Real-time Recognition of U.S. Speed Signs

Christoph Gustav Keller[1], Christoph Sprunk[1], Claus Bahlmann[2], Jan Giebel[3] and Gregory Baratoff[3]

[1] Computer Science Department
Albert-Ludwigs-University Freiburg
79110 Freiburg, Germany
{ckeller,sprunkc}@informatik.uni-freiburg.de

[2] Siemens Corporate Research, Inc
755 College Road East
Princeton, NJ 08540 USA
claus.bahlmann@siemens.com

[3] VDO Automotive AG
Peter-Dornier-Str. 10
88131 Lindau, Germany
{Gregory.Baratoff, Jan.Giebel}
@continental-corporation.com

***Abstract*— In this paper a camera-based system for detection, tracking, and classification of U.S. speed signs is presented. The implemented application uses multiple connected stages and iteratively reduces the number of pixels to process for recognition. Possible sign locations are detected using a fast, shape-based interest operator. Remaining objects other than speed signs are discarded using a classifier similar to the Viola-Jones detector. Classification results from tracked candidates are utilized to improve recognition accuracy. On a standard PC the system reached a detection speed of 27fps with an accuracy of 98.8%. Including classification, speed sign recognition rates of 96.3% were achieved with a frame rate of approximately 11fps and one false alarm every 42s.**

I. Introduction

Advanced driver assistance systems (ADAS) for cars have been evolving rapidly over the last decades. Due to the growth in computational performance these systems are able to handle more and more tasks. As with most car assistance systems, their goal is to improve the driver's safety and comfort. Detecting traffic signs can be used to foster the driver's awareness of the current road situation and warn of dangerous crossings, such as those indicated by stop signs. A system that is able to detect speed signs can ensure the driver is always aware of current speed limits and warn of overseen ones. Integrated into an adaptive cruise control system (ACC) the driver's cognitive load can be reduced, and safe driving is supported.

Traffic sign recognition systems (TSR) can also be used to ease the task of road maintenance. Assuring the visibility and readability of traffic signs is an ongoing task necessary to maintain safety on roads. However, street signs can be covered by obstacles, damaged, soiled or misaligned. Equipping service cars (e.g., police cars) with a sign detection system and the according database containing pre-located positions of important signs can reduce and automate maintenance tasks. Most work in recent years has focused on circular speed signs, which are used in Europe, Asia, and Australia. In this contribution, we describe a TSR system which is adapted for rectangular U.S. speed signs. Examples of U.S. speed signs can be found in Figure 1.

The author would like to thank Prof. Burkhardt from the Chair of Pattern Recognition and Image Processing at the University of Freiburg for his personal and professional support. This work was conducted while the author was with Siemens Corporate Research, Princeton, NJ, USA.

II. Related Work

To solve the problem of traffic sign detection most systems follow the typical approach used in computer vision systems. Generally, the task at hand is structured into the subproblems of pre-processing the acquired image data using different segmentation techniques, detecting signs in the image and finally classifying them. In some systems classification is improved by exploiting the available temporal information.

Color based Segmentation: Traffic sign colors have been chosen by the traffic authorities to assure an immediate focus on the sign. Hence, pre-segmenting the image using color thresholding techniques is widely described in publications (e.g., [8], [12], [17], [22]) in order to identify a region of interest. Because of the distinct colors used in traffic signs (e.g., red, blue, yellow) most authors prefer a linear or non-linear mapping from RGB to other color coordinate systems (i.e. HSV, HSI, YIQ, etc.) which are less sensitive to illumination changes by decoupling color and intensity information. Kehtarnavaz [12] claims that a transformation into the YIQ color space provides the best segmentation outcome for red and yellow traffic signs. He also points out that none of the different color systems provide adequate distinction qualities between black-and-white signs and the background. Unfortunately, most U.S. traffic signs have black letters on white background.

Shape based Segmentation: Another popular technique is exploiting shape information derived from edge features. The main advantage is its robustness with respect to different lighting conditions and the sign's degradation. A traditional method to detect circular structures, for example circular speed limit signs [6] is the circular Hough transform. Detecting straight line segments can be realized using the generalized Hough transform. Grouping detected peaks [11] can

Fig. 1: Example Images of U.S. Speed Signs

be used to detect rectangles [18]. Although hardware implementations for the generalized Hough transform exist [21], calculating the transformation and extracting matching peaks on large images is computationally too complex for real-time processing.

