

Design of a visual perception model with edge-adaptive Gabor filter and support vector machine for traffic sign detection

Jung-Guk Park, Kyung-Joong Kim *

Cognition and Intelligence Lab, the Computer Science and Engineering, Sejong University, Seoul, South Korea

ARTICLE INFO

Keywords:

Machine learning
Gabor filter
Traffic sign detection
Ensemble learning
Generalization
Scale-space
Support vector machine

ABSTRACT

Traffic sign detection is a useful application for driving assistance systems, and it is necessary to accurately detect traffic signs before they can be identified. Sometimes, however, it is difficult to detect traffic sign, which may be obscured by other objects or affected by illumination or lightning reflections. Most previous work on this topic has been based on region of interest analysis using the color information of traffic signs. Although this provides a simple way to segment signs, this approach is weak when a sign is affected by illumination or its own color information is distorted. To overcome this, this paper introduces a robust traffic detection framework for cluttered scenes or complex city views that does not use color information. Moreover, the proposed method can detect traffic sign in the night. We establish an edge-adaptive Gabor function, which is derived from human visual perception. It is an enhanced version of the original Gabor filter, and filters out unnecessary information to provide robust recognition. It decomposes the directional information of objects and reflects specific shapes of traffic signs. Once the extracted feature is obtained, a support vector machine detects the traffic sign. Applying scale-space theory, it is possible to resolve the scaling problem of the objects that we want to find. Our system shows robust performance in traffic sign detection, and experiments on real-world scenes confirmed its properties.

© 2012 Elsevier Ltd. All rights reserved.

1. Introduction

Recently, driving assistant systems that incorporate traffic sign detection, pedestrian detection (Gerónimo, López, Saa, & Graf, 2010; Lie, Ge, Zhang, Li, & Zhang, 2011), traffic lane tracking (Cheng et al., 2010; Huang & Teller, 2011; McCall & Trivedi, 2006), and/or night vision features (Lim, Tsimhoni, & Liu, 2010) have been widely developed to improve safety. Traffic sign recognition has largely been studied using color models to obtain a region of interest (ROI) because color information can be readily obtained from cameras (Gomez-More, Maldonado-Bascon, Gil-Jimenez, & Lafuente-Arroyo, 2010; Tsai et al., 2008). However, this strategy is uncommon in practice, due to the high sensitivity of color to car lighting or natural illumination effects. In the real-world, unexpected noise may change the pixels of objects, resulting in the loss of functionality of color-based models.

With recent advances in the required hardware, it is possible to mobilize computing devices in vehicles. However, there remains the crucial problem of conditions that complicate detection in natural scenes. Several computer vision techniques have shown poor

performance when operating in real-world conditions. The ROI color-based approach mentioned above is a pioneer method in the field of object detection, and it allows for easy and effective segmentation. There are several ways to choose an ROI using color information. In Nguwi & Cho, 2010, an eigen map was introduced, produced by a radial basis function (RBF) neural network to accurately obtain the ROI. Similarly, Le et al. (2010) proposed a method for obtaining the color of interest of traffic signs using the block feature and a support vector machine (SVM). Another related method using color pixels and modeling was proposed in Nguwi and Cho (2010), which used hue saturation value (HSV) color space, and in Maldonado-Bascon, Lafuente-Arroyo, Gil-Jimenez, Gomez-More, and Lopez-Ferreras (2008), the achromatic decomposition method was used to overcome the effects of illumination on hue and saturation components. Although these approaches can determine changes in the pixels of traffic signs, they still underperform under unexpected real-world conditions. It makes other pre-processing methods or segmentation problems additionally.

In computer vision, a sliding window method that moves within a whole image to find the object is typically used, and it can detect characters in natural scenes quite accurately (Kim et al. 2003). For face detection, a convolution neural network with a sliding window is a successful scheme (Garcia & Delakis, 2004). Recently, in Peemen and Corporaal H. (2011), speed sign detection was

* Corresponding author. Tel.: +82 2 3408 3838.

E-mail addresses: machinelearning@sju.ac.kr (J.-G. Park), kimkj@sejong.ac.kr (K.-J. Kim).

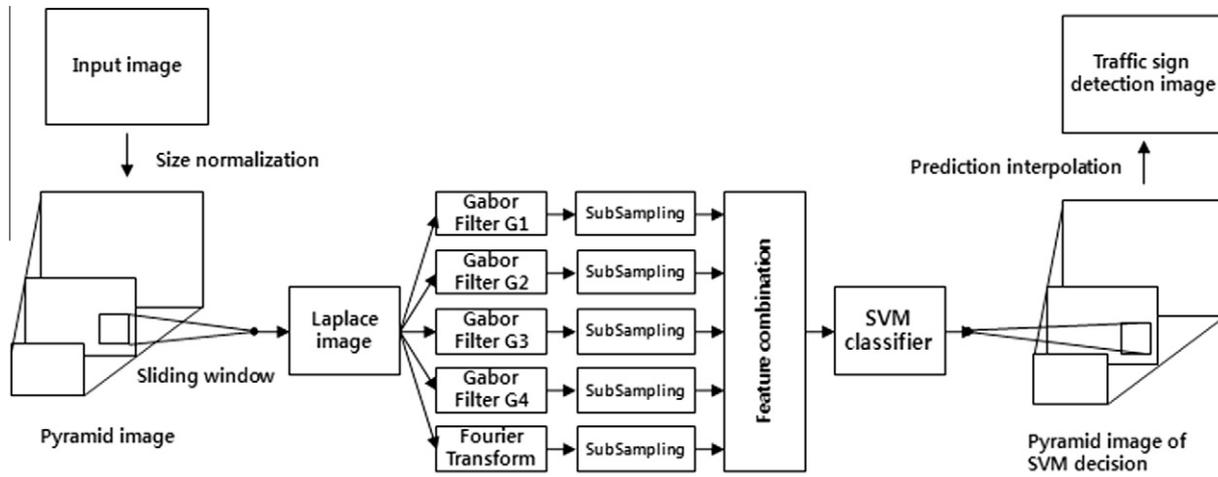


Fig. 1. Overview of the suggested traffic sign detection method.

introduced, using a convolution neural network and CUDA (Compute Unified Device Architecture). In that study, CUDA markedly reduced computation time and displayed high-definition images at 35 frames per second.

