Proceedings of International Joint Conference on Neural Networks, Dallas, Texas, USA, August 4-9, 2013

Detection of Traffic Signs in Real-World Images: The German Traffic Sign Detection Benchmark

Sebastian Houben[1], Johannes Stallkamp[1], Jan Salmen[1], Marc Schlipsing[1], and Christian Igel[2]

***Abstract*— Real-time detection of traffic signs, the task of pinpointing a traffic sign's location in natural images, is a challenging computer vision task of high industrial relevance. Various algorithms have been proposed, and advanced driver assistance systems supporting detection and recognition of traffic signs have reached the market. Despite the many competing approaches, there is no clear consensus on what the state-of-the-art in this field is. This can be accounted to the lack of comprehensive, unbiased comparisons of those methods. We aim at closing this gap by the "German Traffic Sign Detection Benchmark" presented as a competition at IJCNN 2013 (International Joint Conference on Neural Networks). We introduce a real-world benchmark data set for traffic sign detection together with carefully chosen evaluation metrics, baseline results, and a web-interface for comparing approaches. In our evaluation, we separate sign detection from classification, but still measure the performance on relevant categories of signs to allow for benchmarking specialized solutions. The considered baseline algorithms represent some of the most popular detection approaches such as the Viola-Jones detector based on Haar features and a linear classifier relying on HOG descriptors. Further, a recently proposed problem-specific algorithm exploiting shape and color in a model-based Hough-like voting scheme is evaluated. Finally, we present the best-performing algorithms of the IJCNN competition.**

I. Introduction

Many real-world computer vision applications require accurate detection of context-relevant objects in video images. Traffic sign recognition is a challenging example, in which the algorithms have to cope with natural and thus complex dynamic environments, high accuracy demands, and real-time constraints. Therefore and because of the high industrial relevance, many approaches for traffic sign detection and recognition have been proposed. Advanced driver assistance systems featuring traffic sign recognition, usually limited to a subset of possible signs, have been deployed by the automotive industry. Against this background it is surprising that an extensive unbiased comparison of traffic sign detection systems has been missing and that no sufficiently large benchmark data sets are freely available. Therefore, we propose the *German Traffic Sign Detection Benchmark* (GTSDB). It comprises a large data set of real-world images as well as a systematic evaluation protocol, which is supported by a public web interface.

The traffic sign recognition process involves two main stages, detection of the sign in an image or video stream and the subsequent recognition (*i.e.*, classification) of the detected signs. In this study, we focus on the detection step for several reasons. First, although it is desirable in practice that both stages share computational resources (*e.g.*, operate on the same features), they can indeed be considered and evaluated independently. This provides a better understanding of the processing chain and locates the shortcomings – and thus perspectives for improvement – of a system. Second, high quality, freely available benchmarks for assessing the classification performance on traffic sign images do already exist (*e.g.*, [1]). It has been shown that state-of-the-art classification methods lead to human-competitive performance given previous optimal detection [1], [2]. One might therefore argue that the final sign classification problem can be regarded as solved.

Assessing the performance of traffic sign detection algorithms is more difficult than benchmarking the classification stage. Since many systems focus on certain categories of traffic signs such as speed limits, we propose to evaluate algorithms based on their performance on three major categories of signs. It has been shown that a website featuring online submission and evaluation is able to highly excite participation over long periods of time (*e.g.*, think of the Middlebury stereo vision benchmark [3]). Thus, we set up a web interface that allows to upload, evaluate, and rank solutions.

Both very problem-specific [4], [5], [6] as well as rather general object detection approaches [7], [8], [9] have been proposed for traffic sign detection. We implemented and evaluated a representative choice of them as baseline algorithms. From the latter category, we consider a Viola-Jones-type detector based on Haar-like features [10], linear discriminant analysis relying on HOG descriptors [11], and a color template matching approach. As traffic signs are constructed to be easily "detectable" by humans, there are well-defined cues (such as color and shape) that can be utilized in order to design powerful machine vision algorithms. Among the specialized algorithms exploiting such cues, we consider a recent method proposed in [6]. In our experimental comparison, we are especially interested in the question how the general approaches and the problem-specific algorithm compare. It is to be expected that the latter achieves better results.

In the following, we briefly review successfully established benchmarks in the domains of driver assistance systems. Then we introduce our data set in Sec. III and our evaluation procedure in Sec. IV. Section V describes the considered baseline detection algorithms and the empirical results. At the time of this document's writing, the IJCNN competition

[1] Institute for Neural Computation, University of Bochum, Germany, `firstname.lastname (at) ini.rub.de`, [2] Department of Computer Science, University of Copenhagen, Denmark, `igel (at) diku.dk`

phase has just ended. We present the results of the best-performing approaches in Sec. VI.

II. Related work

Many benchmarks in the domain of driver assistance have been widely accepted once they were made available. In most cases, the contribution was not limited to publishing novel data, but included the definition of appropriate evaluation methodologies.

