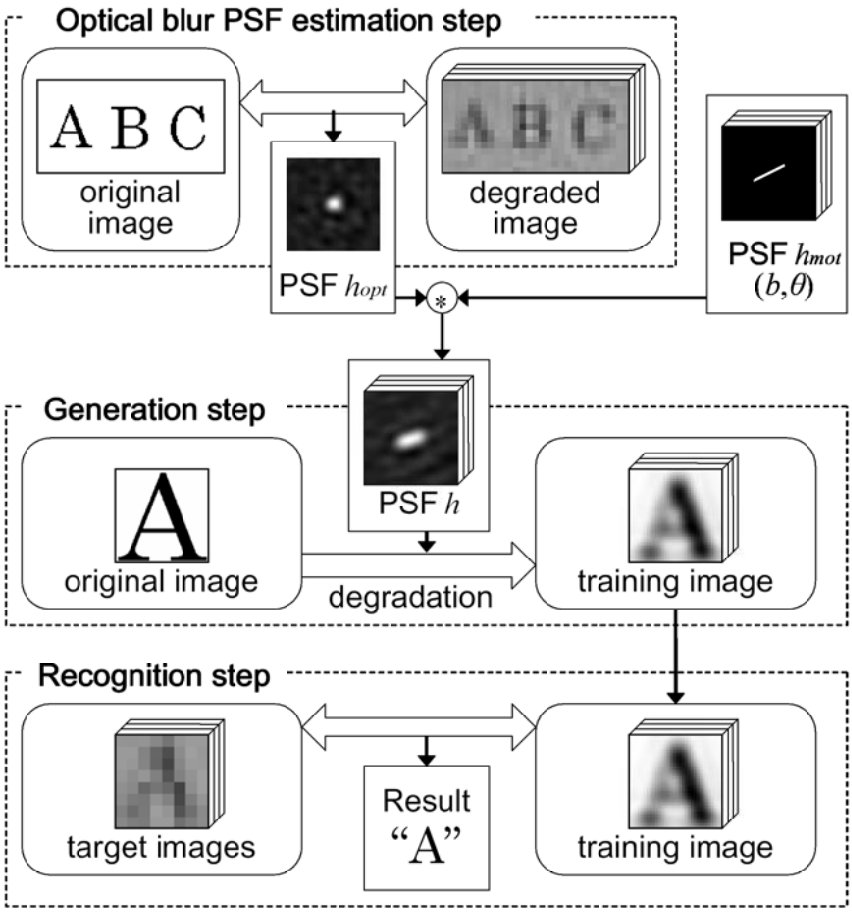


Figure 2.4: Flow of the character recognition method using generative learning method.

Then finally, the optical blur PSF is estimated. The estimation process is described briefly. Assume that the image degradation by the optical blur PSF is represented as

$$g_i(x, y) = f(x, y) * h_{opt}(x, y) + n(x, y), \tag{2.1}$$

where $f(x, y)$ is the original image, $g_i(x, y)$ is the optically blurred image of $f(x, y)$, $h_{opt}(x, y)$ is the optical blur PSF that we need, and $n(x, y)$ is the noise function. Applying two-dimensional Fourier transformation to this equation, we obtain

$$G(u, v) = F(u, v)H_{opt}(u, v) + N(u, v). \tag{2.2}$$

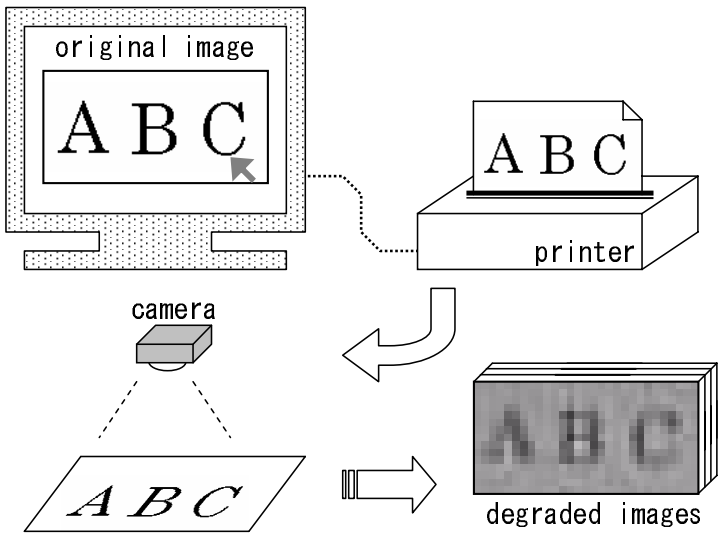


Figure 2.5: Flow of the preparation for PSF estimation.

It is rearranged to

$$H_{opt}(u,v) = \frac{G(u,v)}{F(u,v)} - \frac{N(u,v)}{F(u,v)}. \tag{2.3}$$

Since the noise component is unknown, $h_{opt}(x,y)$ cannot be estimated from a single $g_i(x,y)$. The compound method averages multiple blurred images to restrain this noise. Assuming that we have I images $g_i(x,y)$ $(i = 1, 2, \cdots, I)$, the optical blur component in spatial frequency $\hat{H}_{opt}(u,v)$ is estimated as

$$\hat{H}_{opt}(u,v) = \frac{1}{I}\sum_{i=1}^{I}\frac{G_i(u,v)}{F(u,v)} - \frac{1}{I}\sum_{i=1}^{I}\frac{N_i(u,v)}{F(u,v)}. \tag{2.4}$$

Provided that I is large enough, the second term of this equation converges to 0 because no relation exists among the noise components $N_i(u,v)$ of each image, and then

$$H_{opt}(u,v) \approx \frac{1}{F(u,v)}\frac{1}{I}\sum_{i=1}^{I} G_i(u,v). \tag{2.5}$$

The optical blur PSF $h_{opt}(x,y)$ is obtained from an inverse Fourier transform of $H_{opt}(u,v)$. Figure 2.6 shows some examples of PSFs estimated from digital video

camera, digital camera, and camera equipped in cellular phones. Because of the difference of the optical characteristics, the wave-shapes differ one from another.

2.4.2 Motion blur PSF

In addition to the optical blur, the motion blur is considered as a major reason causing the image degradations. The motion blur is added artificially also by means of a PSF. For convenience, both speed and orientation of the motion blur appearing on one frame are assumed to be constant. This assumption allows to employ the motion blur model proposed by Potmesil [44].

A blurred image $z_2(x, y)$ is generated from the image $z_1(x, y)$ with a blur extent parameter b and a blur angle parameter θ $(0 \leq \theta < \pi)$ by,

$$z_2(x, y) = \int_{-\frac{1}{2}}^{\frac{1}{2}} z_1(x - bt\cos\theta, y - bt\sin\theta)dt. \tag{2.6}$$

This operation can be simplified in the form of a convolution with a motion blur PSF $h_{mot(b,\theta)}(x, y)$ using the two-dimensional Fourier transformation. The blur component is separated from the term $z_1(x, y)$ as

$$H_{mot(b,\theta)}(u, v) = \frac{\sin\left[\pi b(u\cos\theta + v\sin\theta)\right]}{\pi b(u\cos\theta + v\sin\theta)}. \tag{2.7}$$

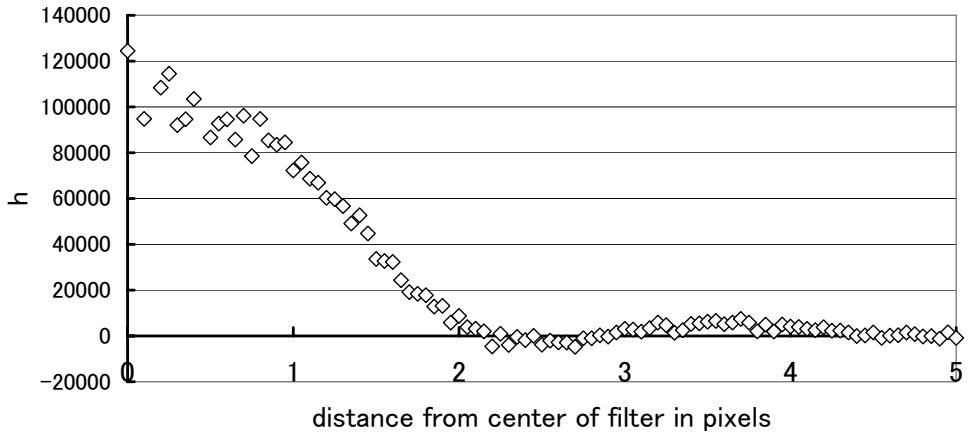
Consequently, Eq. (2.6) is replaced by

$$z_2(x, y) = z_1(x, y) * h_{mot(b,\theta)}(x, y) \tag{2.8}$$

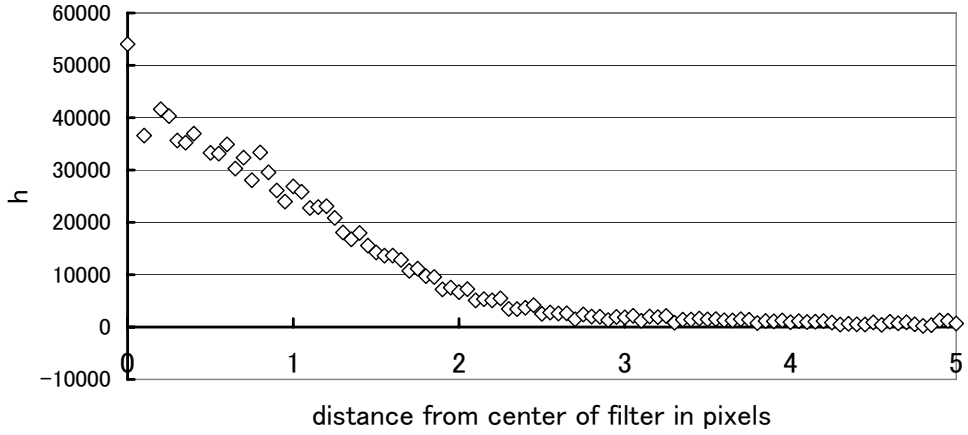
and $h_{mot(b,\theta)}(x, y)$ is obtained by inverting $H_{mot(b,\theta)}(u, v)$. Unlike the optical blur PSF $h_{opt}(x, y)$, this $h_{mot(b,\theta)}(x, y)$ is determined by given two parameters.

2.4.3 Resolution transformation

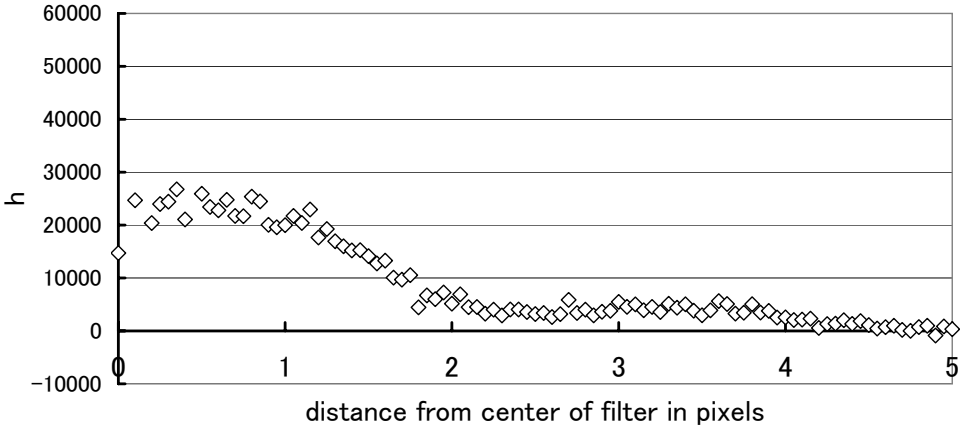
Here a degradation parameter d is introduced. It is identical to the expansion rate of a PSF filter. If $d = 0$, the spatial resolution of the generated image is equivalent to that of the original image, while if $d = 1$, the generated image is identical to an image given by the convolution with a PSF.



(a) Digital video camera (SONY DCR-PC105)

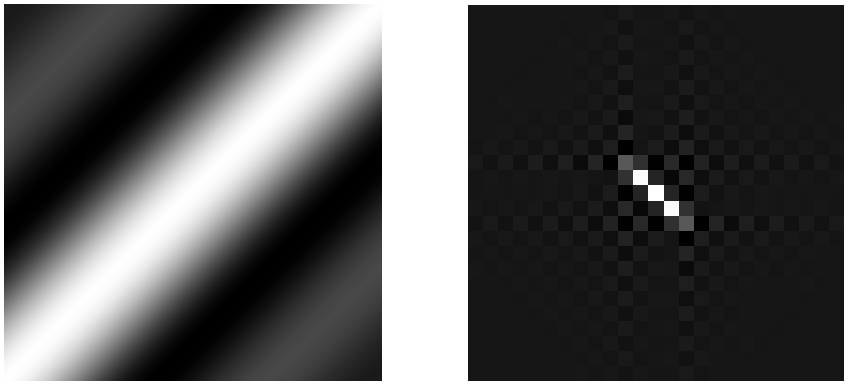


(b) Digital camera (Panasonic DMC-FX9)



(c) Camera equipped in a cellular phone (NTT DoCoMo F901i)

Figure 2.6: Optical blur PSFs. The horizontal axis indicates the distance from the center of a PSF filter $h_{opt}(0,0)$.



(a) $H_{mot(b,\theta)}(u, v)$ (b) $h_{mot(b,\theta)}(x, y)$

Figure 2.7: Motion blur PSF ($b = 5, \theta = 0.25\pi$). (a) shows the parameters of the PSF in spatial frequency. (b) shows the $h_{mot(b,\theta)}(x, y)$ obtained from an inverse Fourier transform of $H_{opt}(u, v)$.

2.4.4 Generation of training images

Training images are generated using the estimated optical blur PSF $h_{opt}(x, y)$ and the parameters introduced above. The parameters are described in a vector form as

$$\boldsymbol{p} = (h_{opt}, b, \theta, d). \tag{2.9}$$

Note that the optical blur PSF $h_{opt}(x, y)$ is included in the parameter vector to keep the notation uncluttered.

Figure 2.8 shows the generation process by the PSFs. Let $\mathrm{x}_{\boldsymbol{p}}^{(c)}(x, y)$ be a training image generated from category c's original image $f^{(c)}$ by a parameter vector $\boldsymbol{p}$. This image is given by

$$\mathrm{x}_{\boldsymbol{p}}^{(c)}(x, y) = \sum_{i,j} h(i, j) f_{seg}^{(c)}(x - id, y - jd), \tag{2.10}$$

where

$$f_{seg}^{(c)}(x - x_o, y - y_o) = f^{(c)}\left(\frac{x}{a} + \Delta x, \frac{y}{a} + \Delta y\right) \tag{2.11}$$

and the united PSF $h(x, y)$ is calculated by

$$h(x, y) = h_{opt}(x, y) * h_{mot(b,\theta)}(x, y). \tag{2.12}$$

Examples of the generated training images are shown with corresponding values of $\boldsymbol{p}$ in Fig. 2.9.

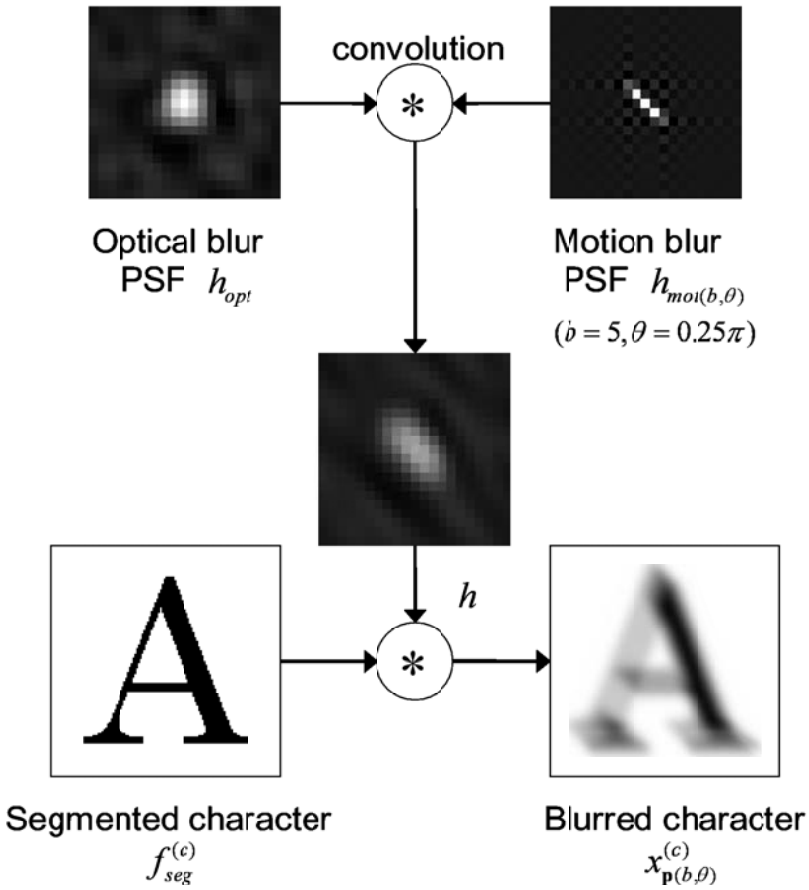


Figure 2.8: Generation of a training image using the estimated PSFs.

2.5 Preliminary experiments

Preliminary experiments were conducted to evaluate the performance of the generative learning method using PSFs. Other than the PSFs, various degradation models were compared in order to examine their applicability to low-resolution character recognition applications. In the experiment in 2.5.1, the robustness against resolution reduction is tested. In the experiment in 2.5.2, the robustness against camera movement is tested. These preliminary experiments focus on the generation stage of training images. Recognition methods using the generative learning method will be presented and experimentally tested in Chapters 3–5.

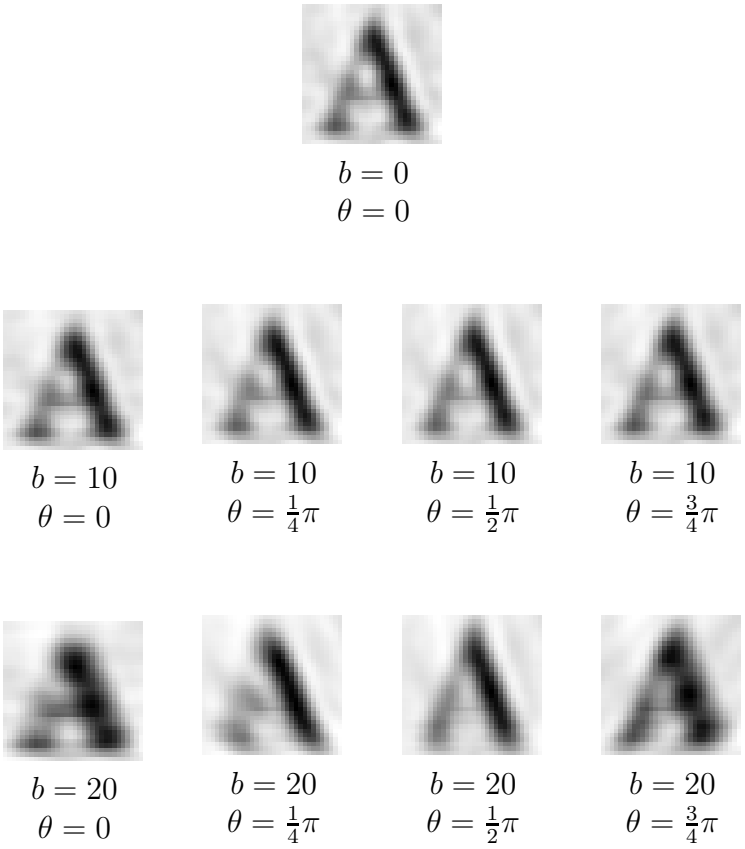


Figure 2.9: Examples of the generated images for category "A" ($\Delta x = \Delta y = 0$, $a = 15/16$, $d = 2$). Motion blur parameters b and θ are changed as shown under each image.

2.5.1 Performance on low-resolution images

An experiment was conducted to evaluate the performance of the optical blur PSF used in the generative learning method. For this purpose, the following four methods were compared.

(Method A) Matching to an original image

This is the simplest method, because it does not use any degradation model. Original character images are normalized in size to 32×32 pixels and used directly as training images. A single training image is used for each category. Unlike the other methods below, the normalized coefficient between a training image and an input image is evaluated as the similarity in the recognition step.

(Method B) Resolution transformation only

Training images are low-resolution characters (8×8 – 32×32 pixels) generated from the original character images. The generation of the training images is performed by reducing the resolution of the images without using the PSFs. Twenty-five training images are generated for each category. In order to use the subspace method for recognition, the size of the training images is normalized again to 32×32 pixels by the nearest-neighbor interpolation (method B1) or the bilinear interpolation (method B2).

(Method C) Generative learning method using a Gaussian blur function

Training images are blurred images (32×32 pixels) generated by the convolution with a two-dimensional Gaussian blur function instead of the PSFs. The standard deviation of the Gaussian blur is changed from 0.5 to 10.0 by 20 steps, which gives 20 training images for each category.

(Method D) Generative learning method using an optical blur PSF

Training images (32×32 pixels) are generated using the optical blur PSF as described in 2.4.1. The optical blur PSF is estimated for each camera listed in Table 2.1. The camera distance is set as presented in Table 2.2. In the generation stage, the degradation parameter d is changed from 0.05 to 1.00 by 20 steps, which gives 20 training images for each category.

In methods B–D, the subspace method [45] and multiple frame integration [46], which shall be detailed in Chapter 3, were used for calculating similarities between training images and an input image. A ten-dimensional subspace is constructed from the generated images for each category. Examples of the generated images are presented in Fig. 2.10.

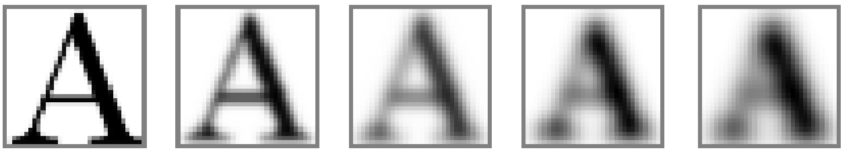
Test images containing 62 different characters (A–Z, a–z, 0–9) were captured 50 times with different types of cameras listed in Table 2.1. The characters were printed with an alphanumeric 'Century' font. Tables 2.3 shows the relation between camera distance and the average size of the test images. After segmentation, the size of all characters was normalized to 32×32 pixels. Figure 2.11 shows some examples of the test images.

Recognition rates are presented in Fig. 2.12 and in Fig. 2.12. Method D, which uses the optical blur PSF, exhibited high recognition rates compared with the other methods. This result shows that the optical blur PSF was effectively used for the

Method A
Matching to an original image

Method B1
Resolution transformation only

Method B2
Resolution transformation only



Method C
Generative learning method using a Gaussian blur function

Method D
Generative learning method using an optical blur PSF
(PSF of Digital camera)

Figure 2.10: Training images for each method.

Table 2.1: Specification of the cameras.

Camera	Resolution	Frame rate
DV camera	720×480pixels	30 fps
Digital camera	640×480pixels	30 fps
Phone camera	162×220pixels	7.5 fps

Table 2.2: Camera distance in the phase of PSF estimation.

Camera	Distance
DV camera	70 cm
Digital camera	70 cm
Phone camera	32 cm

generation of the degraded training images. The recognition rates of the methods B1 and B2 were almost comparable where the size of the target characters was over 10×10 pixels. However, their performance was not sufficient for blurred characters captured by the digital camera, regardless of the normalization methods. Method C using the Gaussian blur function was also effective against strong blur, although it did not outperform method D where the camera distance was not large.

2.5.2 Performance on blurred images

An experiment was conducted to evaluate the usefulness of the motion blur PSF used in the generative learning method. For this purpose, the following two methods were compared.

(Method I) Generative learning method using the optical blur PSF

The estimated optical blur PSF h_{opt} is used. Changing the degradation parameter d by 480 steps ($d = \frac{1}{240} \times 1, \frac{1}{240} \times 2, ..., \frac{1}{240} \times 480$), training images ($32 \times 32$ pixels) are generated.

(Method II) Generative learning method using the optical blur PSF and the motion blur PSF

Table 2.3: Size of characters in pixels.

(a) DV camera and digital camera

Distance	22 cm	35 cm	50 cm	60 cm	70 cm
DV camera	16×16	10×10	7×7	6×6	5×5
Digital camera	17×17	13×13	10×10	9×9	8×8

(b) Camera equipped in a cellular phone

Distance	20 cm	32 cm
Camera equipped in a cellular phone	7×7	5×5

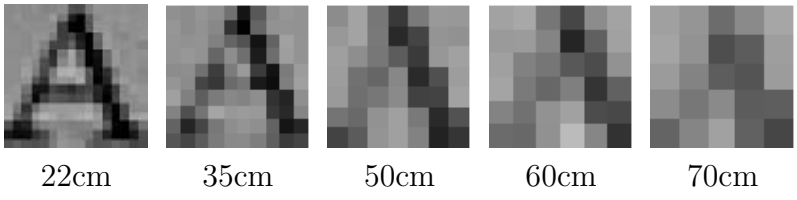
Both the estimated optical blur PSF h_{opt} and the motion blur PSF $h_{mot(b,\theta)}$ are used. The degradation parameter d is changed by 8 steps ($d = 0.25, 0.50,$ ...,2.00), the blur extent parameter b by 5 steps ($b = 0.0, 0.5$,...,2.0), the blur angle parameter θ by 12 steps ($\theta = 0, \pi/12, ..., 11\pi/12$), which results in generating 480 training images (32×32 pixels), where the number is the same as in method I.