In this paper, we introduce a method for robust traffic sign detection that does not use color information. The method is based on a model of mammalian retina responses and machine-learning techniques. It partially segments an input image via a sliding window at every location using a Gabor filter. Jones and Palmer (1987) discovered that simple cells in the visual cortex of the brain of mammals can be modeled by Gabor filters. In our method, we extract robust features from images using a Laplacian operator called the edge-adaptive Gabor filter. By incorporating this function, more robust local and directional features than the original Gabor filter provides can be obtained. Moreover, a power spectrum is used for feature discrimination. More specifically, the proposed method consists of three parts: creating a pyramid scale, detecting objects via the edge-adaptive Gabor filter and verifying them by SVM, and assembling the prediction results with respect to the scale using scale-space theory (Tony, 1998). Fig. 1 shows an overview of the proposed method with n layers.

Because a single classifier suffers from lack of generalization (Park and Kim, 2012), our method employs an ensemble predictor to obtain a generalized classification via machine learning, thereby improving detection performance. A single predictor composed of a number of predictors would be expected to improve detection performance and reduce false positives. Indeed, in the present study, using bias-variance analysis for classification, we found that ensemble methods produce more accurate detection by increasing the number of SVMs. We analyzed our system using the machine learning algorithm Adaboost and tested its performance on images from Google Street View.

The remainder of this article is organized as follows. Section 2 details the proposed work and Section 3 presents the experimental results. Section 4 concludes the study and suggests future research directions.

2. Proposed method

Once an input image is given, the framework scans for traffic signs using an $n \times n$ pixel sliding window at each pixel. This is done with scale-space to overcome the scale of traffic signs. The scanning requires exhaustive trials to verify the segmented image and determine the traffic sign category to which it belongs.

To improve the generalization of the verification, the proposed filters discriminate local directional features and then machine

learning is used to establish the detection model based on each feature.

2.1. Edge-adaptive Gabor filtering for traffic sign detection

In computer vision, the shape of an object contains useful information for visual perception. Our method focuses on the edges of input images to remove unnecessary information.

The edge-adaptive Gabor filter convolves the input image as follows:

$$I_{G,\theta_k}(x,y) = I(x,y) \otimes G_{\theta_k}(x,y) \quad (1)$$

where \otimes is the convolution operator and I_{G,θ_k} is the feature image extracted by the filter as follows:

$$G_{\theta_k}(x,y) = G_{\theta_k}(x,y) \otimes L(x,y) \quad (2)$$

where $G_{\theta_k}(x,y)$ represents the edge-adaptive Gabor filter proposed in Daugman (1985), expressed as:

$$G_{\theta_k}(x,y) = \frac{1}{2\pi\sigma_x\sigma_y} \exp \left\{ -\frac{1}{2} \left(\frac{x_{\theta_k}^2}{\sigma_x^2} + \frac{y_{\theta_k}^2}{\sigma_y^2} \right) + j2\pi Wx_{\theta_k} \right\} \quad (3)$$

where $x_{\theta_k} = x \cos \theta_k + y \sin \theta_k$, $y_{\theta_k} = -x \sin \theta_k + y \cos \theta_k$, and σ_x^2 and σ_y^2 denote variances on the x - and y -axes, respectively. The family function of Eq. (3) is as follows:

$$G_{\lambda,\theta,\phi,\sigma,\gamma}(x,y) = \exp \left(-\frac{x'^2 + \gamma^2 y'^2}{2\sigma^2} \right) \cos \left(2\pi \frac{x'}{\lambda} + \phi \right) \quad (4)$$

in which x' and y' are the same as x_{θ_k} and y_{θ_k} in Eq. (3), respectively; λ denotes the strength of the wavelet; ϕ is the value that specifies the phase offset of the cosine factor of the function; γ affects the ellipticity of the Gaussian function; and σ is calculated using the additional variable, denoting the bandwidth of the filter. Features are extracted using four Gabor filters at the different orientation angles of 0° , 45° , 90° , and 135° to yield local directional information. It is possible to roughly approximate traffic signs using these four directions. The Laplacian operator $L(x,y)$ in Eq. (2) is obtained from:

$$L(x,y) = \frac{\partial^2 I(x,y)}{\partial x^2} + \frac{\partial^2 I(x,y)}{\partial y^2} \quad (5)$$

and we can derive the feature image I_{G,θ_k} . This is approximated to the 3-dimensional $[1, -2, 1]$ mask with respect to horizontal and vertical direction.

In machine learning, there have been efforts to reduce large feature dimensions to preserve useful information while eliminating

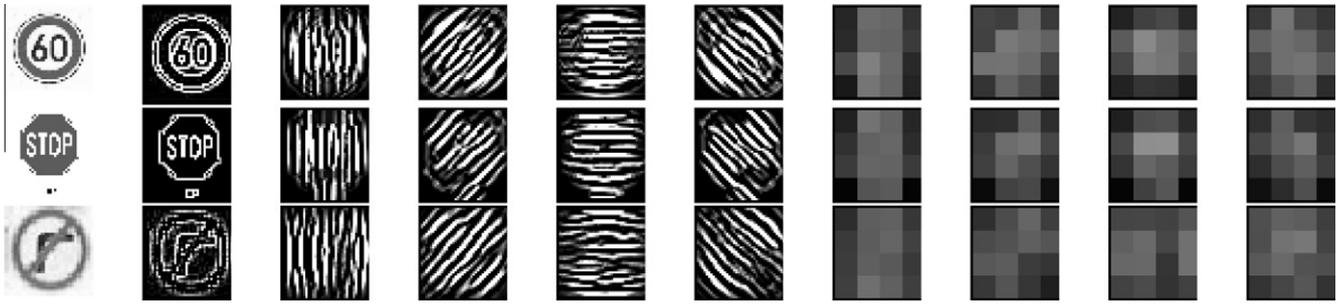


Fig. 2. Examples of the proposed feature of traffic sign images. The first column shows input image scaled to 32 by 32, second is for Laplace image, next images describe the four Gabor images and the corresponding subsampled images.