Barnes and Zelinsky [4] introduce a radial symmetry detector which uses the gradient image to detect (circular) Australian speed signs in real-time. Loy and Barnes [14] presented a modification of this algorithm to detect regular polygon-shaped signs (e.g., triangle, square, octagon). The regular polygon detector is implemented and extended in this work in order to detect rectangular speed signs in the United States of America.

Different template matching methods exist to detect sign shaped structures. Because of the computational complexity, they are often applied to smaller areas that are predetermined by color thresholding [22]. To improve performance, several techniques were proposed to speed up template matching. Betke and Makris [5] use a simulated annealing algorithm for fast template matching, and Gavrila [9] exploits matching properties in the distance-transform of the image. However, these methods adapted to the task of detecting U.S. speed signs are time-consuming due to template shape and size necessary to detect these signs.

Detection performance has been increased in many object detection applications with help of the Viola-Jones detector [23]. This detector has been applied in [1] to detect circular German traffic signs. Its application will be extended in this work to detect rectangular U.S. speed signs.

For a more accurate determination of the exact sign location usually a combination of color and shape segmentation is used. Edge [19], ring [20] and other features are used to detect the outline of a sign with relative robustness to occlusion and distortion.

Classification: The final classification of the sign is usually done by one of the common classifiers (Support Vector Machines [13], Neural Networks [22], Nearest Neighbor [5], Radial Basis Functions [10], etc.). Most often, the intensity image of the candidate region is passed on to the classifier.

Tracking: Tracking candidate signs from frame to frame allows redefining and restricting the region of interest to speed up the classification process [7]. In [16] the tracking of detected signs is realized by using a Kalman-filter framework with the assumption of a constant straight car movement. Bahlmann et al. [1] update the classification confidence by considering and re-weighting classification decisions from previous frames.

These techniques have mainly been used to detect circular speed signs. To the best of our knowledge only Moutarde et al. [15] applied a combination of shape based detection and neural network classification on rectangular U.S. speed signs.

III. Traffic Sign Recognition System Architecture

The system workflow can be divided into three different stages: Detection, classification, and tracking. In the detection stage each input frame is processed to detect speed signs of a fixed size. The search size is selected from a predefined range of sign sizes and is changed for every frame. Figure 2 describes the detection process for a speed sign. A fast shape based interest operator (III-A) is applied to the input image to detect rectangles matching the shape of a speed sign. Candidates generated by the shape detector are passed on to an AdaBoost based classifier (III-B) to remove rectangular signs and objects that are not speed signs. Remaining speed signs are then passed on to the classification stage where their size and rotation is computed (III-C) to allow a correct masking of the imprinted speed limit. Once the speed limit is classified (III-D) the signs are tracked and classification results are propagated to the next frame (III-E) to improve the final classification result.

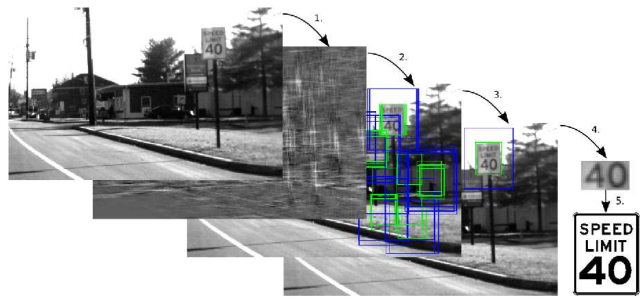


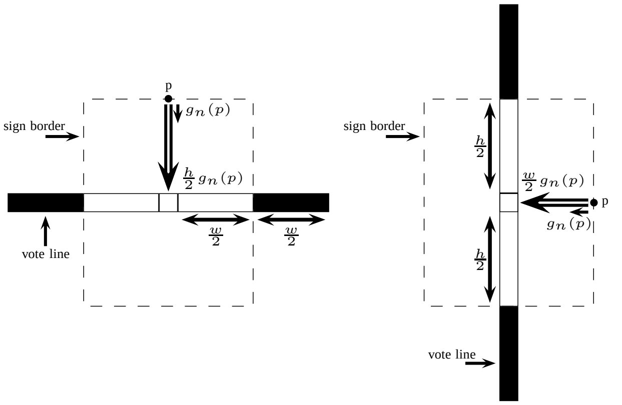
Fig. 2: Speed sign detection and classification: First rectangle structures are detected. Then the generated candidates are classified to contain a speed sign. Finally speed signs are aligned and classified.

