Automatic Detection and Recognition of Signs From Natural Scenes

Xilin Chen, *Member, IEEE*, Jie Yang, *Member, IEEE*, Jing Zhang, and Alex Waibel, *Member, IEEE*

***Abstract*—In this paper, we present an approach to automatic detection and recognition of signs from natural scenes, and its application to a sign translation task. The proposed approach embeds multiresolution and multiscale edge detection, adaptive searching, color analysis, and affine rectification in a hierarchical framework for sign detection, with different emphases at each phase to handle the text in different sizes, orientations, color distributions and backgrounds. We use affine rectification to recover deformation of the text regions caused by an inappropriate camera view angle. The procedure can significantly improve text detection rate and optical character recognition (OCR) accuracy. Instead of using binary information for OCR, we extract features from an intensity image directly. We propose a local intensity normalization method to effectively handle lighting variations, followed by a Gabor transform to obtain local features, and finally a linear discriminant analysis (LDA) method for feature selection. We have applied the approach in developing a Chinese sign translation system, which can automatically detect and recognize Chinese signs as input from a camera, and translate the recognized text into English.**

***Index Terms*—Affine rectification, optical character recognition (OCR), sign detection, sign recognition, text detection.**

I. Introduction

We work, live, and play in a so-called information society where we communicate with people and information systems through diverse media in increasingly varied environments. A wealth of information is embedded in natural scenes. Signs are good examples of objects in natural environments that have high information content. They make our lives easier when we are familiar with them, but they pose problems or even danger when we are not. For example, a tourist might not be able to understand a sign in a foreign country that specifies warnings or hazards. Automatic sign translation, in conjunction with spoken language translation, can help international tourists to overcome these barriers.

Signs are everywhere in our lives. A sign is an object that suggests the presence of a fact. It can be a displayed structure bearing letters or symbols, used to identify something or advertise a place of business. It can also be a posted notice bearing a designation, direction, safety advisory, or command. In this research we focus on detecting and recognizing text on signs. An automatic sign translation system utilizes a video camera to capture an image containing signs, detects signs in the image, recognizes signs, and translates results of sign recognition into a target language. Automatic detection and recognition of text from natural scenes are prerequisites for an automatic sign translation task.

Automatic detection and recognition of text from natural scenes is a very difficult task. The primary challenge lies in the variety of text: it can vary in font, size, orientation, and position. Text can also be blurred from motion or occluded by other objects. As signs exist in three-dimensional space, text on signs in scene images can be distorted by slant, tilt, and shape of objects on which they are found [18]. In addition to the horizontal left-to-right orientation, other orientations include vertical, circularly wrapped around another object, slanted, sometimes with the characters tapering (as in a distinct angle away from the camera), and even mixed orientations within the same text area, such as text on a T-shirt or wrinkled sign. Although many commercial OCR systems work well on high quality scanned documents under a controlled environment, they have a much higher error rate for sign recognition tasks because of low quality images.

In this paper, we propose an approach for automatic detection and recognition of text from natural scenes. The proposed approach embeds multiresolution and multiscale edge detection, adaptive searching, color analysis, and affine rectification in sign detection. Compared with the existing text detection algorithms, this new framework can better handle the dynamics of text detection in natural scenes. Instead of using only binary information as most other OCR systems, we extract features from the intensity image directly, and avoid potentially losing information during the binarization processing, which is irreversible. We propose a local intensity normalization method to effectively handle luminance variations of the captured character image. We then utilize a Gabor transform to obtain local features and an LDA method for feature selection. We have successfully applied the proposed approach to a Chinese-English sign translation system, which can automatically detect and recognize Chinese signs captured from a camera, and translate the recognized text into English.

The rest of this paper is organized as follows: In Section II, we discuss problems in automatic detection and recognition of signs with text, the related work, and the approach to the problems. In Section III, we present the method for text detection, including multiresolution and multiscale edge detection, adaptive searching, color analysis, affine rectification, and layout

Manuscript received June 27, 2002; revised July 7, 2003. This work was supported in part by DARPA under TIDES project. The associate editor coordinating the review of this manuscript and approving it for publication was Dr. Jianying Hu.

X. Chen, J. Yang, and A. Waibel are with the School of Computer Science, Carnegie Mellon University, Pittsburgh, PA, 15213 USA (e-mail: xlchen@cs.cmu.edu; yang@cs.cmu.edu; waibel@cs.cmu.edu).

J. Zhang is with the Mobile Technologies, LLC, Pittsburgh, PA, 15213 USA (e-mail: jingzhang@mobytrans.com).

Digital Object Identifier 10.1109/TIP.2003.819223

analysis technologies. In Section IV, we introduce the intensity-based OCR for Chinese character recognition. We describe a local normalization algorithm for enhancing robustness against lighting variations. In Section V, we present some experiments on the proposed approach and evaluation results. Section VI concludes the paper.

II. Problem Description

In this research, we are interested in automatic detection and recognition of signs with text. The application scenario is as follows. A user uses a camera to capture an image or a sequence of images. The sign detection algorithm automatically detects various signs in the scene and then provides information about the location of signs within an image and some of their attributes, such as sign color distributions, sign shapes, etc. The detected regions are then fed into the recognition module for classification.

The work is related to existing research in text detection from general backgrounds [9], [18], [28], video OCR [20], and recognition of text on special objects such as license plates and containers. Mullot *et al.* reported early attempts on container and car license plate recognition in 1991 [17]. Some other researchers published their efforts on container text detection and recognition later [1], [3], [12]. The latest work on container recognition can be found in Kumano *et al.* [11]. In those applications, bar code and passive radio based technology could be a good substitution for the vision-based technology. In recent years, some researchers have attempted to deal with the sign recognition and translation with video and photography. The early work was focused mainly on the concept itself and required manual selection of the area containing the sign [22], [29]. The recent attempts have moved toward automatic detection and recognition of signs from natural scenes [30], [33].

"Video OCR" was motivated by digital library and visual information retrieval tasks. The text in video, especially in the form of subtitles, provides such meaningful information that video OCR has become an important tool in video labeling and retrieving. The text in a video stream can be a part of the scene, or come from computer-generated text, which is overlaid on the image (e.g., captions in broadcast news programs). In such a text detection and recognition task, an image sequence provides a lot of useful information that can be used to detect text and enhance the image's resolution [13], [14], [20], [27]. Compared with video OCR tasks, text detection from natural scenes faces more challenges. Non-professional equipment usually makes the image poorer than that of other video OCR tasks, such as detecting captions, which are unified when they are generated. The photographer or videographer's movement can also cause unstable input images. In addition, real-time requirement and limited resources, such as the applications in a palm-size PDA (Personal Digital Assistant), have increased challenges in algorithm implementation.

Two different methods, area analysis and edge analysis, have been successfully used for detecting text in an image. Area analysis is based on analyzing certain features in an area, e.g., texture and color analysis [11], [28]. Discrete cosine transform (DCT) and wavelet transform are widely used for area analysis [13], [15]. A major advantage of the DCT area analysis method is that DCT coefficients can be obtained directly from a JPEG or MPEG image, while on the other hand the wavelet transform can provide more stable features compared with DCT method. A disadvantage of the area-based methods is that they are sensitive to lighting and scale changes. They often fail to detect text if the size is too large or too small. The other method, edge analysis, is based on more stable edge features [34], and is thus more suitable for text detection from natural scenes. However, we have to pay special attention to filtering out "noise," because noises can add extra edges. Usually, a statistical classifier or neural network is applied to make these decisions in the area-based method, and a syntactic classifier is used for the edge-based method. In order to detect different sizes of text, the multiresolution/multiscale method has been used in text detection algorithms [13], [14], [28].

OCR is one of the most successful subsets of pattern recognition. For clearly segmented printed materials, state-of-the-art techniques offer virtually error-free OCR for several important alphabetic systems including Latin, Greek, Cyrillic, Hebrew, and their variants. However, when the size of the character set in the language is large, such as in the Chinese or Korean writing systems, or the characters are not separated from one another, such as in Arabic or Deva-nagari print, the error rates of OCR systems are still far from that of human readers, and the gap is exacerbated when the quality of the text image is compromised, for example by a camcorder or camera. The image captured from a camera in a natural environment will be rich in noise while the image obtained from a scanner in the classic OCR can provide high resolution under controllable lighting conditions. In order to address these challenges, we need more robust features for classification. Although most current OCR systems, including commercial systems, use binary features, we believe that intensity-based features are more reliable for our task. Insufficient resolution is a major problem for an OCR system using a camera as its input device. Intensity-based OCR, first introduced by Pavlidis [19], can avoid information loss during binarization, which is irretrievable in subsequent procedure(s). The disadvantage of the intensity-based method is that it is computationally more expensive.

We have successfully applied the proposed methods in developing a Chinese sign translation system [31]. We choose Chinese for several reasons. First, Chinese is one of the major languages of the world and quite different from European languages. Second, a western tourist might face a serious language barrier in China because English is not commonly used there. Third, statistics shows that more people will visit China in the future than ever before. Finally, technologies developed for Chinese sign translation can be easily extended to other languages. In the rest of the paper, we will use some examples of Chinese signs to illustrate the recognition technologies, and we also use some English and Arabic signs to test the multiple linguistic detection capability.

