[14] Russian traffic sign images dataset

V.I. Shakhuro [1], A.S. Konushin [1,2]
[1] NRU Higher School of Economics, Moscow, Russia,
[2] Lomonosov Moscow State University, Moscow, Russia

Abstract

A new public dataset of traffic sign images is presented. The dataset is intended for training and testing the algorithms of traffic sign recognition. We describe the dataset structure and guidelines for working with the dataset, comparing it with the previously published traffic sign datasets. The evaluation of modern detection and classification algorithms conducted using the proposed dataset has shown that existing methods of recognition of a wide class of traffic signs do not achieve the accuracy and completeness required for a number of applications.

Keywords: *traffic sign dataset, traffic sign classification and detection, cascade of weak classifiers, convolutional neural network.*

Citation: *Shakhuro VI, Konushin AS. Russian traffic sign images dataset. Computer Optics, 2016; 40(2): 294-300. DOI: 10.18287/2412-6179-2016-40-2-294-300.*

Introduction

Consider traffic sign recognition task. Recognition algorithm gets sequence of video frames. Algorithm should output bounding boxes and classes of traffic signs in frames. That task has several important applications: 1) Advanced Driver Assistance Systems, ADAS; 2) authomatic preparation and updating of navigational maps; 3) monitoring systems for road management service.

In this work we suppose that algorithm works on each frame independently, i.e. information from neighboring frames is not used. Then algorithm can be divided into two steps: detection and classification. On the first step all signs on image are selected with bounding boxes, on the second step all selected signs are classified.

Traffic sign has standardized appearance and size. Despite of that fact, traffic sign recognition remains a difficult task for several reasons:

1. There are a lot of traffic sign classes (156 in dataset presented in this paper); they differ with shape and images. Existing ADAS implementations detect limited number of classes (speed limit, stop, yield road, pedestrian crossing) and that simplifies the task.
2. For the tasks of road monitoring and map preparation solution should work with high recall (near to 100%) and 1 false positive per minute of video. Such number of false positives requires precision greater than 90%.
3. ADAS implementation should work in real time. Traffic signs on image may be small (from 16×16 pixels) and for detecting them multiscale image pyramid is constructed. In that pyramid signs are detected with a sliding window with a small step (for instance, 4 pixels). As the result, we obtain millions of different window positions and algorithm should evaluate windows very quickly to work in real time.

Modern images object recognition algorithms use machine learning. Quality of trained algorithms significantly depends on size and quality of the training sample. Main result of this work is a new dataset collected on Russian roads; its' comparative analysis with existing public traffic sign datasets and experimental evaluation of selected modern object recognition methods. Dataset will be useful for researching and improving quality of traffic sign recognition algorithms. Dataset is available by address http://graphics.cs.msu.ru/en/research/projects/rtsd

1. Review of recent work

1.1. Traffic sign datasets

We consider the biggest public traffic sign datasets: German (GTSDB and GTSRB), Swedish (STS), Belgian (BTSD) and American (LISA). Sample frames from these datasets are shown on fig. 1a-d.

Quantitative description of these datasets in comparison with presented RTSD dataset is shown in Table 1. Analysis of these statistics shows that existing datasets has the following features:

- small number of frames (GTSDB), which makes dataset insufficiently representative for testing traffic sign detector (for instance, some sign classes exist only in training sample);
- limited number of sign classes (STS), which makes impossible to measure generalization of algorithms on big number of sign classes;
- number of images per one sign class is small

(BTSD, LISA) and that complicates evaluation of classifiers which need big training samples (for instance, convolutional neural networks).
In the end existing datasets either not representative for detection, or for classification and don't suit for integrated training and testing traffic sign recognition system (detector and classifier).

1.2 Traffic sign recognition methods that work on a single image

Object recognition methods can be divided into two groups: heuristic algorithms and algorithms based on machine learning. Heuristic algorithms use prior knowledge about traffic signs color and shape. In [3] input image is transformed into edge map and then compared with sign pattern by Fourier transform. In [6] signs with red borders are detected. Input image is converted into HSV color space, compared with threshold and filtered from noise. Final hypotheses are obtained with generalized Hough transform. In [7] round signs are detected with color dominance channels and circular Hough transform.

Table 1. Statistics of public traffic sign datasets

	GTSRB [1]	GTSDB [2]	STS [3]	BTSD [4]	LISA [5]	RTSD
Number of frames	-	900	4000	25630	6610	179138
Number of classes	43	43	7	108	47	156
Physical signs	1728	1213	-	4565	-	15630
Images of signs	51839	1213	3488	13444	7855	104358

a) GTSDB; STS; c) BTSD; d) LISA

Fig. 1. Sample frames from traffic sign datasets

Heuristic algorithms have two main drawbacks: instability on blurred input images and complexity of construction with many sign classes of different color and shape. Now we review object recognition algorithms based on machine learning.

