Automatic Sign Detection and Recognition in Natural Scenes

Piyanuch Silapachote, Jerod Weinman, Allen Hanson, Richard Weiss†, and Marwan A. Mattar

Department of Computer Science
University of Massachusetts Amherst
Amherst, MA 01003

†School of Cognitive Science
Hampshire College
Amherst, MA 01002

{pla, weinman, hanson, mmattar}@cs.umass.edu, rswCS@hampshire.edu

Abstract

Visually impaired individuals are unable to utilize the significant amount of information in signs. VIDI is a system for detecting and recognizing signs in the environment and voice synthesizing their contents. The wide variety of signs and unconstrained imaging conditions make the problem challenging. We detect signs using local color and texture features to classify image regions with a conditional maximum entropy model. Detected sign regions are then recognized by matching them against a known database of signs. A support vector machine classifier uses color to focus the search, and a match is found based on the correspondences of corners and their associated shape contexts. Our dataset includes images of downtown scenes with several signs exhibiting both illumination differences and projective distortions. A wide range of signs are detected and recognized including both text and symbolic information. The detection and the recognition components each perform well on their respective tasks, and initial evaluations of a complete detection and recognition system are promising.

1. Introduction

An automated sign detection and recognition system provides a visually impaired person the chance to obtain useful information from signs the same way a sighted individual does. With an embedded language translator [25], it further eliminates language barriers. Integrated into a Personal Digital Assistant platform, the system becomes a mobile traveling aid [26].

Several techniques for sign detection and recognition have recently been studied. Much of the previous work has focused on a constrained domain of standardized traffic signs (e.g., [18, 2]). Many systems rely on the knowledge that traffic signs are designed to be easily noticeable, with colors that stand out from the background. Others [15] ignore color cues. Instead, edge orientations and hierarchical templates are used to search for specific geometrical shapes of signs, which are then classified by a decision tree and a moment-based shape descriptor.

Many approaches to text detection have been developed [11, 12, 13, 24]. Some have created systems specifically to detect and read text from signs in natural images. Chen and Yuille [9] demonstrate a visual aid system for the blind that employs a chain of AdaBoost classifiers to detect text regions using highly selective features. After binarization, an OCR system is used to read or reject the detected text. In other similar work, Chen et al. [8] detect Chinese characters in natural images, which are then recognized and translated into English.

Although both text and road signs play an important role in navigation and contain crucial information for drivers and pedestrians, other commercial signs, such as restaurants and banks, are no less significant to the visually impaired. Broader than previously proposed systems, our goal is to detect and recognize a wide variety of "sign", including traffic, government, public, and commercial signs. In this unconstrained domain, signs may be of arbitrary colors and shapes. They may be composed solely of text, symbolic signs containing logos alone and no text, or a combination of both text and logos. While approaches to detecting and recognizing such signs appear to be similar to those used for traffic signs or text, detection and recognition of signs under this broader definition is more challenging. Complications arise not only from how signs are encountered in the environment – their sizes, orientations, and possible occlusions – but also from the broad variations in both text and symbols as well as colors and shapes.

Our system is called VIDI (for Visual Integration and Dissemination of Information). It will be a wearable device with a head mounted camera attached to a mobile com-

0-7695-2372-2/05/$20.00 (c) 2005 IEEE

putational platform, allowing a visually impaired user to receive useful information about the presence of signs in the immediate environment. The process consists of three stages: sign detection, sign recognition, and speech synthesis. The detection stage uses a discriminative maximum entropy model to classify image regions based on local color and texture features. The recognition matches hypothesized sign regions against a database of signs in two steps. First, color information is used to limit the number of signs considered. Second, salient corner features and shape information are used to rank possible matches for the query. The content of the best matched sign is determined and conveyed to a user via speech synthesis, where every sign has a vocalized message, such as the reading of text or its symbolic meaning when no text is present.

2. Detection

Generic sign detection is a challenging problem. Signs may be found at any size, anywhere in an image. Our goal is to detect signs containing an extraordinarily broad set of fonts, colors, arrangements, graphics, etc. As a result, we need a representation that treats images somewhat generically, yet captures the broad traits of signs in order to adequately distinguish them from uninteresting background.