Detecting humans in images is a very challenging task that is needed for various security relevant applications. Therefore, many datasets for pedestrian detection have been published (*e.g.*, [12], [13], [14], [15]). The *.enpeda..* project[1] (see also [16]) offers nine datasets for the evaluation of different computer vision techniques including stereo vision, estimation of optical flow, object detection and tracking.

Traffic sign detection is a currently well-studied and broad field of research. The survey by Møgelmose et al. [17] provides detailed analysis on the most recent developments.

Most approaches make use of two prominent features of traffic signs: color and shape. Due to diverse natural lighting conditions the treatment of color is difficult and many heuristics have been applied [6], [18]. Regarding shape, one can say that two paradigms are currently persued: model-based and Viola-Jones-like methods.

The model-based approaches rely on robust edge detection and aim at connecting them to regular polygons or circles [19], [20], usually via a Hough-like voting scheme or template matching. The Viola-Jones-like detectors compute a number of fast and robust features and try to identify trained patterns by use of different possibly weak classifiers [21].

Beyond our benchmark, there are publicly available datasets that are important to mention here: the *Summer Swedish Traffic Signs* dataset [22][2] provides the remarkable amount of 20,000 images from video sequences of which 20% have been annotated by the authors. The *MASTIF project*[3][23] has started to assemble data packages with traffic sign sequences every year since 2009 containing 1,000 to 6,000 images. The *stereopolis* database[4] (*cf*. [24]) comprises 847 images with 273 road signs from 10 different classes.

However, these datasets consist of continuous video sequences, mostly recorded on a single tour and day. Accordingly, the same traffic sign instance will repeat several times in the dataset. More important, lighting conditions and driving scenario (rural, urban, highway) have little variance. We try to adress this issue by providing single images and presenting most traffic sign instances only once[5] (cf. Sec. III).

[1] `http://www.mi.auckland.ac.nz/index.php?option=com_content&view=article&id=43`

[2] The currently available URL is `http://www.cvl.isy.liu.se/research/traffic-signs-dataset/download`

[3] `http://www.zemris.fer.hr/~ssegvic/mastif/datasets.shtml`

[4] `www.itowns.fr/benchmarking.html`

[5] There is actually one traffic sign instance that happened to turn up twice. We did not realize this before the dataset was published

III. Dataset

The images for our benchmark dataset have been selected from sequences recorded near Bochum, Germany, on several tours in spring and autumn 2010. They capture different scenarios (urban, rural, highway) during daytime and dusk featuring various weather conditions. Figure 1 shows several examples. The recorded traffic signs are normed by the Vienna Convention on Road Signs and Signals that harmonizes their appearance in 62 countries[6].

A. Data collection and format

We used a *Prosilica GC 1380CH* camera with automatic exposure control, recording *Bayer*-pattern [25] images with a resolution of 1360×1024 pixels. For the final benchmark dataset, the images were clipped to 1360×800 pixels as the lower part mainly shows the front lid and is therefore not task-relevant.

All images in the dataset were converted to *RGB* color space employing an edge-adaptive, constant-hue demosaicking method [26], [27] and were stored in raw PPM file format. All relevant traffic signs that are visible in the images were labelled manually. The ground-truth data is stored in an external CSV file and is additionally included in each PPM file comment for convenience.

B. Data organization

The traffic sign sizes vary between 16 and 128 pixels w.r.t. the longer edge. Bounding boxes are not necessarily square due to the aspect ratio of the sign types and perspective distortions. Our final dataset comprises 900 full images containing 1206 traffic signs. We split the dataset randomly into a training (600 images, 846 traffic signs) and an evaluation set (300 images, 360 traffic signs). If images contain the same real-world traffic sign, it is ensured that they are assigned to the same set. Nevertheless, most traffic sign instances occur only once in our dataset. Thus, the training set can be split further, *e.g.*, for cross-validation.

Every image is annotated with the rectangular regions of interest (ROIs) of the visible traffic signs and the specific traffic sign class (*e.g.*, stop sign, speed limit 60, speed limit 80, etc.). Although we clearly want to distinguish the task of detection and classification, we found it useful to divide the signs into three competition-relevant categories that would suit the properties of several known traffic sign detection algorithms. The categories are prohibitive signs, mandatory signs, and danger signs (*cf*. Tab. I). A minority of the annotated signs do not fall into any of those categories and are, thus, not important for the competition itself. Nevertheless, we do provide these annotations for the sake of completeness.

IV. Evaluation Procedure

The benchmark requires all participants to build and train their detector using the training set only. All learning and optimization such as classifier training, model selection,

[6] `http://www.unece.org/fileadmin/DAM/trans/conventn/signalse.pdf`

Fig. 1. Some example images from the dataset. They represent the variance in weather, lighting, and driving scenarios.