III. Text Detection

Fig. 1 illustrates some images of text signs. It is a challenging problem for an automatic sign detection system to detect text

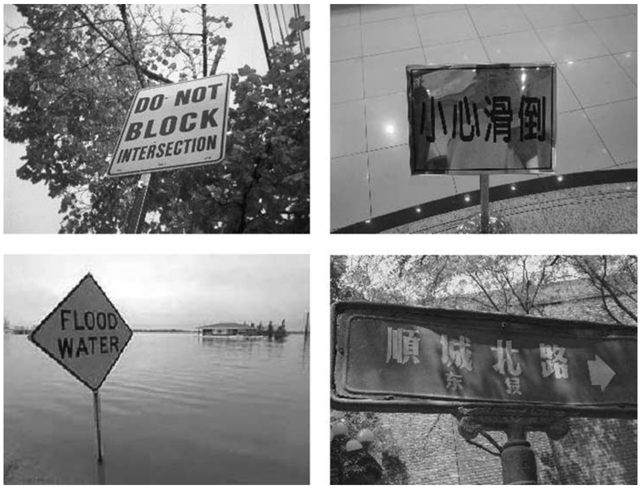


Fig. 1. Some examples of signs captured from natural scenes.

TABLE I
EFFECTS ON TEXT IN AN IMAGE CAUSED BY VARIOUS IMAGING CONDITIONS

Text in an image Imaging condition	Size	Affine	Intensity	Color	Highlight
Lighting	No	No	Yes	Maybe	Yes
Distance	Yes	Yes	No	No	No
Orientation	Yes	Yes	Yes	Yes	Yes
View angle	Yes	Yes	Maybe	Maybe	Yes

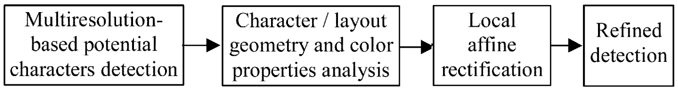

Multiresolution-based potential characters detection
Character / layout geometry and color properties analysis
Local affine rectification
Refined detection

Fig. 2. Multiresolution text detection with affine rectification schema.

from these images because of affine deformations, highlights, shadows, and specularity. We have to deal with these variations. Table I lists the possible effects on text in an image when lighting, orientation, location, and view angle change, where orientation is the angle between the normal of a sign and optical axis of the camera, and the view angle is view scope. In general, a long focus lens has a narrow view angle and a short focus lens has a wide view angle. An auto white-balanced camera will cause additional changes.

To work around these changes in an image, we use a hierarchical detection framework that embeds multiresolution and multiscale edge detection, adaptive searching, color analysis, and affine rectification algorithms. We combine multiresolution and multiscale edge detection technique to effectively detect text in different sizes. We employ a Gaussian mixture model (GMM) to represent background and foreground, and perform color segmentation in selected color spaces. We use affine rectification to recover deformation of the text regions caused by an inappropriate camera view angle. After affine rectification for each sign region in the image, we perform text detection again in rectified regions within the image to refine detection results. This scheme is shown in Fig. 2. We will discuss the detection method in more detail in the following sub-sections.

A. Detection of Candidate Text

Normally, intensity of an image is a major information source for text detection, but it is sensitive to lighting variations. On the other hand, the gradient of the intensity (edge) is less sensitive to lighting changes. Therefore, we use edge-based features in the coarse detection phase. Keeping in mind the purposes of signs, we assume the following:

1) The text is designed with high contrast to its background in both color and intensity images.
2) Each character is composed of one or several connected regions.
3) The characters in the same context have almost the same size in most of the cases, especially for Chinese.
4) The characters in the same context have almost the same foreground and background patterns.

Based on these assumptions, the main idea of the detection algorithm for coarse detection is as follows:

A multiscale Laplacian of Gaussian (LOG) edge detector is used to obtain the edge set. The properties of the edge set associated with each edge patch, such as size, intensity, mean, and variance, within the surrounding rectangle, are then calculated. Some edge patches will be excluded from further consideration based on certain criteria applied to the properties, and the rest will be passed to a recursive procedure. The procedure attempts to merge adjoining edge patches with similar properties and re-calculate the properties recursively until no update can be made. A similar idea for candidate search has been reported in [35]. We have enhanced the algorithm by adding more criteria for edge candidate filtering and using a pyramid structure to handle various variations in our implementation."

With LOG, we can obtain enhanced correspondences on different edge scales by using a suitable deviation σ. Since characters in the same context share some common properties, we can use them to analyze the layout and affine parameters, and refine detection results. Color distribution of the foreground and background is one such important property.

B. Color Modeling

Signs are designed for humans to view at a distance. Therefore they have highly distinguishable colors on their foregrounds and backgrounds, and also a high intensity contrast in their gray scale images. This property seems to make it easy to segment text and to describe characters using marginal distributions in a color space. However, it is almost impossible to obtain uniform color distributions of the foreground and background because of lighting sources, shadows, dirt, etc. In this research, we use a GMM to characterize color distributions of the foreground and background of a sign, and more specifically, for each character, we use the following model:

$$f(c)=(1-\alpha)G_{Back}(\mu_b,\Sigma_b)+\alpha G_{Front}(\mu_f,\Sigma_f),(0\leq \alpha\leq 1) \tag{1}$$

where $G_{Back}(\cdot)$ is the color distribution of the background, and $G_{Front}(\cdot)$ is the color distribution of the foreground. In a hieroglyphic system, e.g., Chinese, a character with more strokes

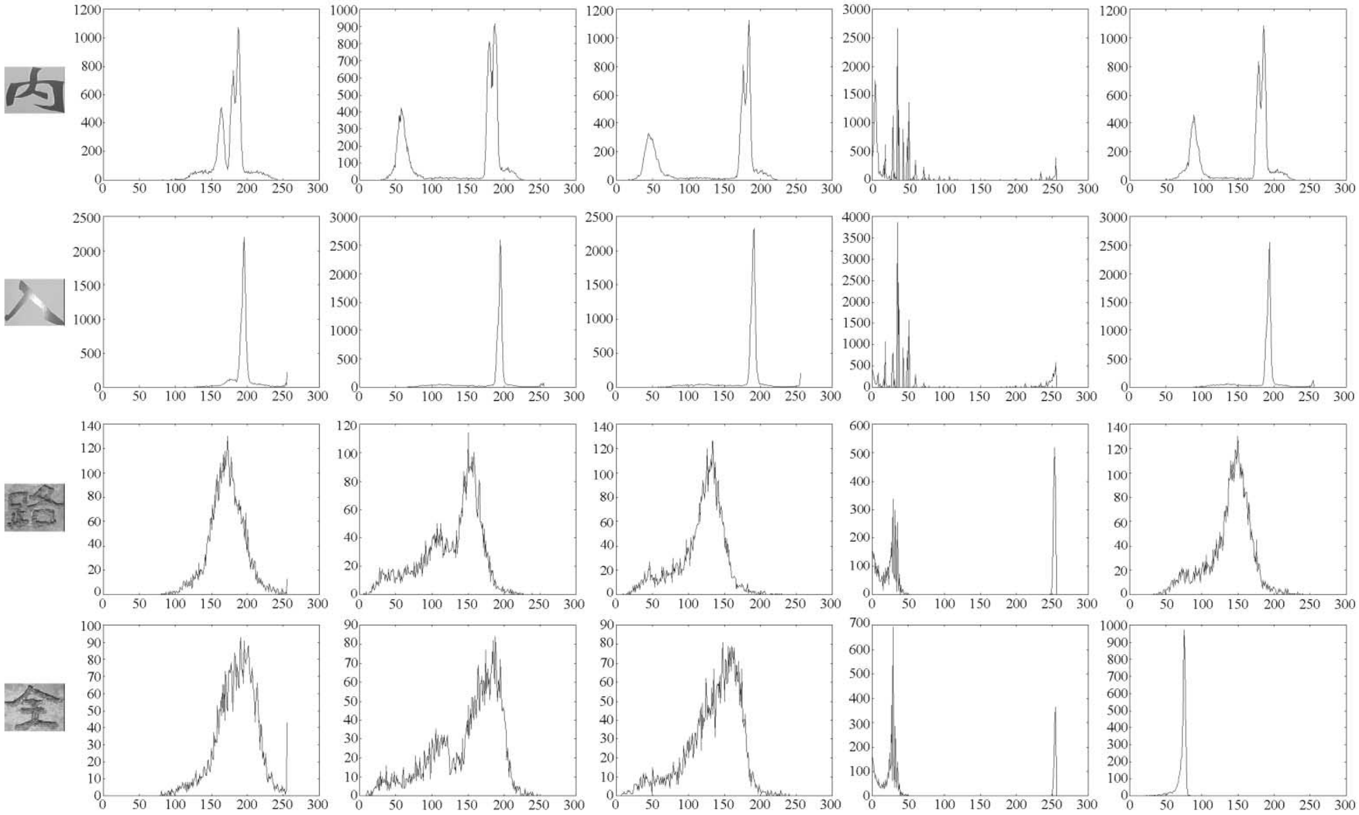


Fig. 3. Color distributions of different fonts, backgrounds, and lighting conditions.

has a higher percentage in its foreground than that of a character with fewer strokes. Furthermore, since transitions between foreground and background are similar for a fixed size character with different font styles, a character with bold font has less variance in its foreground than the same character without bold, because a bold font character occupies a relatively larger area in its foreground compared to a normal font character. The background is the opposite: a bold font character has a larger variance in its background than a normal font character. Therefore, in (1), α provides a cue on the complexity of the character, $\|\mu_f - \mu_b\|$ indicates the contrast for a color space invariant to the lighting condition, and Σ_b, Σ_f yields the font style. Thus, we can describe the character with $(\alpha, \mu_b, \Sigma_b, \mu_f, \Sigma_f)$.