Methods based on cascade of weak classifiers are widely used after work of Viola and Jones [8], which solved task of real-time face detection. They used easy to compute integral features and shallow decision trees (weak classifiers) which then are combined by boosting in cascade (strong classifier). Strong classifier sequentially applies weak classifiers. After every weak classifier several windows are discarded. Full cascade is passed only by windows with objects and complex examples of background. Variants of cascade method show high quality and speed on tasks with low intra-class variability: pedestrians [9-11], traffic signs [12, 13].

Another approach is based on histograms of oriented gradients (HOG) and support vector machine (SVM). In [14] that approach show effectiveness on pedestrian detection task. Image description with HOG is shown to effective in multiclass classification tasks. In [13, 15] comparative analysis of different HOG descriptors, kernels in SVM and other classifiers on traffic sign classification task is conducted.

The last approach, deep learning, is heavily researched in several last years. It became popular after work [16] in which convolutional neural networks was successfully used for classification of ImageNet dataset in 1000 classes. In [17] ensemble of convolutional neural networks was used for traffic sign classification and it surpassed human in classification accuracy. In [18] traffic sign classifier is trained on synthetic training sample. Convolutional neural network trained on such sample shows quality similar to neural network trained on real data. Usage of synthetic training data helps to solve problem of representative training samples and rare sign classes. But experiments in [19] show that traffic sign detector trained on synthetic data obtains low quality figures. In [20] cascade of three neural networks is used for quick and quality face detection on an image. This method is also perspective for traffic sign detection task.

2. Description of Russian Traffic Sign Dataset

RTSD dataset contains frames provided by Geocenter Consulting company (http://geocenter-consulting.ru). Frames are obtained from widescreen digital video recorder which captures 5 frames per second. Frame resolution is from 1280×720 to 1920×1080. Frames are captured in different seasons (spring, autumn, winter), time of day (morning, afternoon, evening) and in different weather conditions (rain, snow, bright sun). Sample frames are shown on fig. 2a-f.

Sign labeling on frames was spent in two steps. On first step tracks of physical objects were selected on sequential frames. On the second step indistinguishable signs were discarded and every physical sign was assigned a class. Interfaces of programs used for labeling are shown on fig. 3a-b. Source code of programs for labeling tracks and classes of traffic signs are distributed with RTSD.

Algorithms were tested on several samples. Samples contain following groups of sign classes: mandatory (blue circles), danger (triangles with red border), prohibitory (circles with red border), main road (yellow rhombs), service (rectangles with blue border), special instructions (blue rectangles). Sign classes outside of these groups were not used in samples. In addition, we excluded rare sign classes, i.e. classes that have fewer than 3 physical signs or fewer than 20 images of signs. For detection sample formation we took one image per physical signs, i.e. frames were thinned such that every physical sign have only one image. Sample RTSD-D1 is similar to GTSDB sign dataset in sign groups. Sizes of samples for evaluation of detectors RTSD-D1, RTSD-D2, RTSD-D3 are shown in Table 2. We formed two samples for classification: RTSD-R1 and RTSD-R3. They have same sign classes as RTSD-D1 and RTSD-D3 and contain cropped images of all physical signs. Statistics of classification samples are shown in Table 5.

Fig. 2. Frames from RTSD with different seasons, weather and light conditions

a) object tracks labeling *b) sign classes labelling*

Fig. 3. Screenshots of programs used for labeling RTSD

Table 2. Statistics of RTSD samples and detector evaluation results

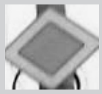
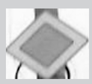
							
RTSD-D1	training (3821 frames)	1054	1594	1842			
	testing (1274 frames)	396	578	605			
Detection quality (AUC)		0.79	0.90	0.83			
RTSD-D2	training (4786 frames)	1033	1617	1848	1268		
	testing (1596 frames)	455	591	626	329		
Detection quality (AUC)		0.82	0.89	0.80	0.92		
RTSD-D3	training (9065 frames)	1164	1800	2099	1678	1235	6843
	testing (3022 frames)	501	651	684	431	474	2085
Detection quality (AUC)		0.80	0.86	0.72	0.90	0.83	0.76

Table 3. Results of detection experiment with sparse pyramid and ignoring signs with different classes when computing detection quality on sample RTSD-D2

				
Sparse pyramid	0.82	0.89	0.80	0.92
+ ignore signs width different classes	0.86	0.90	0.82	0.94

RTSD samples, as in GTSDB, use common frames for different groups of sign classes. It is possible therefore to model complete traffic sign recognition system, in which signs are selected with several detectors (one per sign class group) and then in aggregation recognized with classifier. Training and testing parts of samples were split in ratio 3:1.