A. Rectangle Detection

Loy and Barnes [14] have introduced an algorithm to detect the center of regular polygons. The algorithm operates on the normalized gradients of a gray scale image and uses an accumulator image that receives votes for possible centers of a regular polygon. All gradients with a magnitude above a certain threshold [2] and a certain orientation [3] will be considered in a scan-line order. Our algorithm is based on the same basic idea that a rectangular structure yields gradients with high magnitudes at its borders. U.S. speed signs are not regular but rectangular with fixed aspect ratio. By assuming an upright orientation of a speed sign the algorithm is extended to consider the orientation of a gradient. Depending on the orientation the length of the voting line is adjusted to allow a detection of U.S. speed signs. Figure 3 illustrates the voting process for gradients with horizontal or vertical orientation.

B. Speed Sign Detection

Using the rectangle detector already reduces the number of possible positions containing a traffic sign. However, it cannot distinguish between structures within the image that are rectangular and other rectangular traffic signs. Unfortunately, there are various structures in natural video scenes that can be confused with the rectangular shape of a traffic sign. Additionally, many traffic signs beside speed signs can be detected as false positives because of their rectangular shape. To assure that only the speed limit signs will be further analyzed, a classifier is needed to reject objects in the image that are no speed signs. Again this classifier must fulfill the demand to process in real-time. To address this issue, a classifier based on the method of Viola-Jones [23] is used. We will not further explain its usage but focus on the training and test results in section IV-B.



(a) (b)

Fig. 3: Voting process for one border point. Notation is largely taken from [14] and [2]. Positions receiving a negative vote are visualized by black segments and positive votes by white segments of the vote line. Length of the voting line and offset are adjusted depending on the gradient orientation.

C. Sign Alignment

The sign classification stage assumes accurately aligned sign images, as will be shown later. This constraint is in general not fulfilled, partly because traffic signs are often slightly rotated, partly because the AdaBoost detection operates on a coarse scale resolution. To this end, our architecture employs an explicit processing that estimates location and rotation parameters of the speed sign using robust voting. Subsequently, the image can be accurately normalized according to those parameters.

1) Rotation: To estimate the rotation we make use of the traffic sign's edges, one of its most distinctive characteristics. An upright (rotationally aligned) sign should yield horizontal and vertical gradients for the sign's outer edges as they usually contrast strongly with the sign's inner area, and sometimes also the background (compare Figure 1). In this respect, a rotation of the traffic sign by an angle θ translates into a rotation of its outer edges' gradients by the same θ.

In our system, θ is estimated using histogram voting. We aim to combine votes from all four main gradient directions in a rectangle (i.e., horizontal up/downwards, vertical left/right) into one single bin. Hence, the gradients are first transformed by

$$\alpha' = \alpha \bmod 90^\circ. \tag{1}$$

Here, we assume $\cdot \bmod 90^\circ$ mapping to $(-45^\circ, 45^\circ]$.

Then, the transformed gradient directions α' perform a Gaussian kernel vote with bins of width 1°. Figure 4 shows a histogram for an example image. The rotation angle $\hat{\theta}$, estimated from the histogram maximum, can then be used to rectify the sign.

Similar to the rectangle detection (III-A), computational speed and accuracy can be increased by restricting the voting to pixels that meet constraints in gradient direction α' and magnitude m, more specifically,

$$|\alpha'| \le \epsilon \tag{2}$$

$$m > t. \tag{3}$$

ϵ and t are parameters, to be determined.