Based on the fact that signs are designed for human perception, their foregrounds and backgrounds should be separable using one marginal distribution in a color space under an ideal lighting condition. If there exist multiple lighting sources and shadows, contrasts of foreground and background might change significantly across the entire sign. Therefore, we model the distribution of each character separately rather than the entire sign as a whole. Most cameras use an *RGB* color space; other color spaces, such as *XYZ, YUV, YCrCb, HSI*, etc., can be easily converted into the *RGB* space via linear or nonlinear color conversion, or vice versa [5]. However, the *RGB* space is not necessarily the best one for representing colors for machine perception. As we know, a triple $[r, g, b]$ in the *RGB* space can represent not only color but also brightness. If the corresponding elements in two points, $[r_1, g_1, b_1]$ and $[r_2, g_2, b_2]$, are proportional, they will have the same color but different brightness. Each space has its special properties. For example, *HSI* space can better handle lighting changes than *RGB* space. Equation (2) gives the relation between *RGB* and *HSI* [5]. We normalize *HSI* within the range of [0, 255] in computation

$$
\begin{aligned}
I &= \frac{R+G+B}{3},\\
S &= \begin{cases} 1-\frac{\min(R,G,B)}{I} & \text{if } I \neq 0,\\ 0 & \text{if } I = 0,\end{cases}\\
H &= \begin{cases} \arccos \frac{2R-G-B}{2\sqrt{(R-G)^2+(R-B)(G-B)}} & \text{if } S\neq 0 \text{ and } B \leq G,\\ 2\pi - \arccos \frac{2R-G-B}{2\sqrt{(R-G)^2+(R-B)(G-B)}} & \text{if } S \neq 0 \text{ and } B > G,\\ undefined & \text{if } S = 0.\end{cases}
\end{aligned}
\tag{2}
$$

Fig. 3 contains the $R/G/B/H/I$ histograms of two pairs of Chinese characters from two signs in our sign database. It can be observed from Fig. 3 that at least one of the $R/G/B$ components can be used for distinguishing the foreground and background if the lighting is uniformly distributed and the surface is smooth, e.g., the first line in Fig. 3. Otherwise $R/G/B$ and even I may fail in segmentation. For example the character in the second line of Fig. 3 can only be well segmented by component H because of the highlight. The characters in lines three and four can also only be segmented by component H because the original sign was carved on a rough surface of a stone. The disadvantage of using H is that the hue of a pixel likely diffuses to its adjoined pixels, but it works for most of the cases except a black sign on a white background or vice versa, in which cases,

TABLE II
GMM PARAMETERS FOR THE CHARACTERS IN FIG. 3

	Background		Foreground			Background		Foreground		
	μ_b	σ_b	μ_f	σ_f	α	μ_b	σ_b	μ_f	σ_f	α
R	188.57	11.64	156.37	14.50	0.35	198.55	12.20	169.28	14.22	0.13
G	185.71	18.75	64.49	9.31	0.35	196.39	10.78	120.92	26.59	0.16
B	181.60	9.23	54.76	21.29	0.35	192.55	12.26	120.58	26.87	0.15
H	42.90	14.83	4.63	9.68	0.50	40.32	10.15	255.16	16.40	0.36
I	184.75	9.85	90.90	13.86	0.34	195.39	11.63	135.95	20.54	0.15

	Background		Foreground			Background		Foreground		
	μ_b	σ_b	μ_f	σ_f	α	μ_b	σ_b	μ_f	σ_f	α
R	154.18	15.67	190.82	15.59	0.49	209.72	13.73	167.21	20.96	0.57
G	160.57	12.88	96.56	33.42	0.52	195.16	9.25	132.46	42.33	0.74
B	143.26	13.69	96.57	28.08	0.54	169.83	11.70	116.81	32.50	0.66
H	27.69	6.83	0.01	4.63	0.53	28.41	4.91	1.94	5.58	0.44
I	161.83	13.41	118.56	25.63	0.53	191.55	11.07	139.63	30.71	0.68

component S can be used to identify the situations, and then component I is used for segmentation.

Table II lists the GMM parameters of the characters in Fig. 3. It is obvious that the GMM parameters can reflect how easily each component can be segmented and described in terms of character properties. Thus, we can define each component's *confidence* as (3). Intuitively, a component with one or more sign characters should have a high confidence if it has an obvious two-class distribution, which means the distance between the two classes is large, and the variances of each class is small

$$C_i = \begin{cases} \frac{\left|\mu_b^i - \mu_f^i\right|}{\sigma_b^i + \sigma_f^i} & i = R, G, B, I, \\ \frac{\min\left(\left|\mu_b^i - \mu_f^i\right|, 256 - \left|\mu_b^i - \mu_f^i\right|\right)}{\sigma_b^i + \sigma_f^i} & i = H \end{cases} \tag{3}$$

where C_i is the confidence for component i ($i = R, G, B, H, I$). The higher the value C_i, the greater the confidence of the corresponding component.

These GMM parameters are used for layout analysis and estimation of affine parameters. They can also be used for character conversion from color to intensity image. We apply an expectation maximization (EM) algorithm to estimate these GMM parameters. To differentiate background and foreground, we enlarge the boundary of the character by 2 pixels on each side and calculate the color distribution of the region between the original boundary and the enlarged boundary. This distribution should be the same or similar to the background distribution. We then can determine the background distribution in the GMM by comparing distributions in the GMM to this distribution. The confidence can be used for measuring the performance of each subspace in segmentation. Fig. 4 shows some examples of segmentation using different components.

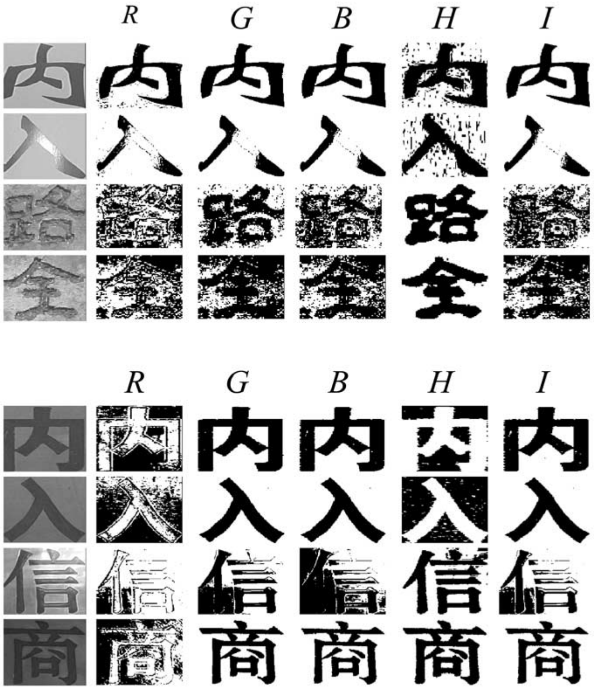


Fig. 4. Examples of segmentation using different components.

C. Layout Analysis and Affine Parameter Estimation

The objective of layout analysis is to align characters in an optimal way, so that characters that belong to the same context will be grouped together. A text layout has two cluster features: intrinsic and extrinsic features. The intrinsic features are those which do not change with the camera position and the extrinsic features are those ones which change with the camera position. The intrinsic features includes font style, color, and contrast; the extrinsic features include character size, sign shape, etc. Both the intrinsic and extrinsic features can provide cues for layout analysis. Moreover, the extrinsic features can also provide some information on the affine transform that occurred while snapping the image.

The text on signs is usually aligned to form a row-based or column-based structure. Even if a sign is deformed (as by an affine transform), the text on the sign might still be in a line. However, the line may be tilted to a certain degree, which will cause problems in layout analysis and further recognition. However, the text characters which are in the same context should have a similar size. Therefore, text alignment will provide some cues to recover deformation of the sign if there exists an affine transform for the text.

1) Layout Analysis: The layout analysis algorithm is listed in Algorithm 1:

```
Algorithm 1 (Layout Analysis)
Input: The candidates of text regions and
the associated attribute sets:
$R = \{r_i = surround\ (e_i) | e_i$ is the edge of the
        candidate characters, $i = 0, 1, \ldots, n-1\}$,
and $A$ =
$\{a_{e_i} = (x, y, h, w, \mu_F, \Sigma_F, \mu_B, \Sigma_B) | i = 0, 1, \ldots, n-1\}$,
where $(x,y)$, $h$, and $w$ are the center, height, and width of the surrounding rectangle respectively, and $\mu_F$, $\Sigma_F$, $\mu_B$, $\Sigma_B$ are the mean vectors and covariance matrices corresponding to the foreground and background color distributions;
Output: The Layout regions
$L = \{(x_i^1, y_i^1, x_i^2, y_i^2, x_i^3, y_i^3, x_i^4, y_i^4) | i = 0, 1, \ldots, m-1\}$.
1. Clustering $r_i$ according to the attribute $\mu_F$, $\Sigma_F$, $\mu_B$, $\Sigma_B$ and form the candidate layout set
   $L = \left\{ LR_j = \left\{ r_i^j | r_i^j \in R \right\} \middle| j = 0, 1, \ldots, m \right\}$.
2. If $\exists r_i, r_k \in LR_j$, $r = r_i \cup r_k$, satisfy $\max(h(r), w(r)) \leq hw_{\max}$, where $hw_{\max} = \max(h(r_i), w(r_i)), r_i \in LR_j$, then $r_i = r, LR_j = LR_j \backslash \{r_k\}$.
3. If $|LR_j| > 2$, using Hough transform to find all possible line segments $l_j^k$ $(k = 0, 1, \ldots, K-1)$, which are fitted by the centers of $r_j^i$. These line segments will form several compatible sets:
   $CL_j^m = \{l_j^k | k \in \{0, 1, \ldots, K-1\}, \text{and}$
   $\qquad \forall l_j^{k_1}, l_j^{k_2}, \text{s.t.} l_j^{k_1} \cap l_j^{k_2} = \varnothing\}, \ m = 0, 1, 2, \ldots, M.$
   Only one of the $CL_j^m$ will be the winner of the candidate layout direction. The winner $CL_j^{win}$ should satisfy the following criteria:
   a. $\left|CL_j^{win_1}\right| = \min\left\{\left|CL_j^m\right|, m = 0, 1, \ldots, M-1\right\}$, and $\left|CL_j^{win_2}\right| = \min\left\{\left|CL_j^m\right|, m = 0, 1, \ldots, M-1, m \neq win_1\right\}$;
   b. Let $\mu_j^{win_1}$ and $\mu_j^{win_2}$ be the mean length of all line segments exclusive the shortest one in $CL_j^{win_1}$ and $CL_j^{win_2}$. If $\mu_j^{win_1}/\mu_j^{win_2} < T$, where $T$ is a threshold, $CL_j^{win_2}$ will be the winner of this layout region. If $\mu_j^{win_1}/\mu_j^{win_2} > 1/T$, $CL_j^{win_1}$ will be the winner of this layout region. Otherwise,
   $CL_j^{win} = \arg\min \left\{ \sum \left| l_j^k - \mu_j^i \right|, l_j^k \in CL_j^{win_1}$
   $\qquad \backslash \left\{ \text{The shortest } l \text{ in } CL_j^{win_1} \right\} \right\}$.
4. Removing all layouts with only some small candidate text regions.
5. Finding the vertices of the surrounding quadrangle for each layout region.
```

The threshold T for Algorithm 1 needs to be selected experimentally. For our Chinese sign database, the value was set as 0.6.

