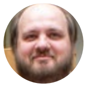
Real-time traffic sign recognition system



Real-time Traffic Sign Recognition System

Zsolt T. Kardkovács, Zsombor Paróczi, Endre Varga
U1 Research Ltd.
2 Gábor Dénes str.
INFOPARK, Budapest, Hungary H-1117
Email: {kardkovacs,paroczi,evarga}@u1research.org

Ádám Siegler, Péter Lucz
Top-Map Plc.
105–113 Bartók Béla str.
Budapest, Hungary H-1115
Email: {adam.siegler,peter.lucz}@topmap.hu

Abstract—**Traffic sign recognition (TSR) is one of the most important background research topics for enabling autonomous vehicle driving systems. Autonomous driving systems require special handling of input data: there is no time for complex transformations or sophisticated image processing techniques, they need a solid and real-time analysis of a situation. This challenge get more difficult to meetin a city like environment where multiple traffic signs, ads, parking vehicles, pedestrians, and other moving or background objects make the recognition much more difficult. While numerous solutions have been published, solutions are tested on autoways, country-side, or at a very low speed. In this paper, we give a short overview on main problems and known strategies to solve these problems, and we give a general solution to tackle real-time issues in urban traffic sign recognition.**

I. Introduction

Autonomous vehicle driving systems (AVDS) recognize potential dangers, threats, driving limitations and possibilities. One of the key factors for a successful AVDS development is to identify appropriate traffic rules valid on a certain road sector or in a junction. Such a visual recognition helps autonavigation or navigation assisting systems to be more safe, because the most of car accidents occur due to lack of concentration and failures to notice important traffic signs.

A large number of traffic sign recognition systems have been developed since the 1980's. First solutions were focusing on optical based micro-programmed hardware in order to avoid computational complexity and other contemporary mobile computing related limitations [1]. Later on, software based solutions have emerged with the first in-car integrations [2], [3]. In-car embedding required real-time image processing, nevertheless they still used parallel hadware components for accelaration and very low camera resolution and frame rate to lower data size complexity. Web cameras were getting cheaper and high resolution at the middle of 2000's which boosted traffic sign recognition research in recent years. On the other hand, that is why high precision real-time traffic sign recognition is still considered to be a hard task, because data size increment is quadratic by using high resolution cameras while computational power increases linearly according to Moore's law. Computational power limits applications even further in mobile environments.

In this paper, we propose a novel approach to tackle with real-time problems in high resolution video streams. The paper is organized as follows: Section II shows by example scenarios the most important problems to be solved for real-time traffic sign recognition systems. A detailed discussion on previous works are summarized in Section III. Section IV describes a novel model which deals with high resolution data in poor computational power environments. Finally, Section VI discusses traffic sign recognition system performance.

II. General Problems

Traffic sign recognition is about to understand vision based real life scenarios in an artificially controlled environment. While limitations help engineers to build AVDS by having an extensive knowledge on traffic situations when traffic rules are not broken, real life scenarios still differ in many ways (see Figure 1), i.e. the problem space is quasi infinite.

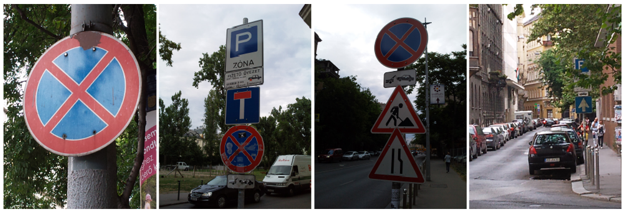


Fig. 1. Example images for ageing, vandalism, occlusion, and an occlusion with rotation, respectively, showing different lighting conditions

- One of the most discussed problem is how to detect and to compensate for the change in ambient lighting conditions, including weather changes, daylight, and vehicle turns. Cameras are optimized for human view (in)abilities, so they change displayed color representations according to lighting angle and brightness. For example, cameras add the least important color (usually blue) to capture the sense of shade, fog, rainy, or cloudy weather, or to compensate greyish patterns in images. Another trick is to reduce the red components (and increase grey) in indirect lighting conditions: human eyes still sense extensive red pixels while pictures get the impression to be more authentic. If cameras are set against incoming lighting, i.e. one drives against the Sun, cameras compensate high luminance values to white by turning all colors into grey. One must add there is almost no project on night time condition TSR [4].
- Occlusions of traffic signs due to the presence of objects such as trees, buildings, vehicles, pedestrians, or another

signs are also important factor to be considered. If part of the object is covered then successful detection and classification require recognition tasks to relax some preconditions that also increase the number of non traffic sign objects to be recognized as road signs. A general approach is to calculate and record a large number of features on image objects, however, because of the computational complexity real-time criteria can be met only at a very low speed or frame rate.