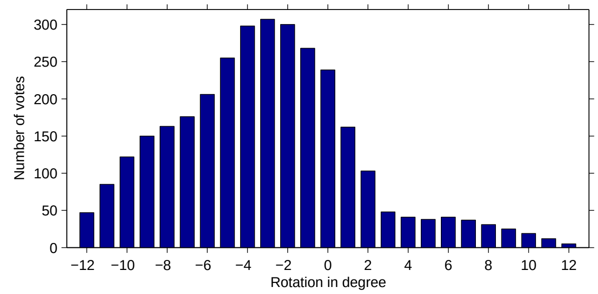
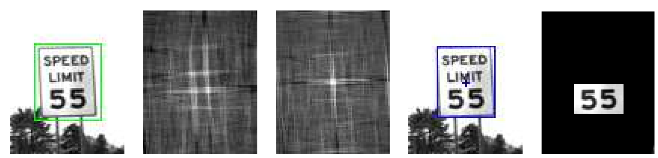


Fig. 4: Rotation estimation: Votes for the speed sign in Figure 5 (a). The estimated sign rotation is $\hat{\theta} = -3^\circ$ with the convention of clockwise rotations being positive.

2) Sign Size: The detector size from the sign detection stage provides a rough estimate of the sign size and center. More accurate values of these parameters can be obtained by re-applying the rectangle detection in a small, local neighborhood around the initial parameters, and computing the maximum in this 3D position-scale space.

Using the computed position, size and rotation the actual speed limit can be masked, rotationally rectified and passed on to the classification. Figure 5 illustrates the alignment steps.



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

Fig. 5: Alignment using the rectangle detector and estimated rotation. (a) Rough size and position estimate provided by speed sign detection. (b)-(d) Retrieval of exact sign position and size through repeated application of the rectangle detector: (b) and (c) show response maps for different rectangle sizes, (d) displays rectangle corresponding to best response. (e) Alignment data allows cropping of the rotationally rectified area of interest for classification.

D. Speed Limit Classification

At a final stage speed signs are classified into a set of given speed limits. For this, we assume a unimodal Gaussian distribution for each class' data samples, and employ a normal distribution classifier following an LDA feature transformation. This assumption is justified by the small intra-class variations of the signs and the previous accurate alignment stage. A vector $\vec{x} \in \mathbb{R}^8$, consisting of the most discriminative basis vectors of the LDA, is classified by maximizing the likelihood of a correct classification according to:

$$\hat{k} = \arg\min_k \{ d_{\Sigma_k}(\vec{x}, \vec{\mu'_k}) \}$$

where μ'_k represents the class mean in the feature space and

$$d_{\Sigma_k}(\vec{x}, \vec{y}) = \sqrt{(\vec{x} - \vec{y})^T \Sigma_k^{-1} (\vec{x} - \vec{y})}$$

is the Mahalanobis distance using the class covariance Σ_k in feature space.

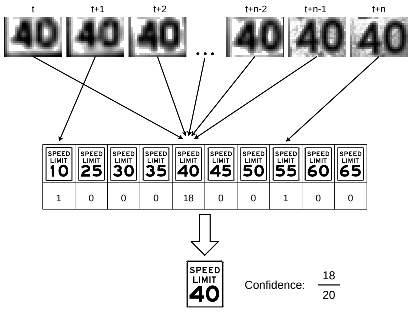


Fig. 6: Accumulated classifier decisions for a tracked speed sign using a majority voting scheme.

E. Temporal Integration

Speed signs are tracked within a temporal information propagation framework, which takes into account 2D image location and scale of the sign, the camera homography, and vehicle CAN bus data. The independent classifier decisions $Y = [y_1 \dots y_n]$ for a track of length n can be fused into a single decision, since the class membership of a track is not expected to change over n. We require the fusion to be robust to outliers, as a speed sign can be occluded or simply misaligned. Majority voting is known to fulfill this requirement. Given

$$c(k) = \frac{\sum_{i=1}^{n} [y_i = k]}{n},$$

describing the fraction of votes for each class, majority voting determines the classifier response $\hat{k}$ by

$$\hat{k} = \arg\max_k c(k).$$

Figure 6 illustrates the class assignment for a tracked speed sign. In our experiments this majority voting scheme proved to be very robust against accidental misclassifications with high confidence.