We begin by dividing the image into square patches that will be the atomic units for a binary classification decision on whether the patch contains sign or not. A patch could consist of just one pixel, but such fine-grained decisions are not necessary. Since the image is being arbitrarily discretized into decision regions, they must only be small enough to ensure that the smallest signs to be detected are always well-covered.

In the following we describe the features that are calculated for each patch and the classification method that yields our detection results.

2.1. Features

Our overall approach to sign detection operates on the assumption that signs belong to some generic class of textures, and we will seek to discriminate such a class from the many others present in natural images.

It is well-established that a bank of scale and orientation-selective filters are effective as the first stage of image processing and mimics the so-called "simple cells" of mammalian visual systems. We follow this general framework, using the statistics of filter bank responses to describe local texture. Specifically, we use the statistics proposed by Portilla and Simoncelli [17], which are based on the steerable pyramid decomposition of an image into scale and orientation components. In addition to the usual central moments of filter responses, the features include correlations between the responses at different scales and orientations that prove necessary for texture synthesis. The pyramid is computed once for the entire image, and then these statistics are computed on a "sub-pyramid" corresponding to a region around each patch.

Scale and orientation-selective filters respond indiscriminately to singular step-edges and one or more bars. However, the text prominent in many signs may be thought of as a series of short bars that are similar to a grating. Recently, neurons were discovered in the visual cortex of monkeys that discriminate between one bar and a grating of several bars. One computational model for these so-called "grating cells" has been proposed by Petkov and Kruizinga [16]. We employ a slightly modified version of this model grating cell to measure more distinct local features, especially as an aid in detecting signs that contain text.

For a given scale and orientation, our modified version of the non-linear grating filter requires at least three proximal and nearly equal responses to a simple filter of that scale and orientation. The "receptive field" (or support) of a grating filter is a line segment orthogonal to the simple filter's orientation and six times the scale length (since a grating is defined as three bars). The three strongest simple filter response maxima are found along the line, and if they are all within some fraction (i.e., 90 percent) of the maximum response in the receptive field, then that maximum becomes the filter output; otherwise the response is zero. More details may be found in [23].

Additionally, histograms of patch hue and saturation are used. According to a likelihood-gain feature ranking [10], the top three most discriminative of these features turn out to be (in order): (i) the level of green hue (easily identifying vegetation as background) (ii) mean grating cell response (easily identifying text) and (iii) correlation between a vertically and diagonally oriented filter of moderate scale (the single most useful other textural feature).

2.2. Classification

Once features are calculated at each patch, we must classify them as sign or background. For this we employ a discriminatively-trained maximum entropy probability model, commonly used in language modeling [3]. Class-conditional maximum entropy models have been widely used for texture synthesis (e.g., the FRAME method of Zhu, Wu, and Mumford [27]) and require slow approximation methods for training and inference or sampling. For classification, an alternate approach is to train an *image*-conditional maximum entropy model. For the binary case,

$$p\left(y \mid \mathbf{x}, \Lambda\right) = \frac{1}{Z\left(\mathbf{x}\right)} \exp\left(\delta\left(y-1\right) \Lambda \cdot F\left(\mathbf{x}\right)\right),$$

where $y \in \{0, 1\}$ represents the class of a patch in image $\mathbf{x}$, Λ is a parameter vector of weights, and F is the feature

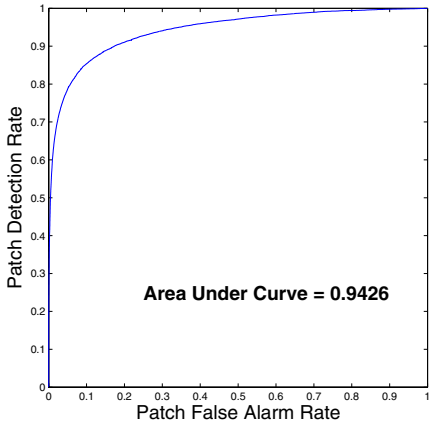

Figure 1. ROC curve for the probability that a patch is sign.

vector for the patch (as described above). Since the normalizing partition function Z is simply a sum over the two class labels rather than a sum over all possible images, no expensive approximation methods are necessary. With a labeled training sample of patch features, the maximum-likelihood estimate of parameters Λ may be found by convex optimization. A prior on the parameters is commonly used for regularization, leading instead to a MAP estimate. We follow this strategy using the typical Gaussian prior [7].