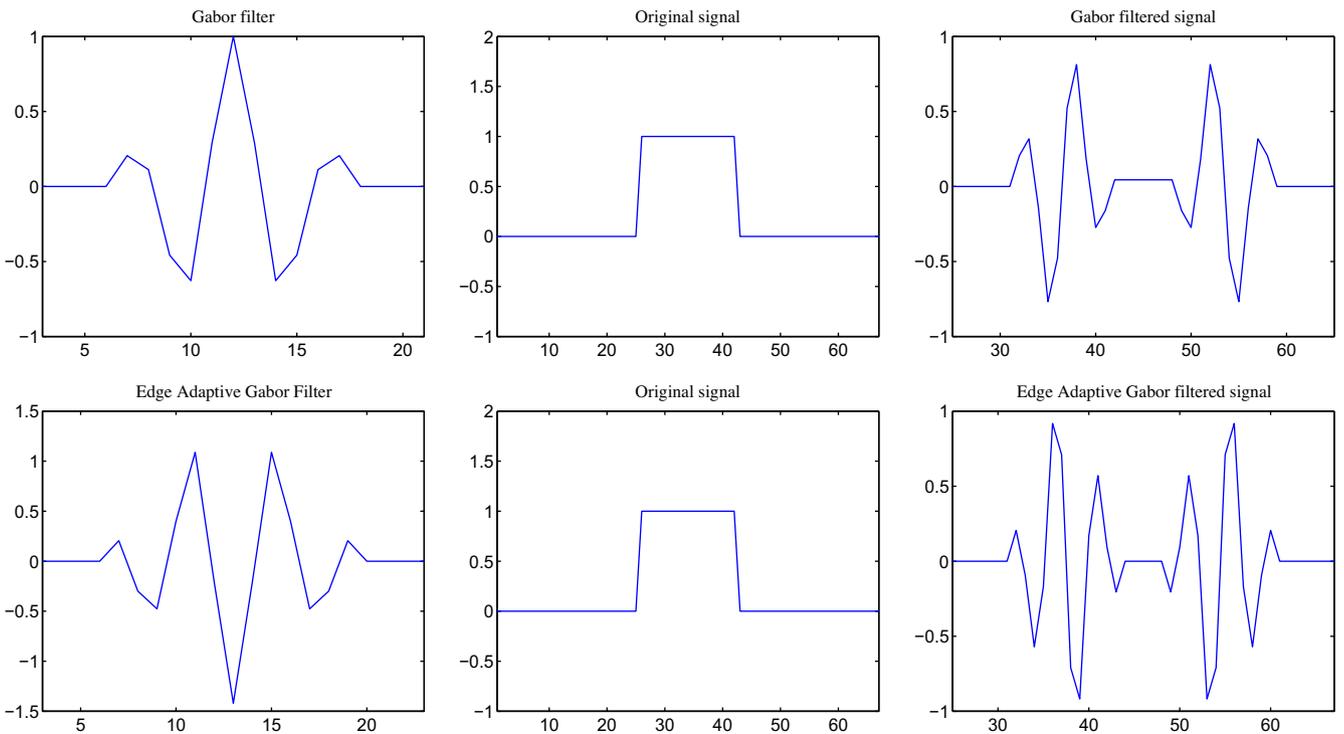


Fig. 3. Filtering illustration. The first row and the second row represent the original Gabor filter and the proposed filter, respectively. The first column represents filters, the second shows original signals and the third are filtered signals.

unnecessary dimensions. With feature dimension reduction, the detection system becomes more discriminative. Hence, our system obtains local information using a simple and low-computation subsampling method that reduces the noise of local information. The filtered image is subsampled as follows:

$$S_{G,\theta_k}(x,y) = \frac{1}{\beta^2} \sum_{i=-\frac{\beta}{2}}^{\frac{\beta}{2}} \sum_{j=-\frac{\beta}{2}}^{\frac{\beta}{2}} I_{G,\theta_k}(x+i,y+j) \quad (6)$$

where $S_{G,\theta_k}(x,y)$ denotes a subsample image from the filtered images I_{G,θ_k} and is the input for SVM, and β represents the length of the subsampling interval. Fig. 2 shows examples of a subsampled image with $\beta = 8$ and Fig. 3 illustrates the proposed edge-adaptive filter and the original Gabor filter.

2.2. Power spectrum feature

To make the system more discriminative, discrete Fourier transform (DFT) is used to create a power spectrum from the input

image. DFT converts the signal from a spatial domain into a frequency domain, and is expressed as follows:

$$F(k,l) = \sum_{x=0}^{N-1} \sum_{y=0}^{N-1} f(x,y) e^{-j2\pi(\frac{kx}{N} + \frac{ly}{N})} \quad (7)$$

where N denotes both the height and width of the input image and $f(x,y) = I(x,y) \otimes L(x,y)$, which represents the Laplace image in the spatial domain. The exponential term is the basis function, corresponding to each point $F(k,l)$ in the Fourier space. This can be interpreted as the value of each point $F(k,l)$ obtained by multiplying the spatial image by the corresponding base function and summing the results. The spectrum image is also subsampled by:

$$S_F(x,y) = \frac{1}{\beta^2} \sum_{i=-\frac{\beta}{2}}^{\frac{\beta}{2}} \sum_{j=-\frac{\beta}{2}}^{\frac{\beta}{2}} F(x+i,y+j) \quad (8)$$

For fast Fourier transform, let λ be the feature vector $\lambda = \{\lambda_1, \lambda_2, \lambda_3, \dots\}$ obtained via the edge-adaptive Gabor filter as follows:

$$\lambda_i = S_{G,k}(x,y) \quad \text{where} \quad i = k(xH + y) \quad (9)$$

The feature vector of the power spectrum is computed from:

$$\lambda_i = S_F(x,y) \quad i = xH + y \quad (10)$$

where i is the vector index and H denotes the height of the subsampled image.