IV. Experiments and Results

To evaluate the performance of the modules 119 minutes of video, recorded on several trips between noon and 4pm were manually labeled. The videos contained a total of 152 different instances of speed limit signs in 4354 frames. Video data was acquired using a gray scale camera with a resolution of 750×400 pixels at a frame rate of 25 frames per second. Depending on the module task the labeled data is split into training and validation data to evaluate the module performance. Results obtained from the evaluation are used to adjust the system parameters. System performance is evaluated using a newly recorded test set, disjoint to the training and validation set.

A. Rectangle Detection Performance

Performance of the rectangle detector was measured using a test set consisting of 3369 image patches containing one speed sign and 12000 randomly cropped patches containing no speed sign. Speed signs are centered in the patch and scaled to the same size. To decide if a sample contains a rectangle it is transformed, as outlined in section III-A, and local maxima are located involving non-maxima suppression. If a peak received a number of votes above a certain threshold t, it is classified as containing a rectangle. Because only rectangles matching the search size will generate a substantial peak there is no further constraint on the position of the peak.

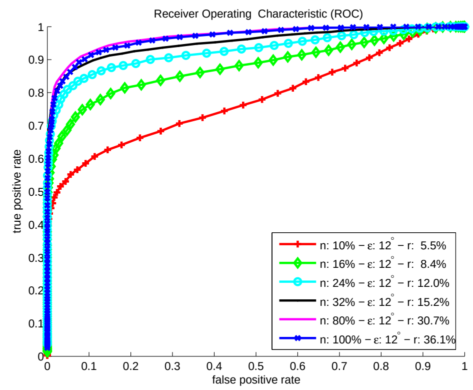


Fig. 7: Performance of the rectangle detector for a fixed angle threshold ϵ, and different percentages n of largest gradients to include in computation. Depending on the threshold the percentage r of gradients from the input image used in the computation vary.

Detection performance and speed is further improved by using only gradients fulfilling Equation (2) in the computation. A value $\epsilon = 12$ representing the maximum sign rotation due to misaligned signs and car motion was derived from the labeled data. As described in [3] restricting the used gradients not only results in a speed up but also improves the detection rate. By using only the n percent largest gradients in the voting process detection speed is further increased. Figure 7 shows the performance of the rectangle detector. By choosing a threshold of $n = 32\%$ only 15% of the images gradients are used in the computation. Minimizing the number of gradients used in the computation is crucial to allow real-time processing. Especially when searching large signs the necessary number of votes reduces the detector speed. Because this detector is only integrated as interest operator the detector threshold is adjusted for a high true positive rate of 98%. The remaining large false positive rate of 48% is handled by the sign detector.

B. Speed Sign Detection Performance

A total of five differently sized detectors have been trained, covering the range of speed signs appearing in approximately 10m up to 50m distance, given camera homography. In this respect, the smallest size is 16 and 20 pixels of sign width and height, respectively, the largest is 38 and 48. To train the detector a total of 2880 positive samples containing signs with different speed limits were used. A total of 0.7 times the number of positive examples were randomly cropped for every detector size from a set of frames containing no traffic signs. The number of negative samples was extended using boostrapping by 15% during every iteration. Figure 8 illustrates the performance of the detector on a test set that consists of 1233 positive patches and 100000 negative patches. In training rounds one to four the size of the wavelet dictionary was limited to 3000 features. By allowing AdaBoost to select from a total of 10000 features in round five the detector performance additionally increased.

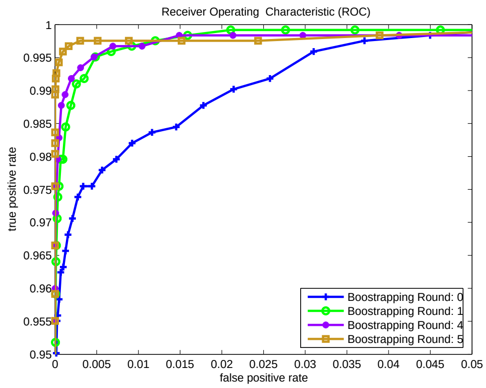


Fig. 8: Performance of the Speed Sign Detector for different bootrapping rounds. Not that the size of the wavelet dictionary was limited to 3000 in the first rounds and extended to 10000 in the last.