After training, classification involves checking whether the probability that an image patch is sign is above a threshold. For a MAP estimate of the label, the threshold is $1/2$, and the test can be determined simply by the sign of the dot product $\Lambda \cdot F(\mathbf{x})$.

2.3. Experiments

Tests of our detection method are performed on a hand-labeled database of 309 images collected from the downtown area of Amherst, MA with a still camera (available online [20]). The ratio of background to sign patches is more than 13:1 in our data set. Patches are 64 pixels on a side, and we break the 1024×768 images into 713 overlapping regions. For evaluation, we randomly split the images into ten sets, training on nine and testing on the held out set. We present the overall results for the entire data set with each of the ten sets held out in turn.

Figure 1 shows the ROC curve for the probability that a patch is sign. Although the curve does not directly correspond to the number of signs detected, it nicely illustrates the trade-off between false positives and the detection rate. If a higher false positive rate can be tolerated, we may lower the probability at which we accept a patch as sign and pass the resulting area to the recognizer.

Figure 2. Example detection results: green (solid) boxes indicate detected sign patches, and red (dashed) boxes are false positives.

Table 1. Detection performance at different probability thresholds. Sign level detection rate is the percentage of signs in which at least one patch is detected. When a sign is detected, coverage describes the percentage of the sign that is found.

Classification Threshold	$p \geq 0.50$	$p \geq 0.07$
Sign Detection Rate	84.46%	96.07%
Avg. Coverage	81±23%	94±13%
Median Coverage	91.75%	100%
False Alarms/Image	2±2	6±3

Figure 3. Examples of signs that went undetected. Most missed signs are either small and blurry, have projective distortions, specular reflections, low contrast, or uncommon orientation.

Overall detection performance is given in Table 1. Many of the signs that are labeled in the ground truth are found at the conservative MAP threshold $p \geq 0.5$. On average, there are only two false positive regions (connected patches) per image. As we show next, such false positives can be easily identified by their high match cost during recognition. These results are illustrated in Figure 2, showing MAP classification of images from the test set. Note that the signs contain characters in unconventional fonts, foreign characters, or no text at all.

We have determined 85 of the 87 signs that are missed completely suffer from one or more of the following conditions: small and blurry (34%), projective foreshortening (13%), low contrast (42%), behind glass with specular reflection (33%), and text at an uncommon orientation (13%). Horizontal and vertical text ($\pm 30°$) is most common, so the textural properties are learned by the model, but any remaining text ($45 \pm 15°$) does not appear often enough to be learned well. Signs do not need to be parallel to the image plane, as many signs with foreshortening are detected. However, failure can occur when the projective distortion shrinks the space between letters enough to virtually eliminate the edges and grating effect. The MAP threshold is conservative, keeping the false alarm rate very low, but causing these signs to be missed. By lowering the threshold, we can detect 94% of the signs, although with more false alarms.

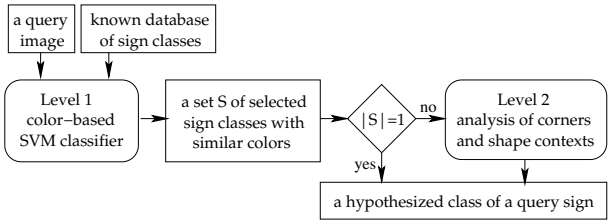

Figure 4. The structure of the sign recognition system.

3. Recognition

Recognition of a sign in a query image involves a two level hierarchical framework, as outlined in Figure 4. First a support vector machine (SVM) classifier narrows the search based on the apparent colors of a query image. Then the query and the relevant subset of signs in the database are compared in detail by ranking the correspondences of corners and their shape contexts.

Color and shape features contain mostly complementary image information. Colors may not always be present or usable and may be highly affected by natural lighting variations. Corners and shape contexts can provide more stable recognition by capturing the spatial relations among salient image points.