The affine parameters can be obtained from each layout region when the region includes multiple characters. We can use two different heuristics for this: one is the size change of the characters and the other is the parallel hypothesis of the characters, provided that all characters in the same layout are in the same text plane.

There are at least two ways to obtain the spatial parallel lines within each sign frame from a given image:

1) If the sign is inside a rectangle frame, we should be able to extend the background area to the boundary of the frame. If the boundary of the sign region is within the image scope, we will use lines to fit it; otherwise we will try the next method. If we can fit the boundary with four lines, we assume that they are two pairs of parallel lines, which will be used for estimation of affine parameters. Some examples are shown in Section III-D. This method assumes the risk that the frame is not a rectangle but just a general trapezium. Although this seldom happens, we can detect it using the cue from the next step.
2) If the boundary of the sign is outside of the image scope or not four lines that can be found to fit the boundary, it is still possible to fit parallel lines from text only. We start from fitting the corner of the surrounding rectangle of the characters along the winner direction obtained from Algorithm 1. We select the corner with higher average edge intensity between upper-left and upper-right corners to fit the upper line, and do the same with the lower line. If we can get more than two aligned anear lines of text with similar properties, such as size, contrast, color distribution, etc., we can get the second pair of parallel lines using the similar method. However, a sign is usually a short phrase or sentence. Therefore, it might be difficult to get the second pair of parallel lines. If that is the case, we can use the left boundary of the left-most character and right boundary of the right-most character in the same text row for estimating the second pair of parallel lines. Compared with the parameter estimation from the rectangle frame, this method is less accurate in the vertical direction, but our experiments indicate that the error has little effect on detection rate and recognition accuracy in most cases. In Section V, we show some examples in which part of boundary of the sign is outside the image, and we have to use the text itself to estimate affine parameters.

2) Affine Parameter Estimation: Several methods exist for obtaining affine parameters from texture [2], [23], [26]. However, texture based methods require rich texture in the region of the estimation. Texture in a sign region is sometimes not rich enough for those algorithms. In this research, we use a nontexture based method as discussed below.

As shown in Fig. 5, suppose that we have the following two lines l_1, l_2 associated with text in the image plane, which are mapped from the spatial parallel lines L_1 and L_2

$$l_i : a_i x + b_i y + c_i = 0 \quad i = 1, 2. \tag{4}$$

If l_1 and l_2 are not parallel in the image plane, and the vanish (intersection) point is (x_1, y_1), then we can get the normalized spatial direction of L_1 and L_2 as

$$\begin{pmatrix} r_1 \\ s_1 \\ t_1 \end{pmatrix} = \frac{1}{\sqrt{x_1^2 + y_1^2 + f^2}} \begin{pmatrix} x_1 \\ y_1 \\ f \end{pmatrix} \tag{5}$$

where f is the focal length of the camera, and can be obtained from calibration. We also assume that the focal length is much smaller than the distance of the object from the camera.

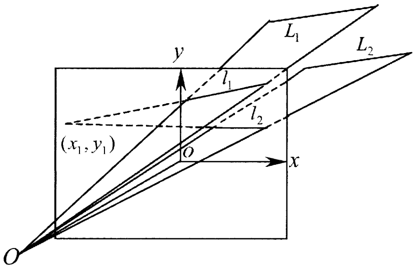


Fig. 5. Two spatial parallel lines and their imaging in the image plane.

If the two lines l_1 and l_2 are parallel in the image, we can get the normalized spatial direction of L_1 and L_2 as

$$\begin{pmatrix} r_1 \\ s_1 \\ t_1 \end{pmatrix} = \frac{1}{\sqrt{a_1^2 + b_1^2}} \begin{pmatrix} b_1 \\ -a_1 \\ 0 \end{pmatrix} \tag{6}$$

where we must keep $r_1 > 0$; otherwise, the direction of this vector needs to be inverted.

Assume that the second pair of spatial parallel lines can be obtained from the following equation:

$$l_i : a'_i x + b'_i y + c'_i = 0 \quad i = 1, 2. \tag{7}$$

As in (4), we can obtain the normalized spatial direction from (8) when two projected lines are not parallel in the image plane, and the vanish point is (x_2, y_2)

$$\begin{pmatrix} r'_2 \\ s'_2 \\ t'_2 \end{pmatrix} = \frac{1}{\sqrt{x_2^2 + y_2^2 + f^2}} \begin{pmatrix} x_2 \\ y_2 \\ f \end{pmatrix}. \tag{8}$$

Otherwise, we can obtain the direction from (9) when the projected lines in the image plane are parallel

$$\begin{pmatrix} r'_2 \\ s'_2 \\ t'_2 \end{pmatrix} = \frac{1}{\sqrt{a'^2_1 + b'^2_1}} \begin{pmatrix} b'_1 \\ -a'_1 \\ 0 \end{pmatrix} \tag{9}$$

where $s'_2 > 0$ must hold, otherwise, the direction of the vector must be inverted. Then, we calculate the normal of the text plane as

$$\begin{pmatrix} r_3 \\ s_3 \\ t_3 \end{pmatrix} = \begin{pmatrix} r_1 \\ s_1 \\ t_1 \end{pmatrix} \times \begin{pmatrix} r'_2 \\ s'_2 \\ t'_2 \end{pmatrix}. \tag{10}$$

Since there is no guarantee $(r'_2 \quad s'_2 \quad t'_2)^T$ is orthotropic to the other two vectors, we need to perform the following conversion:

$$\begin{pmatrix} r_2 \\ s_2 \\ t_2 \end{pmatrix} = \begin{pmatrix} r_3 \\ s_3 \\ t_3 \end{pmatrix} \times \begin{pmatrix} r_1 \\ s_1 \\ t_1 \end{pmatrix}. \tag{11}$$

This will guarantee that all three vectors are orthotropic, and we can then use them in the subsequent affine rectification.

D. Affine Rectification

It is possible to reconstruct a front view of the sign if we know the normal of the sign plane under the camera coordinate system. Fig. 6 depicts an image in four coordinate systems:

1) The camera coordinate system, $OXYZ$, is the basic coordinate system.
2) The text plane coordinate system, $O_t X_t Y_t Z_t$, applies the text plane as $O_t X_t Y_t$ plane, and uses $(r_1 \quad s_1 \quad t_1)^T$, $(r_2 \quad s_2 \quad t_2)^T$ and $(r_3 \quad s_3 \quad t_3)^T$ as its axes. The origin of the system can be selected randomly on the $O_t X_t Y_t$ plane, and is located at (X_0, Y_0, Z_0) under the $OXYZ$ coordinate system. However, it is desirable that we select a point at a character that is as close to the origin of $OXYZ$ as possible.
3) The ideal text plane, $O_{it} X_{it} Y_{it} Z_{it}$, is located at the same origin as $O_t X_t Y_t Z_t$ but uses the vectors (1 0 0), (0 1 0), and (0 0 1) as its axes.
4) The image coordinate system oxy is a 2-D coordinate system while the other three are 3-D coordinate systems.

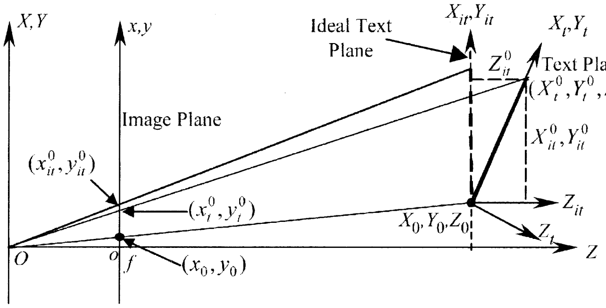


Fig. 6. Illustration of four coordinate systems.