2.3. Learning examples

The previous subsections dealt with the Edge Adaptive Gabor filter and power spectrum features as feature extraction part in machine learning. Machine learning mimics the biological learning scheme that experiences a condition and cope with the corresponding appropriate action. This scheme has been studied due to the fact that flexible fitting of their property on a given situation. It provides promising solutions with mathematic properties. SVM (Christopher, 1998) has the outstanding generalization ability in learning technique of discriminative method. Since SVM yields the large margin decision and convergences to global solutions, there are studies that show better performance comparing with other learning methods. In this study, SVM is used to verify the segmented image via the sliding window and entirely predicts an input scene at every scale. SVM provides the best solution for binary problem and ensures the optimal hyper plane with a large margin. The designed framework employs the detection problem into binary problem in which the observation belongs in the one of the categories of the traffic signs or other objects. Thus, the training dataset should include the traffic sign image and the non traffic sign image. We can crop the non traffic sign image with random sizes and locations from the scene where the traffic sign does not exist. To crop traffic sign image, Singapore's traffic sign dataset (Nguwi & Cho, 2010) is employed and external our work collects additional traffic sign images. Fig. 4 shows the cropped example image of the collection from real-world scenes. The real-world scenes to make non traffic sign image are collected from the Google

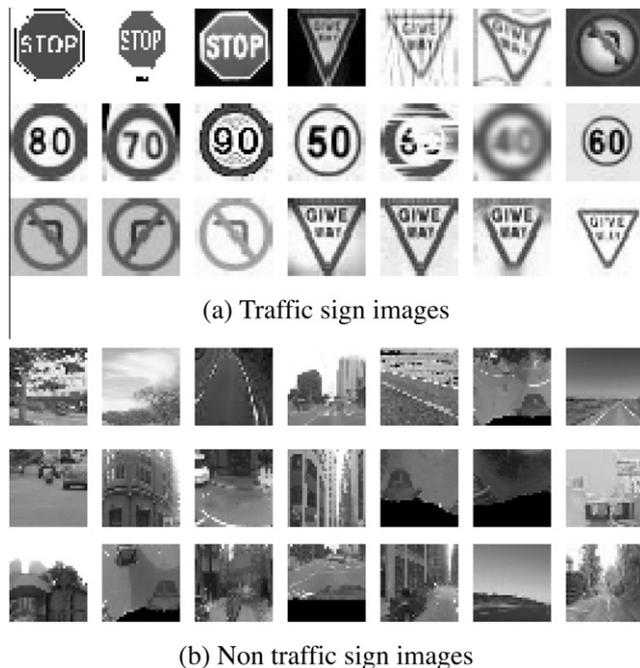


Fig. 4. Training images for support vector machine. First three rows represent traffic sign images and next three rows show non traffic sign images, (a) and (b) depict positive class and negative class for learning SVM, respectively.

Table 1
Numbers and types of traffic signs for SVM training.

Type		# of images	Singapore dataset	Google images	Sum
Inverted triangle	Yield	80		8	88
Circle	No entry	80	680	103	783
	No left turn	50			
	No right turn	50			
	Speed 10	50			
	Speed 20	50			
	Speed 30	50			
	Speed 40	50			
	Speed 50	50			
	Speed 60	50			
	Speed 70	50			
	Speed 80	50			
	Speed 90	50			
	Speed 100	50			
Octagon	Stop sign	80		20	100
Sum		840		131	971

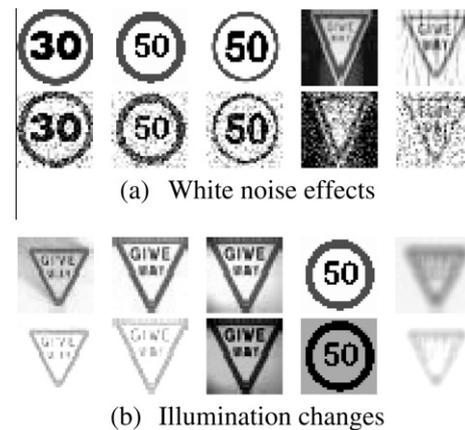


Fig. 5. Validation dataset images and original images, (a) and (b) show white noise effects and illumination changes, respectively. Each first row represents original image (training set) and second row illustrates their distortion (validation set).

website. Table 1 summarizes the training images into the three types of traffic signs.

2.4. Decision on prediction map

The final detection is achieved by interpolating each prediction into one scale, which is the same as the input image. Each prediction map at each scale is obtained via SVM, represented as follows:

$$P_k(x,y) = \mathbb{F}(\lambda_{x,y}) \quad (11)$$

where k is the Gaussian pyramid level, x, y are the coordinates of the prediction map, P_k is the prediction map at the k level, $\mathbb{F}(\cdot)$ represents the SVM, and $\lambda_{x,y}$ denotes the feature vector from the k th pyramid image at x, y . While the prediction runs over all scales, we can mark a black pixel, which represents a positive class as a candidate detection point in the initial white (negative class) map if SVM predicts that there is a traffic sign. The next step is to interpolate all candidate points to the input image scale. The prediction points (pixels) are rescaled via nearest-neighbor interpolation. The predicted marks at the interpolated scale are clustered by the connected components labeling method with eight neighborhoods (Shapiro, 2001). The candidate traffic sign locations are represented

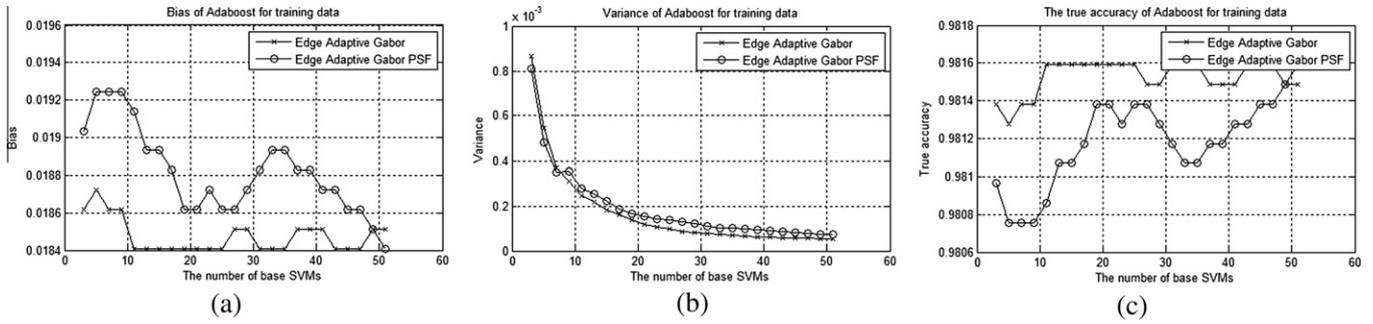


Fig. 6. Proposed Feature analysis with Adaboost result for training data. (a) Represents 1/0 bias analysis. (b) Shows 1/0 variance analysis. (c) Shows Classification accuracy.