Although sufficient for recognition, the computational complexity of matching via correspondence of corners and their associated shape contexts makes it impractical to apply to a large database. For this reason, the color-based SVM classifier is an important focus of attention mechanism.

In the following we briefly describe the two stages of the recognition system; for further details the readers may consult [19].

3.1. Color-Based SVM Classifier

Humans are remarkably good at adapting to many color variations caused by natural illumination effects such as reflections, refractions, specularity, and shadowing. In principle, color can serve as a highly discriminative feature. However, the reliability of color information is limited due to its instability under uncontrolled outdoor settings. Accordingly, our sign recognizer relies on color only for a quick, coarse classification.

A query image is represented by two 1-dimensional histograms of hue and saturation from the hue-saturation-value color space [21]. The value component is ignored because illumination can vary considerably. The contribution of a pixel to the hue histogram is weighted by the amount of white light its color contains, i.e., its saturation. Furthermore, to account for the instability of hue when there is insufficient color, only pixels with saturations above a predefined threshold contribute to the hue histogram.

An SVM classifier [14] based on these color features restricts the subsequent search to a few signs with similar colors. The SVM is an approximate implementation of the Structural Risk Minimization induction principle in which a generalization error is minimally bounded and a margin is maximized. Since the SVM is formulated for binary classification, multi-class categorization is handled by a one-against-all voting scheme.

3.2. Matching Corners and Shape Contexts

In addition to color, shape is another discriminative feature in the general sign domain. Since it is neither possible nor practical to explicitly model the extremely large set of arbitrary shapes comprising signs, we employ a statistical shape descriptor. A sign is represented by a set of local corners whose geometric relations convey global shape information.

Geometrically speaking, a corner is the intersection of lines. We detect corners on the output of the Canny edge operator, a grayscale image whose brightness represents edge strength [22]. This edge map is thresholded to eliminate pixels with low edge strength. Since every edge should be of considerable length, neighboring pixels along an edge's orientation must also be marked as edges, otherwise it is disregarded as a non-edge pixel. This double thresholding of the edge map effectively reduces noise intensified by an edge detector.

Adopting the definition of Chabat, Yang, and Hansell [5], corners are points with strong gradient intensity and no single dominant gradient orientation. Non-maximum suppression is performed on a resulting cornerness map, and only those corners with sufficient arm supports (i.e., at least two associated edges) are accepted.

Every corner has a shape context defined by the distribution of the positions of all other corners relative to itself [1]. We measure the similarity between two images by computing correspondences between the corners across both images. We use the normalized cross-correlation between the shape contexts of two corners to measure the distance between them. Specifically, these distances are used by the Hungarian algorithm [4] to find the optimal correspondences between the two sets of corners.

This second stage of our recognition system begins with the detection of corners and the construction of their shape contexts. A query is compared to the relevant database images, as determined in the first stage. The relevant signs are ranked by their similarity to the query, and the top-ranked match is the output.

3.3. Data Collection and Experimental Setup

Frontal images of signs were taken around downtown Amherst, MA. Natural lighting effects were captured by taking one picture of each sign from approximately the same location at five different times throughout the day. Each image is hand-segmented and rotated in the plane from -90° to +90° at 10° intervals.

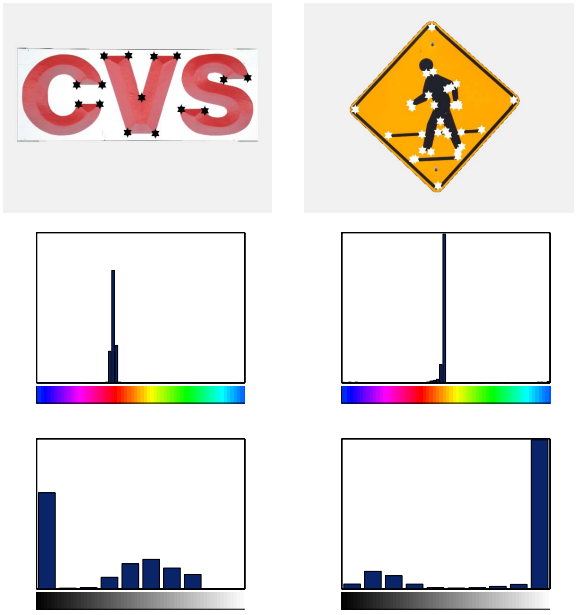

Figure 5. Image features for two signs: (Top) Detected corners overlaid on the signs; (Middle) Hue histogram: from blue to cyan colors along the x-axis; (Bottom) Saturation histogram: from low to high values along the x-axis.