TABLE I

EACH TRAFFIC SIGN IS ASSIGNED TO ONE OF THREE CATEGORIES. THE FOURTH COLLECTION CONTAINS ALL SIGNS WHICH ARE NOT RELEVANT TO THE COMPETITION

	prohibitory signs
	danger signs
	mandatory signs
	other signs

design decisions, parameter tuning, *etc.* is performed on the training data. The evaluation data set is solely meant for performance assessment and does not contain ground-truth data. It could be used for semi-supervised [28] and transductive learning [29], but this is explicitly not in the scope of our benchmark. The final evaluation as well as the comparison with other algorithms is performed separately on the web server (*cf*. Sec. IV-B).

A. Methodology

The output of typical object detection algorithms on a given image contains a list of rectangular regions of interest. Each submitted ROI S is evaluated against every ground-truth G by applying the Jaccard similarity coefficient[7]

$$J(S,G) = \frac{|S \cap G|}{|S \cup G|} \in [0,1],$$

using a binary *loss function* J_b with a threshold (*e.g.*, $J_b = 0$ if $J < 0.6$ and $J_b = 1$ otherwise). If more than one submitted ROI intersects a ground-truth ROI with a Jaccard coefficient above the given threshold, the one with maximum value is used, the others are ignored, *i.e.*, they neither count as hit nor miss.

By choice of a category for the competing detection algorithm, the set of ground-truth traffic signs is fixed. This also means that detecting a sign from another but the relevant category is counted as false positive.

[7] also referred to as Pascal measure

B. Benchmark website

We provide a benchmark website (*cf*. Fig. 2) that allows participants to evaluate their results online: *http://benchmark.ini.rub.de*. It requires the upload of a result file and the selection of a traffic sign category.

Results are computed server-side and displayed instantly in the precision-recall plot of the particular category.

It is allowed and desired to let the participants provide several result files to build up a connected front for the performance graph. As this could allow to overfit the parameters to the evaluation dataset the number of results a participant may upload for one detection method is limited. Furthermore, the evaluation during the competition phase is performed only on a subset of the whole ground truth data. The final results computed from the complete dataset are revealed after the competition has ended.

To define a linear ranking to nominate a clear winner among the teams we use the area under the precision-recall curve (0 – 100%) as a final score.

In conclusion, the server-side evaluation combined with a given upload limit and a preliminary partial evaluation provides objective assessment of all submitted results and at the same time prevents cheating during the benchmark. However, manual annotation of all evaluation images cannot be efficiently prevented.

V. Baseline algorithms

We provide a number of baseline algorithms. These serve as example how the benchmark data is meant to be used and will incent the competitive element in the benchmark initially. We focus on the use of the training set for learning and parameterizing these algorithms.

A. Baseline methods

Three established detection algorithms are provided, namely: a Viola-Jones detector, a linear classifier based on HOG features, and a model-based approach representing several similar algorithms that have been proposed during the last years [4], [5], [20].

Vision algorithms for driver assistance systems usually need to fulfill strong real-time constraints. Hence, we draw a particular focus on real-time capability of the algorithms evaluated here. This is, however, not required in order to participate in the competition. On the contrary, we assume that the comparison of real-time and non-real-time algorithms can provide important hints for the future development of fast detectors.

1) The Viola-Jones detector: The detection approach by Viola and Jones introduced in 2001 has become one of the most popular real-time object detection frameworks [10]. Original results were presented on the face detection problem but the approach can easily be transferred to other domains.

The detector is basically a cascade of binary linear classifiers which are subsequently applied to the sliding window input. An example is passed through the cascade as long as it is positively classified by the current stage. Each stage consists of a certain number of weighted 1-dimensional threshold classifiers that are fed with a single feature.

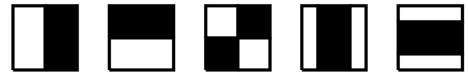

Fig. 3. Basic types of Haar wavelet features used for the Viola-Jones detector.

During the stage-wise training, initially selected detection and false-positive rates are guaranteed to be met by each cascade stage which is trained using *AdaBoost* [30]. Thus, one is able to estimate the final performance given the number of stages. The training set for stage *n* is given by all positive examples and the false-positives remaining after stage *n–1*, where those for the first stage are chosen randomly from the full images.

The real-time capability of the approach is mainly enabled by two properties: Most sliding windows are only evaluated by the first stages which contain few classifiers / features. The features offered during training are simple Haar-like filters which can be evaluated cheaply using a pre-calculated integral image of gray values. In fact, once the pre-calculation is done on the full image, responses of all basic types of Haar-like features (see Fig. 3) is computed by 5–8 additions / subtractions and a single division, independent of position and size.

During detection, the sliding window was scaled to cover all possible sign sizes. In order to achieve robustness towards intermediate sized examples, positive samples were randomly scaled within the selected range. The same was done for translation, which is introduced during training to allow for larger step sizes of the sliding window.