For an illustration of one exemplary classifier the first twelve features selected by AdaBoost are displayed in Figure 9. Note that the patches have twice the width and height of the speed sign. As can be seen, many Haar wavelets focus on the sign boundary, while some lie on the inside of the sign.

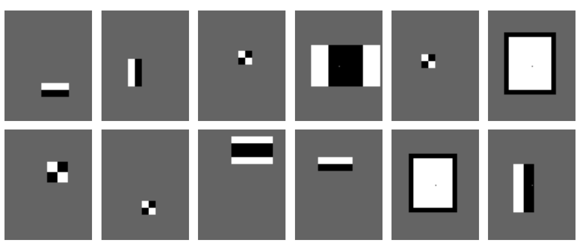


Fig. 9: Top twelve features selected by Adaboost

This classifier is integrated in the detector cascade and adjusted to have a low false positive rate (0.01% on the test set). Remaining responses are assumed to be speed signs and are therefore passed on to the following stages of alignment and speed limit classification.

C. Speed Limit Classifier Performance

The classifier was trained using 2880 aligned positive samples. Figure 10 displays the class means derived from the aligned training data. Evaluating the classification performance on the aligned test set consisting of 1233 different speed signs results in a classification error of 8.6%. Clustering of the data in the first three dimensions of the feature space is illustrated in Figure 11.

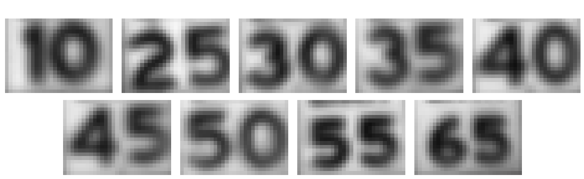


Fig. 10: Class means of the training data

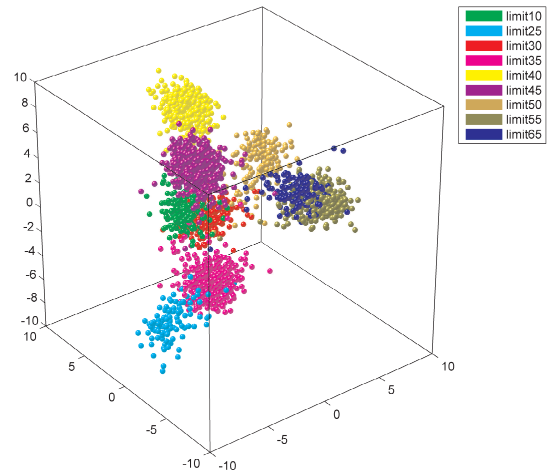


Fig. 11: Clustering of the feature vectors in the first three dimensions of the LDA-space

D. System Performance

System performance is evaluated using video sequences consisting of 16826 frames (673s) and 80 different speed sign instances. Video data was recorded between 2pm and 6pm on a route which was not used to obtain training data. Each recording was started as soon as a sign appeared in view; just perceptible for the human observer. Accordingly, signs are visible in the video sequence several seconds later. After the sign was passed the recording was stopped. In the following the system detection and classification rate are evaluated separately to gain a better understanding of their performance.

TABLE I: System test data and performance

Total length of test videos:	673s
Number of frames:	16826
Number of different speed sign instances:	80
Missed signs:	1
Detected but falsely classified:	2
False alarms:	every 42s
Detection rate:	98.75%
Classification rate:	97.5%
Recognition rate:	96.25%

1) Detection Performance: A sign will be called detected when it was tracked for more than two frames and passed on to the classification module. From all appearing speed signs in the video sequence only one was missed resulting in a detection rate of 98.75%. Figure 12 illustrates the reason for the mis-detection. Due to insufficient border gradients, in all frames in which the sign was visible the rectangle detection module rejected the sign. Detecting speed signs (without speed limit classification) was possible with an average speed of 27fps and a false alarm every 42 seconds.