- Different in-plane and out-of-plane rotations, and the viewing angle can cause problems for recognition as both shape based recognizer, and classification methods are not robust enough for n-dimension non-affine transformations. For example, squared traffic signs are rather look like trapezes from the driver point of view, and even circles turn to be ovals according to cameras non-linear optical transformations. That is why, a non-linear, adaptive normalization is necessary in the most of the cases, however, in order to find a perfect transformation one should identify the object for reference.
- Ageing and vandalism also affect the image perception. Deterioration or intentional deformations (e.g. corrosion, paintings) result in either a loss or a misinterpretation of information. These problems can not be resolved using camera systems only. [5] deals with random degradation problems to low resolution images, however, their generative model rather fits for degradation function than real life deteriorations. [6] models deteriorations as if they would be long distance recognition problems. They stated that luminance is a good feature for distance invariant recognition which does not help locally damaged traffic signs to be identified but a retroreflective layer holds information to help the recognition process. That is why, speed limit recognition systems use auxiliary light emission and they extract these retroreflective information where they are available.
- Videos are sequences of images taken at different speeds which introduce focusing or blurring problems, and sometimes inhibit traffic signs to be captured at a reasonable or "recognizable" (e.g. at least 12px) size. For example, cameras take images in every 0.5m at a speed of 50km/h which causes traffic signs to appear at most two frames at sharp right turns. High speed camera recordings can be compensated by increasing frame rate until real-time limits are reached.
- Last but not least, color codes, shapes and pictograms used for traffic signs are different from town to town but countries. For example, "one way street" traffic signs are long, laying, blue rectangles in Germany, white rectangles in US, and squared, blue rectangles in Hungary. Stop signs and other traffic signs with textual information can be vary even further[1]. In some countries (e.g. Macedonia) traffic signs have yellow backgrounds, while others (e.g. Hungary) use white backgrounds only. As far as we know, our approach is the first to try solving these problems.

III. Architecture for Traffic Sign Recognition

Real-time AVDS must face all aformentioned problems while they solve the three basic steps of recognition (see Figure 2):

1) color segmentation or adjusment which includes video/image decoding and color space transformations
2) selection of regions of interests (ROI) where traffic signs are present in the image
3) identification of traffic signs, i.e. feature extraction techniques used with data mining classifiers.

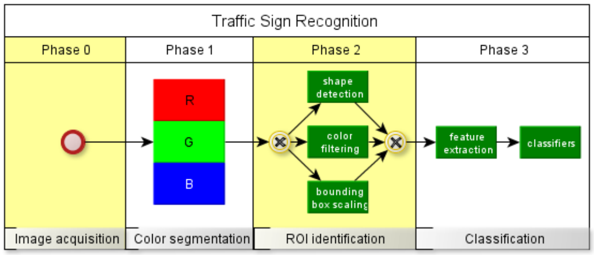


Fig. 2. General architecture for traffic sign recognition

a) Color segmentation: There are three different groups of solutions: those who use natively available color spaces like RGB[7], RGBY, YUV, or YC$_b$C$_r$, others use special color space transformations like HSL, HSI, or HSV/HSB[8] and new researches focus on more complex color spaces, e.g. CIELAB[9], CIECAM, and CIELUV which are very good at modelling human color perception.