In this experiment, the digital camera in Table 2.1 was used. The images used for the optical blur PSF estimation were taken from a distance of 35cm. As in the previous experiment, the subspace method [45] was used for the recognition.

Test images were captured by six unexperienced persons. These persons were told to capture the test images for a period of 20 seconds (600 frames), keeping the camera as stable as possible. Some examples of the test images are presented in Fig. 2.13. Along with the images, camera speed v_b was calculated from 10 successive frames. Averaging shifts of the location during 10 frames, the camera speed of the i-th frame is given by

$$v_b = \frac{1}{9} \sum_{j=i-4}^{i+4} \sqrt{(w_j - w_{j-1})^2 + (h_j - h_{j-1})^2} \tag{2.13}$$

using the location of the target character (w_j, h_j) in the image. This camera speed was used to investigate the relationship between the motion blur and the recognition accuracy.

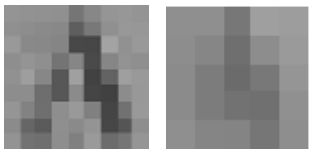


22cm 35cm 50cm 60cm 70cm

(a) Digital video camera

22cm 35cm 50cm 60cm 70cm

(b) Digital camera



20cm 32cm

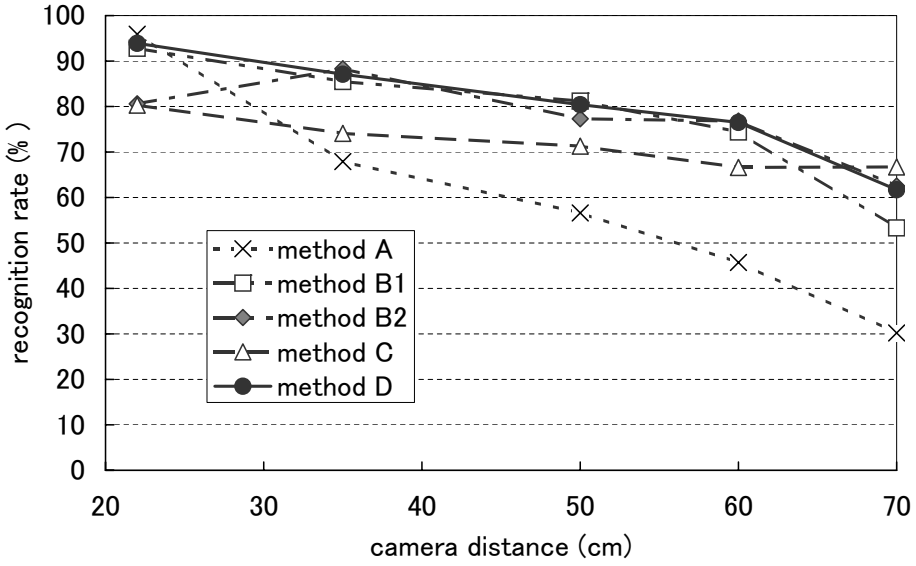
(c) Camera equipped in a cellular phone

Figure 2.11: Test images according to the distance.

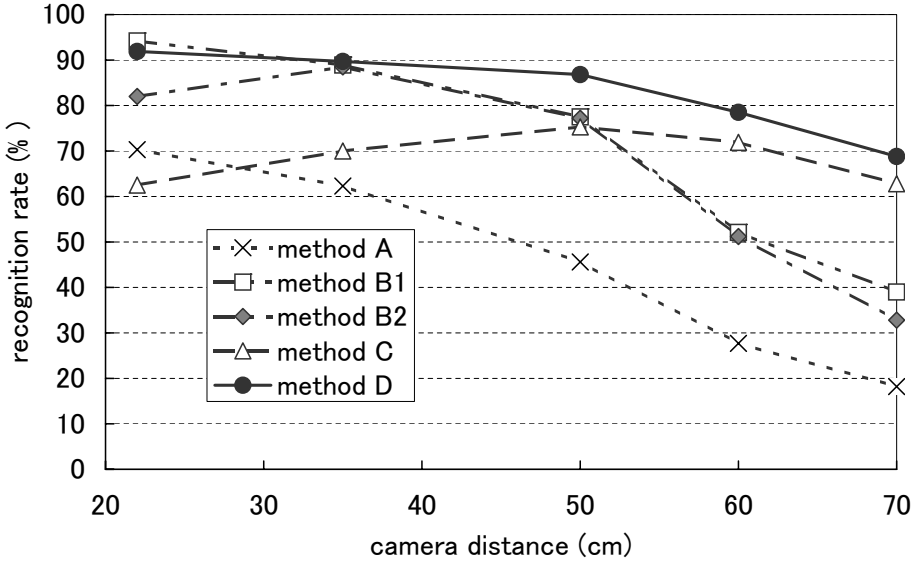
The result is presented in Fig. 2.14, where the horizontal axis of the graph shows the camera speed of the test images. Method II outperformed method I while $0.5 \leq v_b \leq 2.5$, indicating that the use of the motion blur PSF is effective in the presence of a moderate camera movement. The distribution of the calculated camera speed is presented in Fig. 2.15. Since the camera speed v_b concentrated within $0.5 \leq v_b \leq 2.0$, it can be stated that the motion blur PSF is useful for camera-based character recognition systems.

2.6 Summary

In this chapter, a generative learning method was introduced. The generative learning method is a synthesis-based approach for acquiring the training images for the recognition. It allows to cope with the problems related to a camera-based recog-

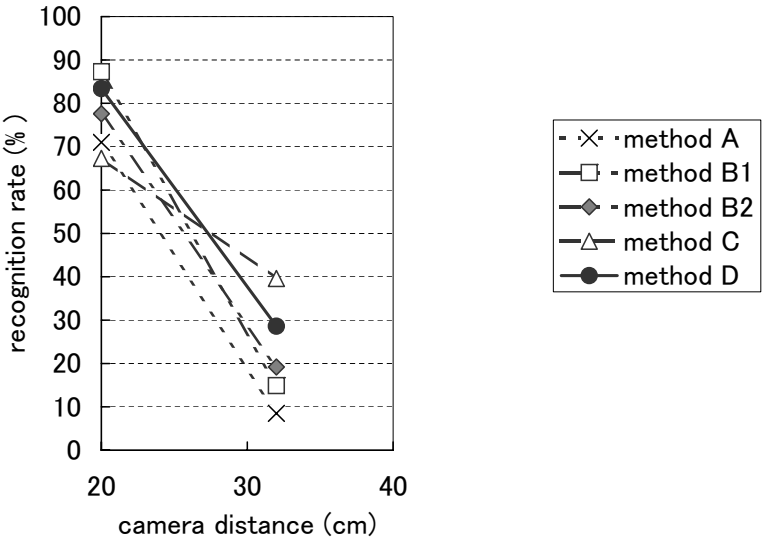


(a) Digital video camera



(b) Digital camera

Figure 2.12: Recognition results.



(c) Camera equipped in cellular phone

Figure 2.12: Recognition results.

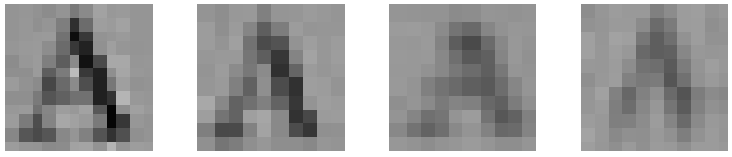


Figure 2.13: Test images undergoing motion blur. Camera speed v_b (pixel / frame) is shown below.

nition system by means of simulation of the degradation factors. As generation models, two types of PSF are defined: (1) the optical blur PSF describes static degradation factors peculiar to the optical system; (2) the motion blur PSF, which is controlled by two motion parameters, describes the degradation effects caused by a camera movement. The effectiveness of the generative learning method was examined by experiments using three types of cameras. The results show that the use of these PSFs in the generation step contributes to higher-quality recognition of low-quality images undergoing resolution reduction and blurring. In case of apply-

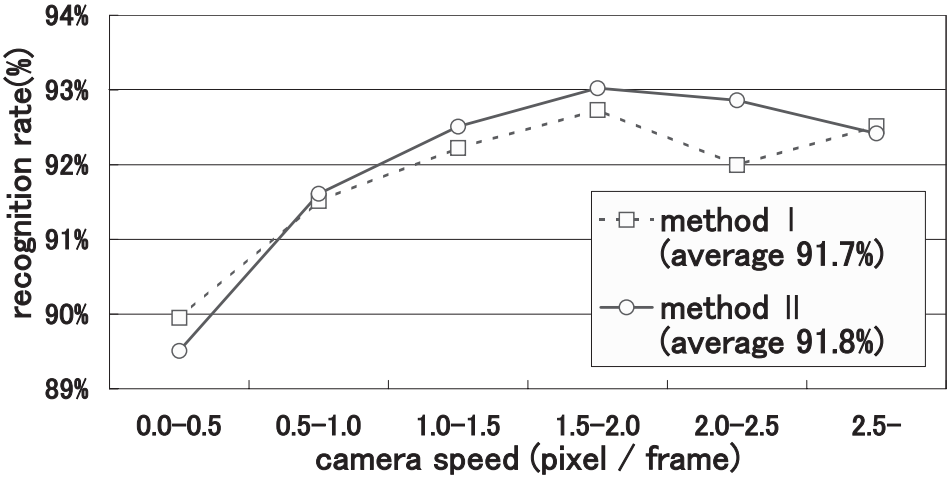


Figure 2.14: Recognition rates depending on the camera speed. The average recognition rates are also shown.

ing the generative learning method to actual recognition tasks, it is important to examine various models that are not limited to the PSFs introduced in this chapter. Appropriate generation models should be employed depending on the application.

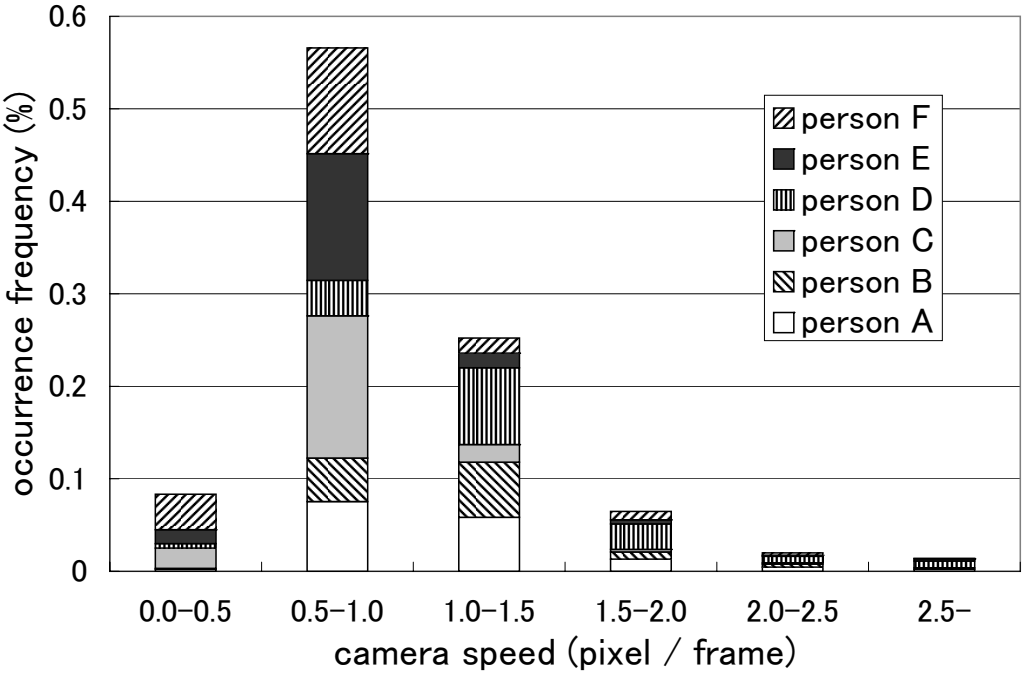


Figure 2.15: Distribution of the camera speed.

Chapter 3

Recognition of low-quality characters

3.1 Overview

Character recognition technologies using portable digital cameras have gained attention in recent years in proportion to the diffusion of portable digital imaging devices. As described in the previous chapter, image degradation is an unavoidable problem peculiar to camera-based character recognition. This problem becomes more serious when a photographer backs the camera away from the target document, trying to capture a larger part of it. Even image restoration techniques are not effective if the characters are so small that their stroke width falls below 1 pixel. This chapter presents a recognition method that does not need any restoration. It instead, copes with the degradations by the generative learning method.

In the previous chapter, models for generating training images were presented. Results from the preliminary experiments have proven that the proposed degradation models can be used effectively as generation models. However, the use of generation models was limited to the training step. Input images were compared only with subspaces constructed from the training images. The classification was performed regardless of the individual training images with various degradation parameters. In contrast, this chapter focuses on the recognition step. The method proposed in this chapter uses the subspace in the first step of recognition, but in the second step, the input images are directly compared with the generated images. In order to use the generation parameters effectively for the recognition, the eigenspace method [43] is employed.

Figure 3.1 illustrates the flow of the proposed method. The training step is based

on the generative learning method, where training images undergoing various speeds and orientations of motion blur are generated. The recognition method consists of two steps. The first step employs the subspace method [45], whose effectiveness for low-resolution character recognition is demonstrated in [46]. However, the subspace method constructs a single subspace from the training images with various levels of speed and orientations of blur, which often yields misclassification among structurally similar characters. The eigenspace method [43] is more effective for such characters, since the similarity to each training image is evaluated. A reclassification process based on the eigenspace method is employed in the second step to improve the recognition accuracy of such characters. This second step reclassifies characters by effective use of the motion blur. For this purpose, motion blur parameters are estimated from camera motion; the similarity between the input characters and training images generated using the corresponding motion blur parameters is evaluated in the recognition step.

This chapter is organized as follows: The generation models and the parameters used for the character recognition are introduced in Section 3.2. The first step of the recognition using the subspace method is detailed in Section 3.3. The second step of the recognition using the eigenspace method is detailed in Section 3.4. The performance of the proposed recognition method is demonstrated through an experiment in Section 3.5. Section 3.6 summarizes this chapter.

3.2 Generative learning method in character recognition

3.2.1 Generation models

Four generation models are defined along with parameters that control the degradation level. Three of them are the same as introduced in Chapter 2. Additionally, a segmentation model is used here to cope with wrongly segmented characters. These models used for this work are listed below.

- Optical blur model (→ 2.4.1)
- Motion blur model (→ 2.4.2)
- Resolution transformation model (→ 2.4.3)

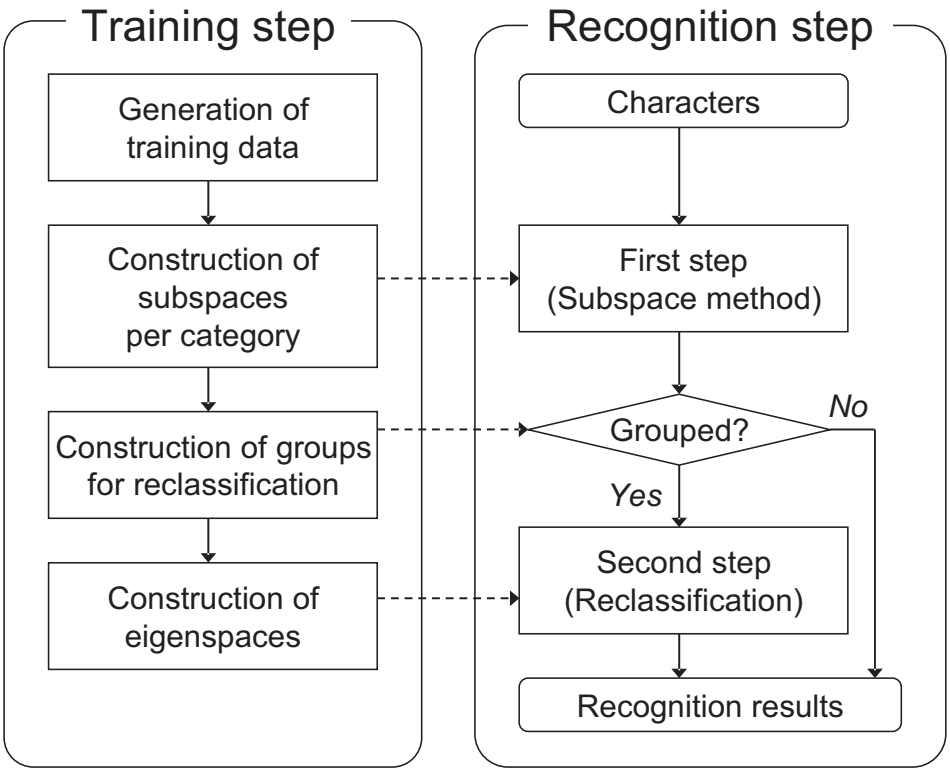


Figure 3.1: Flow of the proposed camera-based character recognition method.

- Segmentation model

  The segmentation model involves translation and expansion of characters in an image. A character area is defined as the minimum square region that contains the whole character. Let (x_o, y_o) be the center of the character area, and l be its side length. A segmented image area is determined by a horizontal gap parameter Δx, a vertical gap parameter Δy, and an expansion rate a as illustrated in Fig. 3.2.

A parameter vector $\boldsymbol{p}$ consisting of the parameters introduced above is described as

$$\boldsymbol{p} = (h_{opt}, b, \theta, \Delta x, \Delta y, a, d). \tag{3.1}$$

Note that the last four parameters are appended here newly to Eq. (2.9). Training images are generated using this parameter vector $\boldsymbol{p}$ including the estimated optical blur PSF $h_{opt}(x, y)$ as described in 2.4.4.

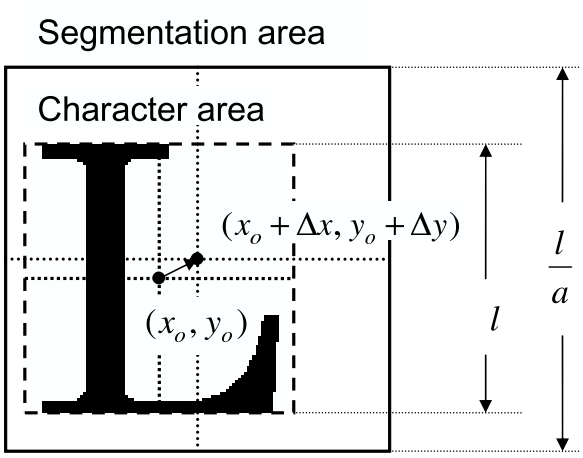


Figure 3.2: Parameters which determine the area for segmentation.

3.3 First step of recognition: Recognition by the subspace method

3.3.1 Construction of a subspace

In the training step, a subspace is constructed from various training images for each category. The constructed subspaces have ability to classify low-quality characters robustly, except for those in some structurally similar categories.

Let $\mathcal{P}$ be a set of N different parameter vectors $\boldsymbol{p}_n$ $(n = 1, 2, \cdots, N)$, where N is the number of training images used for constructing a subspace of a category. N training images are generated from parameter vectors $\boldsymbol{p}_n \in \mathcal{P}$. For each training image $\mathrm{x}^{(c)}_{\boldsymbol{p}_n}$, a vector $\boldsymbol{x}^{(c)}_{\boldsymbol{p}_n}$ is constructed from pixel values of the image as described below. First, $\mathrm{x}^{(c)}_{\boldsymbol{p}_n}$ is converted to a vector $\widetilde{\boldsymbol{x}}^{(c)}_{\boldsymbol{p}_n}$ such that the mean of its elements becomes 0 by

$$\widetilde{\boldsymbol{x}}^{(c)}_{\boldsymbol{p}_n} = \Big[\begin{array}{ccccc} \mathrm{x}^{(c)}_{\boldsymbol{p}_n}(0,0) - \bar{x}^{(c)}_{\boldsymbol{p}_n} & \cdots & \mathrm{x}^{(c)}_{\boldsymbol{p}_n}(w-1,0) - \bar{x}^{(c)}_{\boldsymbol{p}_n} \\ \cdots \quad \mathrm{x}^{(c)}_{\boldsymbol{p}_n}(0,h-1) - \bar{x}^{(c)}_{\boldsymbol{p}_n} & \cdots & \mathrm{x}^{(c)}_{\boldsymbol{p}_n}(w-1,h-1) - \bar{x}^{(c)}_{\boldsymbol{p}_n} \Big]^\top, \end{array} \tag{3.2}$$

where w and h are the width and the height of the image, respectively, and

$$\bar{x}^{(c)}_{\boldsymbol{p}_n} = \frac{1}{wh} \sum_{x=0}^{w-1} \sum_{y=0}^{h-1} \mathrm{x}^{(c)}_{\boldsymbol{p}_n}(x, y).$$

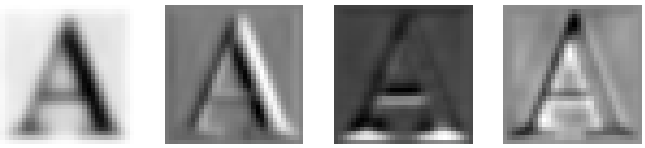


Figure 3.3: Top four eigenvectors for category "A".

Secondly, this vector is normalized to $\boldsymbol{x}_{\boldsymbol{p}_n}^{(c)}$ whose norm is 1 by

$$\boldsymbol{x}_{\boldsymbol{p}_n}^{(c)} = \frac{\widetilde{\boldsymbol{x}}_{\boldsymbol{p}_n}^{(c)}}{\left\|\widetilde{\boldsymbol{x}}_{\boldsymbol{p}_n}^{(c)}\right\|}. \tag{3.3}$$

An auto-correlation matrix $\boldsymbol{Q}_1^{(c)}$ is then calculated by

$$\boldsymbol{Q}_1^{(c)} = \frac{1}{N}\boldsymbol{X}_1^{(c)}\left(\boldsymbol{X}_1^{(c)}\right)^\top, \tag{3.4}$$

where the matrix $\boldsymbol{X}_1^{(c)}$ is represented by a list of the N vectorized training images $\boldsymbol{x}_{\boldsymbol{p}_n}^{(c)}$ as

$$\boldsymbol{X}_1^{(c)} = \left[\boldsymbol{x}_{\boldsymbol{p}_1}^{(c)} \ \cdots \ \boldsymbol{x}_{\boldsymbol{p}_N}^{(c)}\right]. \tag{3.5}$$

Next, the eigenvalues and corresponding eigenvectors of this matrix $\boldsymbol{Q}_1^{(c)}$ are calculated. The eigenvectors are sorted in order of the magnitude of their corresponding eigenvalues, and the largest R_1 ($R_1 < N$) eigenvectors $\boldsymbol{e}_{r_1}^{(c)}$ ($r_1 = 1, 2, \cdots, R_1$) are used for the recognition. Examples of the eigenvectors are illustrated in Fig. 3.3.

3.3.2 Character recognition using multiple frames

In the recognition stage of the subspace method, similarities to the constructed subspaces are calculated; a category which maximizes this similarity is accepted as the recognition result. Yanadume et al. demonstrated that integrating multiple frames improves the recognition accuracy of low-resolution characters [46]. Given M frames of the same character, and letting $\boldsymbol{z}_m$ denote the vectorized and normalized target image in the m-th frame, the recognition result in the first step $\hat{c}_1$ is determined from the inner product to the R_1 eigenvectors $\boldsymbol{e}_{r_1}^{(c)}$ ($r_1 = 1, \cdots, R_1$), by

$$\hat{c}_1 = \arg\max_{\forall c} \sum_{m=1}^{M} \sum_{r_1=1}^{R_1} (\boldsymbol{e}_{r_1}^{(c)\top} \boldsymbol{z}_m)^2. \tag{3.6}$$

3.4 Second step of recognition: Reclassification using blur information

The recognition results obtained in the first step tend to involve misclassification within certain groups of structurally similar categories. The second step attempts to reclassify such dubious results to the correct category using the eigenspace method [43]. The blur parameters estimated from camera motion are used for the matching of characters in this step. This attempt is based on the idea that the blur parameters should supply supplementary information for differentiating structurally similar categories.

3.4.1 Difference between subspace and eigenspace methods

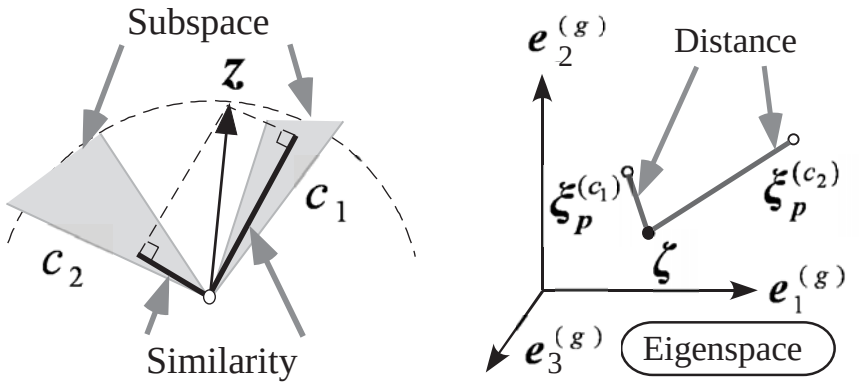
The difference between the subspace method and the eigenspace method is described here.