2) Detection based on HOG features: Histograms of orientated gradients (HOG) have been proposed for pedestrian detection yielding high performance [11]. They have also been successfully applied for other tasks including traffic sign detection and classification [9]. Based on image gradients, different histograms were calculated: first for small non-overlapping *cells* that cover the whole image and then for larger *blocks* that integrate over multiple cells. Computation of histograms involves strong normalization introducing robustness towards intensity changes. The coarse spatial sampling leads to translation invariance.

Detection was performed using a sliding-window approach at different scales employing a linear classifier. In contrast to the original authors, we trained the classifier using linear discriminant analysis [31] (LDA) instead of using a linear support vector machine (SVM). The main advantage of LDA in the given scenario is that no hyperparameter tuning is required.

In order to identify adequate negative examples, the training was performed iteratively. For the first iteration, P negative examples were collected randomly from the training set, where P is the number of positive examples. After applying LDA, we extended the training set by P negative examples that were still detected by the current intermediate

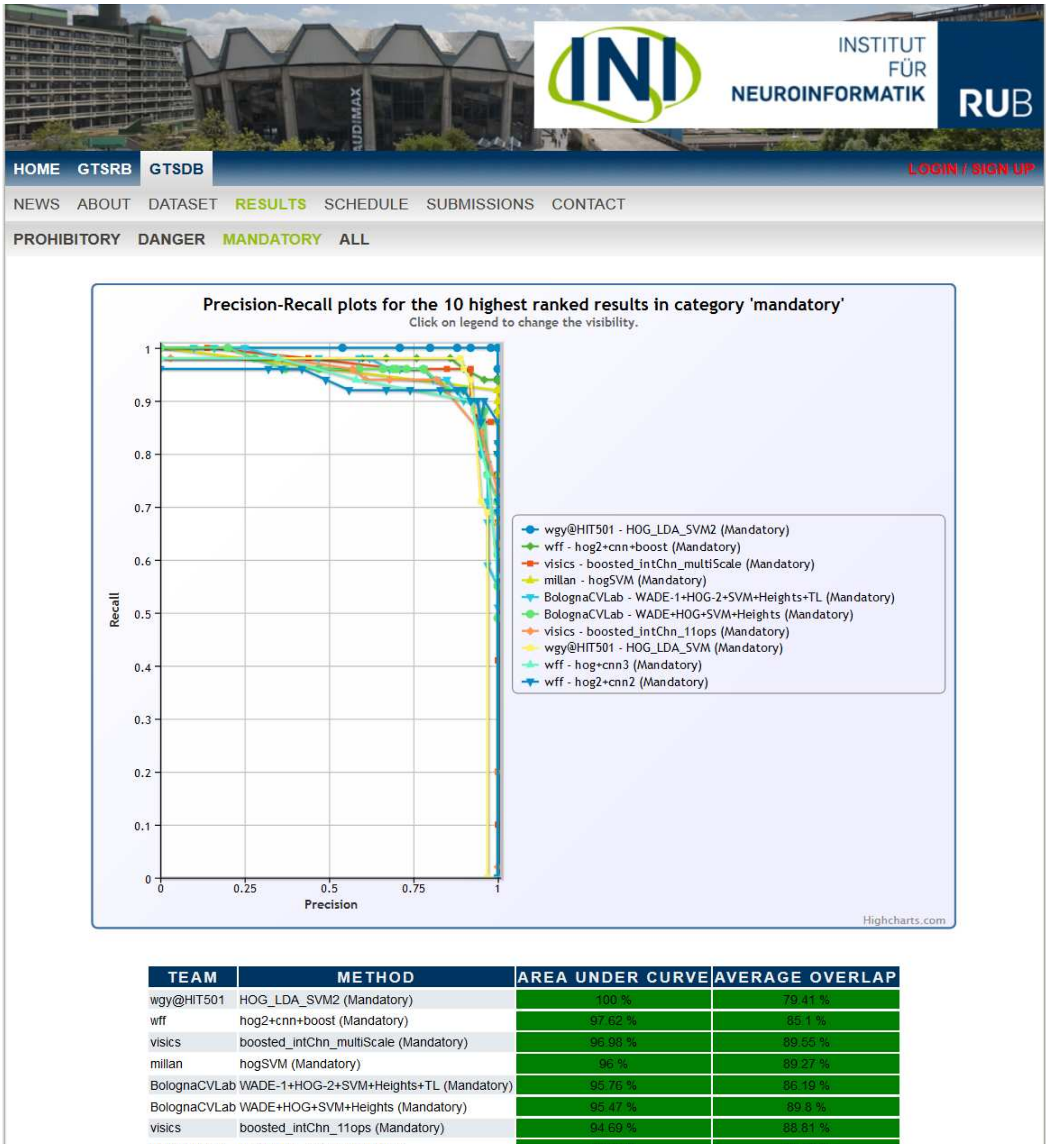

Fig. 2. The web interface during the competition: *http://benchmark.ini.rub.de*. The precision-recall curves of the submitted approaches are shown, as is the ranking by means of the area-under-curve measure.

classifier. After 10 iterations the final detector is trained (with $11 \cdot P$ examples).