RGB and YUV are obvious choices for traffic sign recognition, the most common video codecs natively support these color spaces, therefore no extra processing time is necessary. On the other hand, such gains in processing time is generally lost whenever colors like red, blue, or yellow are to be extracted in natural environments. This problem can be solved easily using HSL and similar color spaces at a cost of a calculation of a matrix product. There is no hard proof whether RGB or HSL color space is better suited for traffic sign recognition both have some benefits and drawbacks[10]. In the case of CIECAM, calculating CIECAM requires a considerable computation time, but the results of [9] show that this color space can improve the overall classification accuracy rate by 8% in sunny, cloudy, and rainy weathers compared to HSL.

It seems that the most important colors for traffic sign recognition, namely red, blue, yellow, black, and white, can not be easily captured in a single color space. [11], [12] noted that hue (H) component which holds color information in HSV and CIELAB color spaces depend on distance, weather conditions, and age of traffic signs. In addition, all of these color spaces focus on determining color of a single pixel, they do not deal with additive color mixing problem, i.e. when adjacent pixels

[1] See http://en.wikipedia.org/wiki/Stop_sign for examples.

(e.g. grey and orange, or black and white) are viewed as another color (e.g. red, or grey, respectively) from a certain distance.

b) Selection of regions of interests: The most common approaches are using color, shape, and patch based filtering to select regions in images which may contain traffic signs. Color filters select region candidates where a specific distribution and quantity of valid traffic sign colors are present at a valid size (see e.g. [8], [9], [12]). As we stated before, color perception is a fuzzy process. In city like environments red, blue, and white colors are too frequents, e.g. ad panels, phone boxes, biking robes are often use this color, so color filters usually identify too many interesting regions which slows down image processing. Color based filters are generally used for highways only.

Shape filters use well-known edge detection algorithms, however, they are sensitive to noises, so a Laplacian filter is often required for smoothing. Different types of Hough transformations are also popular, but since their complexity is $O(n^2)$ or higher if rotations are taken into account where n is the number of pixels. As a consequence, shape filters are not used in real-time application in their pure form [13]. Note that, decreasing n significantly improves the overall performance. One way to do that is about using color filters for preselect ROIs, and another way is use some specific points or features, and calculate transformation on these points only. The latter is called patch based filtering.

The most of patch based approaches are motivated by the work of [14] based on a cascade of boosted Haar's features. Haar's features, in this case, are linear equation on pixel values within rectangles which are classified by AdaBoost algorithm. Since these rectangles fit for only parts traffic signs the Viola-Jones approach can identify rotated and occluded traffic signs as well. Viola-Jones approach complexity is $O(fsn)$ where f is the number of features, s is the size of rectangle, and n is the number of pixels in an image, so if and only if $fs \ll n$ it is quasi linear, i.e. only it is fast for finding small objects in high quality images.

c) Identifying traffic signs: Identification usually consists of two closely related steps: a preprocessing and a classification stage. The well-known classifiers like Multi-layer Perceptrons (MLP), Radial Basis Functions, SVMs, k-nearest neighbours, decision trees are sensitive to input data discrepancies, e.g. data shifting, rotation or pin changes, occlusions or fading, scaling problems, specially if input data are 2D objects. Classifiers require input data to be normalized in any way possible that is why it is important to use feature extraction techniques to transform input data to a uniform feature space.

Some of the solutions use fast, simple, either color distribution or rule based pattern matching. While they are surprisingly robust for occlusions, they can not handle rotations or multiple traffic signs properly. Others try to use well-known data mining feature selection technique like Principal Component Analysis (PCA), Singular Value Decomposition (SVD), or different kinds of Fourier Transformations, however, these solutions are not only computationally expensive but can be applied to no occlusions, and are very sensitive to color representations. SIFT[15] and SURF[16] algorithms were published to deal with the most common object recognition problems by introducing key factors to recognize in images. Both algorithms and their derivates are very robust they even handle 20° change in view angle, nevertheless, they can be applied for small object recognition or tracking in real-time applications.

IV. Proposed System

Our system architecture block diagram is in Figure 3. We propose a generic system which can be used in both personal and mobile computer environments based on a high quality web camera video stream. Web camera outputs a YUV420 encoded MPEG4 video at 1600x1200 frame resolution with 25fps. 2 million pixels to be processed in 40ms requires all algorithms to be linear or at least quasi linear. The basic idea behind our architecture was to reduce the overall processing time by prefiltering all regions, candidate objects, and color schemes which are definitely not traffic signs. Obviously, if the problem space is small enough, even time consuming operations can applied for real time video processing.