Figure 3.4 illustrates the recognition schemes of these methods. In the subspace method, an input image is compared with subspaces of all categories and then classified according to similarities to the subspaces. These subspaces are constructed from training images with various degradation parameters. Consequently, the subspace method has a general capability to recognize the degraded images, although the similarities to each training image cannot be evaluated.

Unlike a subspace, an eigenspace is constructed for a set of categories. All the images, regardless of training or input, are projected onto the same eigenspace. The input image is classified to the nearest category in the eigenspace. The eigenspace method has the advantage that a distance to individual training images can be evaluated as a dissimilarity. If we have some parameters which characterize the input image, the eigenspace method allows us to choose the training images used for the distance calculation, whereas the subspace method cannot use parameters in the recognition step.

3.4.2 Grouping similar characters

For each category g, characters that are frequently misclassified to a category g are grouped and described as $\mathcal{G}^{(g)}$. Such groups are organized by applying the first step of recognition to a certain amount of samples. Let $\rho(g|c)$ denote the rate at which a character in category c is classified to category g in the first step. The category c is grouped if $\rho(g|c) \geq \tau$, where τ is a grouping threshold; and of course, g itself also



(a) Subspace method (b) Eigenspace method

Figure 3.4: While the subspace method evaluates similarities to the subspace, the eigenspace method evaluates distances (dissimilarities) to each training image. (a) In the subspace method, the similarity between input image $\boldsymbol{z}$ and the subspace $\{\boldsymbol{e}_r^{(c)}\}$ is defined as $\sum_r (\boldsymbol{e}_r^{(c)\top}\boldsymbol{z})^2$. (b) In the eigenspace method, the distance between an input image $\boldsymbol{\zeta}$ and a training image $\boldsymbol{\xi}_{\boldsymbol{p}}^{(c)}$, both of which are projected on the eigenspace, is defined as $\|\boldsymbol{\zeta} - \boldsymbol{\xi}_{\boldsymbol{p}}^{(c)}\|$.

needs to be a member of $\mathcal{G}^{(g)}$. Therefore $\mathcal{G}^{(g)}$ is given by

$$\mathcal{G}^{(g)} = \left\{\ c \ \middle|\ \rho\,(g|c) \geq \tau \right\} \cup \{g\}. \tag{3.7}$$

3.4.3 Construction of an eigenspace in groups

An eigenspace used for this second step of recognition is constructed in each group. Similar to the step in which the subspaces were constructed, a covariance matrix $\boldsymbol{Q}_2^{(g)}$ of group g is calculated by

$$\boldsymbol{Q}_2^{(g)} = \frac{1}{KN}\boldsymbol{X}_2^{(g)}\big(\boldsymbol{X}_2^{(g)}\big)^\top, \tag{3.8}$$

except that a matrix $\boldsymbol{X}_2^{(g)}$ is represented by all the training images of categories $c_k \in \mathcal{G}^{(g)}$ $(1 \leq k \leq K = |\mathcal{G}^{(g)}|)$ and their mean $\boldsymbol{\mu}^{(g)}$ by,

$$\boldsymbol{X}_2^{(g)} = \left[\widetilde{\boldsymbol{X}}_1^{(c_1)}\ \cdots\ \widetilde{\boldsymbol{X}}_1^{(c_K)}\right], \tag{3.9}$$

where

$$\widetilde{\boldsymbol{X}}_1^{(c_k)} = \left[\boldsymbol{x}_{\boldsymbol{p}_1}^{(c_k)} - \boldsymbol{\mu}^{(g)}\ \cdots\ \boldsymbol{x}_{\boldsymbol{p}_N}^{(c_k)} - \boldsymbol{\mu}^{(g)}\right] \tag{3.10}$$

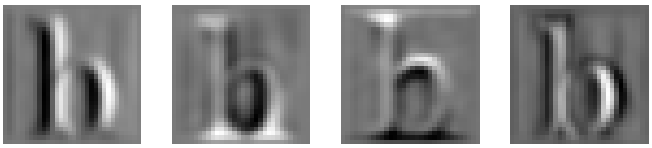


Figure 3.5: Top four eigenvectors for group $\mathcal{G}^{(\text{“h”})} = \{\text{“b”},\ \text{“h”}\}$. These eigenvectors compose an eigenspace that is suitable for differentiating characters “b” and “h”.

and

$$\boldsymbol{\mu}^{(g)} = \frac{1}{KN} \sum_{k=1}^{K} \sum_{n=1}^{N} \boldsymbol{x}_{\boldsymbol{p}_n}^{(c_k)}. \tag{3.11}$$

Next, eigenvalues and corresponding eigenvectors of this matrix $\boldsymbol{Q}_2^{(g)}$ are calculated. The eigenvectors are sorted in order of the magnitude of their corresponding eigenvalues, and the largest R_2 ($R_2 < KN$) eigenvectors $\boldsymbol{e}_{r_2}^{(g)}$ ($r_2 = 1, 2, \cdots, R_2$) are used. Examples of the eigenvectors are illustrated in Fig. 3.5.

3.4.4 Projection of the training images to the eigenspace

All the training images are projected onto the eigenspace as points. The following operation projects category c's training images $\boldsymbol{x}_{\boldsymbol{p}}^{(c)}$ ($c \in \mathcal{G}^{(g)}$, $\boldsymbol{p} \in \mathcal{P}$), and thereby the projected points $\boldsymbol{\xi}_{\boldsymbol{p}}^{(c)}$ in the eigenspace are obtained.

$$\boldsymbol{\xi}_{\boldsymbol{p}}^{(c)} = \left[\boldsymbol{e}_1^{(g)}\ \cdots\ \boldsymbol{e}_{R_2}^{(g)}\right]^{\top} \left(\boldsymbol{x}_{\boldsymbol{p}}^{(c)} - \boldsymbol{\mu}^{(g)}\right) \tag{3.12}$$

Here the vector $\boldsymbol{x}_{\boldsymbol{p}}^{(c)}$ has wh elements (see Eq. (3.2)). Since $\left[\boldsymbol{e}_1^{(g)}\ \cdots\ \boldsymbol{e}_{R_2}^{(g)}\right]^{\top}$ is a matrix composed of $R_2 \times wh$ elements, the feature dimension is reduced from wh to R_2.

3.4.5 Character recognition using blur information

The second step of recognition utilizes blur information obtained from camera motion. A recognition result from the first step is denoted by g. If $|\mathcal{G}^{(g)}| = 1$, there are no other candidates for consideration as a recognition result. Hence the final recognition result also becomes g, whereas if $|\mathcal{G}^{(g)}| \geq 2$, we dismiss the results from the first step once and compute the final result as described below.

1. Projection to the eigenspace

First, the input image is projected onto the eigenspace of group $\mathcal{G}^{(g)}$. An input image in the m-th frame is transformed into a normalized vector and denoted by $\boldsymbol{z}_m$. A projected point $\boldsymbol{\zeta}_m$ corresponding to the image is obtained by

$$\boldsymbol{\zeta}_m = \left[\boldsymbol{e}_1^{(g)} \ \cdots \ \boldsymbol{e}_{R_2}^{(g)}\right]^\top \left(\boldsymbol{z}_m - \boldsymbol{\mu}^{(g)}\right). \tag{3.13}$$

2. Estimation of blur information

   For blur information, the extent and the angle of the motion blur are estimated from a camera motion. Let x_m and y_m represent the location of the target character in the image of the m-th frame. The blur extent parameter $\hat{b}_m$ and the blur angle parameter $\hat{\theta}_m$ are estimated as follows:

$$\hat{b}_m = \sqrt{(x_m - x_{m-1})^2 + (y_m - y_{m-1})^2}, \tag{3.14}$$

$$\hat{\theta}_m = \tan^{-1} \frac{y_m - y_{m-1}}{x_m - x_{m-1}}. \tag{3.15}$$

3. Calculation of similarity

   In the eigenspace method, distances between projected points are calculated for the recognition. The smaller the distance is, the more similar the original images are. Here we need to evaluate the distances of $\boldsymbol{\zeta}_m$ to points $\boldsymbol{\xi}_{\boldsymbol{p}}^{(c)}$ $(b = \hat{b}_m, \theta = \hat{\theta}_m)$ of categories $c \in \mathcal{G}^{(g)}$. However, the estimated $\hat{b}_m$ and $\hat{\theta}_m$ generally may not coincide with parameters $\boldsymbol{p} \in \mathcal{P}$ in the trained set, and more importantly, these estimated values can differ to a certain extent from the actual blur PSF. Accordingly, reference points are selected from training sets with a limited parameter range $\mathcal{B} \subset \mathcal{P}$ given by

$$\mathcal{B} = \left\{ \boldsymbol{p} \in \mathcal{P} \;\middle|\; 0 \le b \le \hat{b} + \Delta b, \ \hat{\theta} - \Delta\theta \le \theta \le \hat{\theta} + \Delta\theta \right\}, \tag{3.16}$$

   assuming that b is not much greater than the estimated $\hat{b}$, and that the difference between θ and $\hat{\theta}$ is not large. The classification is based on the nearest neighbor rule. Figure 3.6 illustrates the classification scheme of the proposed method. For each category, a distance to the nearest reference point is evaluated. The final result $\hat{c}_2$ is computed by

$$\hat{c}_2 = \arg \min_{c \in \mathcal{G}^{(g)}} \sum_{m=1}^{M} \min_{\boldsymbol{p} \in \mathcal{B}} \|\boldsymbol{\zeta}_m - \boldsymbol{\xi}_{\boldsymbol{p}}^{(c)}\|. \tag{3.17}$$

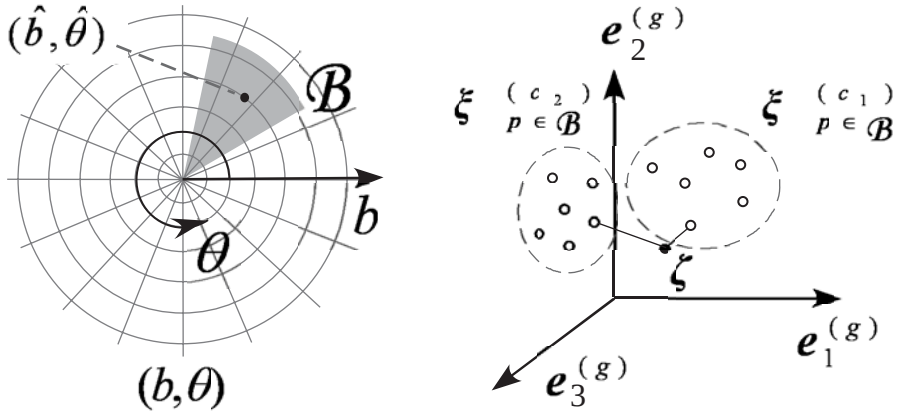


Figure 3.6: Reference points used for the classification (right) are selected from parameter range $\mathcal{B}$ (left) that is restricted by estimated blur parameters.

3.5 Experiments

3.5.1 Conditions

The performance of the proposed method was experimentally tested. The experiment was performed using a digital camera (Panasonic DMC-FX9) that provides the ability to record video with a spatial resolution of 640×480 pixels at the rate of 30 frames per second. As test samples, 62 characters (A-Z, a-z, 0-9 : Century font) printed on a paper were captured. The average size of the printed characters was 5 mm^2. The distance to the paper was 30 cm, and the focal length of the camera was 5.8 cm; the average character size in the captured images was 11×11 pixels. The segmented area of each character was a minimum square that includes the whole character. The anti-blur function of the digital camera was kept off during this experiment.

3.5.2 Training step

First, the optical blur PSF of the camera was estimated. Two hundred images were taken for the PSF estimation from a distance of 30 cm. In the generation step, the parameters were controlled so that the training images should vary. The degradation parameter d was changed in 4 steps ($d = 0.5, 1.0, 1.5, 2.0$), the blur extent parameter b by 11 steps ($b = 0, 2, \cdots, 20$), the blur angle parameter θ by

Table 3.1: Groups (g: $\mathcal{G}^{(g)}$) within which characters are reclassified. If the recognition result in the first step is g, the final result is determined from a category set $\mathcal{G}^{(g)}$. Cases $\tau = 0.05$ and $\tau = 0.10$ gave the same grouping results.

$\tau = 0.01$		$\tau = 0.02$		$\tau = 0.05, 0.10$		$\tau = 0.20$	
L:	{L, t}	S:	{S, 8}	V:	{V, v}	V:	{V, v}
O:	{O, o}	V:	{V, v}	W:	{W, w}	h:	{b, h}
R:	{F, R}	W:	{W, w}	h:	{b, h}	l:	{I, i, l, 1}
S:	{S, 8}	h:	{b, h}	l:	{I, i, l, 1}		
V:	{V, v}	l:	{I, i, l, 1}				
W:	{W, w}	1:	{i, 1}				
h:	{b, h}						
l:	{I, i, j, l, 1}						
1:	{i, 1}						

12 steps ($\theta = 0, \pi/12, \cdots, 11\pi/12$), the expansion rate parameter a by 3 steps ($a = 14/16, 15/16, 1$), and the segmentation parameters Δx and Δy individually by 3 steps ($-a, 0, a$). Changing the parameters as above, $4 \times 11 \times 12 \times 3 \times 3 \times 3 = 14,256$ training images (32×32 pixels) per category were obtained. The original images were also in Century font. The top ten eigenvectors were used for the first step of the recognition ($R_1 = 10$). The number of eigenvectors was determined such that the cumulative contributions of all subspaces were over 97.5%.

Next, the groups used for the reclassification were organized. In the same way as the test data, 300 image sequences composed of 10 successive frames each were taken for the purposes of grouping. The resulting confusion matrix is shown in Table 3.2. Here, five levels of grouping threshold in Eq. (3.7) were tested ($\tau = 0.01, 0.02, 0.05, 0.10, 0.20$). Table 3.1 lists the organized groups in these cases. The eigenspaces were constructed as described in Section 3.4. The value of R_2 in Eq. (3.17) was determined such that the cumulative contribution of eigenvectors was over 80%.

The sets of points, which were used for distance evaluation in the second step of the recognition, were computed by projecting the training images onto the eigenspaces. The parameter range in Eq. (3.16) was set as $(\Delta b, \Delta\theta) = (2, \pi/6)$.

Table 3.2: Confusion matrix used for the category grouping. Values in the matrix represent $\rho(g|c)$. Only the categories c, which satisfy $\rho(g|c) \geq 0.01$, are shown here.

	recognition result g																		
input c	F	I	L	O	R	S	V	W	b	h	i	j	l	o	t	v	w	1	8
F	0.99				0.01														
I		0.04																	
L			1.00																
O				1.00															
R					1.00														
S						1.00													
V							1.00												
W								1.00											
b									0.39	0.61									
h										1.00									
i											0.46		0.51					0.03	
j												0.99	0.01						
l													1.00						
o				0.01										0.99					
t			0.01												0.99				
v							1.00												
w								0.12									0.88		
1													0.31					0.69	
8						0.03													0.97

3.5.3 Comparison with other methods

In order to evaluate the performance of the proposed method, it was compared with the subspace method and with the original eigenspace method.

Recognition results of the subspace method are obtained by Eq. (3.6). The proposed method is equivalent to the subspace method if the second step of the recognition is not employed, namely if $\tau > 1$.

In the original eigenspace method compared here, all the results are obtained by the distance calculation in a single eigenspace. It is also known as the universal eigenspace [43]. The recognition process is basically the same as described the previous section, except that eigenspaces of category groups are replaced by an universal eigenspace. The proposed method is equivalent to the original eigenspace method if all categories g have a group $\mathcal{G}^{(g)}$ consisting of all the categories, namely if $\tau = 0$.

3.5.4 Conditions and recognition results

Three photographic conditions were set for this experiment. Image sequences for the test samples were captured under the following conditions.

Condition A : Still camera on a tripod.

Condition B : Camera held as stable as possible.

Condition C : Camera held by vibrating hand.

Image sequences composed of ten successive frames ($M = 10$) were used for testing. The number of the image sequences for Conditions A, B, and C were 300, 1 736, and 503, respectively. The image sequences for Conditions B and C were captured by six persons. Figure 3.7 shows some examples of the test images. Figure 3.8 shows occurrence rates of the estimated blur extent parameter $\hat{b}$ given by

$$\hat{b} = \frac{1}{M} \sum_{m=1}^{M} \hat{b}_m. \tag{3.18}$$

We can see from the graph that $\hat{b}$ is not always zero even under Condition B. The results are shown in Table 3.3.

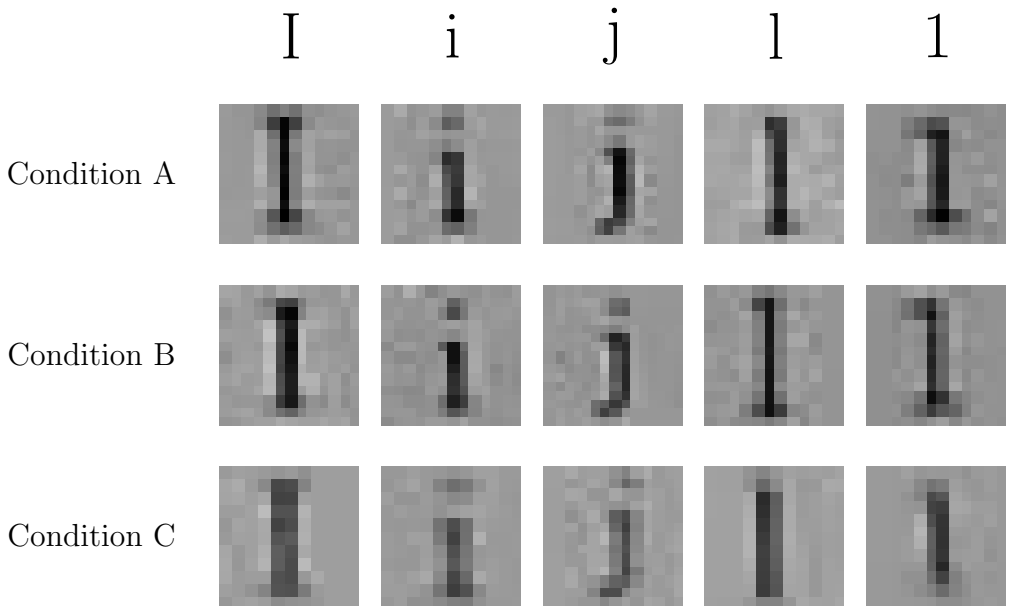


Figure 3.7: Examples of the test images.

3.5.5 Discussion

The availability of the second recognition step is demonstrated by comparing to the results of the subspace method. According to the results, the recognition rates were improved by using the second recognition step. The proposed method was effective particularly under Conditions B and C. This result indicates that some blurred characters, which were misclassified in the first step, were correctly reclassified in the second step. It is also worth remarking that Condition B was more appropriate for the recognition than Condition A. The major reason is that integrating time-varying images by Eqs. (3.6) and (3.17) was effective for recognizing low-resolution characters.

While the usefulness of the second step using the eigenspace method was shown, the recognition rates from the original eigenspace method were low under any conditions. One reason is that the number of categories was too large. An universal eigenspace is not suitable for purposes such as character recognition. The eigenspace is useful if it is constructed among a small number of structurally similar categories and if the image appearance varies depending on parameters. The proposed method uses the eigenspace effectively for the classification within groups of such characters.

Another problem connected with grouping is the determination of threshold τ.

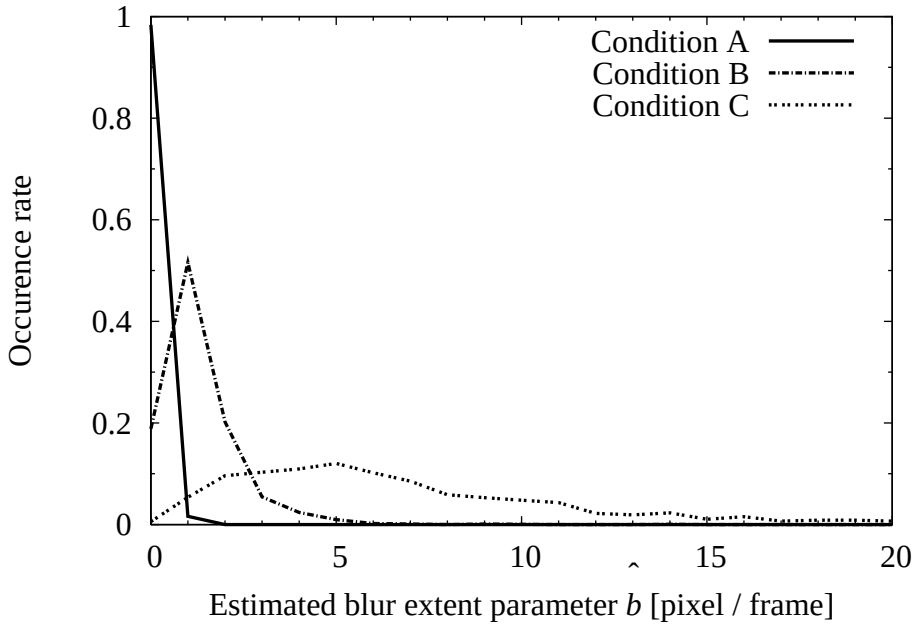


Figure 3.8: Distributions of estimated blur extent parameter $\hat{b}$ by conditions.

As discussed above, desirable grouping contributes to a better performance. If we assume that the recognition performance of the second step is identical to that of the original eigenspace method, its rate is about 80% due to the results in Table 3.3. Accordingly, it can be effective to determine τ such that $\tau \leq 1 - 0.8 = 0.2$. When $\tau = 0.01$, however, the performance under Conditions A and C was lower than the subspace method. Figure 3.9 shows the ratio of characters classified correctly or wrongly in each step. Setting τ too small increased the number of misclassified characters in the second recognition step. We can see from Table 3.2 that the number of groups increases as τ decreases, which can result in over-grouping. These results indicate that the second recognition step is effective for frequently misclassified categories and therefore over-grouping should be avoided.

3.6 Summary

In this chapter, a camera-captured character recognition method is presented. The training images are obtained by the generative learning method. Generation models are the optical blur model, the motion blur model, the resolution transformation model, and the segmentation model. In the recognition stage, the blur parameters

Table 3.3: Recognition rates (%) of characters under various capturing conditions. The proposed method is compared also with the subspace method (SM) and the original eigenspace method (EM). Various grouping thresholds τ were tested.

Method	EM	Proposed method				SM
				$\tau = 0.05$		
Grouping threshold	$(\tau = 0)$	$\tau = 0.01$	$\tau = 0.02$	$\tau = 0.10$	$\tau = 0.20$	$(\tau > 1)$
Condition A	77.22	97.01	97.37	97.39	97.39	97.30
Condition B	77.49	98.38	98.73	98.69	98.69	98.21
Condition C	75.03	93.58	94.29	94.29	94.30	93.76

are estimated from an image sequence. A reclassification step using these parameters was used for reducing classification errors. It was experimentally proved that the effective use of the blur parameters improves the recognition accuracy of blurred characters. Since the training step is based on the generative learning method, this method can be applied easily to characters of any size and font. Evaluating the method's performance under various deformation is a future work, together with improving the reclassification accuracy. Parameters from other models such as a rotation model could also be valuable as supplementary information for the classification.

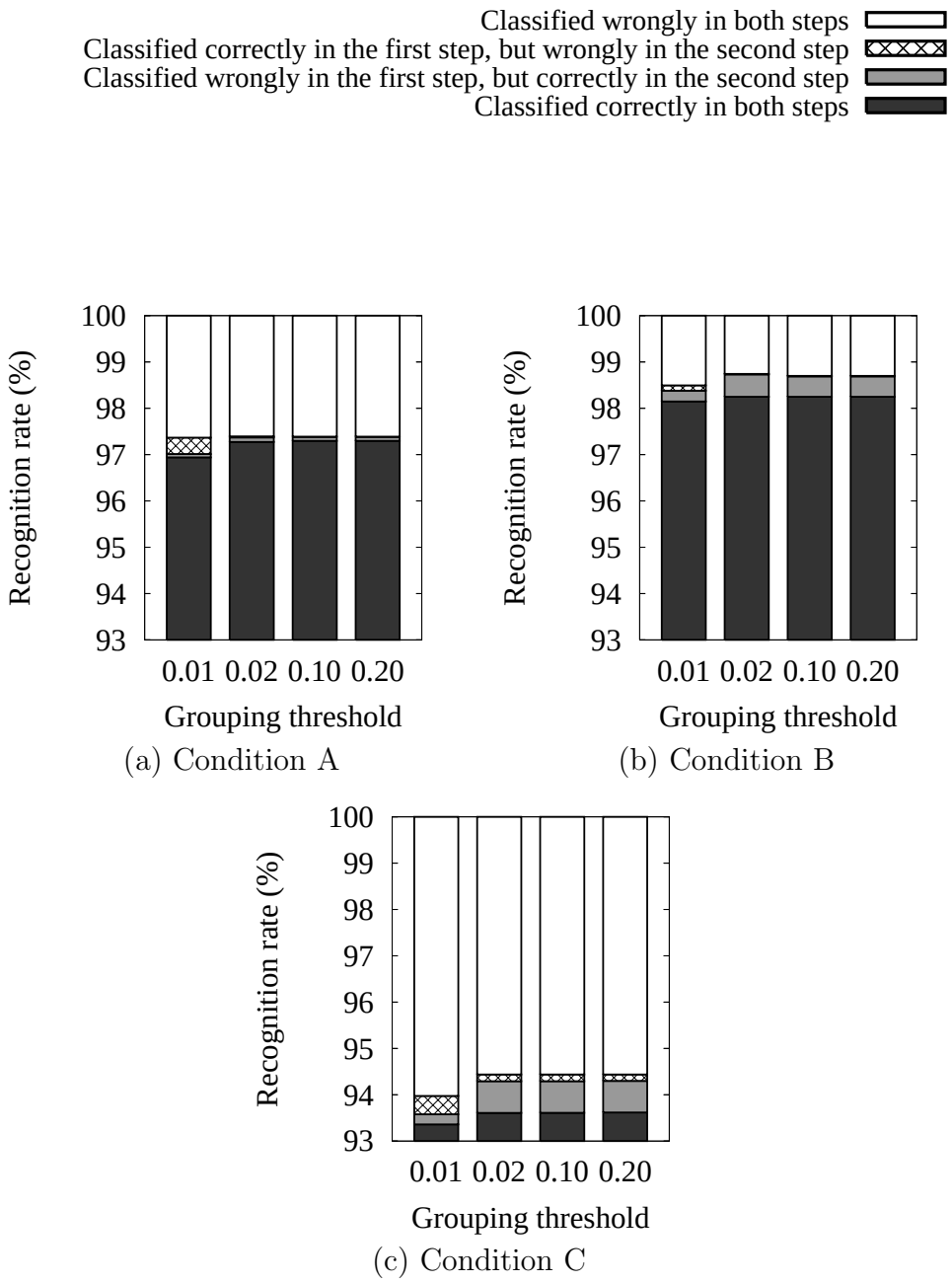


Figure 3.9: The number of misclassified characters in each step. Recognition rates for various levels of τ are compared.