3) Model-based method: Most of today's model-based methods take advantage of two salient features of traffic signs: color and shape. Color is often used to decrease the problem size and only regard image regions within the desired color range, whereas shape features are then examined to distinguish traffic signs from other similarly colored objects. We chose the scheme described in [6] to represent several methods from other publications that work on similar principles.

In brief, we applied a pixel-wise transformation to the input image resulting in a color likelihood image that yields every pixel's probability to carry a certain traffic sign color. On the transformed image, a Canny edge detector [32] was deployed and the result is thresholded for the most prominent edges. The remaining set of edge pixels was then searched for triangles and circles by an adapted version of a Hough shape detector.

Regarding the shape detection attempt, we used a Hough-like voting scheme that searches for exactly as many edge features in the expected orientation as needed to pinpoint the shape's extent and position (2 edge pixels for circles / ellipses, 3 for triangles). In conclusion, the vote space is less cluttered and it is easier to locate the maxima representing detected shapes. The intensity of an edge pixel is considered in the weight of its corresponding vote. Therefore, edges in proximity to traffic sign colors happen to induce stronger votes.

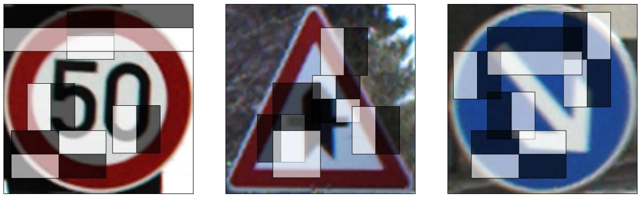

Fig. 4. Five most significant Haar features selected for the first stage of each of the 3 trained detectors.

Fig. 5. A choice of false positive results of prohibitive (top row) and danger sign detection (bottom row).

TABLE II

DETECTION RATE OF ALL ALGORITHMS AT A PRECISION OF 10 %.

Algorithm			
HOG + LDA	91.3 %	90.7 %	69.2 %
Hough-like	55.3 %	65.1 %	34.7 %
Viola-Jones	98.8 %	74.6 %	67.3 %

B. Setup

We used the 600 images of the training dataset to train all baseline approaches independently for three of the eight subsets.

The Viola-Jones detector was mainly trained as proposed by the original authors. Nevertheless, we randomly generated new negative examples after each stage in order to improve robustness. Moreover, we added a final training step where all missed examples left in the training set are determined by applying the sliding window search.

Each stage of the cascade was trained until it fulfilled a detection rate of 99% and a false positive rate of 1%, ending after 10 stages. Our Haar-feature set contains 5 basic types in 3 scales and all possible positions within the window, resulting in a pool of 5445 features. The search window was scaled by a factor of 1.25 – starting with 24×24 pixels – and translated by $1/12$ of the respective windows size.

Refer to Fig. 4 for an illustration of the most salient Haar-features. The evaluation of the Viola-Jones detector is based on our own implementation. Nevertheless, there is one freely available within the OpenCV[8] library.

For the HOG feature approach, we used the implementation provided by Dalal and Triggs.[9] We employed HOG cells of size 5×5 pixels and a block size of 2×2 cells. For the search window we considered 6×6 cells which translates to a sizes of 30×30 pixels. Detection was performed on an image pyramid with scale factors $\sqrt{2}^i$ where $i = 0, \ldots, -5$.

For the model-based approach, we found the YUV color space to yield best results in preliminary experiments (in accordance with [6]). All traffic sign examples from the training set were segmented to capture their main color (red or blue). The medians of each sign's color values are afterwards used as the centers of a mixture of Gaussians distribution that provide the pixel-wise probability for the initial image transformation.

C. Results

We assessed the proposed algorithms on the three categories of the evaluation dataset (*cf*. Sec. IV for the performance measure). In order to construct precision-recall frontlines for all methods (*cf*. Fig. 6, 5, *cf*. Tab. II), we chose varying thresholds on the classifier output (HOG), Hough vote (model-based approach), and the answer of the final classifier stage (Viola-Jones). Since the precision of the Viola-Jones method turned out to be very high we additionally truncated the classifier cascade in order to achieve a higher recall.

The detection rate of the Viola-Jones approach was highest in our pool of methods independent of the chosen category. The HOG classifier performed comparably well (*cf*. Fig. 5 for a choice of misdetections from the Viola-Jones and the HOG detector). It is also notable that the general performance dropped for mandatory (blue circular) and danger signs (red triangular). We account this to the higher variability as rotation and perspective changes have a larger effect (*e.g.*, mandatory signs are placed at differing heights and angles due to the corresponding lane). Both the model-based and the HOG method could handle this difficulty better due to the use of higher-order shape features.

In summary, a classic general-purpose detector yielded very promising results and clearly outperformed a state-of-the-art model-based approach. However, the performance on special subsets (*e.g.*, mandatory signs) is yet too low for a possible industrial application.