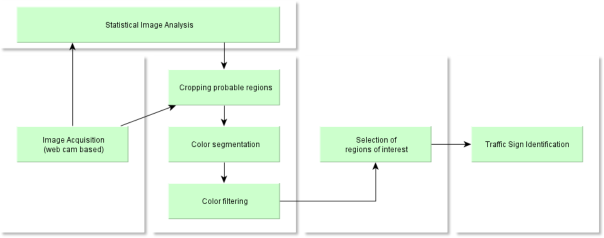

Statistical Image Analysis
Cropping probable regions
Image Acquisition (web cam based)
Color segmentation
Selection of regions of interest
Traffic Sign Identification
Color filtering

Fig. 3. General architecture for traffic sign recognition

A. Filtering Regions

The most common colors belong to the environment, e.g. at the middle on the right hand side of images called focus zone, green patches are usually trees or grass outside a town, while mass grey values are almost always buildings in a city like environment. Traffic signs are not common, nonetheless, they are much more frequent and regular than ad panels, phone boxes, or other background objects. Tail analysis indicates that the 12-22 most frequent colors are traffic sign colors. Since color values vary in weather, distance, and even in different seasons, we introduce a statistical image analysis component which examines image color distributions in specific areas, and periodically refresh interesting color values. Note that, statistical image analysis does not filter regions without any explicit knowledge what colors are in traffic signs, i.e. interesting color values are assumed to be superset of traffic sign colors. Secondly, statistical image analysis provides information for color normalization which is very useful e.g. in different weather conditions, or to decide whether we are in city or country-side.

If interesting colors of the focus zone do not appear in other regions then we eliminate non relevant areas by cropping. Crop is a fast operation that dramatically decreases the problem space, and makes the slow color segmentation even faster. We use web camera as input device optimized for human vision. As a consequence, during YUV420 encoded video color segmentation we must take into account that addition of adjacent pixel colors is perceived differently by human eyes (Figure 4). We make a color resampling for input images using an MLP neural network, and values are transformed into CIELAB color space used for color filtering. Color filtering provides output images with colors appear only in traffic signs.

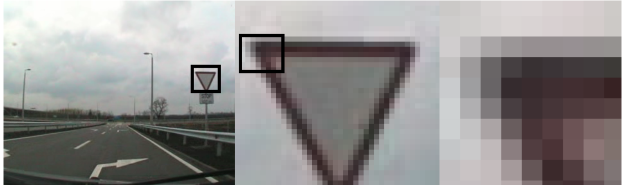


Fig. 4. Human color perception is a fuzzy process from a computer's point of view. While the first image clearly shows a red traffic sign for human eyes, one can see how color information changes by magnification.

Naturally, Hough-like transform for proper determination of bounding boxes is a good choice if precision is the only parameter to be taken into account. Although problem space is dramatically reduced in our approach still any well-known shape based filter consumes at least 15ms processing time on a dual-core mobile processor based laptop which is not acceptable for real time applications. In order to meet the real time criterion we use color density based filtering in the following way. For each pixel at i, j coordinate we introduce a chromatic density value:

$$c_{i,j} = x_{i,j} + \frac{x_{i+1,j-1} + x_{i+1,j} + x_{i+1,j+1} + x_{i,j+1}}{4} \tag{1}$$

where

$$x_{i,j} = \begin{cases} 1 & \text{iff } x_{i,j} \text{ is a traffic sign color,} \\ 0 & \text{otherwise.} \end{cases} \tag{2}$$

For each row, and column we omit all contiguous lines in each direction (top/bottom, and left/right respectively) where there are no two adjacent pixels for which $c_{i,j} > 1$. If the cropped area is not large enough, i.e. it has at most 18 pixels in some direction, then it is dropped. Note that, chromatic density value is a monotonously increasing function for continuous color points from right to left, and from bottom right to top left corner, thus it also encodes some contours of traffic signs. Using diagonal run length values, there is an easy way to disjoin occluded or adjacent traffic signs in the cropped area.