Chapter 4

Recognition of character-strings

4.1 Overview

Throughout the previous chapter, word segmentation was not considered. It is clear, however, that this problem cannot be avoided when considering camera-based document recognition. In this chapter, a method for camera-based character-string recognition is proposed. Since the extraction of character-strings is relatively easy owing to a larger space among them, this method considers character-string images extracted from a document image. It is assumed also that they are rectified [47, 48] or dewarped [49, 50] if such processes are needed.

In order to recognize character-strings, the characters need to be segmented properly. Since the categories of the characters are unknown, it is reasonable to employ a framework of recognition-based segmentation. Conventional recognition-based segmentation methods [22, 23] used a hypothesis graph to search an optimal segmentation result. They worked well on relatively high-resolution character-strings. When they are applied to low-resolution character-strings, however, several practical problems come up. First, the hypothesis graph is sensitive to image degradation. A plausible hypothesis is not obtained if characters contained in an image are even slightly different from their original templates. In the proposed method, the generative learning method is adopted to cope with this problem. Training images of all categories are generated and used as templates. Second, the construction of the hypothesis graph is computationally infeasible if a large number of templates are needed for matching. Against the second problem, the subspace method is combined with the hypothesis graph in the proposed method, since the template matching by the subspace method is fast and robust to degradation.

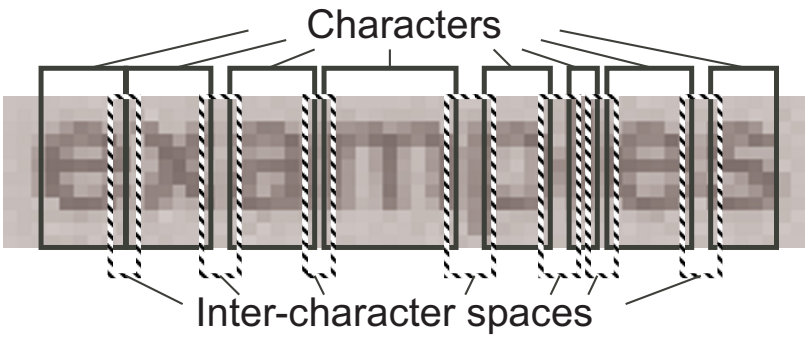


Figure 4.1: Two types of features available in character-string recognition

As discussed so far, the approaches in Chapter 3 can be useful also for the character-string recognition task. Nevertheless, the correct segmentation of characters is still difficult because the boundary of adjacent characters tends to be positioned ambiguously. In addition to the features for individual character recognition, features for boundary identification need to be used. The proposed method focuses on the use of inter-character spaces which are shown in Fig. 4.1, where their features are evaluated more positively and effectively by recognizing them simultaneously with characters

This chapter is organized as follows. The generation stage of the templates is described in Section 4.2. The construction of the hypothesis graph is described in Section 4.3, together with the application of the subspace method. The character-string recognition using inter-character spaces is described in Section 4.4. Results are presented in Section 4.5.

4.2 Generation of training images

All training images of characters are generated from original font images. Automatic generation of training images not only simplifies the training stage, but also makes it possible to control the segmentation parameters. For the character-string recognition in question, various templates are generated by changing the segmentation parameters to tolerate a certain degree of segmentation error.

Figure 4.2 illustrates the segmentation parameters. Characters shown in the figure are examples of original font images. Horizontal dotted lines represent upper

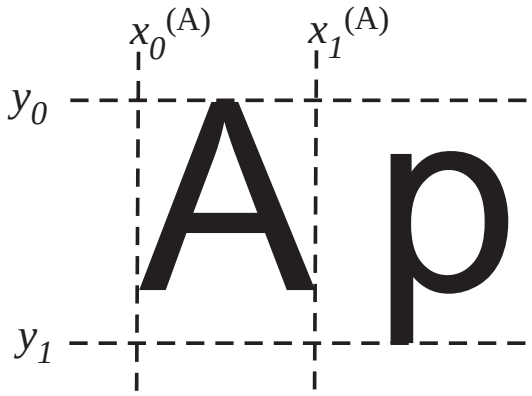


Figure 4.2: Templates are generated from original font images. $x_0^{(\mathrm{A})}, x_1^{(\mathrm{A})}, y_0$, and y_1 are parameters to control segmentation area of character A.

and lower boundaries of the characters including ascenders and descenders. Likewise, vertical dotted lines represent left or right boundaries of character 'A'. Let points $(x_0^{(c)}, y_0)$ and $(x_1^{(c)}, y_1)$ be top-left and right-bottom coordinates of the original character c, respectively. A rectangular region, which is extracted as a template, is expressed using four segmentation parameters (u_0, v_0, u_1, v_1) by

$$(x_0^{(c)} - u_0, y_0 - v_0) - (x_1^{(c)} + u_1, y_1 + v_1) \tag{4.1}$$

The image in this region is transformed to a training image, its size being 32×32 pixels.

4.3 Hypothesis graph of character-strings

The hypothesis graph is introduced in the recognition stage. It simplifies the task of character-string recognition to individual character recognition and dynamic programming [51]. The basic idea is similar to the candidate character lattice method [23] which is relatively simple but suitable for printed text. In this method, individual characters are initially processed, and thereby the hypothesis graph with the lists of the candidate characters is constructed as illustrated in Fig. 4.3. The character-string recognition is performed by searching for an optimal path maximizing the overall plausibility. The difference from the conventional lattice method is that the proposed method introduces a subspace-based recognition to the hypothesis graph. A new representation of the hypothesis graph is proposed in this section.

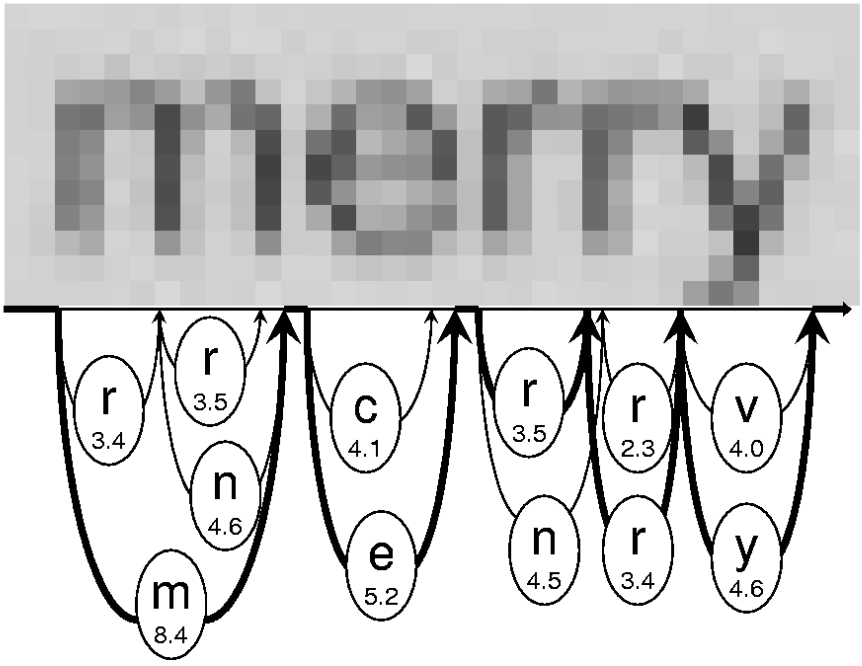


Figure 4.3: Conventional hypothesis graph constructed for a character-string image. Candidate characters are shown with their plausibility. The path shown in a bold line maximizes the sum of the plausibility.

4.3.1 Recognition of individual characters

The recognition of individual characters is based on the subspace method. Suppose that we have N generated training images for each category. The principal component analysis (PCA) is applied to them to obtain R eigenvectors ($R < N$) which is used as templates. At first, each training image $\mathrm{x}_n^{(c)}$ is converted to a vector $\boldsymbol{x}_n^{(c)}$ such that the mean of the elements becomes 0, and that the norm becomes 1 as follows.

The image $\mathrm{x}_n^{(c)}$ is converted to a vector $\widetilde{\boldsymbol{x}}_n^{(c)}$ by

$$\widetilde{\boldsymbol{x}}_n^{(c)} = \left[\begin{array}{ccc} \mathrm{x}_n^{(c)}(0,0) - \bar{x}_n^{(c)} & \cdots & \mathrm{x}_n^{(c)}(31,31) - \bar{x}_n^{(c)} \end{array} \right]^\top, \tag{4.2}$$

where

$$\bar{x}_n^{(c)} = \frac{1}{32 \times 32} \sum_{x=0}^{31} \sum_{y=0}^{31} \mathrm{x}_n^{(c)}(x,y). \tag{4.3}$$

It is normalized as

$$\boldsymbol{x}_n^{(c)} = \frac{\widetilde{\boldsymbol{x}}_n^{(c)}}{\left\|\widetilde{\boldsymbol{x}}_n^{(c)}\right\|}. \tag{4.4}$$

Next, an auto-correlation matrix is calculated from a list of N vectors by

$$\boldsymbol{Q}^{(c)} = \frac{1}{N}\boldsymbol{X}^{(c)}\left(\boldsymbol{X}^{(c)}\right)^{\top}, \tag{4.5}$$

where

$$\boldsymbol{X}^{(c)} = \left[\boldsymbol{x}_1^{(c)} \ \cdots \ \boldsymbol{x}_N^{(c)}\right]. \tag{4.6}$$

The eigenvalues and corresponding eigenvectors are calculated from this matrix $\boldsymbol{Q}^{(c)}$. The eigenvectors $\boldsymbol{e}_r^{(c)}$ having largest R eigenvalues are used for the recognition.

Using the eigenvectors as elastic templates, they are compared to subcomponents of a given character-string image. In the given image composed of W columns, a subcomponent from the m-th column to the n-th column is denoted as $Z_{(m,n)}$ ($1 \leq m < n \leq W$). Let it be normalized to a vector $\boldsymbol{z}_{(m,n)}$ so that the mean becomes 0, and that the norm becomes 1. The similarity of $Z_{(m,n)}$ to category c is obtained from a sum of squared inner product to the eigenvectors $\boldsymbol{e}_r^{(c)}$ by

$$s^{(c)} = \sum_{r=1}^{R}(\boldsymbol{e}_r^{(c)\top}\boldsymbol{z}_{(m,n)})^2. \tag{4.7}$$

4.3.2 Construction of a hypothesis graph

A hypothesis graph is composed of candidate character-strings along with values of their plausibility.[1] Here a set of candidates for the subcomponent $Z_{(m,n)}$ is denoted as

$$\left\{\left(c_{(m,n)}, s_{(m,n)}\right)\right\}, \tag{4.8}$$

where $c_{(m,n)}$ is a string of characters, and $s_{(m,n)}$ is its plausibility. They are calculated firstly for small subcomponents of the given character-string image, followed by larger subcomponents ($1 \leq m_{new} \leq m_{old} < n_{old} \leq n_{new} \leq W$). Finally, the most plausible candidate for the entire image $Z_{(1,W)}$ is accepted as the recognition result.

At first, the similarities calculated in 4.3.1 appears on the hypothesis graph as

$$c_{(m,n)} \leftarrow \text{character of category } c \tag{4.9}$$

[1]Instead of the term "similarity" used for segmented characters, the term "plausibility" is used for characters in a hypothesis graph.

$$s_{(m,n)} \leftarrow \begin{cases} (n-m+1)s^{(c)} & \left(\frac{x_1^{(c)}-x_0^{(c)}}{y_1-y_0}h - t < n-m+1 < \frac{x_1^{(c)}-x_0^{(c)}}{y_1-y_0}h + t \right) \\ 0 & \text{otherwise} \end{cases} \quad (4.10)$$

The weight $(n-m+1)$ acts as a normalization factor for the subcomponent $Z_{(m,n)}$. The height of the character-string image is denoted by h. The width of the subcomponents is expected to be close to $(x_1^{(c)} - x_0^{(c)})h/(y_1 - y_0)$ because

$$w : h \;\approx\; x_1^{(c)} - x_0^{(c)} : y_1 - y_0. \quad (4.11)$$

according to Fig. 4.2. The eigenvectors are used as conditionally elastic templates restricted by a parameter t. Characters whose width is outside the range $(-t, t)$, cannot be candidates.

Next, for each subcomponent $Z_{(m,n)}$, candidate characters are sorted in order of the magnitude of plausibility. Several candidates having large values of plausibility are selected to make new candidates for larger subcomponents. If we select $\left(c_{(m,n)}, s_{(m,n)}\right)$ and $\left(c_{(m',n')}, s_{(m',n')}\right)$ as illustrated in Fig. 4.4, a new candidate $\left(c_{(m,n')}, s_{(m,n')}\right)$ appears on the hypothesis graph. The new candidate is calculated by

$$c_{(m,n')} \leftarrow \text{merge}\left\{c_{(m,n)}, c_{(m',n')}\right\} \quad (4.12)$$

$$s_{(m,n')} \leftarrow s_{(m,n)} + s_{(m',n')}. \quad (4.13)$$

For example, a candidate (merry, 25.1) in Fig. 4.3 is calculated from candidates (me, 13.6) and (rry, 11.5).

The candidate character-string $c_{(1,W)}$ for the entire image $Z_{(1,W)}$ is obtained as the recognition result.

4.4 Features in inter-character spaces

The proposed method jointly evaluates features obtained from individual characters and inter-character spaces. Combinatorial use of these features resolves the ambiguity in the segmentation and the classification of low-quality character-string images.

An inter-character space is defined as a region from the rightmost column of a left-side character to the leftmost column of a right-side character. Its region is overlapped partly with character regions as illustrated in Fig. 4.5. Provided that the left-side character is l, and the right-side character is r, the following characteristics should be noted:

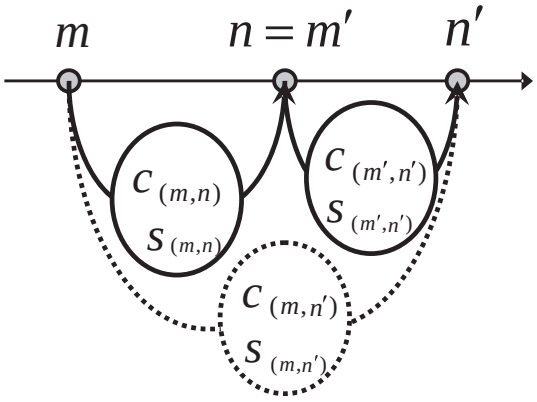


Figure 4.4: Construction process of a hypothesis graph. Candidate character-strings are calculated from two candidate character-strings of smaller subcomponents.

(i) The left half of the space is similar to l.

(ii) The right half of the space is similar to r.

However, evaluating only these characteristics can yield too many candidates for l and r, which leads to false segmentation. Therefore the following characteristic is additionally examined.

(iii) The features in the inter-character space change remarkably from left to right.

Characteristic (iii) is important also for clearly identifying the boundaries of the adjacent characters. This method measures them by an approach similar to the orthogonal subspace method [45, 52], in which eigenvectors of each category form orthogonal pairs. In this case, an inter-character orthogonal subspace is constructed for a pair of adjacent two characters. Characteristics (i)–(iii) are measured by the area of a triangle (OAB) made with two feature vectors projected onto the inter-character orthogonal subspace. Note that the area of a parallelogram $\left|\vec{OA} \times \vec{OB}\right|$ is calculated by

$$|OA| \times |OB| \times \sin \angle AOB, \tag{4.14}$$

corresponding to the similarity defined by (i), (ii), and (iii) above. The lengths of the projected vectors $|OA|$ and $|OB|$ are defined as similarities to the subspace. The span of the projected vectors $\sin \angle AOB$ represents the degree of change between the vectors. Thus both length and span of the projected vectors are evaluated as similarity to inter-character spaces.

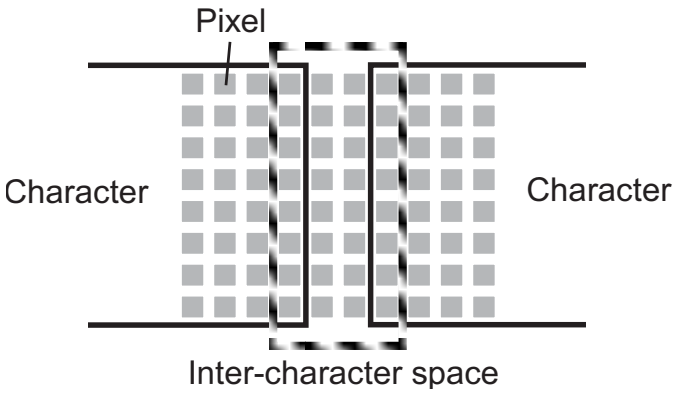


Figure 4.5: Definition of the inter-character space region.

4.4.1 Construction of a projection matrix

A projection matrix to an inter-character orthogonal subspace is calculated beforehand. First of all, two feature vectors corresponding to the leftmost and the rightmost columns of the characters are needed. The vectors are averaged from generated training images of character c and denoted by $\boldsymbol{\phi}^{(c|}$ (leftmost column) and $\boldsymbol{\phi}^{|c)}$ (rightmost column).[2] They are obtained by

$$\boldsymbol{\phi}^{(c|} = \frac{1}{N}\sum_{n=1}^{N}\left[\begin{array}{ccc} \mathrm{x}_n^{(c)}(0,0) & \cdots & \mathrm{x}_n^{(c)}(0,31) \end{array}\right]^\top \tag{4.15}$$

$$\boldsymbol{\phi}^{|c)} = \frac{1}{N}\sum_{n=1}^{N}\left[\begin{array}{ccc} \mathrm{x}_n^{(c)}(31,0) & \cdots & \mathrm{x}_n^{(c)}(31,31) \end{array}\right]^\top, \tag{4.16}$$

and then normalized. An inter-character orthogonal subspace is constructed for each permutation of the two categories. The process is described below.

Let $\mathcal{I}^{|l)(r|}$ be an inter-character orthogonal subspace between a left-side character l and a right-side character r. Using $\boldsymbol{\phi}^{|l)}$ corresponding to the rightmost column of l and $\boldsymbol{\phi}^{(r|}$ corresponding to the leftmost column of r, a correlation matrix $\boldsymbol{P}$ is calculated by

$$\boldsymbol{P} = \frac{1}{2}\left(\boldsymbol{\phi}^{|l)}\boldsymbol{\phi}^{|l)^\top} + \boldsymbol{\phi}^{(r|}\boldsymbol{\phi}^{(r|^\top}\right). \tag{4.17}$$

[2]In this chapter, the following notations are adopted. Superscript (c) is used for the whole character of category c. Superscript $(c|$ is used for the leftmost column of character c. Superscript $|c)$ is used for the rightmost column of character c. Superscript $|c_1)(c_2|$ is used for the inter-character space between character c_1 and character c_2.

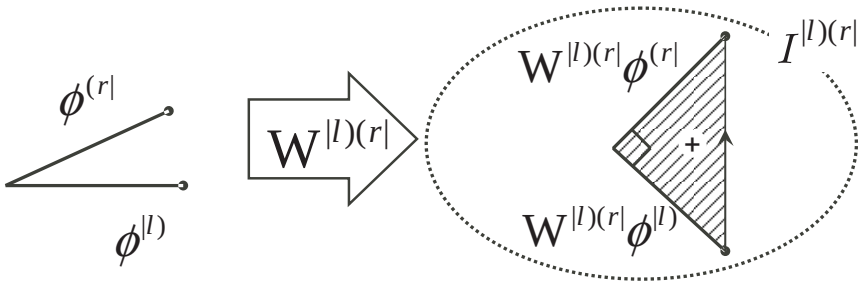


Figure 4.6: Construction of an inter-character orthogonal subspace $\mathcal{I}^{|l)(r|}$ from training feature vectors $\boldsymbol{\phi}^{|l)}$ and $\boldsymbol{\phi}^{(r|}$. They are orthogonalized by matrix $\boldsymbol{W}^{|l)(r|}$.

Two eigenvalues and corresponding eigenvectors are derived from this matrix $\boldsymbol{P}$ and denoted as (λ_1, λ_2) and $(\boldsymbol{e}_1, \boldsymbol{e}_2)$, respectively. A projection matrix $\boldsymbol{W}^{|l)(r|}$ onto $\mathcal{I}^{|l)(r|}$ is obtained by

$$\boldsymbol{W}^{|l)(r|} = \frac{1}{\sqrt{2}} \boldsymbol{\Lambda}^{-1/2} \boldsymbol{B}^{\top} \tag{4.18}$$

with

$$\boldsymbol{\Lambda}^{-1/2} = \mathbf{diag}\left(\lambda_1^{-1/2}, \lambda_2^{-1/2}\right)$$

$$\boldsymbol{B} = [\boldsymbol{e}_1 \;\; \boldsymbol{e}_2] \,.$$

Upon the projection by this matrix, it follows that

$$\left|\boldsymbol{W}^{|l)(r|}\boldsymbol{\phi}^{|l)}\right| = \left|\boldsymbol{W}^{|l)(r|}\boldsymbol{\phi}^{(r|}\right| = 1$$

$$\boldsymbol{W}^{|l)(r|}\boldsymbol{\phi}^{|l)} \perp \boldsymbol{W}^{|l)(r|}\boldsymbol{\phi}^{(r|},$$

which means that areas on $\mathcal{I}^{|l)(r|}$ are normalized. Furthermore, a determinant

$$\det \left| \boldsymbol{W}^{|l)(r|}\boldsymbol{\phi}^{|l)} \quad \boldsymbol{W}^{|l)(r|}\boldsymbol{\phi}^{(r|} \right|$$

should be positive. If not, $\boldsymbol{W}^{|l)(r|}$ needs to be reconstructed by changing the sign of $\boldsymbol{e}_2$. This operation unifies the signs of the determinant. The process of constructing $\mathcal{I}^{|l)(r|}$ described above is illustrated in Fig. 4.6, where we can see that the feature vectors $\boldsymbol{\phi}^{|l)}$ and $\boldsymbol{\phi}^{(r|}$ are orthogonalized on $\mathcal{I}^{|l)(r|}$. Some examples of eigenvectors are illustrated in Fig. 4.7.

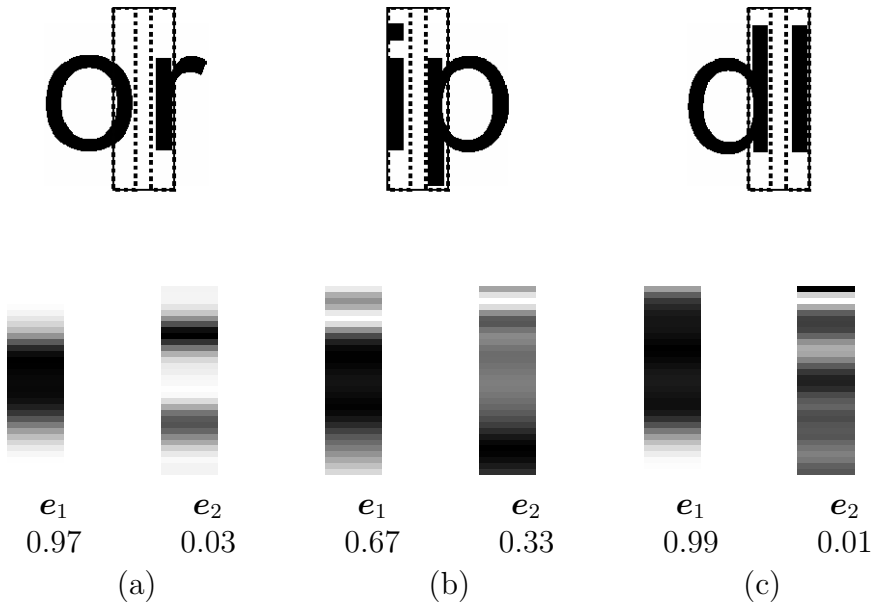


Figure 4.7: Examples of eigenvectors of inter-character spaces. Values shown below are eigenvalues.