3. Experimental evaluation of existing algorithms

3.1 Detector

For traffic sign detection experiments we took implementation of cascade method based on integral channel features from Piotr Dollar's toolbox [21]. Detector was trained with parameters similar to [13]: 10 channel for computing features (LUV, gradient magnitude, six gradient orientations), cascade of 400 decision trees with depth 2, which was trained in four steps with bootstrapping (up to 2000 negative examples on each step), with {50, 100, 200, 400} decision trees trained on each step. For multiscale detection of traffic signs (from 16×16 to 128×128 pixels) we construct pyramid consisting of 50 scales. For every group of sign classes we train a model of size 56×56 pixels. For improving detection accuracy every trained model is scaled and tested five time (with width/height ratio {0.8, 0.9, 1.0, 1.1, 1.2}), final detections are merged.

For computing precision and recall we use PASCAL measure for intersection rectangles. That measure is equal to ratio of intersection area to union area. Detection is correct if it intersects with some rectangle from ground truth more than 0.5. The final quality metric is area under ROC-curve, AUC. This metric is standard in object detection task and is used for evaluating detectors on GTSDB dataset [2, 13]. Detector quality numbers on samples RTSD-D1, RTSD-D2, RTSD-D3 are shown in Table 2.

Our experiments show that scaling model in 5 different ratios as in [13] improve detection quality slightly (approximately 0.005 AUC). We also note that pyramid in detector [13] is very dense, 50 scales. Experiment with 25 scales and 1 model ratio shows that quality changes slightly (about 0.01 AUC).

We conducted experiment with ignoring of other groups of signs. We fix sign classes group, train and test a detector for that group of classes. If the detector selected signs of other groups, than these detections are not included for computing quality of detection. Results of experiment show that blue circles (mandatory) are often confused with circles with red border (prohibitory). That is because some signs with red border have blue background (for instance, prohibition and restriction of parking).

Detector results on sample RTSD-D2 with sparse pyramid and ignoring of other classes are shown in Table 3.

Now we compare obtained results with requirements for application declared in introduction (nearly 100% recall and 90% precision). Only detectors trained for narrow groups of sign classes (main road and red triangles) achieve desired quality. General solution of detection task, which can be obtained as combination of detectors trained for all groups of classes, shows unsatisfactory quality.

Table 4. Convolutional neural network architecture used for sign classification

Layer	Type	Number of channels and neurons	Kernel
0	Input	3 channels with 48×48 neurons	
1	Convolutional	100 channels with 100×100 neurons	7×7
2	Max pooling	100 channels with 21×21 neurons	2×2
3	Convolutional	150 channels with 18×18 neurons	4×4
4	Max pooling	150 channels with 9×9 neurons	2×2
5	Convolutional	250 channels with 6×6 neurons	4×4
6	Max pooling	250 channels with 3×3 neurons	2×2
7	Dense	300 neurons	1×1
8	Dense	43 neurons (number of classes)	1×1

Table 5. CNN classification results on different samples from RTSD

Sample	Classification accuracy (%)
RTSD-D1 training – 4490 signs testing – 1579 signs	85.18
RTSD-R1 (66 classes) training – 25432 signs testing – 7551 signs	90.78
RTSD-D3 training – 14819 signs testing – 4826 signs	90.08
RTSD-R3 (106 classes) training – 70687 signs testing – 22967 signs	92.90

3.2 Classifier

In classification experiments we tested convolutional neural network. Architecture of neural network is introduced in [17], has 8 layers and is shown Table 4. We use Caffe [22] library for the implementation. We checked our implementation on GTSRB dataset and obtained 98% accuracy in comparison with [17], where neural network obtained 98.5% accuracy.

We evaluated neural network on samples RTSD-D1, RTSD-D3, RTSD-R1, RTSD-R3. Last two samples contain cropped sign images from samples for detector. Evaluation results are shown in Table 5. It is easy to notice that bigger training sample (RTSD-D1 compared to RTSD-R1, RTSD-D3 compared to RTSD-R1) improves classification quality. Worse quality of classifier in comparison with GTSDB dataset may be explained by lower sign images quality in RTSD. Many images are blurred a lot and captured in complex lighting conditions).