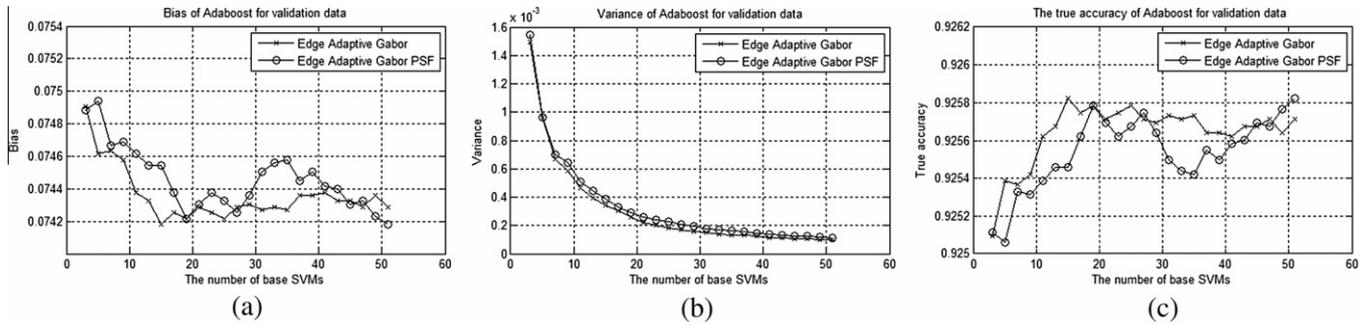


Fig. 7. Proposed Feature analysis with Adaboost result for validation data. (a) Represents 1/0 bias analysis. (b) 1/0 shows variance analysis. (c) Shows classification accuracy.

by the positive prediction of SVM. The final detection rectangle is obtained using the average of the clustered candidates and the corresponding coarsest scale within the cluster (the terms ‘fine’ and ‘coarse’ are from scale-space theory). Thus, we can obtain the detection rectangle, which encompasses the shape of the traffic sign when the candidate points are predicted on the coarse scale.

3. Experimental results

We analyzed our system using the ensemble method (Freund & Schapire, 1999), which aims to improve classification accuracy. A total of 9521 sample images were used for training the SVM, and 971 of these were related to traffic signs. The number of training images of non-traffic sign should be larger than traffic sign class because there are a number of non-traffic sign images in real-world scenes. All images were normalized to 32 × 32 pixels and all images that were categorized as being part of a traffic sign were assigned a value of 1 (positive class); otherwise, a value of 0 (negative class) was assigned. This experiment used gray-scale images and LIBSVM (Chang & Lin, 2011). All images used in this paper appear at http://cilab.sejong.ac.kr/home/doku.php?id=public:traffic_signdb.

3.1. Ensemble predictor

Adaboost is widely used as an ensemble predictor. It resamples training data in a sophisticated fashion, but tends to overfit training data. Hence, we used a validation dataset to evaluate ensemble performance by measuring the generalized error of the ensemble predictor. The dataset included white noise and the addition or deletion of raw pixels to create distortions. White noise follows a normal distribution $N(0, 1)$ and its magnitude was set at five steps ($\pm 2.5, \pm 5, \pm 10, \pm 15, \pm 25$). Pixel changes, which were made to simulate illumination effects, were also added at five steps ($\pm 25, \pm 50, \pm 75, \pm 100, \pm 150$) without white noise. This resulted in 9710 modified traffic sign images (10 variants of each of the 971

training images). Fig. 5 illustrates examples of the validation image set of traffic signs. The rest of the validation dataset consisted of 45,000 non-traffic sign images. These were randomly extracted from various natural scenes that were not used in the training set. To evaluate the results of Adaboost, Domingos (2000) bias and a variance analysis for classification were used. Parameters of Gabor filter are set according to Table 2.

Figs. 6 and 7 compare two feature extraction methods, the edge-adaptive Gabor filter alone and this filter with the additive power spectrum feature. Column (a)'s in these Figures show the analysis of bias and variances of the effects of Adaboost on the training dataset. The variances were gradually reduced and the biases were decreased by adding base classifiers. As shown in Figs. 6 and 7(b), the bias of both the training set and the validation set of the edge-adaptive Gabor method alone were lower than that of the Gabor method incorporating the additional power spectrum feature. As shown in column (c)'s, Adaboost yielded a more accurate classification of the validation set than the adaptive-edge Gabor filter with or without the additional power spectrum feature. Note that the true accuracy was measured via the synthesis decision of the Adaboosted SVMs, and is not the Domingo's error term.

Table 2

SVM classifier performance of validation data using 5-fold cross validation of Edge Adaptive Gabor Filter with respect to strength of Wavelet (row) and ellipticity of Gabor function (column). Bold represents the best accuracy.

	2	3	4	5	6
0.2	91.52 ± 0.20	95.62 ± 0.27	94.57 ± 0.38	95.75 ± 0.27	96.96 ± 0.37
0.3	92.40 ± 0.28	96.05 ± 0.32	96.12 ± 0.36	96.00 ± 0.26	97.13 ± 0.30
0.4	92.94 ± 0.28	96.27 ± 0.24	97.06 ± 0.36	96.66 ± 0.19	97.41 ± 0.33
0.5	92.94 ± 0.27	96.41 ± 0.27	97.30 ± 0.26	97.29 ± 0.18	97.62 ± 0.34
0.6	92.69 ± 0.39	96.49 ± 0.24	97.32 ± 0.25	97.60 ± 0.23	97.69 ± 0.47
0.7	92.42 ± 0.37	96.53 ± 0.18	97.36 ± 0.25	97.71 ± 0.25	97.76 ± 0.36
0.8	92.09 ± 0.35	96.60 ± 0.22	97.43 ± 0.22	97.91 ± 0.25	97.86 ± 0.38
0.9	91.61 ± 0.41	96.68 ± 0.26	97.46 ± 0.22	98.04 ± 0.27	97.93 ± 0.44



Fig. 8. Performance comparison over the (a) original Gabor filter, (b) Edge Adaptive Gabor filter and (c) additional power spectrum feature.