The mapping from a point $\left(X_t^0, Y_t^0, Z_t^0\right)$ in the text plane $O_t X_t Y_t Z_t$ to a point (x_t^0, y_t^0) in the image coordinate system can be written as

$$\begin{pmatrix} x_t^0 \\ y_t^0 \end{pmatrix} = \begin{pmatrix} f & 0 \\ 0 & f \end{pmatrix} \begin{pmatrix} \frac{(X_{it}^0 + X_0)}{(Z_{it}^0 + Z_0)} \\ \frac{(Y_{it}^0 + Y_0)}{(Z_{it}^0 + Z_0)} \end{pmatrix} \tag{12}$$

where,

$$\begin{pmatrix} X_{it}^0 \\ Y_{it}^0 \\ Z_{it}^0 \end{pmatrix} = \begin{pmatrix} r_1 & r_2 & r_3 \\ s_1 & s_2 & s_3 \\ t_1 & t_2 & t_3 \end{pmatrix} \begin{pmatrix} X_t^0 \\ Y_t^0 \\ Z_t^0 \end{pmatrix}. \tag{13}$$

In order to reconstruct a front view of the text plane, we can use an affine rectification

$$\begin{pmatrix} x_{it}^0 \\ y_{it}^0 \end{pmatrix} = \begin{pmatrix} f & 0 \\ 0 & f \end{pmatrix} \begin{pmatrix} \frac{(X_t^0 + X_0)}{Z_0} \\ \frac{(Y_t^0 + Y_0)}{Z_0} \end{pmatrix}. \tag{14}$$

Considering that the text plane is on the $O_t X_t Y_t$ plane, we have $Z_t^0 = 0$. Since the origin (X_0, Y_0, Z_0) of both $O_t X_t Y_t Z_t$ and $O_{ti} X_{it} Y_{it} Z_{it}$ is mapped to the point (x_0, y_0) in the image plane, we have

$$x_0 = f \frac{X_0}{Z_0}, y_0 = f \frac{Y_0}{Z_0}. \tag{15}$$

From (12) to (15), we obtain

$$\begin{aligned} x_{it}^0 =& x_0 + f \frac{\begin{vmatrix} x_0 - x_t^0 & r_2 f - t_2 x_t^0 \\ y_0 - y_t^0 & s_2 f - t_2 y_t^0 \end{vmatrix}}{\begin{vmatrix} r_1 f - t_1 x_t^0 & r_2 f - t_2 x_t^0 \\ s_1 f - t_1 y_t^0 & s_2 f - t_2 y_t^0 \end{vmatrix}}, \\ y_{it}^0 =& y_0 + f \frac{\begin{vmatrix} r_1 f - t_1 x_t^0 & x_0 - x_t^0 \\ s_1 f - t_1 y_t^0 & y_0 - y_t^0 \end{vmatrix}}{\begin{vmatrix} r_1 f - t_1 x_t^0 & r_2 f - t_2 x_t^0 \\ s_1 f - t_1 y_t^0 & s_2 f - t_2 y_t^0 \end{vmatrix}}. \end{aligned} \tag{16}$$

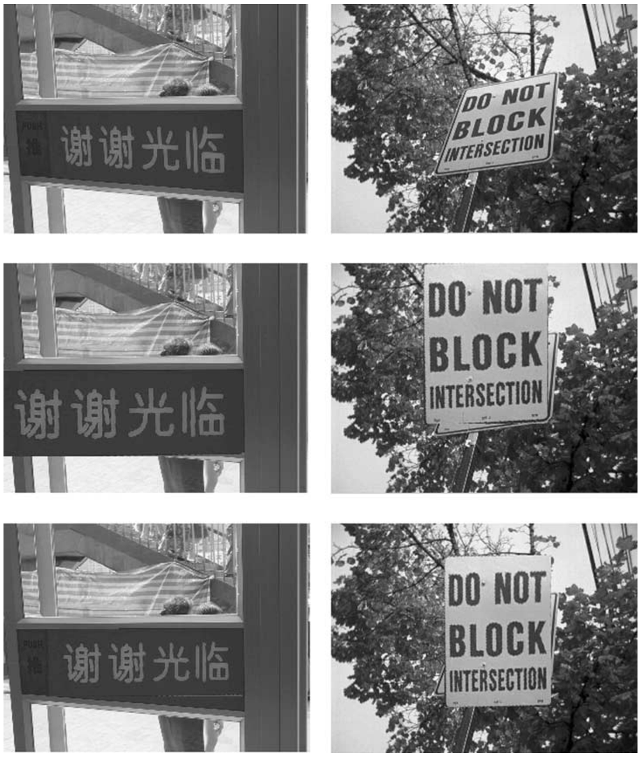


Fig. 7. Restoration from the affined images.

Equation (16) can be used to restore the front view of the text from an affined text image. A proper interpolation should be applied because it is not a one-to-one mapping for the digitalized image. We use a B-spline interpolation in our current implementation. To avoid additional blur in the interpolation, all edge pixels are interpolated only along the edge direction while the other points are done by surface interpolations.

Fig. 7 contains some examples of the restored mapping from deformed images. Fig. 7(a) shows the affined signs. Fig. 7(b) and (c) show the ones recovered with B-spline interpolations using different reference origins in the text plane. In these two examples, the rectangle boundary of the sign frames is used for estimating affine parameters.

IV. Intensity-Based OCR

A sign can contain graphic or text content. In this research, we focus on the text signs only. As mentioned in Section II, OCR for the characters captured from natural scenes faces more challenges than that of document analysis. For a traditional document analysis task, a scanner with a stable embedded lighting system is used to obtain high quality images, which are then easily binarized. For a sign recognition task, however, because the sign image is captured by a camera from natural scenes under various lighting conditions, the signal to noise ratio (SNR) is much lower. If we use binary features for OCR, we cannot guarantee effectively removing noises before the binarization processing. Although by carefully selecting color spaces we can reduce noises to a certain degree, we have no way to distinguish noises and useful information within the image. The binarization processing will weigh noises and useful information the same. Furthermore, the segmentation of foreground and background cannot be perfect because of noises. Fig. 8 illustrates some examples from both binary and intensity images, where the color space is carefully selected and hue is used to obtain both binary and intensity images. It is obvious that the binary images are much noisier, while the intensity images have less of a problem separating the background and foreground. Thus intensity images keep all the information essential to the decision making stage. Wang and Pavlidis showed the advantages of direct feature extraction from gray scale image for OCR [24]. Our experiments also indicate that the intensity based OCR has advantages over binary OCR for images with low SNR. To avoid irretrievably losing information during the binarization processing, we use the intensity character image directly for feature extraction.

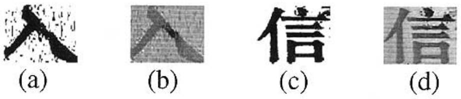


Fig. 8. Comparison of (a), (c) binary characters and (b), (d) gray scale characters.

A. Preprocessing

The characters captured by a camera from natural scenes vary in size, font style, color, and contrast. Furthermore, a character may vary with an affine deformation if the optical axis is not perpendicular to the character plane (see Section III). We use a scaling algorithm to normalize the size of the character images from different signs. Color variation does not cause a major problem in the OCR phase because we don't use a color image directly. In fact, signs are usually designed with high contrast in both color and gray scale images. We can seldom find a sign using pure colors in both foreground and background, e.g., red characters on a green background. Even in that case, we can deal with it using *HSI* color space as discussed in Section III-B.

Our intensity based OCR uses a gray scale image as its input. We convert a color image into a gray scale image before we further process it. Gray scale sign images come in two forms: bright characters on a dark background or dark characters on a bright background. We can resolve these two into a single case by inverting the foreground and background if the text is darker than its background. To determine whether the text is darker or brighter, we apply a method similar to the one described in Section III-B, but only in gray scale (single component).

Lighting sources in natural scenes cause another variation in sign recognition. For example, the sun can cast a highlight point on a sign, and the location of the point will change with time. In addition, many other factors, such as multiple lighting sources and the reflective properties of the surface, will cause uneven intensity distribution of the foreground and background. More specially, the intensity distribution of all strokes within a text region changes in a large dynamic scope: some have obvious contrast, some not, and some are highlighted while others may be dark. We utilize a localized intensity normalization method before feature extraction to reduce intensity distribution changes. The localized intensity normalization is intended to leave the

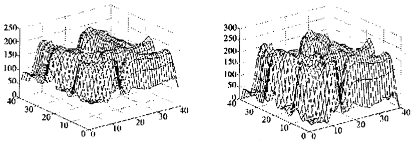

(a) (b)

Fig. 9. Example of localized intensity normalization.

foreground with almost the same intensity distribution. Fig. 9(a) contains the original image distribution of character “Xing” and Fig. 9(b) contains the normalized one.

B. Feature Extraction

We use a Gabor wavelet for feature extraction. Because of its superior mathematic properties, Gabor wavelet has been widely used for data compression [21], face recognition [25], texture analysis [16], handwriting recognition [4] and other image processing tasks in recent years. In OCR applications, Gabor wavelet has been applied to binary images [5], [6] and recently applied to a video stream [32]. Yoshimura and his colleagues even report using Gabor for feature extraction and linear vector quantization (LVQ) for feature selection from a video stream.

Gabor wavelet is a sinusoidal plane wave with a particular frequency and orientation, modulated by a Gaussian envelope. It can characterize a spatial frequency structure in the image while preserving information about spatial relations, and is therefore suitable for extracting orientation-dependent frequency contents from patterns. A complex-valued 2-D Gabor function modulated by a Gaussian envelope is defined as follows:

$$G(x,y,k,\theta) = G_1(x,y) \times \left[\cos(R) - \exp\left(-\frac{\sigma^2}{2}\right)\right] + iG_1(x,y)\sin(R) \quad (17)$$

where

$$G_1(x,y) = \frac{k^2}{\sigma^2}\exp\left[-\frac{k^2(x^2+y^2)}{2\sigma^2}\right],$$

$$R = kx\cos\theta + ky\sin\theta, k = \frac{2\pi}{\tau}.$$

The parameter σ is the deviation of the Gaussian envelope and τ and θ are the wavelength and orientation of the Gabor function, respectively. Fig. 10 illustrates frequency responses for the Gabor filters in four orientations (0°, 45°, 90°, and 135°).

For a given pixel at (x_1, y_1) with intensity $I(x_1, y_1)$ in an image, its Gabor feature can be treated as a convolution

$$J(x_1, y_1, k, \theta) = \int I(x_1 - x, y_1 - y)\, G(x, y, k, \theta)\, \mathrm{d}x\mathrm{d}y. \quad (18)$$

Suppose that m frequencies and n orientations are used to extract a Gabor feature. We can have a vector of $m \cdot n$ complex coefficients for each position. We call this vector a *jet*[10], which is used to represent the position of local features. In our application, we divide a character into 7×7 grids as shown in Fig. 11, which results in a $49m \cdot n$ dimension feature vector for a character.