Conclusion

This work presented Russian Traffic Sign Dataset, RTSD. This dataset surpasses other public traffic sign datasets in number of frames, signs classes, physical signs and images of signs. In addition, dataset contains frames with different weather conditions, lighting and seasons. On this dataset we evaluated detector based on integral channel features and soft cascade and classifier based on convolutional neural network. Analysis of results shows that modern traffic sign detection and classification method show unsatisfactory quality for applications.

Acknowledgements

This work was supported by the Skolkovo Institute of Science and Technology, the contract №081-R, Annex A1.

References

1. Stallkamp J, Schlipsing M, Salmen J, Igel C. Man vs. computer: Benchmarking machine learning algorithms for traffic sign recognition. Neural networks 2012; 32: 323-332.

2. Houben S, Stallkamp J, Salmen J, Schlipsing M, Igel C. Detection of traffic signs in real-world images: The German Traffic Sign Detection Benchmark. The International Joint Conference Neural Networks; 2013: 1-8.

3. Larsson F, Felsberg M. Using Fourier descriptors and spatial models for traffic sign recognition. Image Analysis 2011; 238-249.

4. Timofte R, Zimmermann K, Van Gool L. Multi-view traffic sign detection, recognition, and 3d localisation. Machine Vision and Applications 2014; 25(3): 633-647.

5. Mogelmose A, Trivedi MM, Moeslund TB. Vision-based traffic sign detection and analysis for intelligent driver assistance systems: Perspectives and survey. IEEE Transactions on Intelligent Transportation Systems 2012; 13(4): 1484-1497.

6. Yakimov PYu. Preprocessing of digital images in systems of location and recognition of road signs. Computer Optics 2013; 37(3): 401-405.

7. Ruta A, Li Y, Porikli F, Watanabe S, Kage H, Sumi K. A New Approach for In-Vehicle Camera Traffic Sign Detection and Recognition. Machine Vision and Applications 2009; 509-513.

8. Viola P, Jones M. Rapid object detection using a boosted cascade of simple features. Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition; 2001: 511-518.

9. Dollar P, Appel R, Kienzle W. Crosstalk cascades for frame-rate pedestrian detection. Computer Vision (ECCV); 2012: 645-659.

10. Dollar P, Appel R, Belongie S, Perona P. Fast feature pyramids for object detection. IEEE Transactions on Pattern Analysis and Machine Intelligence 2014; 36(8):1532-1545.

11. Benenson R, Mathias M, Timofte R, Van Gool L. Pedestrian detection at 100 frames per second. IEEE Conference on Computer Vision and Pattern Recognition; 2012: 2903-2910.

12. Overett G, Tychsen-Smith L, Petersson L, Pettersson N, Andersson L. Creating robust high-throughput traffic sign detectors using centre-surround HOG statistics. Machine Vision and Applications 2014; 25(3): 713-726.

13. Mathias M, Timofte R, Benenson R, Van Gool L. Traffic sign recognition – How far are we from the solution? International Joint Conference on Neural Networks; 2013: 1-8.

14. Dalal N, Triggs B. Histograms of oriented gradients for human detection. IEEE Computer Society Conference on Computer Vision and Pattern Recognition; 2005: 886-893.

15. Lisitsyn SO, Bayda OA. Road sign recognition using support vector machines and histogram of oriented gradients. Computer Optics 2012; 36(2): 289-295.

16. Krizhevsky A, Sutskever I, Hinton GE. Imagenet classification with deep convolutional neural networks. Advances in neural information processing systems; 2012: 1097-1105.

17. Ciresan D, Meier U, Masci J, Schmidhuber J. Multi-column deep neural network for traffic sign classification. Neural Networks 2012; 32: 333-338.

18. Moiseev B, Konev A, Chigorin A, Konushin A. Evaluation of Traffic Sign Recognition Methods Trained on Synthetically Generated Data. Advanced Concepts for Intelligent Vision Systems; 2013: 576-583.

19. Chigorin A, Konushin A. A system for large-scale automatic traffic sign recognition and mapping. CMRT13 – City Models, Roads and Traffic 2013 (ISPRS Annals of the Photogrammetry, Remote Sensing and Spatial Information Sciences) 2013; 3: 13-17.

20. Li H, Lin Z, Shen X, Brandt J, Hua G. A convolutional neural network cascade for face detection. Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition; 2015: 5325-5334.

21. Dollar P. Piotr's image and video Matlab Toolbox (PMT). Source: http://vision.ucsd.edu/~pdollar/toolbox/doc/index.html.

22. Jia Y, Shelhamer E, Donahue J, Karayev S, Long J, Girshick R, Guadarrama S, Darrell T. Caffe: Convolutional architecture for fast feature embedding. Proceedings of the ACM International Conference on Multimedia; 2014: 675-678.

COMPUTER OPTICS