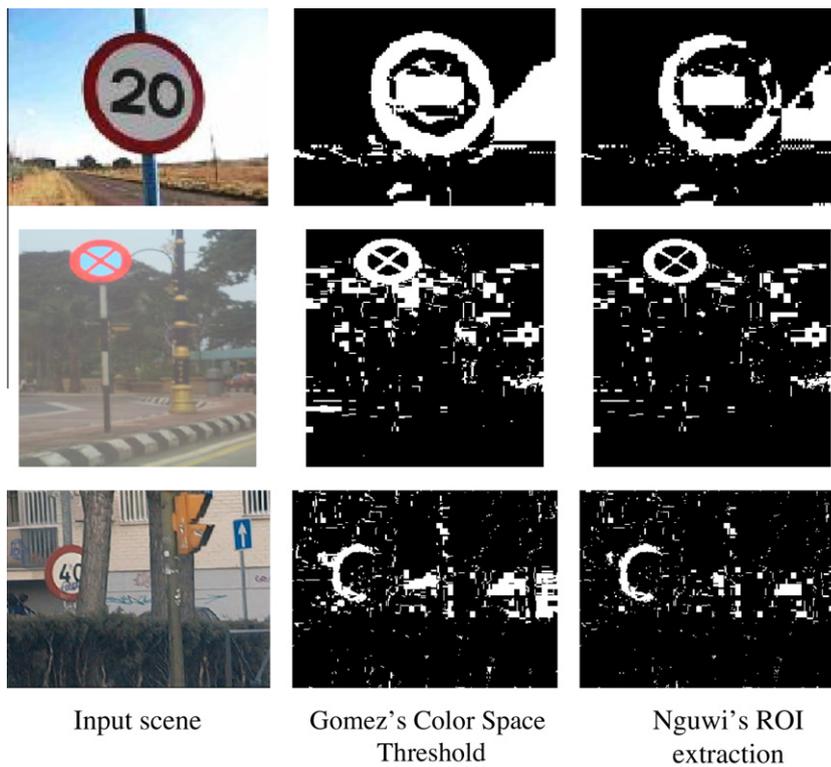


Fig. 9. Result of color model ROI causing segmentation problems.



Fig. 10. Experimental results on Google Street View.

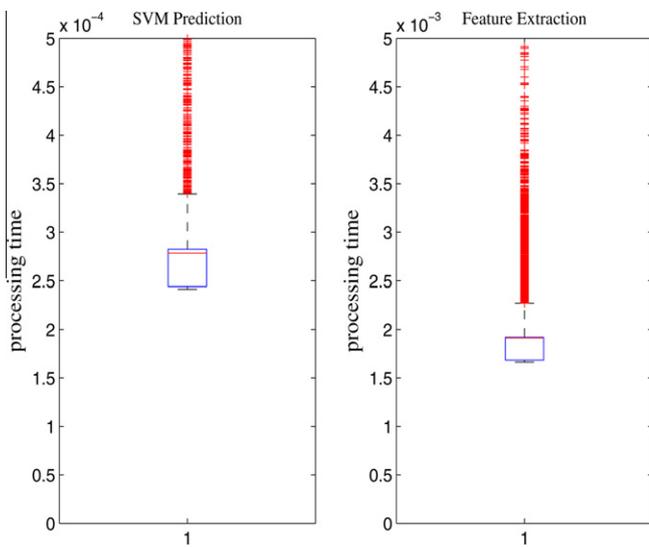


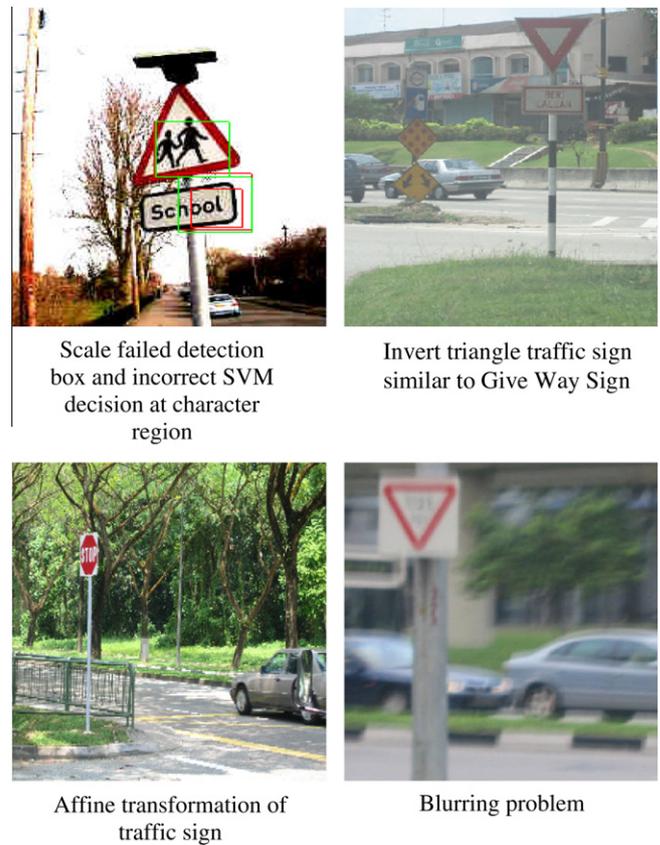
Fig. 11. Processing time of this study.

3.2. Traffic sign detection in real-world scenes

We applied our system to real-world scenes in Nguwi (2010) and Google Street View ICMLA (2011). The pyramid scale parameter was set to 1.2 and scanning runs of two pixels were conducted to verify the segmented image of the sliding window. The sliding window size was 32×32 pixels. Base Gabor filter is chosen with 5 strength of Wavelet and 0.9 ellipticity of Gabor function. Table 2 shows SVM classifier accuracy of our Edge Adaptive Gabor filter with respect to strength of Wavelet and ellipticity of Gabor filter. Fig. 8 compares the original Gabor filter, the edge-adaptive Gabor filter, and the filter incorporating the power spectrum feature. The latter filter significantly reduced false positives and yielded accurate detections. To examine color model for ROI extraction, Fig. 9 shows the results of ROI extraction. These suffer from generalization to color space in illumination effect and make another research topic to group objects in a given scene image.

Fig. 10 shows the resulting scenes extracted from Google Street View provided by ICMLA (International Conference on Machine Learning (2011)). These reflect real driving conditions, and the proposed method exhibited reliable performance.