VI. COMPETITION ALGORITHMS AND PERFORMANCES

The competition attracted 18 teams between the middle of February and the end of March 2013 to submit over 110 results to the online evaluation system. 6 teams decided to publish their approaches in papers that were accepted for publication in the IJCNN proceedings. As a glimpse to the variety of methods we asked the three best-performing teams to summarize their work for this paper: [33], [34], [38].

A. Team LITS1

The model is divided into two modules: the ROI extraction module and the recognition module. The ROI module exploits the common properties of sign borders in each category, and considerably reduces the search space with high efficiency for further processing. It consists of three steps. The first step is the pixel-wise color classification, which transforms the color images to gray images such that the characteristic color for the traffic signs is denoted as high intensity and other colors as low intensity.

[8] http://opencv.willowgarage.com
[9] http://www.navneetdalal.com/software

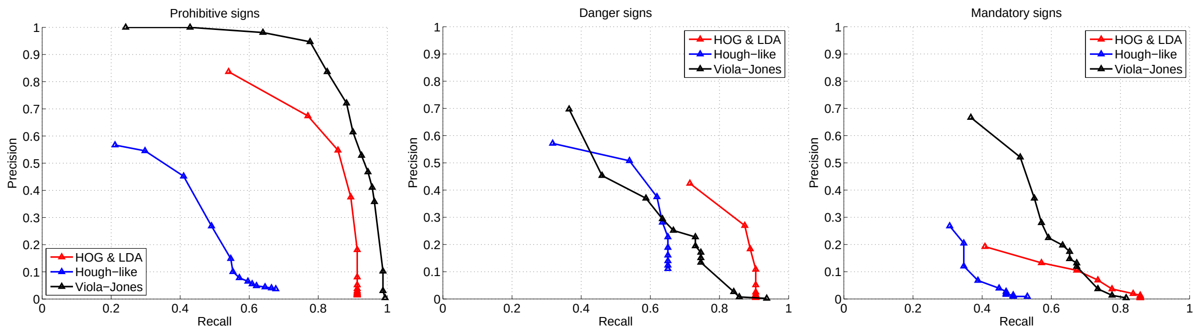

Fig. 6. Precision-Recall plots of all baseline algorithms on the chosen categories.

The second step conducts shape matching over the gray images to find the possible sign locations. The third step refines the ROIs. The recognition module performs finer validations over the ROIs and generates the final detection results. It extracts HOG (histogram of oriented gradients) and color histograms from each ROI, and concatenates them to obtain a rich and robust representation of traffic sign appearance. Support vector machines are trained on them and used to judge whether a ROI is a target sign or not.

B. *Team VISICS*

Recently, variants of the integral channel features detector have shown excellent results in terms of speed (50 fps) and quality for the task of pedestrian detection, improving over most existing methods [35], [36]. Since traffic signs are rigid objects designed to be recognizable, a detector based on a single template, such as the channel features detector, initially proposed by Dollár et al. [37], seems to fit the requirements.

Three classifiers are trained, each for one category of the GTSDB. The training time per classifier is around 45 minutes. Traffic signs are detected using a sliding window approach. Besides searching traffic signs at different scales, we also search for signs with different aspect ratios. This helps to detect slightly rotated traffic signs (around the gravity axis) by approximating the correct perspective transformation. Each detector runs at 2.5 Hz. Using techniques like approximating nearby scales, multi-scale models or even approximating nearby ratios, we expect the be able to reach a much higher detector speed as shown in [35].

C. *Team wgy@HIT501*

We present a coarse-to-fine algorithm for traffic sign detection. Firstly, it roughly finds out all candidate ROIs in a 20 $\times$ 20 sliding window, which referred to as the coarse filtering; secondly, the candidate ROIs are resized to 40 $\times$ 40 windows and further verified, which referred to as the fine filtering; finally, non-maximal suppression is performed to suppress multiple nearby ROIs. The coarse filtering is capable to find out even the smallest signs in the images, but it also outputs many false positives, which are mostly filtered out in the fine filtering.

TABLE III

COMPETITION RANKING BY AREA-UNDER-CURVE (AVERAGE OVERLAP)

Team	Prohibitive	Danger	Mandatory
wgy@HIT501	100% (91%)	99.91% (86%)	100% (79%)
visics	100% (88%)	100% (87%)	96.98% (90%)
LITS1	100% (87%)	98.85% (86%)	92% (89%)
BolognaCVLab	99.98% (85%)	98.72% (87%)	95.76% (86%)
NII-UIT	98.11% (82%)	–	86.97% (82%)
wff	–	99.78% (86%)	97.62% (85%)
milan	–	96.55% (86%)	96% (89%)
Viola-Jones	90.81% (88%)	46.26% (84%)	44.87% (88%)
HOG + LDA	70.33% (78%)	35.94% (79%)	12.01% (77%)
Hough-like	26.09% (76%	30.41% (68%)	12.86% (78%)

The features used in the two filterings are both HOG, and the classifiers used are LDA and IK-SVM respectively. The baseline is enough to give high recall and precision for prohibitory signs, while some extra steps are needed for the other two categories. For danger signs, we perform projective adjustment to the ROIs and re-classify them with HOG and SVM. For mandatory signs, we train a class-specific SVM for each class of mandatory sign, and if any of the SVMs outputs positive response for a ROI, then the ROI is determined to be a true positive. Experimental results show the proposed method give very high recalls and precisions for all the three categories, but the processing time for one image is several seconds, not enough for real-time application.