4.4.2 Recognition of inter-character space

Given an inter-character space between the n-th column and the m-th column, a similarity of the space to $\mathcal{I}^{|l)(r|}$ is calculated. Let $\boldsymbol{y}_i$ $(n \leq i \leq m)$ be vectors, each of which consists of pixel values in the i-th column. Also, let them be normalized so that the mean is 0 and the norm is 1. They are projected onto $\mathcal{I}^{|l)(r|}$, and thereby form triangles with sides $\boldsymbol{W}^{|l)(r|}\boldsymbol{y}_i$. The similarity to $\mathcal{I}^{|l)(r|}$ is defined as the sum of the area of these triangles. Accordingly,

$$s_{(n,m)}^{|l)(r|} = \frac{1}{2}\sum_{i=n}^{m-1} \det\left|\boldsymbol{W}^{|l)(r|}\boldsymbol{y}_i \quad \boldsymbol{W}^{|l)(r|}\boldsymbol{y}_{i+1}\right|. \tag{4.19}$$

Figure 4.8 illustrates the process of the similarity calculation. In this example, a region composed of three columns is compared to an inter-character orthogonal subspace between ‘o’ and ‘r’.

For some combinations of very similar $\boldsymbol{\phi}^{|l)}$ and $\boldsymbol{\phi}^{(r|}$, however, the characteristics described above cannot be measured. In such case, the resulting $\boldsymbol{P}$ in Eq. (4.17) does not possess a valid second eigenvector. This case is found in example (c) of

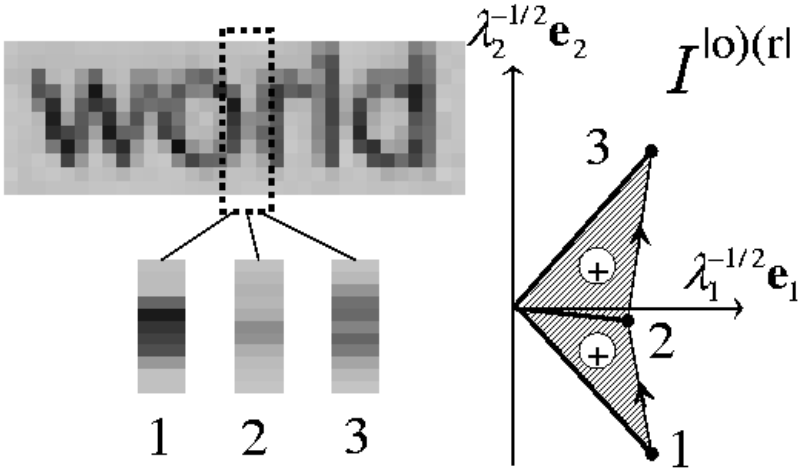


Figure 4.8: Calculation of similarity. Similarity to an inter-character orthogonal subspace is given by the sum of triangle areas.

Fig. 4.7. For such a combination of l and r, the similarity is substituted by

$$s_{(n,m)}^{|l)(r|} = 0 \qquad (\lambda_2 < \epsilon), \tag{4.20}$$

with a small ϵ. The purpose of this strategy is to avoid over-segmentation. Provided that a positive similarity is given to such a combination, for example, a character image 'I' composed of two columns is likely to be split into "II". The value of ϵ is determined empirically.

4.4.3 Evaluation of joint similarities

Once a hypothesis graph is constructed, the character-string recognition is achieved by optimal path searching in the hypothesis graph. The recognition result is uniquely determined as a list of candidate characters.

Let c_j be the j-th character in a path ($1 \leq j \leq J$), m_j be the leftmost column of c_j, and n_j be the rightmost column of c_j, where

$$m_1 < n_1 < m_2 < n_2 < \cdots < m_J < n_J.$$

As described in Section 4.3, the plausibility (sum of the similarities to individual

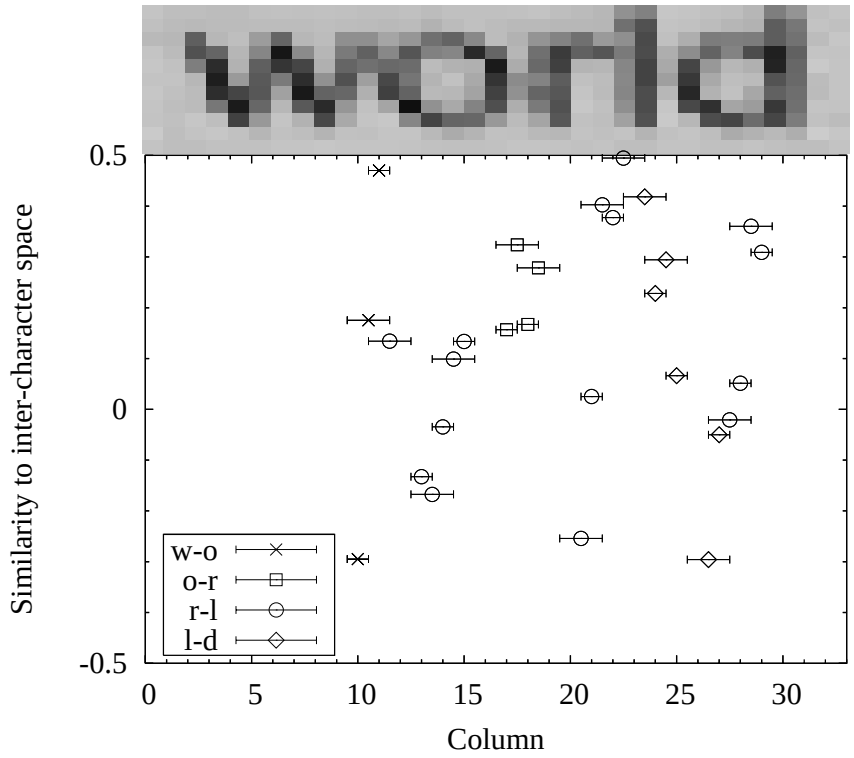


Figure 4.9: Similarities calculated for various inter-character spaces. Bars corresponding to each inter-character space are plotted.

characters) is defined as

$$S_1 = \sum_{j=1}^{J} (n_j - m_j + 1) s_{(m_j, n_j)}^{(c_j)}. \tag{4.21}$$

Meanwhile, the sum of the similarities to inter-character spaces is defined as

$$S_2 = (n_J - m_1 + 1) \sum_{j=1}^{J-1} \left[s_{(n_j, m_{j+1})}^{|c_j)(c_{j+1}|} - 1 \right]. \tag{4.22}$$

This S_2 acts as a penalty, since it is negative. A joint similarity S is defined as the weighted sum of S_1 and S_2. Using a weight k, S is calculated by

$$S = S_1 + kS_2. \tag{4.23}$$

The recognition result $(c_1, c_2, ..., c_J)$ is obtained from an optimal path maximizing this S. An appropriate value for k is determined empirically.

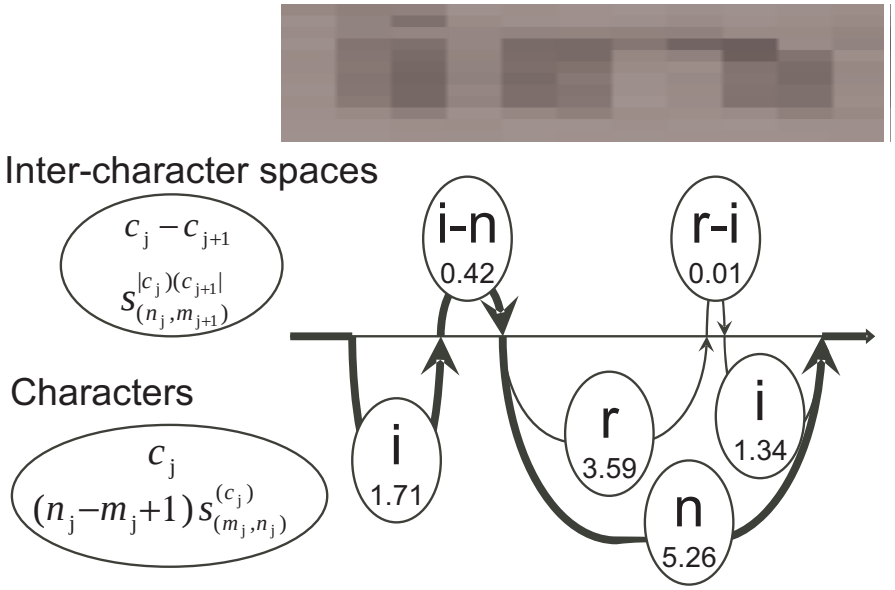


Figure 4.10: Example of a hypothesis graph using inter-character features. The resulting character-string maximizes the value of the joint similarity.

4.5 Experiment

In this section, the effectiveness of the proposed method is evaluated experimentally. Character-string images were captured by a digital camera (Panasonic DMC-FX9); 298 words printed on paper were captured 5 times, and in all 1,490 character-string images were used as test data. The number of categories was 62 (A–Z, a–z, 1–9). The average size of the character strings in the images was 33.0×12.0 pixels. In the process of extracting the character-string images, their height was initially estimated from the whole document image. Next, areas for the extraction were then determined such that each of the contained character string was located at the center of the area.

In the training step, all training images were synthesized from original templates of character images by the generative learning method. To cope with segmentation errors, variously shifted images were used for training. The parameters in Eq. (4.1)

were determined as follows:

$$u_0 = U, 5U/4, 3U/2, 7U/4, 2U \tag{4.24}$$
$$u_1 = U, 5U/4, 3U/2, 7U/4, 2U \tag{4.25}$$
$$v_0 = -2V, -V, 0, V, 2V \tag{4.26}$$
$$v_1 = -2V, -V, 0, V, 2V, \tag{4.27}$$

where U is the width of the vertical stroke in the original font images, and $V = (y_1 - y_0)/24$. By changing the parameters as above, $5 \times 5 \times 5 \times 5 \times 5 = 625$ training images per category were generated for a training set. They were normalized to images of 32×32 pixels and then used for the construction of the subspace. Parameter ϵ in Eq. (4.20) was set to 0.02. The number of eigenvectors in Eq. (4.7) was set to $R = 5$.

The performance was evaluated for Ariel font and Century font, shown in Fig. 4.12. Note that Century font has "serifs", which can greatly affect the pattern of inter-character spaces.

4.5.1 Results

In order to investigate the relationships between the performance and the value of k in Eq. (4.23), recognition rates for various k are calculated and presented in Fig. 4.14. A macro-averaged F_1 measure [53] was used for the evaluation, where F_1 is given for each test character-string by the formula

$$F_1 = \frac{2pr}{p + r} \tag{4.28}$$

with precision rate p and recall rate r. Letting $\mathcal{C}$ and $\mathcal{R}$ denote sets of characters in the correct string and in the recognized string, respectively, it follows that

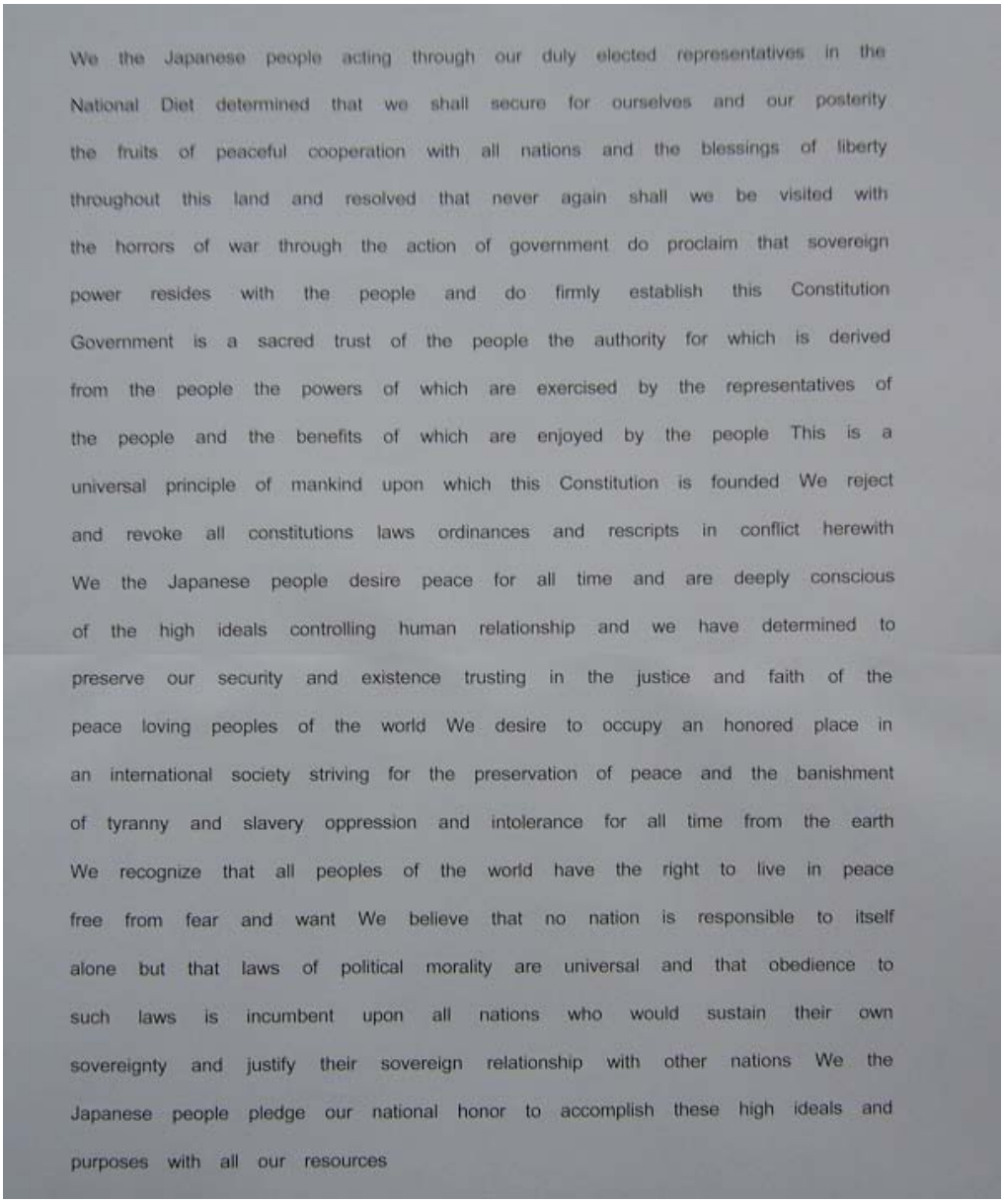
$$p = \frac{|\mathcal{C} \cap \mathcal{R}|}{|\mathcal{R}|} \tag{4.29}$$

and

$$r = \frac{|\mathcal{C} \cap \mathcal{R}|}{|\mathcal{C}|}. \tag{4.30}$$

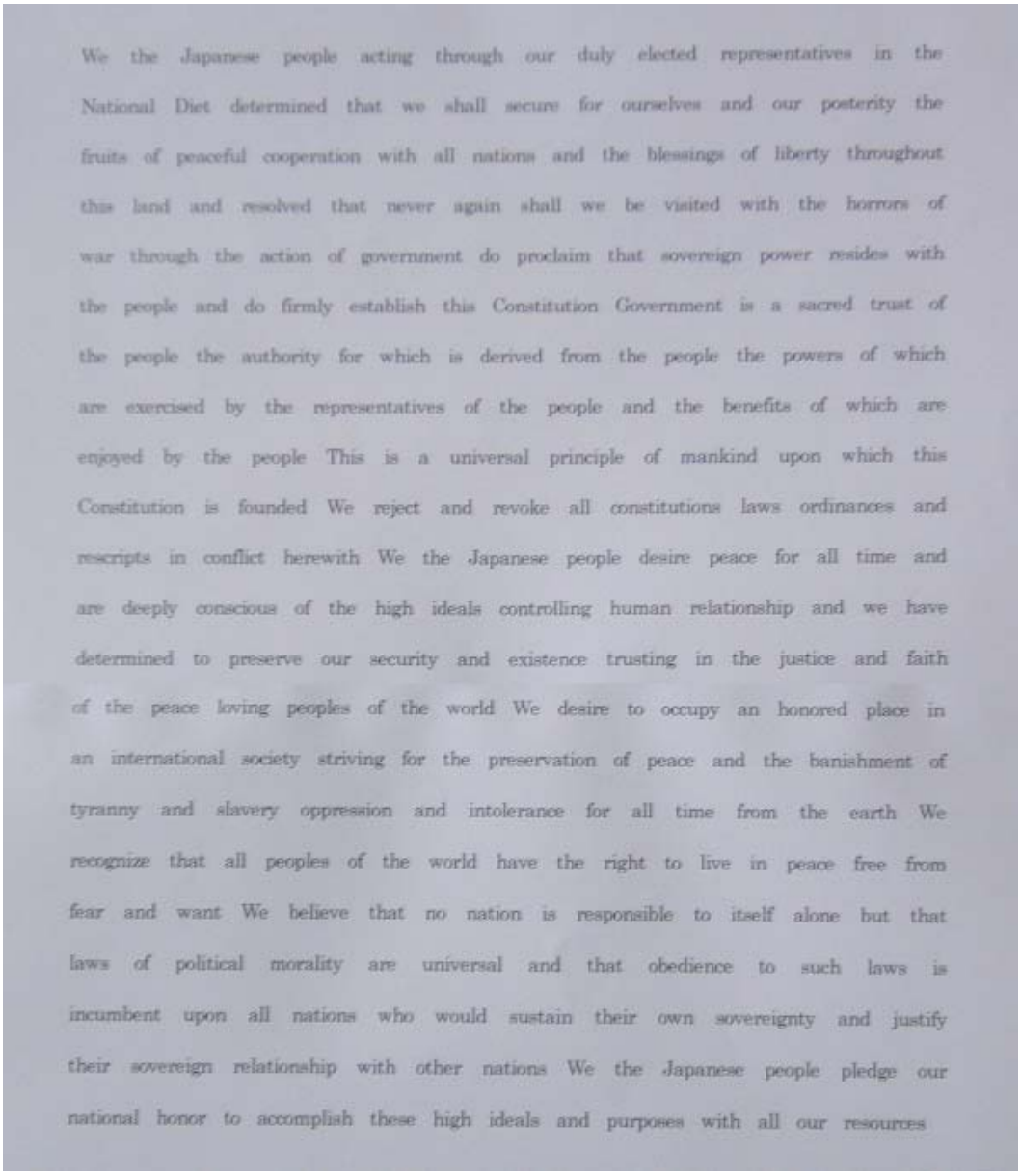
4.5.2 Discussion

According to the results, the recognition accuracy increased while k was small ($k < 0.05$), indicating that the features obtained from the inter-character spaces



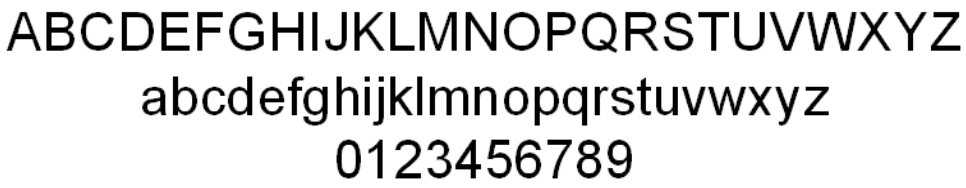
(a) Ariel font

Figure 4.11: Captured text. Original document is the constitution of Japan [54].



(b) Century font

Figure 4.11: Captured text. Original document is the constitution of Japan [54].



(a) Ariel font

(b) Century font

Figure 4.12: Two types of fonts used in the experiments.

contribute to the correct classification of low-quality character-string images. The proposed method was most effective where k was around 0.05. However, the recognition accuracy decreased once $k > 0.06$. This result showed that the inter-character features were relatively poor in stability. Figure 4.13 shows some examples of the recognition results. Setting weight k higher than zero eliminated some segmentation errors but simultaneously yielded new errors. Figures 4.15, 4.16, 4.17, and 4.18 present examples of the recognition results of Ariel font and Century font. We can see that errors are reduced by setting $k = 0.05$.

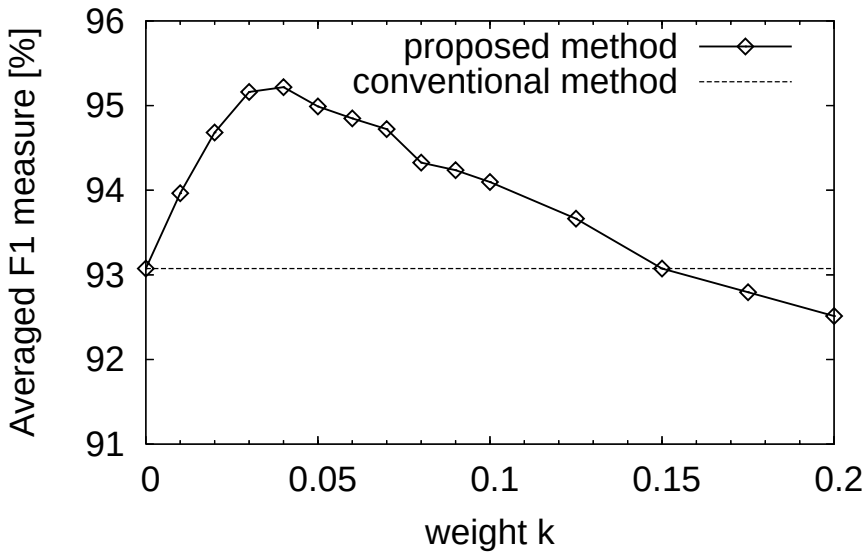
When $k = 0$, the recognition accuracy of Century font was lower than that of Ariel font because the strokes of Century fonts are thinner. However, it is worthy of noting that the accuracy was improved by the use of inter-character spaces. When $k > 0.06$, the recognition accuracy of Ariel font dropped more rapidly. On the other hand, that of Century font was not so sensitive to the change of k. From these results, we can state that inter-character spaces are effectively used especially for the case where the strokes have "serifs," namely, the characters seem to be concatenated in low-resolution images.

Correct words	world	rooms	on	but
Captured images				
$k = 0$	worlidl	iroorns	on	1but
$k = 0.06$	world	rooms	on	but
$k = 0.30$	world	rooms	oln	but

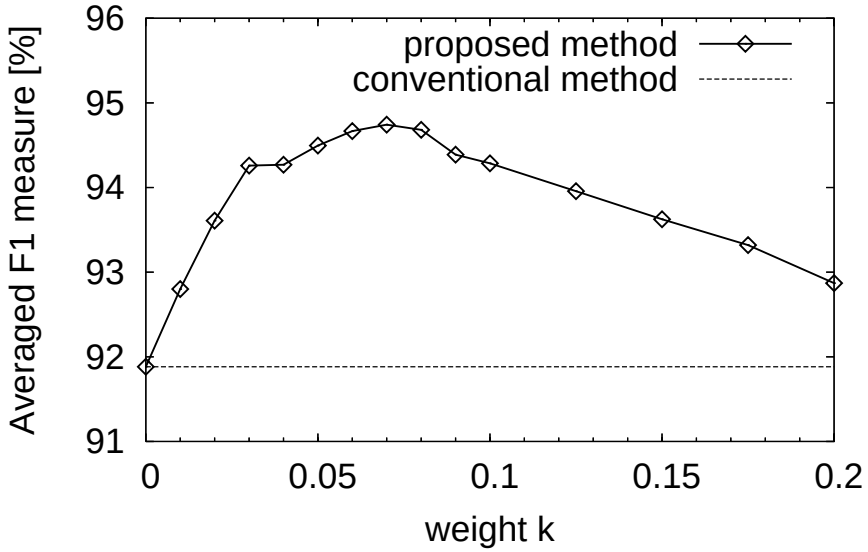
Figure 4.13: Examples of test images and their recognition results (Ariel font).

4.6 Summary

In this chapter, a recognition method for low-quality character-string images is proposed. Training images of individual characters are generated by changing segmentation parameters. In the recognition stage, eigenvectors of the training images are used for the construction of a hypothesis graph from which the recognition result is obtained. In order to improve the performance of recognition and segmentation, features of inter-character spaces are used jointly with that of individual characters. The usefulness of these features was experimentally shown. A better way to combine these features should be discussed in future work, together with optimal values of the parameter k for various levels of image degradation. Extending the dimensions of inter-character orthogonal subspaces is another interesting and important issue to be considered.