Fig. 11 shows the processing time of our system tested under various conditions. The tests were conducted on an Intel Core i5 CPU 750, 2.67 GHz, and 3 GB RAM, and there was no use of code



Scale failed detection box and incorrect SVM decision at character region

Invert triangle traffic sign similar to Give Way Sign

Affine transformation of traffic sign

Blurring problem

Fig. 12. False detection of the proposed method.

optimization or parallel processing methods. For each prediction, the edge-adaptive Gabor filter and spectrum feature took 1.9×10^{-3} s and SVM took 2.7583×10^{-4} s. In the experiments testing the ensemble and real-world scenes, the optimized parameters of SVM had a polynomial kernel with the parameters $\hat{\gamma} = 1$ and $C = 1/\mathbb{N}$ that retrain generalization, where \mathbb{N} denotes the feature dimension. Our proposed system did have some limitations arising from blurring and affine transformations; these are shown in Fig. 12. Fig. 13 indicates the detection of scale invariance for traffic signs. Fig. 14 shows that the proposed framework detected traffic signs not only under bad weather conditions, but also under illumination effects.

Table 3 shows the receiver operating characteristic (ROC) analysis for the proposed method applied to 159 realworld scenes that

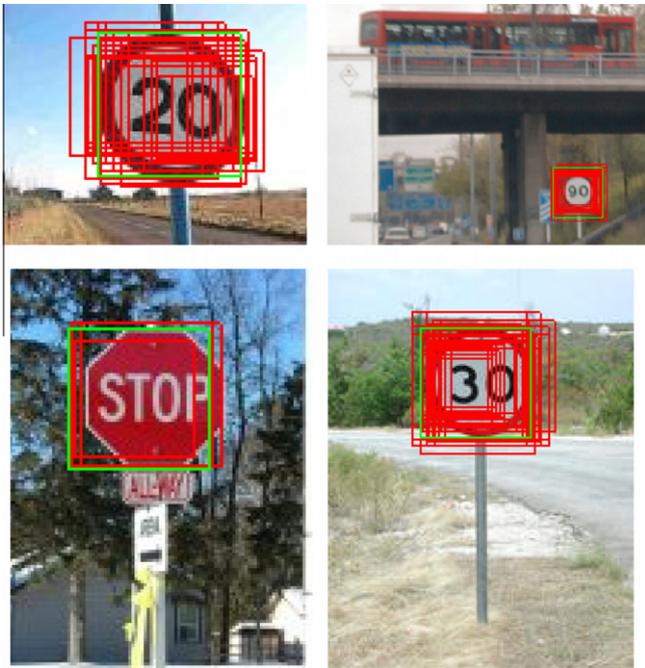


Fig. 13. Experimental results of the proposed method yielding scale invariant property.

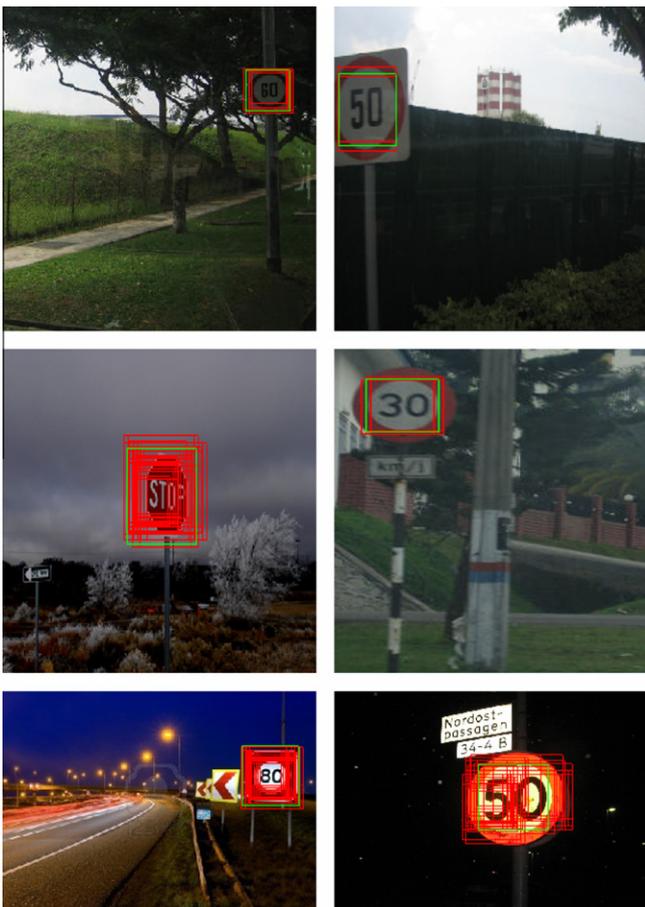


Fig. 14. Experimental results of the proposed method with illumination effects or different weather conditions.

Table 3

ROC analysis of the proposed method. The dataset of test scenes contains 168 traffic signs (sensitivity = TP/P, P: The number of traffic signs, TP: Correct predictions of SVM to traffic signs).

Traffic sign types	True positive rate (sensitivity)	False positive rate per image (false alarm)	False negative rate (miss)
Total	90.41%	8.98%	9.58%

Table 4

ROC analysis of the proposed method on 387 images from Google Street View.

True positive rate (sensitivity)	False positive rate per Image	False negative rate (miss)
85.93%	11.62%	14.06%

should include certain signs in Table 1. The proposed framework and the SVM showed good performance. Table 4 shows the results on the images from Google Street View.

4. Discussion and future research

Unlike other approaches to traffic sign detection, the system proposed herein does not depend on color information, but rather uses machine learning and an ensemble method to overcome illumination effects and improve traffic sign detection accuracy.

Although sliding window-based scanning requires many attempts to verify a segmented image, the proposed framework yields reliable classification and robust traffic sign detection under various illumination and weather conditions. Moreover, a relatively small number of traffic sign images are sufficient to train the SVM due to the discriminative edge-adaptive Gabor filter feature.