D. *Results*

Table III shows a ranking of the seven leading teams for each traffic sign category. Three teams managed to achieve perfect results in a category. The salient algorithms clearly outperform our baseline methods (as they were meant to). One can also observe that the mandatory traffic sign are harder to detect. This can be attributed to their blue color shades, that seems to be hard to distinguish in natural scenes, and the fact that they are installed near the ground which might make them prone to vandalism.

VII. CONCLUSION

Traffic sign detection is a challenging computer vision task of high industrial relevance. However, good benchmarks for traffic sign detection algorithms have been missing so far. We therefore presented a large real-world data set for evaluating such algorithms together with a reasonable performance metric, baseline results, and a web-interface for comparing approaches. We attracted 18 teams to a participate in a competition held at IJCNN 2013.

The publicly available data comprises a training set containing manual annotations as ground-truth and an evaluation set. The results on the latter were reported through a web-interface that provides immediate feedback about the own performance compared to all other participants. In our assessment scheme, we allow to measure the performance on relevant subclasses of signs to benchmark different types of traffic sign detectors.

We have implemented a number of baseline algorithms that we consider to cover the currently favoured techniques. However, they fall short on the performance of the prominent competition algorithms. Regarding the variety of driving and weather conditions these results are very promising and will hopefully take a strong influence on the development of general-purpose industrial traffic sign detection systems.

REFERENCES

[1] J. Stallkamp, M. Schlipsing, J. Salmen, and C. Igel, "Man vs. computer: Benchmarking machine learning algorithms for traffic sign recognition," *Neural Networks*, vol. 32, pp. 323–332, 2012.

[2] D. C. Ciresan, U. Meier, J. Masci, and J. Schmidhuber, "A committee of neural networks for traffic sign classification," in *Proceedings of the IEEE International Joint Conference on Neural Networks*, 2011, pp. 1918–1921.

[3] D. Scharstein and R. Szeliski, "A taxonomy and evaluation of dense two-frame stereo correspondence algorithms," *International Journal of Computer Vision*, vol. 47, pp. 7–42, 2002.

[4] A. Broggi, P. Cerri, P. Medici, P. P. Porta, and G. Ghisio, "Real time road signs recognition," in *Proceedings of the IEEE Intelligent Vehicles Symposium*, 2007, pp. 981–986.

[5] N. Barnes, A. Zelinsky, and L. Fletcher, "Real-time speed sign detection using the radial symmetry detector," *IEEE Transactions on Intelligent Transportation Systems*, vol. 9, no. 2, pp. 322–332, 2008.

[6] S. Houben, "A single target voting scheme for traffic sign detection," in *Proceedings of the IEEE Intelligent Vehicles Symposium*, 2011, pp. 124–129.

[7] C. Bahlmann, Y., V. Ramesh, M. Pellkofer, and T. Koehler, "A system for traffic sign detection, tracking, and recognition using color, shape, and motion information," in *Proceedings of the IEEE Intelligent Vehicles Symposium*, 2005, pp. 255–260.

[8] A. Ruta, Y. Li, and X. Liu, "Real-time traffic sign recognition from video by class-specific discriminative features," *Pattern Recognition*, vol. 43, pp. 416–430, 2010.

[9] F. Zaklouta and B. Stanciulescu, "Warning traffic sign recognition using a HOG-based K-d tree," in *Proceedings of the IEEE Intelligent Vehicles Symposium*, 2011, pp. 1019–1024.

[10] P. Viola and M. Jones, "Robust real-time object detection," *International Journal of Computer Vision*, vol. 57, no. 2, pp. 137–154, 2001.

[11] N. Dalal and B. Triggs, "Histograms of oriented gradients for human detection," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2005, pp. 886–893.

[12] M. Andriluka, S. Roth, and B. Schiele, "People-tracking-by-detection and people-detection-by-tracking," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2008, pp. 1–8.

[13] P. Dollár, C. Wojek, B. Schiele, and P. Perona, "Pedestrian detection: A benchmark," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2009, pp. 304–311.

[14] M. Enzweiler, A. Eigenstetter, B. Schiele, and D. M. Gavrila, "Multi-cue pedestrian classification with partial occlusion handling," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2010, pp. 990–997.

[15] C. Keller, M. Enzweiler, and D. M. Gavrila, "A new benchmark for stereo-based pedestrian detection," in *Proceedings of the IEEE Intelligent Vehicles Symposium*, 2011, pp. 691–696.