Figure 6. An example of ranking results by the matcher. The image in the upper left corner is the query, the remaining 19 images are the matches in rank order: rank 1-4 on the first row, 5-9 on the second, 10-14 on the third, and rank 15, 19, 23, 27, and 31 on the fourth row. The length of the bar on the bottom of each sign represents the matching cost.

The database consists of 35 sign classes for a total of 3,325 sign images (available online [20]). We group the sign classes by color similarity into 8 superclasses, 3 of which contain a single sign class and the other 5 contain 2, 2, 7, 10, and 11 individual sign classes, respectively.

Color histograms (see Figure 5) are used as feature vectors for an SVM classifier [6] with a Gaussian radial basis kernel. The classifier performance is evaluated based on a 5-fold cross validation analysis.

When the SVM classifier labels a query as a superclass containing a single sign class, the recognition is complete. Other queries falling into multi-class superclasses are passed to the matching process, which finds the sign class that is the best match to the query.

The five original (non-rotated) images are considered our prototype, or representative, sign images in a conjectured superclass. With corners detected (see Figure 5) and shape contexts constructed, we perform a pairwise comparison between a query and every prototype. The similarity score provides a rank-order, and the top ranked prototype is selected as the matching sign.

3.4. Results

We first conduct an experiment on 24 sign classes selected from our database. The SVM puts every query into the correct superclass, and the matching process recognizes the sign in all but 2 of 1,900 queries. In both of these misclassified cases the correct match occurs at the second rank.

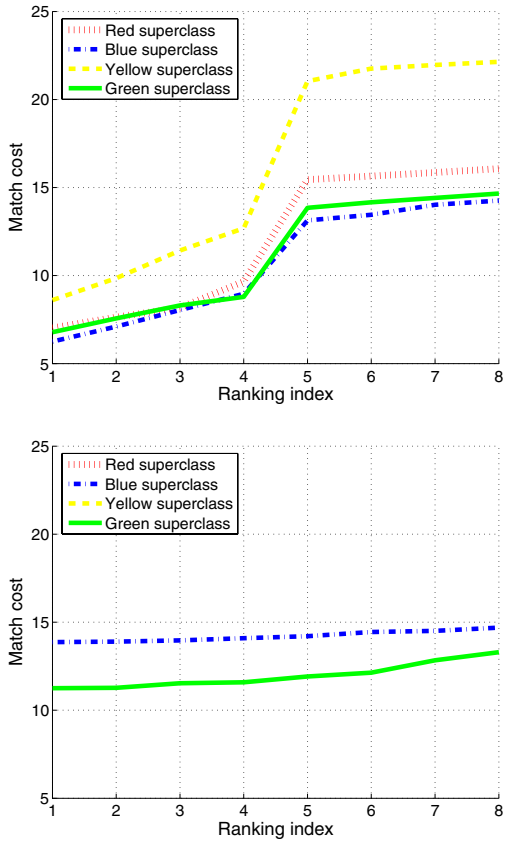

Figure 7. The average match costs by rank: (Top) Correctly classified queries; (Bottom) Misclassified queries.

Another experiment involves our full data set of 35 sign classes. In this case, the SVM classifier has a 97.14% accuracy and 97.83% for the matcher. This result demonstrates how additional complexity in the data weakens the performance of the recognizer. At the same time, it raises a question involving an error propagation from one level of the hierarchy to the next; a recovery of errors made by the coarse SVM classification before entering the matching stage.