B. Solutions for Recognition

Traffic signs are different for different regions and countries. Color codes and pictograms used in some localities are too different, so it seems rather obvious to treat a wide range of colors as traffic sign colors in order to avoid country specific recognition. In addition, traffic signs may have alternative appearance using e.g. alternative color for background or other color combinations fit more to local visibility conditions. Note that, recognition is also dependent on ROI selection algorithms, so a robust recognition method should identify traffic signs based on fragments of images as well.

State of the art research indicates that binary Haar-like features extracted for each chroma channel are ideal to deal with the latter problem. Since the number of traffic sign colors is more than colors used in a single country, one must add features increased by the number of country specific chroma channels. Our tests indicate that a duplication of these features is a must, however, the most of the Haar-like features formulae are the same for different chroma channels. We introduce a color insensitive Haar-wavelets in the following way:

- There is defined a $\mathbf{C} = \{c_1, c_2, \ldots, c_n\}$ set of valid color codes where c_i stands for traffic sign colors.
- Let $\mathcal{T}$ be a set of recognizable traffic signs by the system, and $m : \mathcal{T} \to 2^{2^{C}}$ be a function which gets valid templates for each traffic signs.
- Let $f : H \to 2^{\mathcal{T}}$ be a predefined function that maps Haar-features in images to set of candidate traffic signs contain this Haar-feature. For example, if a red rectangle of a width of w and height h is above of a white rectangle of the same size than $f(H) = \{\text{Give-a-way}\}$. If a blue rectangle is followed by a green on then $f(H) = \emptyset$.

The recognition algorithm is based of f and m functions. For every Haar-features H_i, H_j, H_k at pixels i, j, k respectively a geometric verification is made by using

$$\{m(T) | T : f(H_i) \cap f(H_j) \cap f(H_k)\}$$

templates for which

$$f(H_i) \cap f(H_j) \cap f(H_k) \neq \emptyset$$

and $\arg\max(d(i,j), d(j,k), d(i,k)) \geq t$ where t is a predefined threshold. If there are multiple candidates in the same region then a neural network based voting mechanism chooses between candidate traffic signs based on the number of evidences, and Haar-features.

For each traffic sign candidate a new parallel thread is initiated to recognize its identity. Note that, f function can be easily replaced by AdaBoost which rather works on Haar-wavelet inputs than Haar-like features. For AdaBoost, the function m is implicitly encoded in its states after learned during model building phase. Using AdaBoost also gives us a considerable gain in processing time as Haar-features are calculated once: the overall complexity is $O(nT)$ where n is the number of pixels (Haar-features) to be processed.

V. Experiments and Results

We performed an extensive benchmarking on the above described system based on 10 hours of traffic sign video material:

- taken in urban (66%) and country-side (33%) environments including highway and freeway traffics,
- 2 hours of 388x260 and 8 hours of 800x600 and 2 hours of 1600x1200 frame resolution at 25fps,
- at all four seasons (spring, summer, autumn, winter), and all types of weather conditions,

TABLE I
EXPERIMENT SETUPS' RESULTS

Experiment	TPR	FPR	MR	No. of traffic signs
1^{st} setup	78.4%	12.6%	3%	254
2^{nd} setup	69.7%	12.5%	6.2%	3366
3^{rd} setup	68.1%	16.0%	6.1%	311

- in five countries: Hungary (82.5%), Macedonia (7.5%), Austria (5%), Germany (5%),
- using 6 cars with 10 different camera setups (heights and view angles).

In the first experimental setup we evaluated the overall system. We created 2x10 minutes videos by editing the video database where traffic sign appearance is frequent. Using these videos we built up the traffic sign template database by labeling frames. Each template is extracted using 1-7 distinct training samples, not counting cross-frame appearances. The test data contained a 1 hour labeled video that did not appear in training set. Experimental results are summarized in Table I.

In the second experimental setup an evaluation was made on the whole 10 hours video database using the same training data. Finally, in the third experimental setup we changed our model using information taken from the second setup, and we field tested our solution producing new videos on traffic routes not known by the proposed algorithm. All three setups were processed real-time, i.e. in the first two setups we achieved 35fps processing speed on videos. After the modifications, and changing running environment to a considerable lower performance computer we had 18 frames processed in each second which was compensated by tracking algorithms, i.e. if we recognize a traffic sign, we do not attempt to re-recognize it again. We tested our solution on mobile phones as well, we have got 1.4fps processing time.