2) Classification: Once the sign was detected it is classified as described in Section III-E. For a correct classification of a track the majority vote has to be correct for the last frame in which the sign was visible. Measuring the system speed including the alignment, classification and tracking modules results in an average processing speed of 11fps. Only two

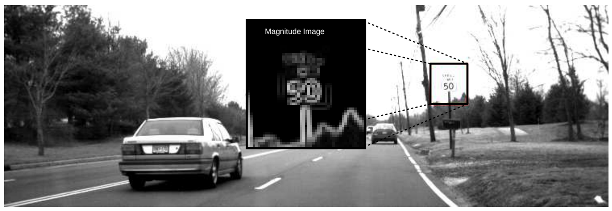


Fig. 12: Mis-detection caused by insufficient gradients at the sign border.

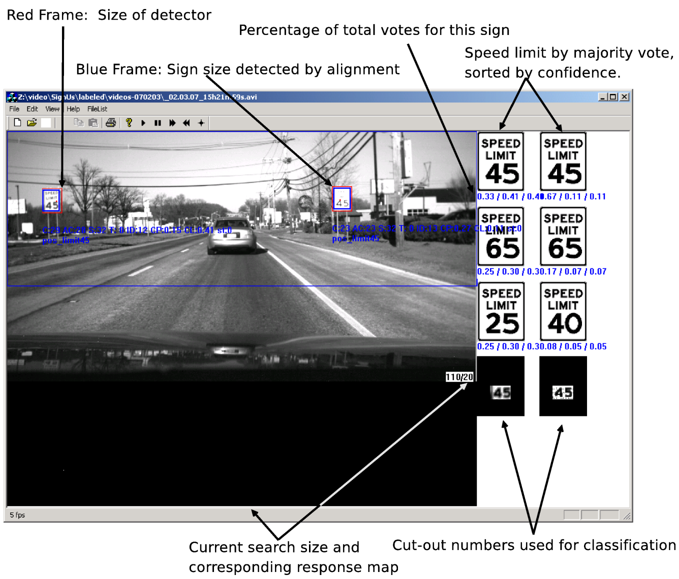


Fig. 13: Two speed limits detected, tracked, and classified on a countryside highway

tracks were mis-classified leading to a classification rate of 97.5%. A result of combining the different modules can be seen in Figure 13. Two speed signs are tracked and classified. Table I summarizes the key features of the test data and the system.

V. Conclusion

We have presented a system for detection and classification of rectangular U.S. speed signs. The system uses a mono grayscale camera with a resolution of 700x400 pixels. Video data can be processed in realtime on a standard 2.16GHz dual-core laptop. Detecting speed signs is possible at an average framerate of 27fps while detecting, tracking, and classifying speed signs requires 11fps. Overall detection rate on a sample 11-minute video with 80 speed sign instances is 98.75% with one false alarm every 42 seconds, classification rate is 97.5%, resulting in an overall recognition rate of 96.25%.

In future work, we specifically want to address a low false alarm rate. An effective, yet easy to implement way lies in utilizing the classifier confidence value for an additional rejection of non speed signs. From an application perspective, the system needs to be adapted for the task of recognizing variable U.S speed limits and other rectangular shaped signs. Those can be straightforwardly integrated by extending the classifier training samples and parameter adjustment.

References

[1] C. Bahlmann, Y. Zhu, V. Ramesh, M. Pellkofer, and T. Koehler. A system for traffic sign detection, tracking, and recognition using color, shape, and motion information. In *IEEE Intelligent Vehicles Symposium (IV)*, 2004.

[2] N. Barnes. Improved signal to noise ratio and computational speed for gradient-based detection algorithms. In *IEEE International Conference on Robotics and Automation (ICRA)*, pages 4661–4666, Apr. 2005.

[3] N. Barnes and G. Loy. Real-time regular polygonal sign detection. In *Field and Service Robotics*, volume Volume 25/2006, pages 55–66. Springer Berlin / Heidelberg, 2006.

[4] N. Barnes and A. Zelinsky. Real-time radial symmetry for speed sign detection. In *IEEE Intelligent Vehicles Symposium (IV)*, pages 566–571, June 2004.

[5] M. Betke and N. Makris. Fast object recognition in noisy images using simulated annealing. Technical report, Massachusetts Institute of Technology, Cambridge, MA, USA, 1994.

[6] Y. B. Damavandi and K. Mohammadi. Speed limit traffic sign detection and recognition. In *IEEE Cybernetics and Intelligent Systems (CIS)*, volume 2, pages 797–802, Dec. 2004.