C. Feature Transforms and Recognition

We would like to reduce the number of dimensions of feature vectors because they are computationally expensive and because not all of them are effective for recognition. LVQ and LDA are two common tools for dimension reduction. We use LDA in this research because it can be used not only for dimension reduction, but also for feature optimization. LDA is a method used to find a transform that can maximize the between-class scatter matrix $\mathbf{S}_b$ and minimize the within-class scatter matrix $\mathbf{S}_w$ simultaneously, as in (19)

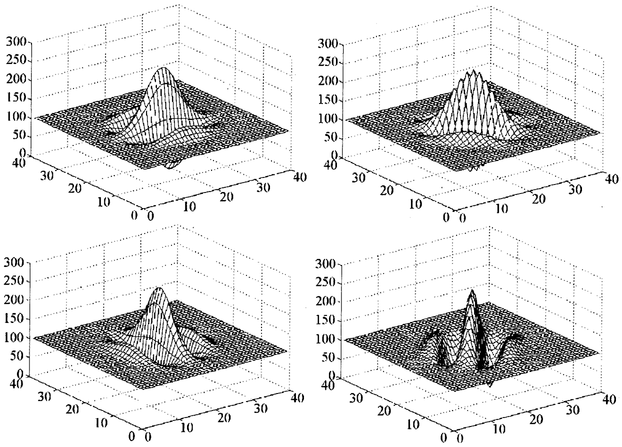


Fig. 10. Gabor filters in four different orientations.

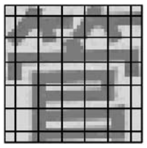


Fig. 11. Regions for feature extraction.

$$\arg_{W}\left\{\max\frac{\mathbf{W}^T\mathbf{S}_b\mathbf{W}}{\mathbf{W}^T\mathbf{S}_w\mathbf{W}}\right\}. \quad (19)$$

The new space, $\mathbf{W}$, is then the most discriminative space. Feature vector $\mathbf{x}$ in the original feature space is projected to this new space and yields the new feature $\mathbf{y}$ using (20), which can be used for classification

$$\mathbf{y} = \mathbf{W}^T\mathbf{x}. \quad (20)$$

We obtain the reference vectors $\mathbf{C}_i$ $(i = 1, 2, \ldots, 3755)$ for all 3755 characters in GB-2312–80, a standard for commonly used Chinese words, from the training set. The training set includes six different fonts: SongTi, HeiTi, KaiTi, LiShu, YaoTi, and YouYuan. All of these character images are gray scale. For each type of font, we generate the standard character images from the standard font library, and then pass them through a low-pass filter that produces the gray scale image. We have compared two types of training sets: *single sample per font* and *multiple samples per font*. We generate the multiple samples (here, we generate 25 samples) by adding noise and skew deformations to the character images. We calculate the average Gabor vector $\mathbf{G}_i^j$ $(i = 1, 2, \ldots, 3755, j = 1, 2, \ldots, 6)$ for each type of font from these images. The within-class and between-class scatter matrixes are

$$\mathbf{S}_w = \sum_{i=1}^{3755}\sum_{j=1}^{6}(\mathbf{G}_i^j - \mathbf{m}_i)(\mathbf{G}_i^j - \mathbf{m}_i)^t \quad (21)$$

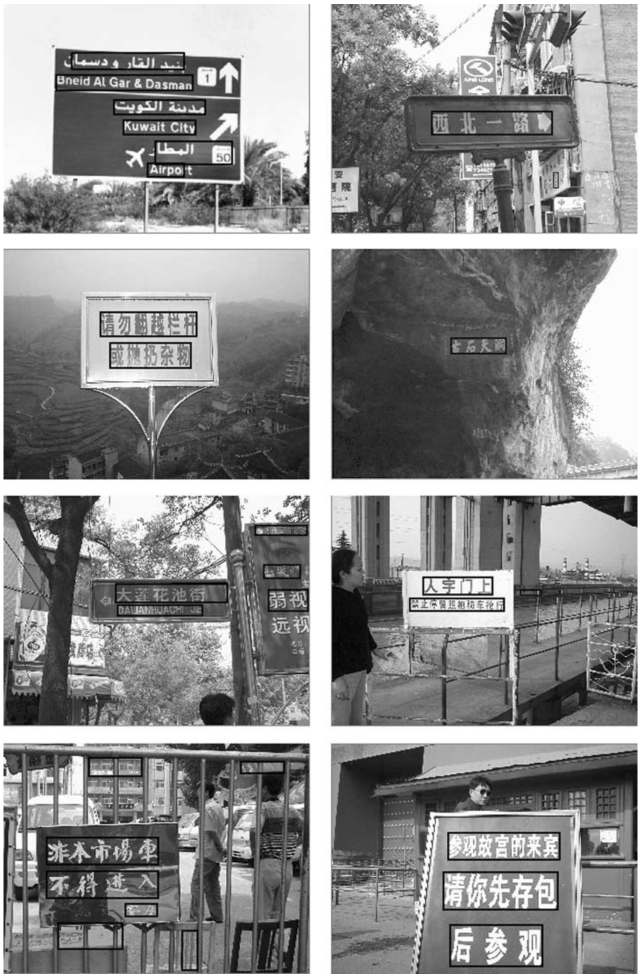


Fig. 12. Examples of sign detection.

$$\mathbf{S}_b = \frac{1}{3755} \sum_{i=1}^{3755} \sum_{j=1}^{6} (\mathbf{m}_i - \mathbf{m})(\mathbf{m}_i - \mathbf{m})^t \tag{22}$$

where $\mathbf{m}_i = 1/6 \sum_{j=1}^{6} \mathbf{G}_i^j$, and $\mathbf{m} = 1/3755 \sum_{i=1}^{3755} \mathbf{m}_i$.

Finally, we obtain the discriminative space $\mathbf{W}$ during the training phase. Usually, a hierarchical classifier can be used for a large class set if the feature is ordered or partly ordered. Considering that the LDA has already provided the maximum discriminant space and that the feature from LDA is not an ordered feature, we must use the nearest neighborhood classifier in recognition; several distance measurements are compared, including street block distance, Eulerian distance, and vector angle distance. We find that the vector angle distance is significantly better than others. The accuracy reaches 92.46% for our testing set when we use the vector angle distance, while it can only reach 82.61% when we use the Eulerian distance.

We have compared the results from two types of training sets: single sample per font and multiple samples per font. The experimental results indicate that these give similar performance in most of the cases, although sometimes the result from the training set of single sample per font is better than that of multiple samples per font. Our explanation is that the single sample method is a center-based method, and the standard image from the font library is the perfect center of all possible images of the character.

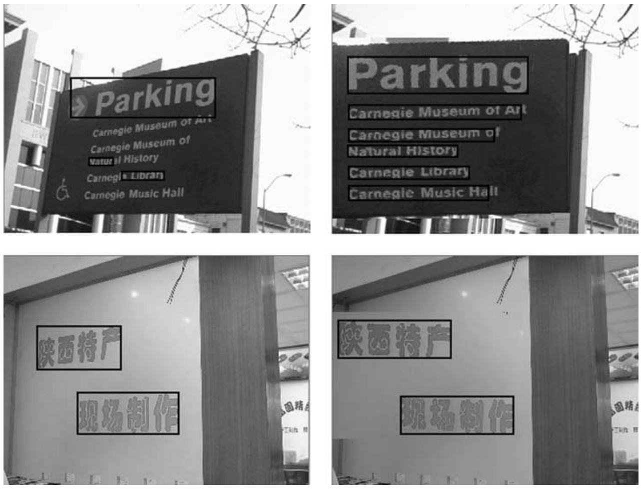


Fig. 13. Examples of sign detection with distortion.

TABLE III
COMPARING OF RECOGNITION RESULT WITH RESPECT TO AFFINE RECTIFICATION

	Total Characters	Recognized Characters	
		Before affine rectification	After affine rectification
γ<20°	24	22	23
30°>γ≥20°	48	43	45
40°>γ≥30°	46	39	43
50°>γ≥40°	88	42	75
γ≥50°	45	25	34

Note: γ is the angle between normal of text plane and Z-axis of the camera.

V. Experiments and Discussion

We have performed several experiments to evaluate the proposed detection and recognition methods. Figs. 12 and 13 contain examples of automatic sign detection, where the rectangles give detection results. These multilingual signs include Chinese, English and Arabic. Signs captured from a camera might have some distortion because of nonfrontal viewing angle, which can reduce detection rate and recognition accuracy. The affine rectification can correct such distortion. Fig. 13 shows two sets of examples of text detection, where the left side of each group [(a) and (c)] is the detection result without affine rectification, and the right side of each group [(b) and (d)] is the result with affine rectification. Observe that only partial text can be detected from distorted images because the affine deformation makes the text tilt along a certain direction and vary in size. For the corresponding right parts, the text has almost the same size and is aligned in nearly horizontally after affine rectification; now the detection algorithm can successfully find all the text regions. In Fig. 13(a), the boundary of the sign frame is completely within the image, and frame information can be used for parameter estimation. In Fig. 13(c), however, part of the sign frame is outside the image. Therefore, we use the parallel lines from text itself to estimate the affine parameters. In this case, we will need to extend the area for rectification beyond just the text area.