VI. CONCLUDING REMARKS

We have describe a new, real-time traffic sign detection, and recognition system. The system integrates color, shape, and motion information. It is built on three main components, a well-known color acquisition framework, an accelerated Hough-like transform based ROI identification, and a country independent recognition module. Our results indicate a lower accuracy to those published in the literature, nevertheless, they are direct consequence of the more thorough field testing in multiple countries, weather conditions, seasons, and other environmental setups.

REFERENCES

[1] M. Lalonde and Y. Li, "Road sign recognition – survey of the state of the art," Centre de recherche informatique du Montreal, Tech. Rep. CRIM-IIT-95/09-35, 1995.

[2] S. Estable, J. Schick, F. Stein, R. Janssen, R. Ott, W. Ritter, and Y.-J. Zheng, "A real-time traffic sign recognition system," in *Proc. IEEE Intelligent Vehicles '94 Symposium*, 1994, pp. 213–218.

[3] V. Rehrmann, R. Lakmann, and L. Priese, "A parallel system for realtime traffic sign recognition," in *International Workshop on Advanced Parallel Processing Technologies '95 (APPT)*, 1995, pp. 72–78.

[4] 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 *Proc. IEEE Intelligent Vehicles 2005 Symposium*, 2005, pp. 255–260.

[5] H. Ishida, T. Takahashi, I. Ide, Y. Mekada, and H. Murase, "Identification of degraded traffic sign symbols by a generative learning method," in *Proc. 18th Int. Conf. Pattern Recognition (ICPR 2006)*, vol. 1, 2006, pp. 531–534.

[6] P. Siegmann, R. J. López-Sastre, P. Gil-Jiménez, S. Lafuente-Arroyo, and S. Maldonado-Bascón, "Fundaments in luminance and retroreflectivity measurements of vertical traffic signs using a color digital camera," *IEEE Transactions on Instrumentation and Measurement*, vol. 57, no. 3, pp. 607–615, 2008.

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

[8] A. de la Escalera, J. M. Armignol, and M. Mata, "Traffic sign recognition and analysis for intelligent vehicles," *Image and Vision Computing*, vol. 21, no. 3, pp. 247–258.

[9] X. Gao, K. Hong, P. Passmore, L. Podladchikova, and D. Shaposhnikov, "Colour vision model-based approach for segmentation of traffic signs," *EURASIP Journal on Image and Video Processing*, vol. 2008, pp. 1–7.

[10] U. L. Jau, C. S. Teh, and G. W. Ng, "A comparison of rgb and hsi color segmentation in real-time video images: A preliminary study on road sign detection," in *Proc. Int. Symp. Information Technology ITSim 2008*, vol. 4, 2008, pp. 1–6.

[11] Y. Aoyagi and T. Asakura, "A study on traffic sign recognition in scene image using genetic algorithms and neural networks," in *Proc. IEEE IECON 22^{nd} Int. Industrial Electronics, Control, and Instrumentation Conf.*, vol. 3, 1996, pp. 1838–1843.

[12] T. Warsop and S. Singh, "Distance-invariant sign detection in high-definition video," in *Proc. IEEE 9^{th} Int. Cybernetic Intelligent Systems (CIS) Conference*, 2010, pp. 1–6.

[13] G. Piccioli, E. de Micheli, P. Parodi, and M. Campani, "Robust method for road sign detection and recognition," *Image and Vision Computing*, vol. 14, no. 3, pp. 209–223.

[14] P. Viola and M. J. Jones, "Robust real-time face detection," *International Journal of Computer Vision*, vol. 57, no. 2, pp. 137–154, 2004.

[15] D. G. Lowe, "Distinctive image features from scale-invariant keypoints," *International Journame of Computer Vision*, vol. 60, no. 2, pp. 91–110, 2004.

[16] H. Bay, T. Tuytelaars, and L. van Gool, "Surf: Speeded up robust features," *Computer Vision and Image Understanding*, vol. 110, no. 3, pp. 346–359, 2008.