In our experiments, machine learning played an important role in making good generalizations and classifying traffic signs. For example, a false positive in one step could be rejected in the next step of identification, decreasing the overall rate of false identifications. The problems shown in Fig. 12 could be solved by adding various training samples that cause false detections. Fig. 14 shows segmentation problems with a color-based ROI model. Without segmentation using an ROI method, our method yielded robust traffic sign detection. Compared to a convolution neural network, our system, which incorporates an edge-adaptive Gabor filter and SVM, has the advantage of a single architecture. In contrast, the convolution neural network which yields reliable performance for object detection has a bipyramidal architecture and involves higher computational costs. It also consists of trainable weights for the feature map of both convolution layers and subsampling layers, whereas our system uses simple fixed weights that are identical in the convolution and subsampling layers. In the learning phase, the convolution neural network uses back-propagation for training weights and iterative boosting (Peemen & Corporaal, 2011) learning, which is essential to reduce false positives. In contrast, our system retains high performance without iterative boosting. Another issue is the computation cost for online processing. However, recently, research on CUDA (Nasse, Thurau, & Fink, 2009; Uetz & Behnke, 2009) has revealed outstanding processing time reduction with parallel computation through graphics processing units. Indeed, using CUDA for SVM, it is possible to decrease the processing time by as much as 22–172 times versus not using CUDA (Carpenter, 2009), and our method can be implemented via CUDA for real-time processing.

Acknowledgements

This research was supported by Basic Science Research Program through the National Research Foundation of Korea (NRF) funded by the Ministry of Education, Science and Technology (2010-0012876). The authors would like to thank Prof. Siu-Yeung Cho for their support on the traffic sign database.

References

- Carpenter, A. (2009). CUSVM, A CUDA implementation of support vector classification and regression. <<http://patternsonscreen.net/cuSVM.html>>.
- Chang, C.-C., & Lin, C.-L. (2011). LIBSVM: A library for support vector machines. *ACM Transactions on Intelligent Systems and Technology*, 3.
- Cheng, H.-Y., Yu, C.-C., Tseng, C.-C., Fan, K.-C., Hwang, J.-N., & Jeng, B.-S. (2010). Environment classification and hierarchical lane detection for structured and unstructured roads. *Computer Vision, IET*, 4, 37–49.
- Christopher, J. C. B. (1998). A tutorial on support vector machines for pattern recognition. *Data Mining and Knowledge Discovery*, 2, 121–167.
- Daugman, J. G. (1985). Uncertainty relations for resolution in space, spatial frequency, and orientation optimized by two-dimensional visual cortical filters. *Journal of the Optical Society of American*, 2, 1160–1169.
- Domingos, P. (2000). A unified bias-variance decomposition for zero-one and squared loss. In *Proceedings of the seventeenth national conference on artificial intelligence*, AAAI Press, pp. 564–569.
- Freund, Y., & Schapire, R. (1999). A short introduction to boosting. *Japanese Society for Artificial Intelligence*, 14, 771–780.
- Garcia, C., & Delakis, M. (2004). Convolutional face finder: A neural architecture for fast and robust face detection. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 26, 1408–1423.
- Gerónimo, D., López, A. M., Saa, A. D., & Graf, T. (2010). Survey of pedestrian detection for advanced driver assistance systems. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 32, 1239–1258.
- Gomez-More, H., Maldonado-Bascon, S., Gil-Jimenez, P., & Lafuente-Arroyo, S. (2010). Goal evaluation of segmentation algorithms for traffic sign recognition. *IEEE Transactions on Intelligent Transportation Systems*, 11, 917–930.
- Huang, A., & Teller, S. (2011). Probabilistic lane estimation for autonomous driving using basis curves. *Autonomous Robots, Springer Netherlands*, 31, 269–283.
- ICMLA (International Conference on Machine Learning and Applications). (2011). StreetView Recognition Challenge. <<http://www.icmla-conference.org/icmla11/>>.
- Jones, J. P., & Palmer, L. A. (1987). An evaluation of the two-dimensional Gabor filter model of simple receptive fields in cat striate cortex. *Journal of Neurophysiology*, 58, 1233–1258.
- Kim, K. I., Chung, K. C., & Kim, J. H. (2003). Texture-based approach for text detection in images using support vector machines and continuously adaptive mean shift algorithm. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 25, 1631–1639.
- Le, T. T., Tran, S. T., Mita, S., & Nguyen, T. D. (2010). Real time traffic sign detection using color and shape-based features, In *Proceedings of the second international conference on intelligent information and database systems*, Lecture Notes in Computer Science, Berlin Heidelberg: Springer, pp. 268–278.
- Lie, G., Ge, P.-S., Zhang, M.-H., Li, L.-H., & Zhang, Y.-B. (2011). Pedestrian detection for intelligent transportation systems combining Adaboost algorithm and support vector machine. *Expert Systems with Applications*, 13.
- Lim, J. H., Tsimhoni, O., & Liu, Y. (2010). Investigation of driver performance with night vision and pedestrian detection systems—Part I: Empirical study on visual clutter and glance behavior. *IEEE Transactions on Intelligent Transportation Systems*, 11, 670–677.
- Maldonado-Bascon, S., Lafuente-Arroyo, S., Gil-Jimenez, P., Gomez-More, H., & Lopez-Ferreras, F. (2008). Road-sign detection and recognition based on support vector machines. *IEEE Transactions on Intelligent Transportation Systems*, 8, 264–278.
- McCall, J. C., & Trivedi, M. M. (2006). Video-based lane estimation and tracking for driver assistance: Survey, system, and evaluation. *IEEE Transactions on Intelligent Transportation Systems*, 7, 20–37.
- Nasse, F., Thureau, C., & Fink, G. (2009). Face detection using GPU-based convolutional neural networks. *Computer Analysis of Images and Patterns, Lecture notes in Computer Science*, 83–90.
- Nguwi, Y.-Y., & Cho, S.-Y. (2010). Emergent self-organizing feature map for recognizing road sign images. *Neural Computing and Applications*, 19, 601–615.
- Park J. -G., & Kim K. -J. (2012). A method for feature extraction of traffic sign detection and the system for real-world scene. In *2012 IEEE international conference on emerging signal processing applications*.
- Peemen, B. M., & Corporaal H. (2011). Speed sign detection and recognition by convolutional neural networks. In *International automotive congress*.
- Shapiro, L. G. (2001). *Computer vision*. Prentice Hall, pp. 56–59.
- Tony, L. (1998). Feature detection with automatic scale selection. *International Journal of Computer Vision*, 30, 77–116.
- Tsai, L.-W., Hsieh, J.-W., Chuang, C.-H., Tseng, Y.-J., Fan, K.-C., & Lee, C.-C. (2008). Road sign detection using eigen colour. *Computer Vision, IET*, 2, 164–177.
- Uetz, R., & Behnke, S. (2009). Large-scale object recognition with CUDA-accelerated hierarchical neural networks. In *2009 IEEE international conference on intelligent computing and intelligent systems*, 1, 536–541.