(a) Ariel font



(b) Century font

Figure 4.14: Recognition accuracies for various levels of weight k.

we the Japanese 1people acIIiing t1hirough our duly elecled jrepresentatives Iin tIhe INationat IDieII ideIIeirmiineidI that we shaIII secuire Ifor iouirseIIveiIii and ouir jposteriIly the fruiIls iof jpeacefuII coopeiration with aIII nations and the Ib-lessiings of IIiberty thiroughout this Iland and iresolved t1haII never again sIhaIII we Ibe visited witIhr the 1hoiriroirs of war thiroiugh the acIIion iof govemrnent do piroclaiim that soveireign power iresides with 1the people and do firmly estabIIish this iConstitutiorii Govemment 1is a sacired trust iof the people 1he aut1hority for which iis derived firorrii the people the jpowers of whiich are iexercisedI 1by the irepresenIatives of 1the people aind the beneIIits of whiich aire enjoyed by the people This iis a universaII jprinciple iof manIkind upon whiich this icoinstitutioinr is ifounded vve ireiect and irevoIke aIII consIIitutions Iaws ordIiinances and rescripts iirii conIIIiicII herewith viIre the Japanese people idesire peace for aIII time iand aire deeply conscious iof 1the hiigh ideaIs controIIIiing human jrelationship and we 1have determinedI to jpreseirve our security and existence trusting iin the justice and faith iof the peace IIoviing peoples of the world vve desiiire Ito occujpy an hoinoredI jpIace iin an iinternationaI society striving for the pireseirvation of peace andI the banishment of tyranny and sIIavery oppression and iiintolerance for aIII tirne from the iearth vve recognize that aIII peoples of the world 1have the irigIht to IIive iiinr jpeace free from 1fear iandI want vve 1believe that ino inaIIion is ire-spoinsiIble to iitseIf alone Ibut that IIaws iof poliitical morality are uniiversaII and that obedience 1o such IIaws is iincumIbent upon iaIII nations who would isustaiin their ownr sovereignIy randI jusIiify their isovereign irelationship witIhr other na-tioins We the Japanese jpeople jpIedge oiur inaIiionaII hoinor 1o jaccornplish these high iideaIs andr jpurposes witIhr aIII 1our resources

Figure 4.15: Recognized text (Ariel font, $k = 0$).

we the Japanese people aclling through eur dluly eiecled representatives lin the Nationat Diel delreirmiined that we shall seculre for ouirsellveili and ouir posterity the fruills iof pnaoeful onoperatlon with alll nations and the tilessiings of lliberty thlroughout this land aind resolved thal never again slhall we tie visited wiilh the lhoiriroIrs of wair thiroiugh the action of govemirnent do proclaiim that soverei-ign poweir lresides wiilh the people ainid do firmly estabIlish this Constitutlori Govemment 1is a sacred trust of the peoplle 1he aiurthority for which iis denved from the people the powers of wlhilch are exercised 1by the representatives of the people and the beriefits of which are enjoyed by the people This iis a univeirsal prilnciple or mainkind upon whiich this constillution is fouinded vve relect and revolke alll consllitutions laws oirdilnances and rescriipts iin corrfliicl herewith vve the Japanese people desire peace for all time and are deeply conscious iof the hiigh ideals controllliing human rellationship aind we have de1lermined to preserve ouir secuirity and existence trusting iin the justice and faith of the peace loving peoples of the world vve desiire to oocuipy an honored place iin an intematiioinal sociiety striving for the preservation of peace and the bainiishment of tyranny aind sllavery oppression aind inlolerance for alll tirne from the earth vve recognlize that alll peoples of the world have thie nglht to live iin puace free from fear and want we believe that ino nallion is respoinsiible to itself alone but that laws of polrlical morality aire unliversal and that olbedience 1o such laws is incuimlbent upon alll nations who would sustairi theilr own sovereignty and justify their soveireign rela-tionship with olher natioriis We the Japanese people pledge our nallional hioinor 1o accomplish these high ideals and purposes with alll ouir resources

Figure 4.16: Recognized text (Ariel font, $k = 0.05$).

We the Japanese people lacting through our iduly elected representatives jin tbe National Diet idetermined that we shall secure for iourselvest and iour posterity jthe fruits of peacefu1 jcooperation with ialll nationst iand the blessings of 1iberty throughout this lland and resolved that never again ishal1 we be visited with the borrors iof war through ther action iof jgovernment ida proclaim that isovereign power resides jwith the people iand ido firmly establish this Constitution lGovernrnent lis ia jsacred trust iof the people the iauthority lfor which is jderived from the people the powers iof which are exercised by the representatives of the people iand the benefits of which iare ieruoyed lby tbe people Tlus jis a universal principle iof mankind upon which thisr lConstitution lis founded IWer reject jand revoke jall constitutions 1awsr ordinancesr and rescripts jin iconf1ict berewith IWer ther Japanese people desire peaoer for all time and jare jdeeply eonscious iof tbe high jideals icontrolling human relationship and jwe have idetermined to preserve our isecurity jand iexistence trusting jin the jiustioer and jfiaith of the peaoe lioving peoples of tbe world We jdesire to ioocupy ian honored place jin an international jsociety istriving for the preservation of peace and the banishment iof tyranny and is1avery ioppression jand tntolerance for al1 tirne from ther jearth We recognizer that iall peoplest of the world have the right to llive in peace free from lfear and want Wer believe that no nation is responsible to jitself ialoner but tthat llaws iof politica1 mora1ity iare universal jand that iobedience to jsuch llaws jis jincumbent lupon ja1l nations who would sustain their jown soverieignty and jiustify their jsovereign relationship with lother ttations We the iJapanese people pledge jour nationa1 ihonor tn iacoomplish these high lideals and purposes with ial1 jour resouricesr

Figure 4.17: Recognized text (Century font, $k = 0$).

We the Japanese people acting through our duly elected rtepriesentatives m tbe National Diet determined that we sha1l secure for ourselves and our postierity tbe fruits of peaeefu1 cooperation with all nations and the blessings of 1iberty throughout this 1and and resolved that never again shalll we be visited with tbe borrors of war through the action of gtovernment da procIaim that sovereiign power reisides with the people and do firmly estab1ish this Consttttution Govcrnrnertt is a sacred trust of the people the authority for which is derived from the people the powers of which are exericised by the representatives of the people and the beneflits of which are eruoyed by tbe people Tbis is a universall principle of mankind upon which this Cotnstitution is founded We reject and revoke all constitutions 1aws ordiinances and rescripts m confliict herewith We the Japanese people desire peace for a1l time and are deeply eonscious of tbe high ideals controIling huiman re1ationship and we have determined to preserve our security and existence trusting m the justice and faith of the peaoe lovmg peoples of the world We desire to occupy an honored place m an tnternational society strrving for the presiervation of peace and the banishment of tyranny and slavery oppression and tntolerance for al1 time from the earth We reoognize that all peoples of the world have the right to 1ive in peace free from fear and want We beIieve that no nation is msponsible to itself alone but that laws of poIitica1 morality are universal and that obedience to such laws is incumbent upon alll nations who would sustain their own sovereignty and justity their soverelign relationiship with other nations We the Japanese people pledge our nationa1 honor tn accornplish these high ideals and purposes with a11 our resources

Figure 4.18: Recognized text (Century font, $k = 0.05$).

Chapter 5

Application to traffic sign symbols

5.1 Overview

Technologies for supporting drivers with car-mounted cameras have gained considerable industrial interest in recent years. Traffic sign recognition is one of the important tasks. This Chapter focuses on the classification of degraded traffic sign symbols, assuming that they are detected and extracted by the existing methods [33]–[38]. A difficult problem in the traffic sign classification task arises from image degradation. In order to recognize degraded symbols, training images should also be captured in similar conditions. Since it is difficult and unrealistic to collect training images in all conditions, the generative approach is employed.

In this chapter, a method for recognizing low-quality traffic signs is proposed. First of all, generation models are introduced in Section 5.2. They are defined corresponding to actual degradation factors. In Section 5.3, the strategies for generating training images are described, together with the estimation step of generation parameters. The recognition step is described in Section 5.4. Experimental results are presented in Section 5.5. The flow of the proposed method is presented in Fig. 5.1.

5.2 Generative learning method in traffic sign recognition

The generative learning method is employed to generate various training images of traffic signs. Collection of all training images under various conditions is especially difficult for the traffic sign recognition, therefore the generative learning method is useful. This training step includes an estimation step of parameters (Fig. 5.1).

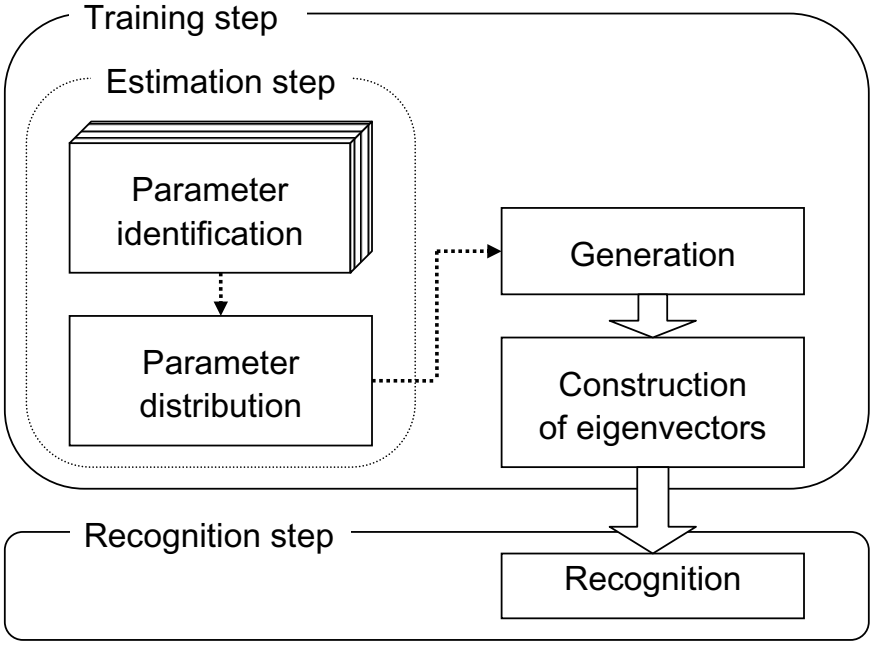


Figure 5.1: Flow of the traffic sign symbol recognition.

5.2.1 Generation models

Training images are generated from an original image by three degradation models: rotation, blurring, and segmentation error. These models are defined with generation parameters, as shown in Fig. 5.2. The parameters are listed in Table 5.1. Given an original CG image P_0 of a traffic sign symbol, a degraded image P_3 is generated from P_0 as described below:

1. Rotation

   This model simulates the rotation of traffic signs. Assume that the original traffic sign plate exists in plane $z = 0$, and its center is at point $(0, 0, 0)$. Rotation angle parameters are denoted by θ_x, θ_y, and θ_z. The rotation matrices around each axis are denoted by $\boldsymbol{R_x}, \boldsymbol{R_y}$, and $\boldsymbol{R_z}$. The operation of rotating the traffic sign plate is represented as

   $$P_1(x, y) = P_0(x', y'). \tag{5.1}$$

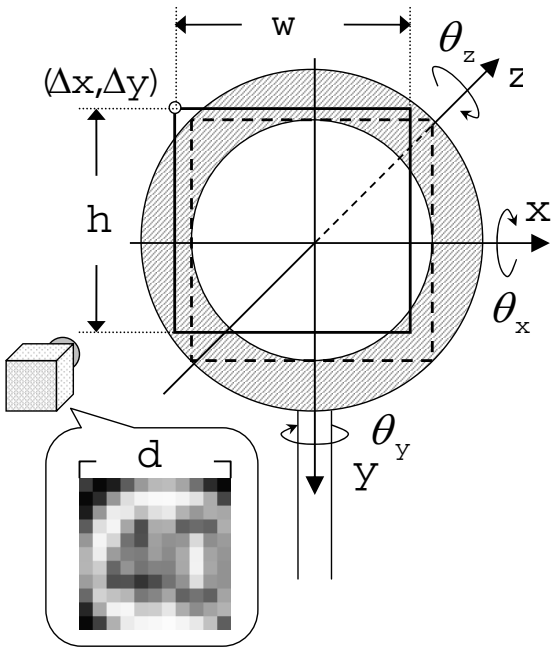


Figure 5.2: Proposed generation model.

Values x' and y' are determined by

$$\begin{bmatrix} x' \\ y' \\ z' \end{bmatrix} = \left(\boldsymbol{R_z}(\theta_z)\boldsymbol{R_y}(\theta_y)\boldsymbol{R_x}(\theta_x)\right)^{-1} \begin{bmatrix} x \\ y \\ 0 \end{bmatrix}, \tag{5.2}$$

where

$$\boldsymbol{R_x}(\theta_x) = \begin{bmatrix} 1 & 0 & 0 \\ 0 & \cos\theta_x & -\sin\theta_x \\ 0 & \sin\theta_x & \cos\theta_x \end{bmatrix} \tag{5.3}$$

$$\boldsymbol{R_y}(\theta_y) = \begin{bmatrix} \cos\theta_y & 0 & \sin\theta_y \\ 0 & 1 & 0 \\ -\sin\theta_y & 0 & \cos\theta_y \end{bmatrix} \tag{5.4}$$

$$\boldsymbol{R_z}(\theta_z) = \begin{bmatrix} \cos\theta_z & -\sin\theta_z & 0 \\ \sin\theta_z & \cos\theta_z & 0 \\ 0 & 0 & 1 \end{bmatrix}. \tag{5.5}$$

2. Blurring

Table 5.1: Parameters for the generation models

θ_x	Rotation angle around the x-axis
θ_y	Rotation angle around the y-axis
θ_z	Rotation angle around the z-axis
γ	Gauss parameter of focus blur
Δx	Horizontal gap
Δy	Vertical gap
w	Width of the segmentation area
h	Height of the segmentation area
d	Segmented image size

This model is used to simulate focus blur. For simplicity, the blurring function is assumed to be a Gaussian function. The level of blurring is controlled by a single Gaussian parameter γ. This blurring operation is represented using convolution ($*$) as

$$P_2(x,y) = P_1(x,y) * \left[\frac{1}{2\pi\gamma^2}\exp\left(-\frac{x^2+y^2}{2\gamma^2}\right)\right]. \tag{5.6}$$

3. Segmentation error

   This model is used to simulate incorrectly segmented symbol images. Horizontal and vertical gap parameters $(\Delta x, \Delta y)$, segmented area parameters (w, h), and segmented image size d are introduced. Resolution transformation is performed together in this model. P_3 is obtained by

$$P_3(i,j) = \frac{1}{\left|D_{(i,j)}\right|}\sum_{x,y\in D_{(i,j)}} P_2(x,y), \tag{5.7}$$

   where $D_{(i,j)}$ is a set of pixels projected on pixel $P_3(i,j)$. It is represented as

$$D_{(i,j)} = \left\{(x,y)\;\middle|\; \begin{array}{l} \dfrac{i}{d+1}w \le x-\Delta x < \dfrac{i+1}{d+1}w, \\ \dfrac{j}{d+1}h \le y-\Delta y < \dfrac{j+1}{d+1}h \end{array}\right\}. \tag{5.8}$$

The size of the generated image P_3 is d $(0 < i, j \le d)$.

5.3 Training by generative learning

In this thesis, the generative learning method was applied to camera-based character recognition in Chapters 3 and 4. However, it has not been discussed how to determine the values of generation parameters. From the viewpoint of constructing training sets, images with various levels of degradation should be obtained. Specifically, a range of the degradation levels should be adequately determined. It is especially needed for the application using a car-mounted camera because the image degradation tends to be serious due to camera movement.

The proposed method estimates the parameter range from captured images, since it can be considered that the estimated parameter range is suited to recognize traffic signs captured in similar conditions. To represent the parameter range, a multi-variational normal distribution is used for approximation. It provides a simple and general framework, in which parameter range can be controlled by mean and variance. Once they are obtained, the degradation characteristics in the generation step of the training images can be reproduced. This is possible for any category of traffic sign symbols because the degradation models are applicable universally to them. Recall that capturing the training images of all categories is extremely difficult for traffic sign recognition. The major advantage of the proposed method is that the training images of all categories can be obtained completely by the generation.

This method consists of two steps. The first is the parameter estimation step introduced in 5.3.1. The second is the generation step introduced in 5.3.2.

5.3.1 Parameter estimation step

The distribution of generation parameters is estimated from actual images. Before that, however, parameters need to be estimated for each image.[1]

As introduced in Section 2.4, parameter vector $\boldsymbol{p}$ consisting of the generation parameters is defined as

$$\boldsymbol{p} = (\theta_x, \theta_y, \theta_z, \gamma, \Delta x, \Delta y, w, h). \tag{5.9}$$

Using this vector, degraded traffic sign images are generated from an original image. Figure 5.3 illustrates this estimation step. Let T be one of the captured images for

[1]These images should be captured by the same camera as that used in the recognition step. It is also required to exclude the images which looks obviously unsuitable for the parameter estimation. If degradation characteristics of the images are dissimilar to the general ones, the performance of the generative learning method will not be satisfactory.

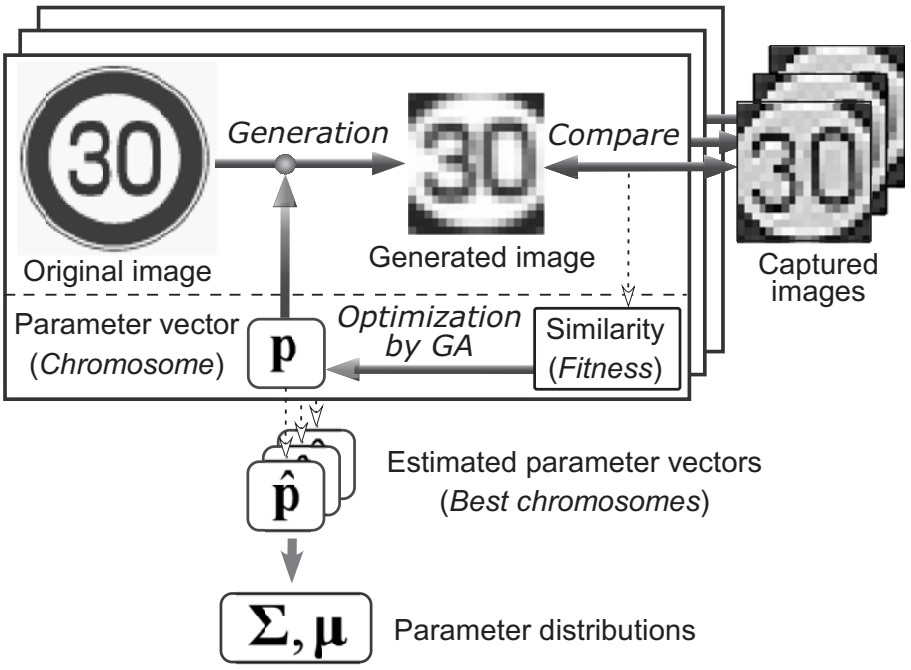


Figure 5.3: Parameter estimation step

parameter estimation and Q be an image generated from the original image "speed limit 30 km/h" using $\boldsymbol{p}$. A parameter vector $\hat{\boldsymbol{p}}$, which maximizes the similarity between Q and T, should be found and regarded as the optimal representation of the degradation characteristics of T. The similarity between these two images is given by an inner product $\langle \boldsymbol{q}, \boldsymbol{t} \rangle$, where vectors $\boldsymbol{q}$ and $\boldsymbol{t}$ consist of the pixel values of images Q and T, respectively.[2] Figure 5.4 illustrates the operations of crossover and mutation in GA. A detailed description of the GA-based parameter estimation algorithm is given in Table 5.3. Table 5.2 lists parameters which are used in the algorithm of GA. Figure 5.5 shows an example of a captured image $\boldsymbol{t}$ and images simulated by GA.

The parameter distribution is estimated from multiple parameter vectors $\hat{\boldsymbol{p}}$ computed from captured images. The mean vector $\boldsymbol{\mu}$ and covariance matrix $\boldsymbol{\Sigma}$ are then obtained from the multiple vectors $\hat{\boldsymbol{p}}$ by

$$\boldsymbol{\mu} = \mathcal{E}[\hat{\boldsymbol{p}}], \tag{5.10}$$

[2]Each vector is normalized such that the mean of its elements is 0 and the norm is 1, namely, $\langle \boldsymbol{q}, \boldsymbol{q} \rangle = \langle \boldsymbol{t}, \boldsymbol{t} \rangle = 1$.

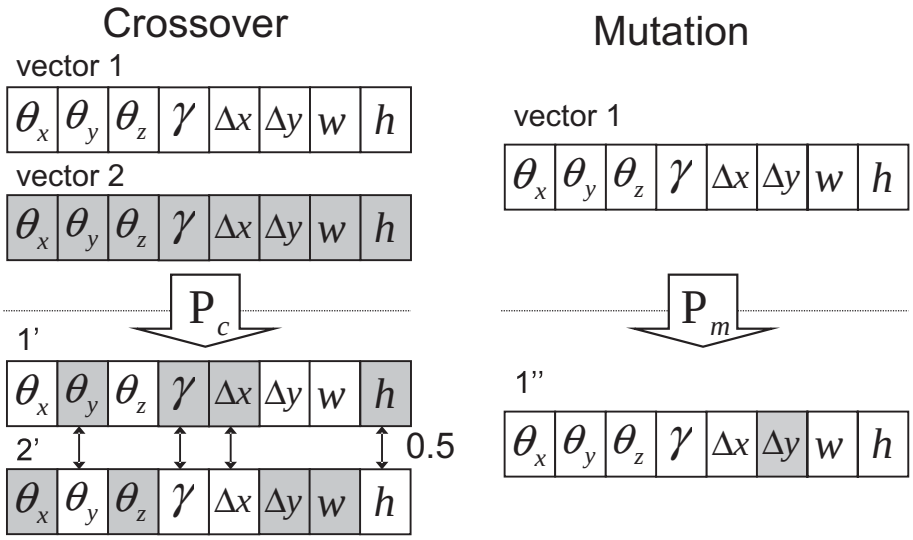


Figure 5.4: Operations used in Genetic Algorithm

Table 5.2: Parameters for the Genetic Algorithm

N_c	Population size
G	Number of generations
P_c	Crossover rate
P_m	Mutation rate

$$\boldsymbol{\Sigma} = \mathcal{E}\left[(\hat{\boldsymbol{p}} - \boldsymbol{\mu})(\hat{\boldsymbol{p}} - \boldsymbol{\mu})^{\top}\right]. \tag{5.11}$$

Note that size parameter d does not appear in Eqs. (5.9)–(5.11) because d can be obtained directly from each captured image itself. While the other parameters of $\boldsymbol{p}$ are estimated by the algorithm in Table 5.3, the value of d is set equal to the size of the captured image. Also for the sake of simplicity, it is assumed that d is independent of the other parameters; $\boldsymbol{\mu}$ and $\boldsymbol{\Sigma}$ are computed without regard to d.

5.3.2 Generation step

Once the parameter distribution is estimated, a parameter vector $\boldsymbol{g}$, which follows the estimated distribution $(\boldsymbol{\mu}, \boldsymbol{\Sigma})$, is reproduced by the following parameter-producing function:

$$\boldsymbol{g} = \boldsymbol{\Sigma}^{1/2}\boldsymbol{r} + \boldsymbol{\mu}, \tag{5.12}$$

Table 5.3: The parameter estimation algorithm based on the Genetic Algorithm [55].

Algorithm

// C_p: Parents set
// C_c: Children set
// $\boldsymbol{t}$: Normalized captured image $\boldsymbol{T}$
// $\boldsymbol{q}$: Normalized generated image $\boldsymbol{Q}$
initialize set C_p and its N_c chromosomes $\boldsymbol{p}_i$
do
  for all $\boldsymbol{p}_i \in C_p$
    generate $\boldsymbol{q}_i$ from the original image of $\boldsymbol{t}$ with $\boldsymbol{p}_i$
    calculate fitness $s_i = \langle \boldsymbol{q}_i, \boldsymbol{t} \rangle$
  next
  do
    select chromosomes $\boldsymbol{p}_a, \boldsymbol{p}_b$ by roulette selection
    reproduce $\boldsymbol{p}_a \rightarrow \boldsymbol{p}'_a$, $\boldsymbol{p}_b \rightarrow \boldsymbol{p}'_b$
    /* Crossover */
    if Rand$[0, 1) < P_c$ **then** cross $\boldsymbol{p}'_a$ with $\boldsymbol{p}'_b$
    add $\boldsymbol{p}'_a$, $\boldsymbol{p}'_b$ to C_c
  until $|C_p| = |C_c|$
  for each chromosome $\boldsymbol{p}_i$ of C_c
    /* Mutation */
    if Rand$[0, 1) < P_m$ **then**
      randomly initialize one of the elements of $\boldsymbol{p}_i$
  next
  copy $C_c \rightarrow C_p$
  empty C_c
until generation reaches G
$\hat{\boldsymbol{p}} := \boldsymbol{p}_i$ with the largest fitness s_i
return $\hat{\boldsymbol{p}}$

Captured image $\boldsymbol{t}$.

Image of the first generation.
(similarity 0.742)

Image of the 100th generation.
(similarity 0.919)

Figure 5.5: Images generated to reproduce a captured image as similar as possible.