[16] T. Vaudrey, C. Rabe, R. Klette, and J. Milburn, "Differences between stereo and motion behaviour on synthetic and real-world stereo sequences," in *Proceedings of the Conference on Image and Vision Computing New Zealand*. IEEE Press, 2008, pp. 1–6.

[17] A. Møgelmose, M. Trivedi, and T. Moeslund, "Vision based traffic sign detection and analysis for intelligent driver assistance systems: Perspectives and survey," *IEEE Transactions on Intelligent Transportation Systems, Special Issue on Machine Learning for Traffic Sign Recognition*, 2012.

[18] A. Arlicot, B. Soheilian, and N. Paparoditis, "Circular road sign extraction from street level images using colour, shape and texture database maps," in *Proceedings of the Workshop on City Models, Roads and Traffic*, 2009.

[19] N. Barnes, G. Loy, D. Shaw, and A. Robles-Kelly, "Regular polygon detection," in *Proceedings of the International Conference on Computer Vision*, 2005, pp. 778–785.

[20] R. Belaroussi and J.-P. Tarel, "Angle vertex and bisector geometric model for triangular road sign detection," in *Proceedings of the IEEE Workshop on Applications of Computer Vision*, 2009, pp. 577–583.

[21] K. Brkić, A. Pinz, and S. Segvić, "Traffic sign detection as a component of an automated traffic infrastructure inventory system," in *Proceedings of the annual Workshop of the Austrian Association for Pattern Recognition*, 2009, pp. 1–12.

[22] F. Larsson and M. Felsberg, "Using Fourier Descriptors and Spatial Models for Traffic Sign Recognition," in *Proceedings of the Scandinavian Conference on Image Analysis*, ser. Lecture Notes in Computer Science, vol. 6688, Heidelberg, 2011, pp. 238–249.

[23] S. Segvić, K. Brkić, Z. Kalafatić, V. Stanisavljević, M. Ševrović, D. Budimir, and I. Dadić, "A computer vision assisted geoinformation inventory for traffic infrastructure," in *Proceedings of the IEEE International Conference on Intelligent Transportation*, 2010, pp. 66–73.

[24] R. Belaroussi, P. Foucher, J.-P. Tarel, B. Soheilian, P. Charbonnier, and N. Paparoditis, "Road sign detection in images: A case study," in *Proceedings of IEEE International Conference on Pattern Recognition*, 2010, pp. 484–488.

[25] B. E. Bayer, *U.S. Patent 3971065: Color imaging array*. Eastman Kodak Company, 1975.

[26] B. K. Gunturk, J. Glotzbach, Y. Altunbasak, R. W. Schafer, and R. M. Mersereau, "Demosaicking: Color filter array interpolation," *IEEE Signal Processing Magazine*, vol. 22, no. 1, pp. 44–54, 2005.

[27] R. Ramanath, W. E. Snyder, G. L. Bilbro, and W. A. Sander, "Demosaicking methods for bayer color arrays," *Journal of Electronic Imaging*, vol. 11, pp. 306–315, 2002.

[28] O. Chapelle, B. Schölkopf, and A. Zien, Eds., *Semi-Supervised Learning*. MIT Press, 2006.

[29] V. Vapnik, *Statistical Learning Theory*. Wiley, 1998.

[30] Y. Freund and R. Schapire, "A decision-theoretic generalization of on-line learning and an application to boosting," *Journal of Computer and System Sciences*, vol. 55, no. 1, pp. 119–139, 1997.

[31] T. Hastie, R. Tibshirani, and J. Friedman, *The Elements of Statistical Learning: Data Mining, Inference, and Prediction*. Springer-Verlag, 2001.

[32] J. Canny, "A computational approach to edge detection," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 8, no. 6, pp. 679–698, 1986.

[33] M. liang, M. Yuan, X. Hu, J. Li, and H. Liu, "Traffic sign detection by supervised learning of color and shape," in *Proceedings of IEEE International Joint Conference on Neural Networks*, 2013.

[34] M. Mathias, R. Timofte, R. Benenson, and L. V. Gool, "Traffic sign recognition - how far are we from the solution?" in *Proceedings of IEEE International Joint Conference on Neural Networks*, 2013.

[35] R. Benenson, M. Mathias, R. Timofte, and L. V. Gool, "Pedestrian detection at 100 frame per second," in *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition*, 2012.

[36] R. Benenson, M. Mathias, T. Tytelaars, and L. V. Gool, "Seeking the strongest rigid detector," in *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition*, 2013.

[37] P. Dollr, Z. Tu, P. Perona, and S. Belongie, "Integral channel features," in *Proceedings of British Machine Vision Conference*, 2012.

[38] G. Wang, G. Ren, Z. Wu, Y. Zhao, and L. Jiang, "A robust, coarse-to-fine traffic sign detection method," in *Proceedings of IEEE International Joint Conference on Neural Networks*, 2013.