In the following, we examine the cases where the superclassification by the SVM is correct. The match score of a query's true class is well-separated from those of other signs when it is ranked first. Figure 6 illustrates one such example where the matcher successfully ranks every prototype of the correct class first. Even though the true class of a query and the second ranked candidate class are both street-name signs whose contexts are very similar, the matcher gives them very different match costs. This behavior is typical of the ranking, as illustrated in Figure 7. When a match is correct, there is a noticeable jump between the 4^{th} and the

Figure 8. Four sample results of the integrated system. The top row is an image scene; green (solid) boxes mark detected sign regions and red (dashed) are false positives. The second row is a set of connected, detected patch components that are passed to the recognition system. The third row shows sign classes successfully recognized.

5^{th} rank match scores, which is not apparent when matches are incorrect. This sharp increase of the match scores indicates the change from the true class of the queries (ranked 1 to 4) to the incorrect ones. Instead of five for each sign class, there are only four prototypes included in the ranking since the prototype used to synthesize the query is excluded from match consideration.

4. End-to-End System

Here we report preliminary work on an integration of the detection and recognition components. Queries to the recognizer are connected image patches given by the automatic sign detection. Four examples of the detection and subsequent recognition are shown in Figure 8. There are both occlusions and background effects in the detected sign regions. Additionally, the sign in the third column has some projective distortion, and the fourth is relatively small in size. These examples show how the detection and the recognition components have robustly located the signs and matched their correct classes.

Although many signs in a number of different scenes are successfully detected and recognized, several challenges are still faced. For instance, the detected sign regions, which become queries to the recognizer, contain some image background and have occlusions. Additionally, natural illumination effects and projective distortions complicate the recognition. Furthermore, juxtaposed signs in a query image may be passed to the recognizer as one region, as shown in Figure 9. An intermediate step would be required to separate these before being passed to the recognition stage.

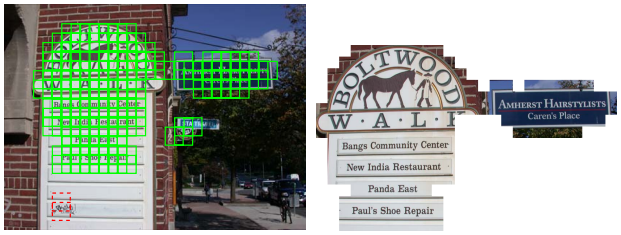

Figure 9. A scene where a detected region covers two signs.

5. Summary and Conclusions

We have proposed algorithms for detecting and recognizing general signs in everyday scenes. The individual components were tested successfully on experimental databases. Preliminary experiments on an integrated system give promising results and pose interesting challenges for further exploration.

We are able to detect signs containing text in non-standard fonts, foreign characters, and even those containing little or no text at all. A hierarchical recognition process reduces computation time by focusing the search for a match and provides flexibility by allowing different features at each level. Recognition on the testbed is accurate and robust to illumination effects and in-plane rotations. We are currently examining possibilities for recovering from errors made by the SVM.

VIDI is a work in progress. Besides an extension of our data set to a larger collection of scenes and sign classes, our immediate goal is to continue integrating the detection and recognition components and add the voice synthesis module. Finally, optimization is to be completed for practical use of the VIDI system as a wearable traveling aid for the visually impaired. We are also exploring the use of wireless telephony for transmitting images from the portable device to a server that would perform the recognition and voice synthesis, returning the result to the user over the same connection.

Acknowledgments

This work is supported by the NFS grant IIS-0100851.

References

[1] S. Belongie, J. Malik, and J. Puzicha. Shape matching and object recognition using shape contexts. *IEEE Transaction on Pattern Analysis and Machine Intelligence*, 24(4), 2002.

[2] M. Bénallal and J. Meunier. Real-time color segmentation of road signs. In *Proc. IEEE Canadian Conf. on Electrical and Computer Engineering*, Montréal, Québec, CA, May 2003.

[3] A. L. Berger, S. A. D. Pietra, and V. J. D. Pietra. A maximum entropy approach to natural language processing. *Computational Linguistics*, 22(1):39–71, 1996.

[4] N. Borlin. Hungarian: Matlab implementation. Dept. of Computing Science, Umeå Univ., Sweden, 1996.

[5] F. Chabat, G. Yang, and D. Hansell. A corner orientation detector. *Image and Vision Computing*, 17(10), Aug. 1999.