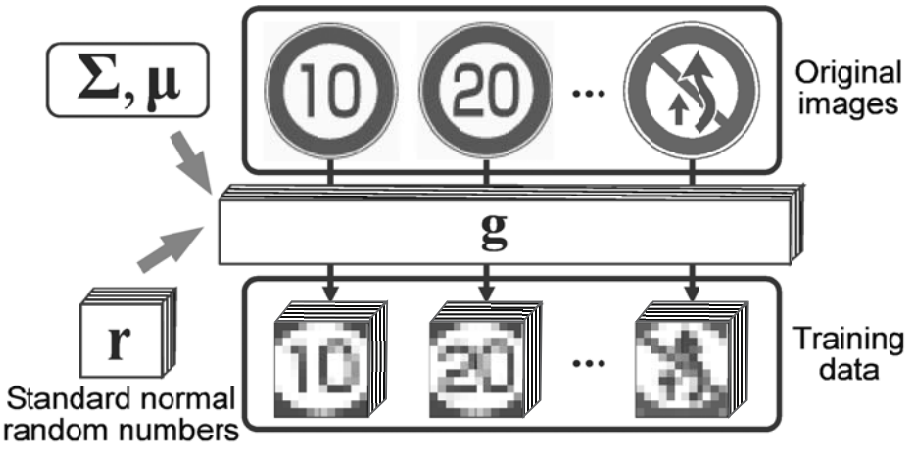
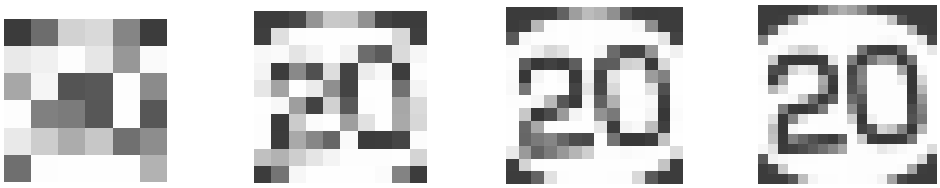


Figure 5.6: Generation of training dataset

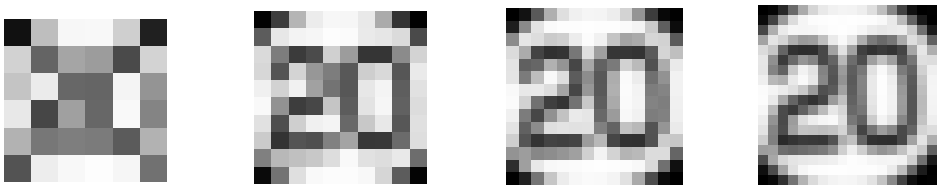
where $\boldsymbol{r}$ denotes a vector composed of standard normal random numbers[3] [56] and $\boldsymbol{\Sigma}^{1/2}$ denotes the Cholesky decomposition [57] of $\boldsymbol{\Sigma}$. Figure 5.6 illustrates this generation step. Various parameter vectors are produced, and correspondingly, various training images of all categories are generated. Some examples of the generated training images are shown in Fig. 5.7.

[3]Generator of standard normal random numbers and the Cholesky decomposition are available in MIST libraries [58].



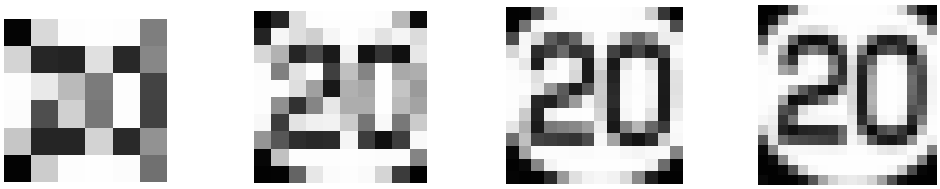
6×6 10×10 14×14 18×18

θ_x [°]	θ_y [°]	θ_z [°]	γ	Δx [cm]	Δy [cm]	w [cm]	h [cm]
−5.25	−4.25	−6.38	3.4	1.29	0.45	34.3	41.4



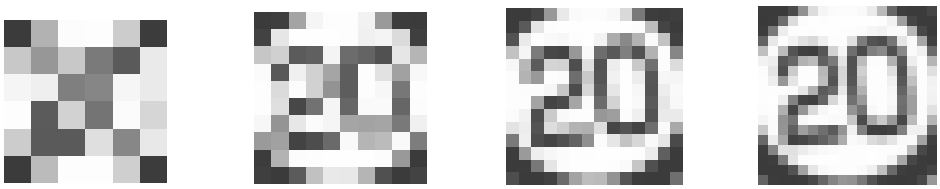
6×6 10×10 14×14 18×18

θ_x [°]	θ_y [°]	θ_z [°]	γ	Δx [cm]	Δy [cm]	w [cm]	h [cm]
−2.00	4.31	−2.25	17.3	2.10	0.91	35.7	38.4



6×6 10×10 14×14 18×18

θ_x [°]	θ_y [°]	θ_z [°]	γ	Δx [cm]	Δy [cm]	w [cm]	h [cm]
−0.88	−4.00	1.19	9.4	−0.28	2.38	36.2	38.6



6×6 10×10 14×14 18×18

θ_x [°]	θ_y [°]	θ_z [°]	γ	Δx [cm]	Δy [cm]	w [cm]	h [cm]
−1.94	4.63	−6.44	9.1	1.01	2.31	39.3	41.8

Figure 5.7: Examples of generated images for “speed limit 20 km/h”.

Figure 5.8: Top three eigenvectors (Speed limit 20 km/h, size 16×16)

5.4 Recognition method

The subspace method [45] is used in the recognition step. The process of constructing a subspace is described in 5.4.1, followed by a description of the recognition step using multiple-frame integration in 5.4.2. A simple algorithm to extract circular traffic signs is outlined in 5.4.3, since it is needed to evaluate the proposed training method.

5.4.1 Construction of a subspace

A subspace is constructed from the generated training images for each category and also for each size.

Let $\boldsymbol{x}^{(c)}_{\{n,d\}}$ be a vector consisting of $d \times d$ pixels of category c's n-th training images; $\boldsymbol{x}^{(c)}_{\{n,d\}}$ is normalized so that its norm is 1, and the mean of its elements is 0. A matrix $\boldsymbol{X}^{(c)}_{d}$ is constructed from N training images ($n = 1, \cdots, N$) by

$$\boldsymbol{X}^{(c)}_{d} = \left[\boldsymbol{x}^{(c)}_{\{1,d\}} \ \cdots \ \boldsymbol{x}^{(c)}_{\{N,d\}} \right]. \tag{5.13}$$

An auto-correlation matrix $\boldsymbol{Q}^{(c)}_{d}$ is computed by

$$\boldsymbol{Q}^{(c)}_{d} = \boldsymbol{X}^{(c)}_{d} \left(\boldsymbol{X}^{(c)}_{d} \right)^{\top}. \tag{5.14}$$

Eigenvectors are derived from $\boldsymbol{Q}^{(c)}_{d}$, of which $\boldsymbol{e}^{(c)}_{\{l,d\}}$ ($l = 1, \cdots, L$) with the largest L ($L < N$) eigenvalues are used for recognition. Figure 5.8 shows examples of the eigenvectors. The reason why the subspaces are constructed for each size d is that size normalization can have an undesirable effect on the matching process. If the image size is changed, the influence of pixel interpolation on very small images is not negligible.

5.4.2 Multiple frame integration

An input image is classified to a category c that maximizes the similarity. In the subspace method, the similarity is given by the sum of the squared inner product between the given image and the eigenvectors. Yanadume et al. demonstrated that integrating similarities from multiple frames improves recognition accuracy [46]. Given M image frames of the same target, let $\boldsymbol{z}_m$ be the m-th input image ($m = 1, \cdots, M$) converted in vector form; the recognition result is obtained by

$$\hat{c} = \arg\max_{c} \sum_{m=1}^{M} \sum_{l=1}^{L} \left(\boldsymbol{e}_{\{l,\bar{d}_m\}}^{(c)\top} \boldsymbol{z}_m \right)^2, \tag{5.15}$$

where $\bar{d}_m$ represents the size of the segmented image $\boldsymbol{z}_m$. In order to distinguish it from the generation parameter d, the size of the captured images is denoted by $\bar{d}_m$.

5.4.3 Circular sign detection

HSV color space [59] is useful for the extraction of symbol regions in circular signs, since H and S are nearly uniform in respect to changes of illumination. A discriminant function for finding the red circumference is defined as

$$\mathrm{red}(x,y) = \begin{cases} 1 & \left(\begin{array}{c} -\pi/9 < H(x,y) < \pi/9 \\ \text{and } 0.2 < S(x,y) \leq 1 \\ \text{and } 30 \leq V(x,y) \leq 255 \end{array} \right) \\ 0 & \text{otherwise} \end{cases}. \tag{5.16}$$

Circular signs is detected by matching a doughnut-shaped structure shown in Fig. 5.9 with segmentation parameters. Here (x_0, y_0) is the center point, R_1 is the symbol area, R_2 is the red circumferential area, and r_1 and r_2 are the radii of R_1 and R_2, respectively. They are represented as

$$R_1 = \left\{ (x,y) \;\middle|\; \sqrt{(x-x_0)^2 + (y-y_0)^2} < r_1 \right\} \tag{5.17}$$

and

$$R_2 = \left\{ (x,y) \;\middle|\; r_1 < \sqrt{(x-x_0)^2 + (y-y_0)^2} < r_2 \right\}. \tag{5.18}$$

The extracted region is the smallest square that includes the entire symbol area. Using Eqs. (5.16), (5.17), and (5.18), segmentation parameters (x_0, y_0) and segmented image size $\bar{d}$ are obtained by

$$\left\{ x_0, y_0, \frac{\bar{d}}{2} \right\} = \arg\max_{\{x,y,r_1\}} \left[\sum_{(x,y)\in R_2} \frac{\mathrm{red}(x,y)}{|R_2|} - \sum_{(x,y)\in R_1} \frac{\mathrm{red}(x,y)}{|R_1|} \right]. \tag{5.19}$$

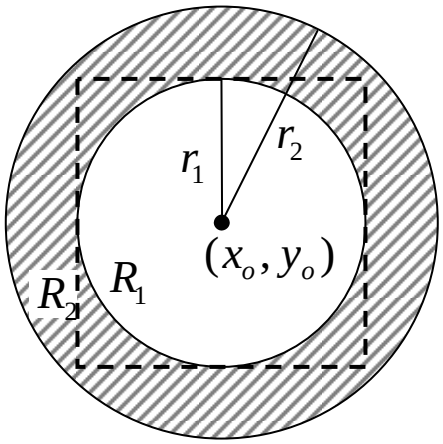


Figure 5.9: Extraction parameters defined for a circular sign

This segmentation algorithm is applied to the input video stream. Searching only neighborhoods of (x_0, y_0) obtained from previous frames is effective for the reduction of computational complexity and false recognition.

5.5 Experiment

An experiment was performed using video streams captured by a car-mounted camera (Table 5.4) during one run on a sunny morning. Figure 5.10 illustrates twenty circular traffic signs commonly used in Japan. The video stream contained fifteen traffic signs: two of No. 2, five of No. 4, three of No. 5, three of No. 12, and two of No. 20. They were divided into five data sets per category, so as to ascertain whether parameters estimated from different category sets were valid for training. Table 5.5 shows these data sets (sets A–E) and the number of their symbol images successfully detected by the algorithm in 5.4.3. In this experiment, each data set was chosen once for the parameter estimation, and the remaining four sets were used for testing. In other words, each bit of data was evaluated as test samples by changing the training sets four times. Figure 5.11 shows the size distribution of the segmented images, and Fig. 5.12 shows examples of the images.

In the training step, the parameter distribution was estimated using the algorithm in Table 5.3 with $N_c = 100$, $G = 100$, $P_c = 0.7$, and $P_m = 0.01$. Instead of Eq. (5.12), training images were generated by a parameter producing function in

Table 5.4: Specifications of the car-mounted camera

Product model	Sony DCR-PC105
Resolution	720×480
Frame rate	30 fps
Focus length	3.7 mm

Table 5.5: Number of symbol images in each data set

Set	Category	Number of symbol images
A	No. 2	174
B	No. 4	356
C	No. 5	214
D	No. 12	214
E	No. 20	115

which $\boldsymbol{\Sigma}^{1/2}$ was weighted on as

$$\boldsymbol{g} = k\boldsymbol{\Sigma}^{1/2}\boldsymbol{r} + \boldsymbol{\mu}, \tag{5.20}$$

where k is considered as a factor that controls the parameter range by weighting on the estimated $\boldsymbol{\Sigma}^{1/2}$. The number of the generated training images was 200 ($N = 200$). Recognition rates in six cases ($k = 0, 1/4, 1/2, 1, 2, 4$) were compared. In the case of $k = 0$, however, only a single training image ($\boldsymbol{g} = \boldsymbol{\mu}$) was obtained from Eq. (5.20). Hence in this case, the input images were classified by

$$\hat{c} = \arg\max_{c} \sum_{m=1}^{M} \left(\boldsymbol{x}_{\bar{d}_m}^{(c)\top} \boldsymbol{z}_m \right) \tag{5.21}$$

with a single training image $\boldsymbol{x}_{\bar{d}_m}^{(c)}$. In the other cases, recognition results were obtained by Eq. (5.15). The case of $k = 1$ was identical to the proposed method, since Eq. (5.20) equals Eq. (5.12). In the recognition step, ten successive frames were integrated ($M = 10$), and ten eigenvectors were used ($L = 10$).

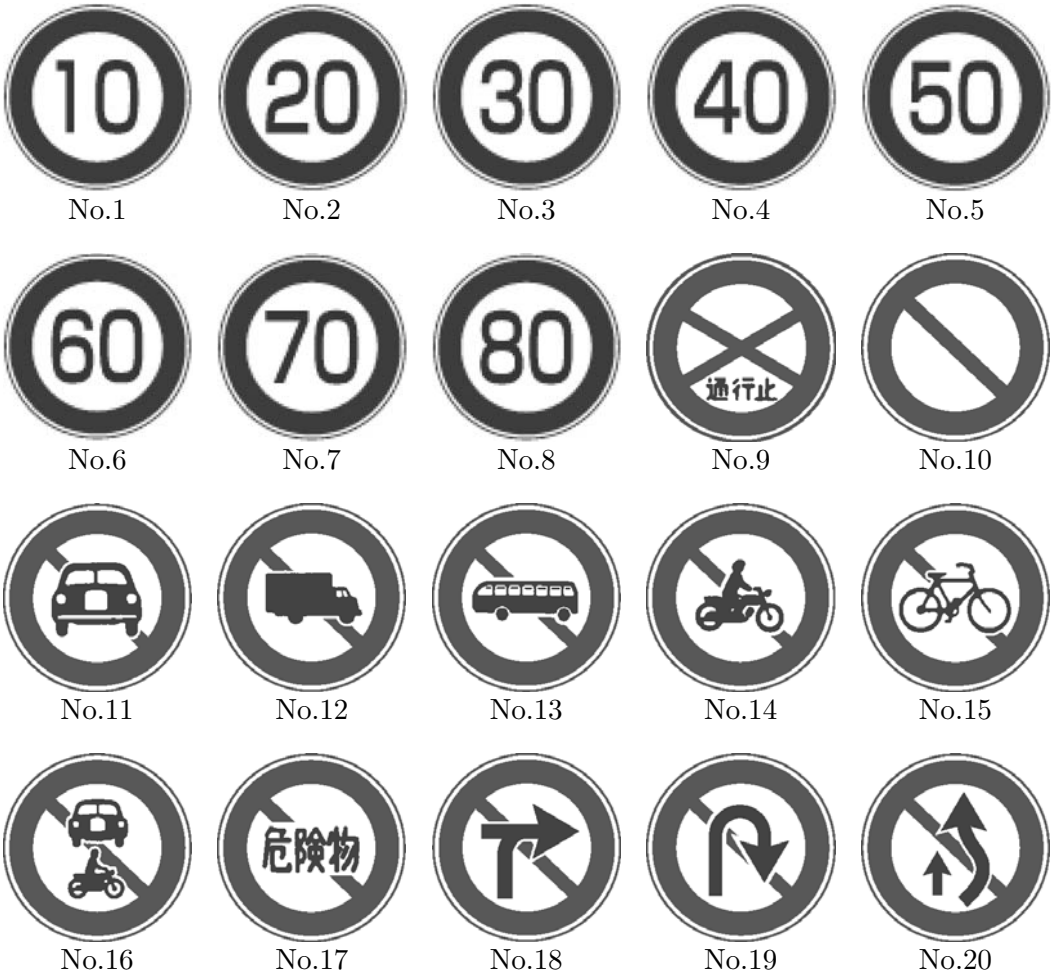


Figure 5.10: Traffic sign categories

5.5.1 Results

Recognition rates are presented in Fig. 5.13, where the horizontal axis in the graph represents the maximum symbol size $\bar{d}_{\max}$ within the integrated M frames. As shown in the results, the recognition rates have strong relationships with the image sizes. The proposed method exhibited high recognition rates; the recognition rate of relatively large symbols ($\bar{d}_{\max} \geq 20$) was 100%. For small symbols ($\bar{d}_{\max} < 10$), it was 84.4%. In Table 5.6, overall recognition rates are presented together with rates from single frame recognition ($M = 1$). Compared with the case of $k = 0$, in which an average pattern was learned, the recognition rates improved drastically. Although the other cases of k ($k = 1/4, 1/2, 2, 4$) also exhibited high recognition

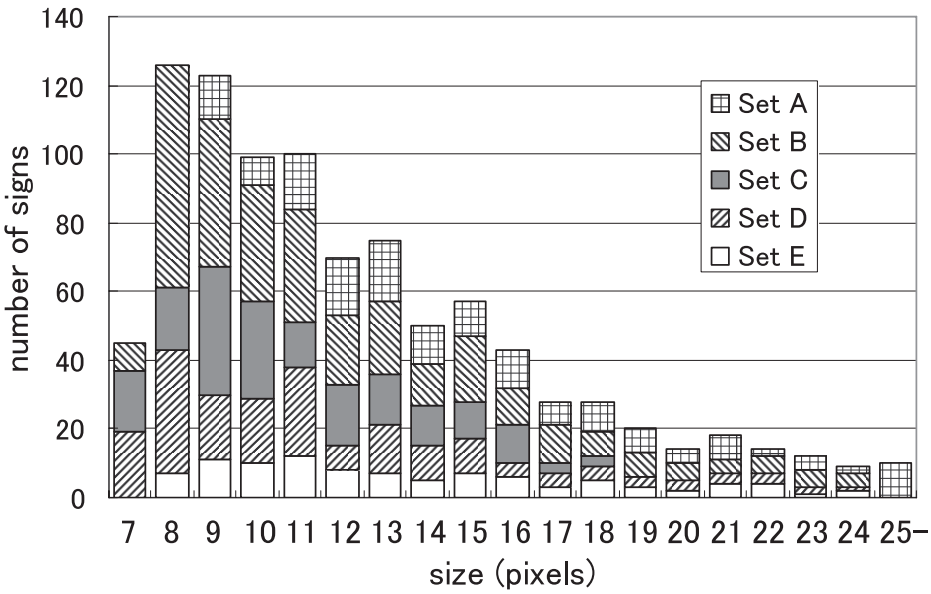


Figure 5.11: Distribution of traffic sign size

Table 5.6: Average recognition rates from single frame ($M = 1$) and multiple frame integration ($M = 10$)

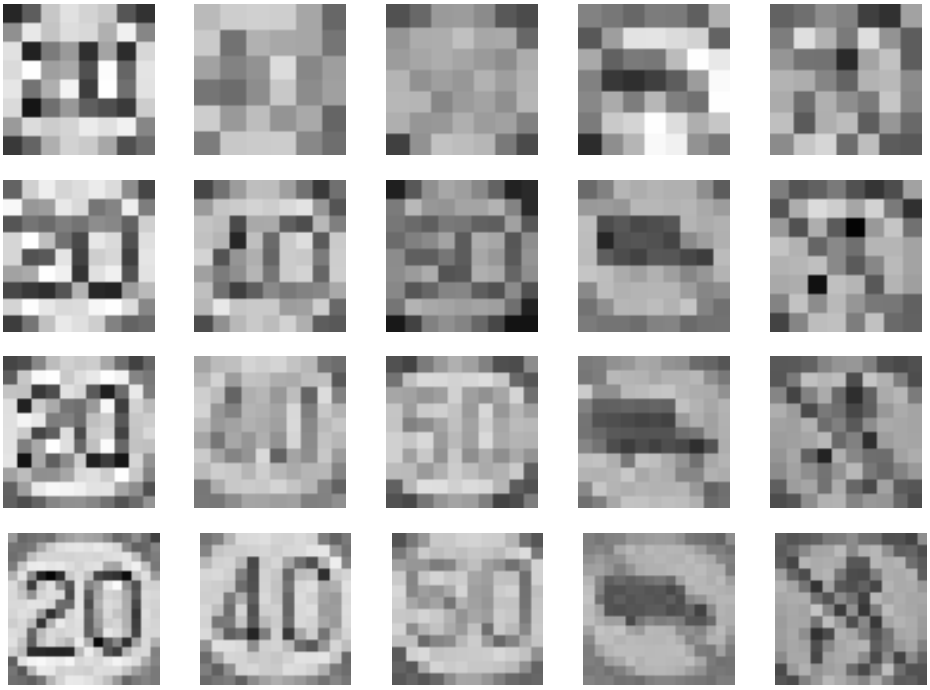
weight$k$	0	1/4	1/2	1	2	4
Single frame	48.0	81.7	83.4	84.3	82.7	82.4
Multiple frames	57.4	89.2	91.7	92.9	91.4	91.2

rates, the case of $k = 1$ was the most effective. Some examples of the recognition results are presented in Fig. 5.14.

5.5.2 Discussion

It is worthy of noting that the case using the estimated distribution ($k = 1$) was the most appropriate for recognizing traffic signs captured in similar conditions. This result indicates that GA-based parameter estimation successfully worked. It exhibited also the superiority of the proposed method over the other values of k.

Since most of the available traffic sign images are small as presented in Fig. 5.11,



Set A Set B Set C Set D Set E

Figure 5.12: Examples of test images

robustness to low-resolution images is important for real-world applications. Nevertheless, the recognition rate was not high enough where the image size was very small ($\bar{d}_{\max} < 10$). One reason is that small signs are especially sensitive to the degradation factors. It implies the dependency of parameters, which are listed in Eq. 5.9, on size parameter d. For the sake of simplicity, the proposed method assumes independence of d from the other parameters. A better representation for parameter distribution should be discussed in future works.

Table 5.7 shows the recognition rates of the proposed method ($k = 1$) for each data set. A sufficient performance should be obtained also from the case where different sets were used for estimation and testing. Non-diagonal elements in Table 5.7 show the results of such cases. However, they did not exhibit high recognition rates especially when sets A and C were used for testing, compared with the case where the same set was used both for estimation and testing. This is partly due to the distribution of traffic sign size in Fig. 5.11; set A was composed mostly of large images, and set C was composed mostly of small images. Moreover, the recognition rates were lower when sets D and E were used for parameter estimation. One expla-

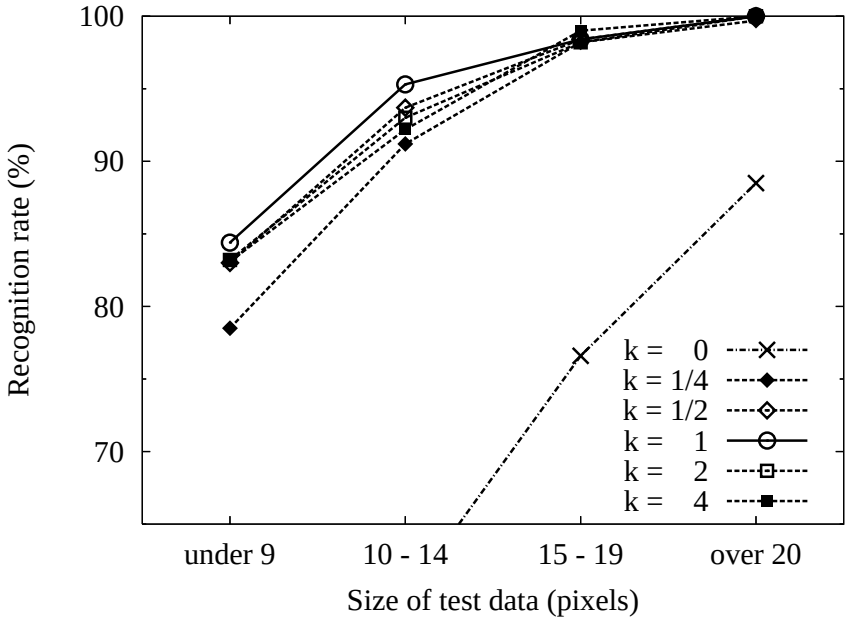


Figure 5.13: Recognition results according to the maximum size of traffic sign symbol images in multiple frames

nation is that the parameter distribution was not satisfactorily estimated because of structural simplicity of the original traffic sign symbols. Table 5.8 shows the complexities calculated for the traffic sign symbols, where the complexity is defined by edge density as introduced in [60]. Altogether, parameters should preferably be estimated from images of various sizes using structurally complex symbols.

5.6 Summary

In this chapter, a method for recognizing traffic sign symbols was proposed. Degradation parameters were defined in order to generate variously degraded training images. Based on the generated models, degradation characteristics were estimated from a small number of captured images. The estimated characteristics were learned via the generated training images. The usefulness of our method for degraded traffic sign symbol images was experimentally demonstrated.