We further tested the improvement of OCR accuracy with affine rectification. Experiments were performed on Chinese

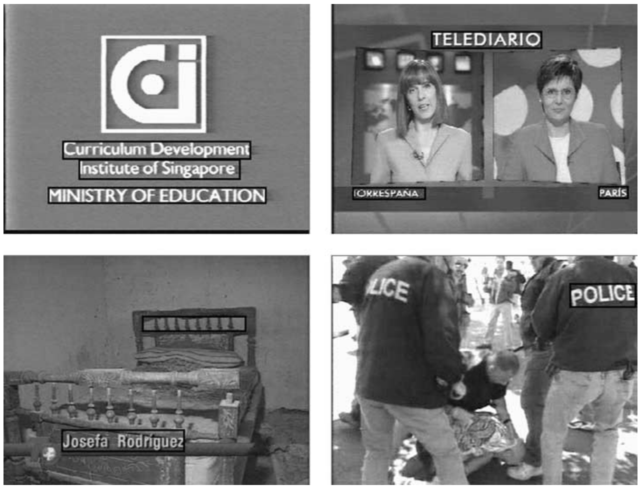


Fig. 14. Examples of text detection from video images.

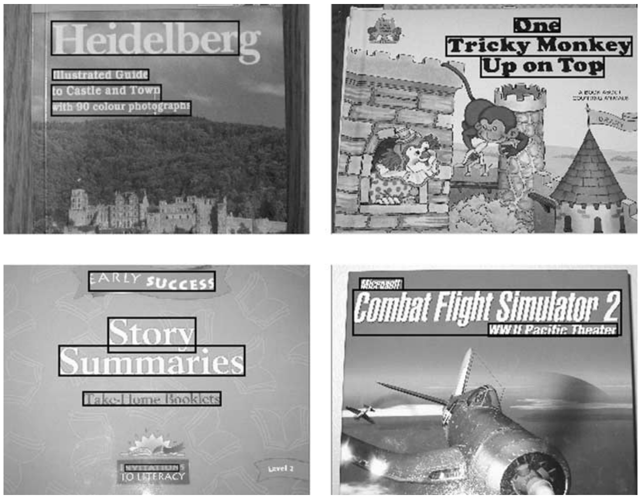


Fig. 15. Examples of text detection on (a)-(c) book covers and (d) CD box cover.

signs. Fifty images, which had different amounts of deformation, were selected from our Chinese sign database (a total of more than 2000 images). These images include a total 251 characters to be recognized. The results are listed in Table III. Without affine rectification, the recognition rate decreases rapidly as the angle between normal of the text plane and Z-axis of camera coordinate system increases. The restoration can improve the results significantly, especially when the angle ranges from 30° to 50°. If the angle is larger than that, the characters cannot be recovered properly due to the amount of the information lost in the imaging procedure.

We also use our sign detection algorithms in other text detection tasks such as text in video and text on paper. Fig. 14 shows some examples of the video images from Microsoft Research Asia [8]. Fig. 15 shows some examples of the text on book covers and a CD box cover.

We have evaluated the proposed approach on a Chinese sign recognition task. The current training set includes all level 1 characters in Chinese national standard character set GB2312–80 (a total of 3755 different characters). We also randomly selected 1630 character images from our sign library,

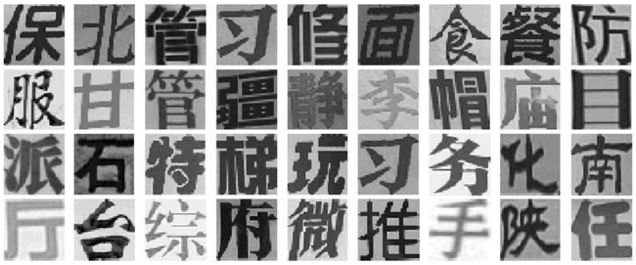
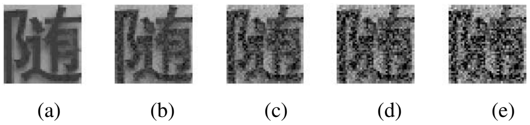


Fig. 16. Various characters from signs in natural scenes.



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

Fig. 17. Character with different noise. (a) Original one, (b)-(e) with 5%, 10%, 15%, and 20% Gaussian noise.

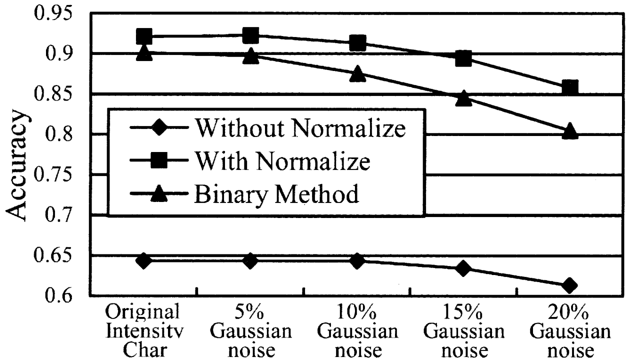


Fig. 18. Performance comparison of different approaches.

which contains more than 8000 characters in more than 2000 sign images from natural scenes, to form the testing set. The different characters in the testing set cover roughly 1/5 of the level 1 Chinese characters. Fig. 16 shows some examples from the testing set. The recognition accuracy is 92.46%.

This is an encouraging result because the character images in the testing set are captured from natural scenes, with variations in fonts, lighting conditions, rotation, and even affine deformation. For further verification of the robustness of the proposed approach against noises, we add the zero mean Gaussian noise to each character image in the testing set. For a given pixel in the character image whose intensity is $L(x,y)$, we have

$$L_{noise}(x,y) = L(x,y) + n(x,y) \tag{23}$$

where

$$n(x,y) \sim N\left(0, (L(x,y)\cdot\gamma)^2\right).$$

Parameter γ represents the intensity of the noise. Fig. 17 illustrates the impacts on a Chinese character from testing set when we add noise $\gamma = 0.05$, $\gamma = 0.10$, $\gamma = 0.15$ and $\gamma = 0.20$, respectively.

The top curve of Fig. 18 shows the recognition rate of 1630 characters in testing set when we add different intensities of Gaussian noise. Compared to the original testing set, this is

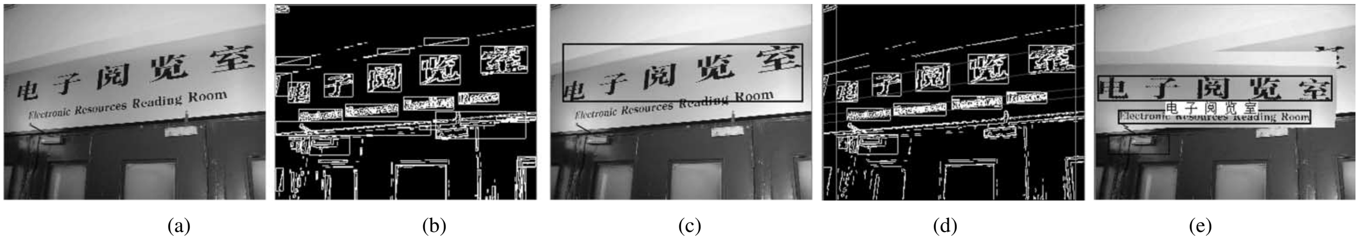


Fig. 19. Example of automatic sign detection and recognition.

only about a 1% decrease in recognition rate when 10% noise is added and about 6.5% decrease when 20% noise is added. This figure also illustrates the effectiveness of local intensity normalization; the recognition accuracy is about 24% to 28% higher than without intensity normalization. We also performed similar experiments on the binarization method. In the binary experiments, we also use the one simple per font training set. The results indicate that the accuracy is slightly lower for the original images, and that the accuracy decreases much faster than that of the intensity-based method when noise level is increased.

Fig. 19 illustrates the process from detection to recognition. Fig. 19(a) is the original input image, and (b) is the combination of detected candidates from two different resolutions. Fig. 19(c) is the detection result without affine rectification. It can be seen that only part of the characters are detected in (c). Similar to the example in Fig. 13(c), no rectangle sign frame can be found within the image. Therefore, the lines are fit from text. Note that the higher edge intensity corners are used for fitting in Fig. 19(d). The detected result with affine rectification is in Fig. 19(e), where all characters are detected and all Chinese characters are recognized correctly.

VI. Conclusion

In this paper, we present an approach for automatic detection and recognition of signs for application to sign translation. The proposed approach can robustly detect and recognize text signs from natural scenes. The edge-based method is used for coarse detection accompanied by a multiresolution scheme for different sign sizes. By combining the layout analysis and affine rectification, we obtain a markedly improved detection rate. An intensity-based OCR method is employed for sign recognition, where we apply a localized normalization method to enhance feature extraction, and use Gabor transform to extract features and LDA to select and reduce the dimensionality of the feature space. The method has been applied to a Chinese OCR task, which includes all level 1 characters in Chinese national standard character set GB2312–80 (total 3755 different characters). The detection and recognition approach has been implemented on different platforms, such as desktop, laptop, and palm-size PDA with an application of a Chinese sign translation system, which can automatically detect Chinese text input from a camera, recognize the text, and translate the recognized text into English. We are combining the traditional 2-D based recognition method with 3-D preprocessing technology to better understand the text in a 3-D world.

Acknowledgment

The authors would like to thank the associate editor and anonymous reviewers for their invaluable comments and suggestions led to a great improvement on this manuscript. They also would like to thank Z. Sloane for his proofreading this manuscript.

References

[1] E. Barnes, "Image recognition for shipping container tracking and I.D.," *Adv. Imag.*, vol. 10, no. 1, pp. 61–62, 1995.

[2] M. S. Brown and W. B. Seales, "Document restoration using 3D shape: A general deskewing algorithm for arbitrarily warped documents," in *Proc. ICCV*, vol. 2, 2001, pp. 367–374.

[3] Y. Cui and Q. Huang, "Character extraction of license plates from video," in *Proc. CVPR*, 1997, pp. 502–507.

[4] D. Deng, K. P. Chan, and Y. Yu, "Handwritten Chinese character recognition using spatial Gabor filters and self-organizing feature maps," in *Proc. ICIP*, vol. 3, 1994, pp. 940–944.

[5] R. C. Gonzalez and R. E. Woods, *Digital Image Processing*. Reading, MA: Addison-Wesley, 1993.