[6] C. Chang and C. Lin. Libsvm - A Library for SVM, 2002.

[7] S. F. Chen and R. Rosenfeld. A survey of smoothing techniques for me models. *IEEE Transactions on Speech and Audio Processing*, 8(1), Jan. 2000.

[8] X. Chen, J. Yang, J. Zhang, and A. Waibel. Automatic detection and recognition of signs from natural scenes. *IEEE Transactions on Image Processing*, 13(1):87–99, 2004.

[9] X. Chen and A. L. Yuille. Detecting and reading text in natural scenes. In *Proc. IEEE Conf. on Computer Vision and Pattern Recognition*, Washington, DC, June 2004.

[10] S. Della Pietra, V. Della Pietra, and J. Lafferty. Inducing features of random fields. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 19(4):380–393, 1997.

[11] C. Garcia and X. Apostolidis. Text detection and segmentation in complex color images. In *Proceedings of 2000 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP2000)*, volume 4, pages 2326–2330, June 2000.

[12] A. Jain and S. Bhattacharjee. Text segmentation using Gabor filters for automatic document processing. *Machine Vision Applications*, 5:169–184, 1992.

[13] H. Li, D. Doermann, and O. Kia. Automatic text detection and tracking in digital video. *IEEE Transactions on Image Processing*, 9(1):147–156, 2000.

[14] E. Osuna, R. Freund, and F. Girosi. Support vector machines: Training and applications. A.I. Memo 1602, Artificial Intelligence Laboratory, MIT, 1997.

[15] P. Paclík and J. Novovicová. Road sign classification without color information. In *Proc. of the 6th Annual Conference of the Advanced School for Computing and Imaging*, Lommel, Belgium, June 2000.

[16] N. Petkov and P. Kruizinga. Computational model of visual neurons specialised in the detection of period and aperiodic oriented visual stimuli: bar and grating cells. *Biological Cybernetics*, 76:83–96, 1997.

[17] J. Portilla and E. P. Simoncelli. A parametric texture model based on joint statistics of complex wavelet coefficients. *International Journal of Computer Vision*, 40(1):49–71, 2000.

[18] L. Sekanina and J. Torresen. Detection of Norwegian speed limit signs. In *European Simulation Multiconference*, 2002.

[19] P. Silapachote, A. Hanson, and R. Weiss. A hierarchical approach to sign recognition. In *Proc. of IEEE Workshop on Applications of Computer Vision*, pages 22–28, Breckenridge, Colorado, January 2005.

[20] P. Silapachote, J. Weinman, M. A. Mattar, and A. R. Hanson. The VIDI project web page. http://vis-www.cs.umass.edu/projects/vidi/index.html. University of Massachusetts Amherst.

[21] M. Swain and D. Ballard. Color indexing. *Intl. Journal of Computer Vision*, 7(1):11–32, Nov. 1991.

[22] E. Trucco and A. Verri. *Introductory Techniques for 3-D Computer Vision*. Prentice-Hall, New Jersey, 1998.

[23] J. Weinman, A. Hanson, and A. McCallum. Sign detection in natural images with conditional random fields. In *Proc. of IEEE Intl. Workshop on Machine Learning for Signal Processing*, pages 549–558, São Luís, Brazil, Sep. 2004.

[24] V. Wu, R. Manmatha, and E. M. Riseman. Finding text in images. In *DL'97: Proceedings of the 2nd ACM International Conference on Digital Libraries, Images, and Multimedia*, pages 3–12, 1997.

[25] J. Yang, J. Gao, Y. Zhang, X. Chen, and A. Waibel. An automatic sign recognition and translation system. In *Proc. Workshop on Perceptive User Interfaces*, FL, Nov. 2001.

[26] J. Zhang, X. Chen, J. Yang, and A. Waibel. A PDA-based sign translator. In *Proc. of the International Conference on Multimodal Interfaces*, 2002.

[27] S. C. Zhu, Y. Wu, and D. Mumford. Filters, random fields and maximum entropy (FRAME)–towards a unified theory for texture modeling. *International Journal of Computer Vision*, 27(2):1–20, 1998.