The proposed method is applicable for any traffic sign by combining it with existing traffic sign detection methods [33]–[42]. In future works, the effectiveness of the proposed method should be evaluated under various weather conditions and at

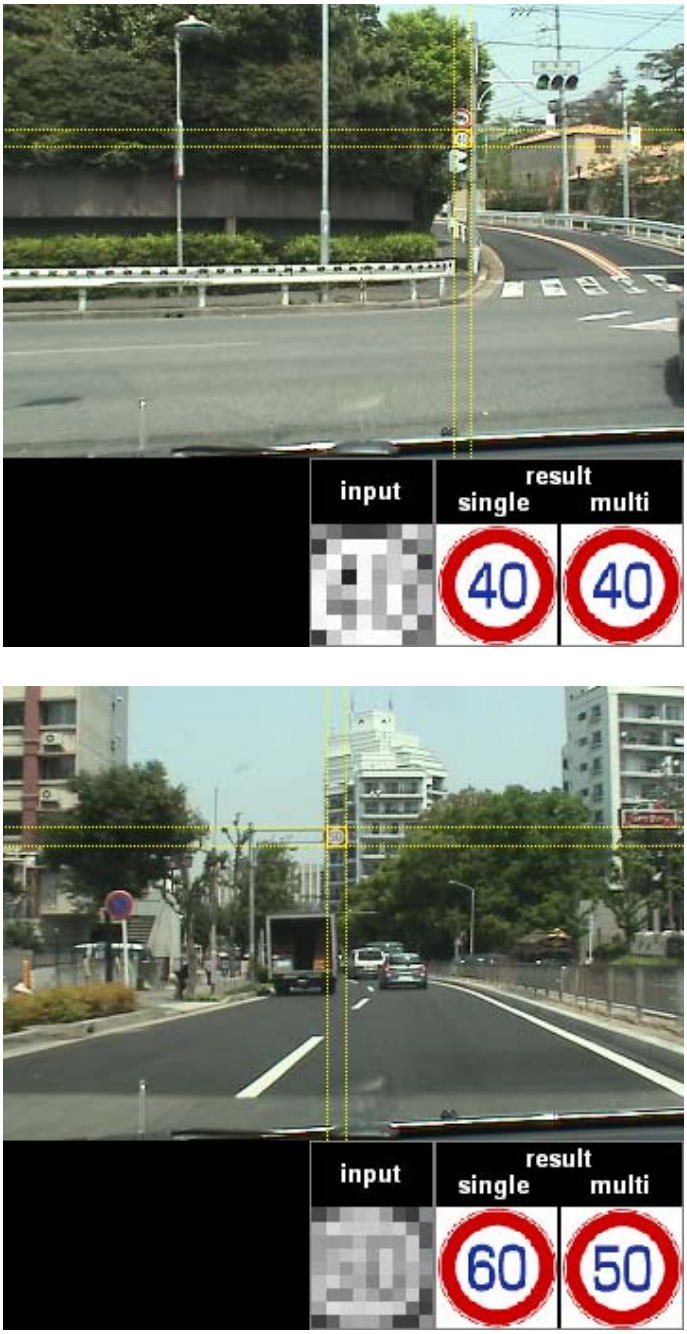


Figure 5.14: Video stream demonstrating the recognition results. Traffic signs shown at the bottom of the images are the extracted symbol, result of single-frame recognition ($M = 1$), and result of multiple-frame recognition ($M = 10$), from left to right. Whereas the single-frame recognition sometimes gave incorrect results, the multiple-frame recognition gave the correct result at higher rates.

Table 5.7: Recognition rates of proposed method ($k = 1$) for each training and test set.

	Recognition rates for test data [%]						
	Single frame						Multiple frames
Training data	Set A	Set B	Set C	Set D	Set E	Average	Average
Set A	97.1	68.0	68.2	98.6	100	82.3	93.5
Set B	97.7	78.4	72.0	100	100	86.9	95.8
Set C	93.7	82.6	82.7	100	100	89.7	98.5
Set D	91.4	83.1	65.4	98.6	100	85.8	91.5
Set E	89.1	76.4	56.1	98.1	100	81.3	90.0

Table 5.8: Edge density measured from original traffic sign symbol images of 56×56 pixels.

Set	A	B	C	D	E
Category	No. 2	No. 4	No. 5	No. 12	No. 20
Edge density	0.064	0.061	0.065	0.046	0.055

various times of day. Also, application to the recognition of other on-road objects will be interesting.

Chapter 6

Conclusion

This thesis presented a generative learning method for camera-based recognition systems and its three real-world applications. The framework of the generative leaning method consists of estimation of degradation characteristics such as point spread functions (PSFs) or parameter distributions. Training images are generated based on the estimated degradation characteristics. Together with consistent application of the generative learning methods, recognition methods for camera-captured characters, character-strings, and for traffic sign symbols were presented. All the methods take advantages of the generative learning method.

In the camera-based character recognition task, blurring and vibration of hands holding a camera seriously affect recognition accuracy. An optical blur PSF is estimated to treat the problem of the optical blur. Also, a motion blur PSF determined by two parameters is introduced. Using these PSFs in the generation stage of training images, the recognition accuracy of low-quality characters is improved. The proposed recognition method uses the blur parameters also for the recognition stage. Such blur parameters are considered to be useful to distinguish structurally similar characters that are often misclassified to each other.

In the character-string recognition task, training images of individual characters are generated. Using the subspace method, eigenvectors of the generated training images are used as elastic templates for the matching, which enables to identify the characters in the low-resolution character-string images. In addition to the individual characters, the features between two adjacent characters are used for segmentation. The hypothesis graph of the recognition results is constructed by combining both features from characters and inter-character spaces. The training samples of the inter-character spaces also are obtained from the generated training images.

In case of the traffic sign recognition task by a car-mounted camera, appropriate parameters should be provided in the generation step of training images. The proposed method estimates the parameter distribution from a small number of captured samples by means of the genetic algorithm. The estimated distributions are considered to be appropriate for the generation of the training images, since the parameters are obtained from the actual images.

Results showed that the proposed methods were effective for these three applications, which indicates also that the generative learning method should be applicable to various camera-based recognition tasks suffering from various image degradations such as low-resolution. However, many challenges still exist, involving shift-variant blur, occlusion, distortion, and extraction error. Future work includes examining a more effective way of modeling and simulation.

Acknowledgements

I thank to Professor Hiroshi Murase for guiding me and for providing me with an opportunity to work on this study. I thank Associate Professor Ichiro Ide for providing me with a lot of valuable advice on this study and for helping me write papers. I thank to Professor Yasuhito Suenaga for encouraging me to work on this study and for his valuable comments on this thesis. I thank to Professor Yoshito Mekada of Chukyo University for teaching me many knowledge about image processing. I thank to Associate Professor Tomokazu Takahashi of Gifu Shotoku Gakuen University for many helpful advices on my works. I thank to Assistant Professor Daisuke Deguchi for providing us students with a wonderful image processing library MIST [58]. I thank to Mr. Shinsuke Yanadume for his advices on the study on character recognition. Finally, I thank to all of the members of Murase Laboratory and Suenaga Laboratory for their assistance.

Parts of this research were supported by the Grants-In-Aid for JSPS Fellows (19-6540) and Scientific Research (16300054).

Publications

Journal papers

- H. Ishida, T. Takahashi, I. Ide, and H. Murase, "Generation of templates for low-resolution text recognition using a hypothesis graph," Pattern Recognition and Image Analysis, MAIK Nauka/Interperiodica distributed, vol. 18, no. 4, pp. 638–642, December 2008.
- H. Ishida, T. Takahashi, I. Ide, Y. Mekada, and H. Murase, "Recognition of camera-captured low-quality characters using motion blur information," Pattern Recognition, Elsevier, vol. 41, no. 7, pp. 2253–2262, July 2008.
- H. Ishida, T. Takahashi, I. Ide, Y. Mekada, and H. Murase, "Generation of training data by degradation models for traffic sign symbol recognition," IEICE Trans., vol. E90-D, no. 8, pp. 1134–1141, August 2007.
- H. Ishida, T. Takahashi, I. Ide, Y. Mekada, and H. Murase, "A generative learning method for the recognition of blurred characters taken by portable cameras (in Japanese)," IEICE Trans., vol. J89-D, no. 9, pp. 2055–2064, September 2006.

Letters

- H. Ishida, T. Takahashi, I. Ide, and H. Murase, "Recognition of character strings in low-quality images using inter-character features," Information Technology Letters, pp. 229–232, September 2007.

Articles

- H. Ishida, T. Takahashi, I. Ide, and H. Murase, "A generative learning method for traffic sign recognition using a car-mounted camera (in Japanese)," Gazo Labo, Nikkan Kogyo Shuppan, vol.19, no.9, pp.36–39, September 2008.

Conference papers (International)

- H. Ishida, T. Takahashi, I. Ide, and H. Murase, "A Hilbert warping method for camera-based finger-writing recognition," Proc. 19th International Conference on Pattern Recognition, ThCT5.2(CD-ROM), Tampa, Florida, USA, December 2008.

- H. Ishida, T. Takahashi, I. Ide, and H. Murase, "A Hilbert warping algorithm for recognizing characters from moving camera," Proc. 8th IAPR Workshop on Document Analysis Systems, pp. 21–27, Nara, Japan, September 2008.

- H. Ishida, T. Takahashi, I. Ide, Y. Mekada, and H. Murase, "Recognition of camera-captured characters in blurred image using motion-blur parameters," Proc. 8th International Conference on Pattern Recognition and Image Analysis, vol. 1, pp. 126–130, Yoshkar-Ola, Russia, October 2007.

- H. Ishida, T. Takahashi, I. Ide, H. Murase, "Recognition of character strings in low-quality images using character and inter-character space patterns," Proc. 9th International Conference on Document Analysis and Recognition, vol. 1, pp. 302–306, Curitiba, Brazil, September 2007.

- H. Ishida, T. Takahashi, I. Ide, Y. Mekada, and H. Murase, "Identification of degraded traffic sign symbols by a generative learning method," Proc. 18th International Conference on Pattern Recognition, vol. 1, pp. 531–534, Hong Kong, China, August 2006.

- H. Ishida, S. Yanadume, T. Takahashi, I. Ide, Y. Mekada, and H. Murase, "Recognition of low-resolution characters by a generative learning method", Proc. 1st International Workshop on Camera-Based Document Analysis and Recognition, pp. 45–51, Seoul, Korea, August 2005.

Domestic symposia

- H. Ishida, T. Takahashi, I. Ide, and H. Murase, "Finger action recognition by a Hilbert warping algorithm (in Japanese)", Proc. Meeting on Image Recognition and Understanding (MIRU) 2008, pp. 1162–1168, July 2008.
- H. Ishida, T. Takahashi, I. Ide, and H. Murase, "Character recognition from image sequences captured by moving camera using Hilbert transform (in Japanese)", Proc. Meeting on Image Recognition and Understanding (MIRU) 2007, pp. 141–148, July 2007.
- H. Ishida, T. Takahashi, I. Ide, Y. Mekada, and H. Murase, "A recognition method for camera-captured low-quality characters using blur information (in Japanese)", Proc. Meeting on Image Recognition and Understanding (MIRU) 2006, pp. 180–186, July 2006.
- H. Ishida, T. Takahashi, I. Ide, H. Murase, and M. Enomoto, "A study on the generation process of training data for traffic sign recognition (in Japanese)", Proc. Meeting on Image Recognition and Understanding (MIRU) 2005, pp. 989–996, July 2005.
- H. Ishida, S. Yanadume, I. Ide, Y. Mekada, and H. Murase, "Generative learning method for the recognition of low-resolution characters using the subspace method (in Japanese)", Technical Report of IEICE, PRMU2004-7, pp. 41–48, May 2004.
- H. Ishida, T. Takahashi, I. Ide, Y. Mekada, and H. Murase, "Recognition of low resolution traffic signs using a generative learning (in Japanese)", Proc. IEICE 2005 Annual Convention, A-17-3, p.308, March 2005.
- H. Ishida, T. Takahashi, I. Ide, Y. Mekada, and H. Murase, "Parameter estimation to generate training images for traffic sign recognition (in Japanese)", 2005 Electric Engineering Related Society Tokai Sectors Joint Symposium, O-241, September 2005.
- H. Ishida, Y. Mekada, I. Ide, and H. Murase, "Recognition of low resolution traffic signs using a subspace method (in Japanese)", 2004 Electric Engineering Related Society Tokai Sectors Joint Symposium, O-346, September 2004.

Awards

- H. Ishida, T. Takahashi, I. Ide, and H. Murase, "Recognition of Character Strings in Low-quality Images Using Inter-character Features," FIT2007 paper award, September 2007.

Bibliography

[1] H. Murase, "Generative learning for image recognition (in Japanese)," Trans. IPS Japan, vol. 46 no. SIG15, CVIM-12, pp. 35–42, October 2005.

[2] K. Ishii, "Generation of distorted characters and its applications," Systems and Computers in Japan, vol. 14, no. 6, pp. 19–27, June 1983.

[3] T. Horiuchi, K. Toraichi, L. Yamamoto, and H. Yamada, "On method of training dictionaries for handwritten character recognition using relaxation matching," Proc. 2nd Int. Conf. on Document Analysis and Recognition, pp. 638–641, Tsukuba, Japan, October 1993.

[4] D. Doermann, J. Liang, and H. Li, "Progress in camera-based document image analysis," Proc. 5th Int. Conf. on Document Analysis and Recognition, pp. 606–616, Edinburgh, Scotland, August 2003.

[5] V. Govindan and A. Shivaprasad, "Character recognition – a review," Pattern Recognition, Elsevier Science Inc., vol. 23, no. 7, pp. 671–683, New York, NY, USA, July 1990.

[6] S. Mori, K. Yamamoto, and M. Yasuda, "Research on machine recognition of handprinted characters," IEEE Trans. PAMI, vol. 6, no. 4, pp. 386–405, July 1984.

[7] S. Kahan, T. Pavilidis, and H. Baird, "On the recognition of printed characters of any font and size," IEEE Trans. PAMI, vol. 9, no. 2, pp. 274–288, March 1987.

[8] H. Murase, H. Kimura, M. Yoshimura, and Y. Miyake, "An improvement of the auto-correlation matrix in the pattern matching method and its application to handprinted HIRAGANA recognition (in Japanese)," IEICE Trans., vol. J64-D, no. 3, pp. 276–283, March 1981.

[9] S. Mori and M. Sawaki, "A survey on robust character recognition and its application (in Japanese)," Technical Report of IEICE, PRMU2001-275, February 2002.

[10] K. Kise, S. Omachi, S. Uchida, and M. Iwamura, "Current status and future prospects of camera-based character recognition and document image analysis (in Japanese)," Technical Report of IEICE, PRMU2004-246, February 2005.

[11] H. Andrew and B. Hunt, "Digital image restoration," Prentice-Hall, Englewood Cliffs, NJ, USA, 1977.

[12] J. Hobby and H. Baird, "Degraded character image restoration," Proc. 5th UNLV Symp. on Document Analysis and Information Retrieval, Las Vegas, USA, pp. 177–189, April 1996.

[13] H. Li and D. Doermann, "Text enhancement in digital video using multiple frame integration," Proc. 7th ACM Int. Conf. on Multimedia, pp. 19–22, Orlando, FL, USA, November 1999.

[14] C. Mancas-Thillou and M. Mirmehdi, "Super-resolution text using the Teager filter," Proc. 1st Int. Workshop on Camera-Based Document Analysis and Recognition, pp. 10–16, Seoul, Korea, August 2005.

[15] N. Tsunashima and M. Nakajima, "Estimation of point spread function using compound method and restoration of blurred images (in Japanese)," IEICE Trans., vol. J81-D-II, no. 11, pp. 2688–2692, November 1998.

[16] K. Fujimoto, K. Fujita, and Y. Yoshida, "Restoration from multi-frame blurred images based on stochastic image models (in Japanese)," IEICE Trans., vol. J82-D-II, No. 5, pp. 863–871, May 1999.

[17] S. Hashimoto and H. Saito, "Restoration of shift variant blur blurred image estimating the parameter distribution of PSF (in Japanese)," IEICE Trans., vol. J77-D-II, no. 4, pp. 719–728, April 1994.

[18] Y. Yitzhaky and N. Kopeika, "Identification of blur parameters from motion blurred images," Graphical Models and Image Processing, Academic Press, vol. 59, no. 5, pp. 310–320, September 1997.

[19] F. Kobayashi, H. Tsuboi, M. Tanaka, and R. Misaki, "Restoration of blurred image by a direct approach (in Japanese)," IIEEJ Trans., vol. 22, no. 3, pp. 247–254, June 1993.

[20] M. Ben-Ezra and S. Nayar, "Motion-based motion deblurring," IEEE Trans. PAMI, vol. 26, no. 6, pp. 689–698, June 2004.

[21] S. Wachenfeld, H. Klein, and X. Jiang, "Recognition of screen-rendered text," Proc. 18th Int. Conf. on Pattern Recognition, pp. 1086–1089, Hong Kong, China, August 2006.

[22] R. Casey and E. Lecolinet, "A survey of methods and strategies in character segmentation," IEEE Trans. PAMI, vol. 18, no. 7, pp. 690–706, July 1996.

[23] H. Murase, T. Wakahara, and M. Umeda, "Online writing-box free character string recognition by candidate character lattice method (in Japanese)," IEICE Trans., vol. J-68-D, no. 4, pp. 765–772, April 1985.

[24] J. Sun, Y. Hotta, K. Fujimoto, Y. Katsuyama, and S. Naoi, "Grayscale feature combination in recognition based segmentation for degraded text string recognition," Proc. 1st Int. Workshop on Camera-Based Document Analysis and Recognition, pp. 39–44, Seoul, Korea, August 2005.

[25] D. Mochiduki, Y. Yano, T. Hashiyama, and S. Okuma, "Pedestrian detection with a vehicle camera using fast template matching based on background elimination and active search (in Japanese)," IEICE Trans., vol. J87-D-II, no. 5, pp. 1094–1103, May 2005.

[26] F. Lindner, U. Kressel, and S. Kaelberer, "Robust recognition of traffic signals," Proc. IEEE 2004 Intelligent Vehicles Symp., pp. 49–53, Parma, Italy, June 2004.

[27] F. Kimura, T. Takahashi, Y. Mekada, I. Ide, H. Murase, T. Miyahara, and Y. Tamatsu, "Measurement of visibility conditions toward smart driver assistance for traffic signals," Proc. IEEE 2007 Intelligent Vehicles Symp., pp. 636–641, Istanbul, Turkey, June 2007.

[28] M. Noda, T. Takahashi, Y. Mekada, I. Ide, and H. Murase, "Recognition of road markings from in-vehicle camera images using generative learning method (in Japanese)," Technical Report of IEICE, PRMU2008-93, pp. 41–48, October 2008.

[29] H. Uchiyama, T. Takahashi, I. Ide, and H. Murase, "Frame registration of in-vehicle normal camera with omni-directional camera for self-position estimation," Proc. of 3rd IEEE Int. Conf. on Innovative Computing, Information and Control, WS01-007(CD-ROM), Dalian, China, June 2008.

[30] N. Shibuhisa, J. Sato, T. Takahashi, I. Ide, H. Murase, Y. Kojima, and A. Takahashi, "Accurate vehicle localization using DTW between range data map and laser scanner data sequences," Proc. IEEE 2007 Intelligent Vehicles Symp., pp. 975–980, Istanbul, Turkey, June 2007.

[31] H. Kurihata, T. Takahashi, Y. Mekada, I. Ide, H. Murase, Y. Tamatsu, and T. Miyahara, "Rainy weather recognition from in-vehicle camera images for driver assistance," Proc. IEEE 2005 Intelligent Vehicles Symp., pp. 204–209, Las Vegas, NV, USA, June 2005.

[32] S. Kamijo, K. Okumura, and A. Kitamura, "Digital road map database for vehicle navigation and road information systems," Proc. Int. Conf. on Vehicle Navigation and Information Systems, pp. 319–323, Toronto, Canada, September 1989.

[33] A. Escalera and M. Salichs, "Road traffic sign detection and classification," IEEE Trans. Industrial Electronics, vol. 44, no. 12, pp. 848–859, December 1997.

[34] D. Gavrila, "Multi-feature hierarchical template matching using distance transforms," Proc. 14th IEEE Int. Conf. on Pattern Recognition, vol. 1, pp. 439–444, Brisbane, Australia, August 1998.

[35] J. Miura, T. Kanda, and Y. Shirai, "An active vision system for real-time traffic sign recognition," Proc. IEEE 2000 Intelligent Transportation Systems, pp. 52–57, Dearborn, MO, USA, June 2000.

[36] G. Mo and Y. Aoki, "A recognition method for traffic sign in color image (in Japanese)," IEICE Trans., vol. J87-D-II, no. 12, pp. 2124–2135, December 2004.

[37] K. Uchimura, H. Kimura, and S. Wakiyama "Extraction and recognition of circular road signs using road scene color images (in Japanese)," IEICE Trans., vol. J81-A, no. 4, pp. 546–553, April 1998.

[38] D. Matsuura, H. Yamauchi, and H. Takahashi, "Extracting circular road signs using specific color distinction and region limitation (in Japanese)," IEICE Trans., vol. J85-D-II, no. 6, pp. 1075–1083, June 2002.

[39] S. Lafuente-Arroyo, P. Gil-Jimenez, R. Maldonado-Bascon, and F. Lopez-Ferreras, "Traffic sign shape classification evaluation I: SVM using distance to borders," Proc. IEEE 2005 Intelligent Vehicles Symposium, pp. 557–562, June 2005.

[40] P. Gil-Jimenez, S. Lafuente-Arroyo, H. Gomez-Moreno, F Lopez-Ferreras, and S. Maldonado-Bascon, "Traffic sign shape classification evaluation II: FFT applied to the signature of blobs," Proc. IEEE 2005 Intelligent Vehicles Symp., pp. 607–612, Las Vegas, NV, USA, June 2005.

[41] C. Bahlmann, Y. Zhu, V. Ramesh, M. Pellkofer, and T. Koehler, "A system for traffic sign recognition, tracking, and recognition using color, shape, and motion information," Proc. IEEE 2005 Intelligent Vehicles Symp., pp. 255–260, Las Vegas, NV, USA, June 2005.

[42] B. Johansson, "Road sign recognition from a moving vehicle," Master's thesis, Center for Image Analysis, Swedish University of Agricultural Science, 2002.

[43] H. Murase, S. Nayar, "Three-dimensional object recognition from appearance — parametric eigenspace method," Systems and computers in Japan. vol. 26, no. 8, pp. 45–54, August 1995. .

[44] M. Potmesil, "Modeling motion blur in computer-generated images," Computer Graphics, vol. 17, no. 3, pp. 389–399, July 1983.

[45] E. Oja, "Subspace methods of pattern recognition," Research Studies, Hertfordshire, UK, 1983.

[46] S. Yanadume, Y. Mekada, I. Ide, and H. Murase, "Recognition of very low-resolution characters from motion images," Proc. 2004 Pacific-Rim Conf. on Multimedia, Lecture Notes in Computer Science, Springer-Verlag, vol. 3331, pp. 247–254, December 2004.

[47] P. Clark and M. Mirmehdi, "Recognising text in real scenes," Int. Journal of Document Analysis and Recognition, Springer-Verlag, vol. 4, no. 4, pp. 243–257, July 2002.

[48] G. Myers, R. Bolles, Q. Luong, J. Herson, and H. Aradhye, "Rectification and recognition of text in 3-D scenes," Int. Journal of Document Analysis and Recognition, Springer-Verlag, vol. 7, no. 2–3, pp. 147–158, July 2005.

[49] H. Ezaki, S. Uchida, A. Asano, and H. Sakoe, "Dewarping of document image by global optimization," Proc. 8th Int. Conf. on Document Analysis and Recognition, pp. 302–306, Seoul, Korea, August 2005.

[50] K. Chua, L. Zhang, Y. Zhang, and C. Tan, "A fast and stable approach for restoration of warped document images," Proc. 8th Int. Conf. on Document Analysis and Recognition, pp. 384–388, Seoul, Korea, August 2005.

[51] T. Breuel, "Segmentation of handprinted letter strings using a dynamic programming algorithm," Proc. 6th Int. Conf. on Document Analysis and Recognition, pp. 821–826, Seattle, USA, September 2001.

[52] T. Kawahara, M. Nishiyama, and O. Yamaguchi, "Face recognition by orthogonal mutual subspace method (in Japanese)," IPS Japan SIG Technical Report, CVIM–151, pp. 17–24, November 2005.

[53] Y. Yang, "An evaluation of statistical approaches to text categorization," Journal of Information Retrieval, Kluwer Academic Publishers, vol. 1, no. 1–2, pp. 69–90, April 1999.

[54] The constitution of Japan (in English), Prefaces, November 1946. (*http://www.solon.org/Constitutions/Japan/English/english-Constitution.html*)

[55] L. Daris, "Handbook of genetic algorithms," Van Nostrand Reinhold, New York, NY, USA, 1991.

[56] J. von Neumann, "Various techniques used in connection with random digits," National Bureau of Standards Series, no. 12, pp. 36–38, 1951.

[57] G. Golub and C. Loan, "Matrix computations," 2nd ed., Cambridge Univ. Press, Cambrigde, UK, 1992.

[58] MIST project, *http://mist.suenaga.m.is.nagoya-u.ac.jp/trac-en/*.

[59] A. Smith, "Color Gamut transform pairs," Computer Graphics, Springer, vol. 12, no. 3, pp. 12–19, August 1978.

[60] K. Hirayama, S. Omachi, and H. Aso, "String extraction from scene images using color and luminance information (in Japanese)," IEICE Trans., vol. J-89-D, no. 4, pp. 893–896, April 2006.