[6] Y. Hamamoto, S. Uchimura, K. Masamizu, and S. Tomita, "Recognition of handprinted Chinese characters using Gabor features," in *Proc. 3rd ICDAR*, vol. 2, 1995, pp. 819–823.

[7] Q. Hou, Y. Ge, and Z. Feng, "High performance Chinese OCR based on Gabor features, discriminative feature extraction and model training," in *Proc. ICASSP*, vol. 3, 2001, pp. 1517–1520.

[8] X. S. Hua, W. Liu, and H. J. Zhang, "Automatic performance evaluation for video text detection," in *Proc. 6th ICDAR*, Seattle, WA, USA, Sept. 10–13, 2001, pp. 545–50.

[9] A. K. Jain and B. Yu, "Automatic text location in images and video frames," *Pattern Recognit.*, vol. 31, no. 12, pp. 2055–2076, 1998.

[10] J. J. Koenderink and A. J. Van Doorn, "Representation of local geometry in the visual system," *Biol. Cybern.*, vol. 55, pp. 367–375, 1987.

[11] S. Kumano, K. Miyamoto, M. Tamagawa, H. Ikeda, and K. Kan, "Development of container identification mark recognition system," *Trans. Inst. Electron., Inform., Commun. Eng. D-II*, vol. J84D-II, no. 6, pp. 1073–1083, 2001.

[12] C. M. Lee and A. Kankanhalli, "Automatic extraction of characters in complex scene images," *Int. J. Pattern Recognit. Artif. Intell.*, vol. 9, no. 1, pp. 67–82, 1995.

[13] H. Li, D. Doermann, and O. Kia, "Automatic text detection and tracking in digital video," *IEEE Trans. Image Processing*, vol. 9, pp. 147–156, Jan. 2000.

[14] R. Lienhart, "Automatic text recognition for video indexing," in *Proc. ACM Multimedia 96*, 1996, pp. 11–20.

[15] Y. Lim, S. Choi, and S. Lee, "Text extraction in MPEG compressed video for content-based indexing," in *Proc. 15th ICPR*, vol. 4, 2000, pp. 409–412.

[16] R. Mehrotra, K. R. Namuduri, and N. Ranganathan, "Gabor filter-based edge detection," *Pattern Recognit.*, vol. 25, no. 12, pp. 1479–1494, 1992.

[17] R. Mullot, C. Olivier, J. L. Bourdon, P. Courtellemont, J. Labiche, and Y. Lecourtier, "Automatic extraction methods of container identity number and registration plates of cars," in *Proc. Int. Conf. Industrial Electronics, Control, Instrumentation*, vol. 2591, 1991, pp. 1739–44.

[18] J. Ohya, A. Shio, and A. Akamatsu, "Recognition of characters in scene images," *IEEE Trans. Pattern Anal Machine Intell.*, vol. 16, no. 2, pp. 214–220, 1994.

[19] T. Pavlidis, "Recognition of printed text under realistic conditions," *Pattern Recognit. Lett.*, vol. 14, no. 4, pp. 317–326, 1993.

[20] T. Sato, T. Kanade, E. K. Hughes, and M. A. Smith, "Video OCR for digital news archives," in *IEEE Int. Workshop on Content-Based Access of Image and Video Database*, 1998.

[21] H. Szu, B. Telfer, and J. Garcia, "Wavelet transforms and neural networks for compression and recognition," *Neural Networks*, vol. 9, no. 4, pp. 695–708, 1996.

[22] Y. Watanabe, Y. Okada, Y. B. Kim, and T. Takeda, "Translation camera," in *Proc. 14th ICPR*, 1998, pp. 613–617.

[23] E.L. Walker and T. Kanade, "Shape Recovering of a Solid of Resolution from Apparent Distortions of Patterns," Carnegie-Mellon Univ., Pittsburgh, PA, vol. CMU-CS-80–133, 1984.

[24] L. Wang and T. Pavlidis, "Direct gray-scale extraction of features for character recognition," *IEEE Trans. Pattern Anal. Machine Intell.*, vol. 15, pp. 1053–1067, Oct. 1993.

[25] L. Wiskott, J. M. Fellous, N. Kruger, and C. Malsburg, "Face recognition by elastic bunch graph match," *IEEE Trans. Pattern Anal. Machine Intell.*, vol. 19, pp. 764–768, July 1997.

[26] A. P. Witkin, "Recovering surface shape and orientation from texture," *Artif. Intell.*, vol. 17, no. 1–3, pp. 17–45, Aug. 1981.

[27] E. K. Wong and M. Chen, "A Robust algorithm for text extraction in color video," *Proc. IEEE Int. Conf. on Multimedia and Expo.*, vol. 2, pp. 797–800, 2000.

[28] V. Wu, R. Manmatha, and E. M. Riseman, "TextFinder: An automatic system to detect," *IEEE Trans. Pattern Anal. Machine Intell.*, vol. 21, pp. 1224–1229, Nov. 1999.

[29] J. Yang, W. Yang, M. Denecke, and A. Waibel, "Smart sight: A tourist assistant system," in *Dig. Papers 3rd Int. Symp. Wearable Computers*, 1999, pp. 73–78.

[30] J. Yang, J. Gao, Y. Zhang, and A. Waibel, "Toward automatic sign translation," in *Proc. Human Language Technology Conf.*, San Diego, CA, Mar. 2001, pp. 269–274.

[31] J. Yang, X. Chen, J. Zhang, Y. Zhang, and A. Waibel, "Automatic detection and translation of text from natural scenes," in *Proc. ICASSP*, vol. 2, 2002, pp. 2101–2104.

[32] H. Yoshimura, M. Etoh, K. Kondo, and N. Yokoya, "Grayscale character recognition by Gabor jets projection," in *Proc. 15th ICPR*, vol. 2, 2000, pp. 335–338.

[33] J. Zhang, X. Chen, A. Hanneman, J. Yang, and A. Waibel, "A robust approach for recognition of text embedded in natural scenes," in *Proc. 16th ICPR*, vol. 3, 2002, pp. 204–207.

[34] Y. Zhong, K. Karu, and A. K. Jain, "Locating text in complex color images," *Pattern Recognit.*, vol. 28, no. 10, pp. 1523–1536, 1995.

[35] J. Gao and J. Yang, "An adaptive algorithm for text detection from natural scenes," in *Proc. CVPR'-1*, vol. , 2001, pp. 84–89.

Xilin Chen (M'98) received the B.S., M.S., and Ph.D. degrees in computer science from Harbin Institute of Technology, China, in 1988, 1991, and 1994, respectively.

He has been a Professor at Harbin Institute of Technology since 1999. He has been a Visiting Scholar at Carnegie Mellon University, Pittsburgh, PA, since 2001. His research interests are image processing, pattern recognition, computer vision and multimodal interface.

Dr. Chen has served as a program committee member for several international and national conferences. He received several awards, including one National Scientific and Technological Progress Award in 2000, for his research work. He is a member of the IEEE Computer Society.

Jie Yang (M'91) received the Ph.D. degree from the University of Akron, Akron, OH, 1994.

He is currently a Senior Systems Scientist at the School of Computer Science in Carnegie Mellon University, Pittsburgh, PA. He pioneered hidden Markov model for human performance modeling in his Ph.D. dissertation research. He joined the Interactive Systems Laboratories in 1994, where he has been leading research efforts to develop visual tracking and recognition systems for multimodal human computer interaction. He developed adaptive skin color modeling techniques and demonstrated software-based real-time face tracking system in 1995. He has involved developments of many multimodal systems in both intelligent working spaces and mobile platforms, such as gaze-based interface, lipreading system, image-based multimodal translation agent, multimodal people ID, and automatic sign translation systems. His current research interests include multimodal interfaces, computer vision, and pattern recognition.

Jing Zhang received the B.S. degree in computer science from Jilin University, China, in 1989, and the M.S. and Ph.D. degrees in computer science from Harbin Institute of Technology, China, in 1992 and 1996, respectively.

She then joined Motorola-NCIC Joint R&D Laboratory, Beijing, China, where she worked on MPEG-2 and DVB technologies. From 1999, she worked in Compunicate Technology, Inc., a design house in Beijing, China, where her work was to design DVB receiver and Palm-size PCs. She joined Mobile Technologies, LLC, Pittsburgh, PA, in 2001, where she worked as a Senior Scientist, and her research focused on recognition technologies on portable computational devices. Her research interests are image processing, OCR, and embedded systems.

Alex Waibel (M'83) received the B.S. degree in electrical engineering from the Massachusetts Institute of Technology, Cambridge, in 1979, and the M.S. and Ph.D. degrees in computer science from Carnegie Mellon University (CMU), Pitssburgh, PA, in 1980 and 1986, respectively.

He is a Professor of computer science at CMU and the University of Karlsruhe, Germany. He directs the Interactive Systems Laboratories (http://www.is.cs.cmu.edu) at both universities with research emphasis in speech recognition, handwriting recognition, language processing, speech translation, machine learning, and multimodal and multimedia interfaces. At CMU, he also serves as Associate Director of the Language Technology Institute and as Director of the Language Technology Ph.D. program. He was one of the founding members of CMU's Human Computer Interaction Institute (HCII) and continues on its core faculty. He was one of the founders of C-STAR, the international consortium for speech translation research and served as its Chairman from 1998 to 2000. His team developed the JANUS speech translation system, the JANUS speech recognition toolkit, and a number of multimodal systems including the meeting room, the Genoa Meeting recognizer, and meeting browser.

Dr. Waibel's work on the time delay neural networks was awarded the IEEE Best Paper Award in 1990; his work on multilingual and speech translation systems the Alcatel SEL Research Prize for Technical Communication in 1994; the Allen Newell Award for Research Excellence from CMU in 2002; and the *Speech Communication* Best Paper Award in 1992.