[7] S. Estable, J. Schick, F. Stein, R. Janssen, R. Ott, W. Ritter, and Y. J. Zheng. A real-time traffic sign recognition system. In *IEEE Intelligent Vehicles Symposium*, pages 213–218, Oct. 1994.

[8] X. Gao, L. Podladchikova, D. Shaposhnikov, K. Hong, and N. Shevtsova. Recognition of traffic signs based on their colour and shape features extracted using human vision models. *Journal of Visual Communication and Image Representation*, 17(4):675–685, Aug. 2006.

[9] D. Gavrila. Traffic sign recognition revisited. In *Mustererkennung 1999, 21. DAGM-Symposium*, pages 86–93, London, UK, 1999. Springer-Verlag.

[10] R. Janssen, W. Ritter, F. Stein, and S. Ott. Hybrid approach for traffic sign recognition. In *IEEE Intelligent Vehicles Symposium (IV)*, pages 390–395, July 1993.

[11] V. Kamat and S. Ganesan. A robust hough transform technique for description of multiple linesegments in an image. In *IEEE International Conference on Image Processing (ICIP)*, volume 1, pages 216–220, Chicago, IL, USA, Oct. 1998.

[12] Kehtarnavaz, N., and A. Ahmad. Traffic sign recognition in noisy outdoor scenes. In *IEEE Intelligent Vehicles Symposium (IV)*, pages 460–465, Detroit, MI, USA, Sept. 1995.

[13] S. Lafuente-Arroyo, P. Gil-Jimenez, R. Maldonado-Bascon, F. Lopez-Ferreras, and S. Maldonado-Bascon. Traffic sign shape classification evaluation I: SVM using distance to borders. In *IEEE Intelligent Vehicles Symposium (IV)*, pages 557–562, June 2005.

[14] G. Loy and N. Barnes. Fast shape-based road sign detection for a driver assistance system. In *IEEE Intelligent Robots and Systems (IROS)*, volume 1, pages 70–75, Sept./Oct. 2004.

[15] F. Moutarde, A. Bargeton, A. Herbin, and L. Chanussot. Robust on-vehicle real-time visual detection of american and european speed limit signs, with a modular traffic signs recognition system. In *Intelligent Vehicles Symposium, 2007 IEEE*, pages 1122–1126, Istanbul, Turkey,, June 2007.

[16] G. Piccioli, E. D. Micheli, and M. Campani. A robust method for road sign detection and recognition. In *European conference on Computer vision (ECCV)*, pages 495–500, 1994.

[17] L. Priese, V. Rehrmann, R. Schian, and R. Lakmann. Traffic sign recognition based on color image evaluation. In *IEEE Intelligent Vehicles Symposium (IV)*, pages 95–100. IEEE, 1993.

[18] H. Shen and X. Tang. Generic sign board detection in images. In *ACM SIGMM international workshop on Multimedia information retrieval*, pages 144–149, New York, NY, USA, 2003. ACM Press.

[19] A. Soetedjo and K. Yamada. Fast and robust traffic sign detection. In *IEEE Systems, Man, and Cybernetics Society*, volume 2, pages 1341–1346, Oct. 2005.

[20] A. Soetedjo and K. Yamada. Traffic sign classification using ring partitioned method. *IEICE Trans. Fundam. Electron. Commun. Comput. Sci.*, E88-A(9):2419–2426, 2005.

[21] R. Strzodka, I. Ihrke, and M. Magnor. A graphics hardware implementation of the Generalized Hough Transform for fast object recognition, scale, and 3d pose detection. In *Proceedings of IEEE International Conference on Image Analysis and Processing (ICIAP)*, pages 188–193, 2003.

[22] J. Torresen, J. W. Bakke, and L. Sekanina. Efficient recognition of speed limit signs. In *Intelligent Transportation Systems, 2004. Proceedings. The 7th International IEEE Conference on*, pages 652–656, Oct. 2004.

[23] P. Viola and M. Jones. Robust real-time object detection. Technical report, Compaq Cambridge Research Laboratory, Feb. 2001. CRL 2001/1.