PAPER ID: NITET14

AUTOMATIC RECOGNITION OF TRAFFIC SIGNS USING FANN AND OPENCV

ANIL KANNUR
Department of Computer Science & Engineeering, A.G.Patil Institute of Technology, Solapur
archanapatil708@gmail.com

ARCHANA S. PATIL
Department of Computer Science & Engineeering, A.G.Patil Institute of Technology, Solapur
archanapatil708@gmail.com

NAYANA N. ARADHYE
Department of Computer Science & Engineeering, A.G.Patil Institute of Technology, Solapur
archanapatil708@gmail.com

SUPRIYA S. MALGE
Department of Computer Science & Engineeering, A.G.Patil Institute of Technology, Solapur

ABSTRACT
Automation Recognition of Traffic Signs is integrated and automation software for Traffic Symbol Recognition. The proposed system detects candidate regions as Maximally Stable Extremely Region (MSERs), which offers robustness to variations in lighting conditions. Recognition is based on Artificial Neural Network (ANN) classifiers. The training data are generated from real footage road signs which will be fetched using camera board and by applying threshold values we get proper training data for each frame. By applying thinning mechanism like erode and corrode and segmentation we can recognize proper shape and symbol. The proposed system is accurate at high vehicle speeds, operates under a range of weather conditions, runs at an average speed of 10 frames per second, and recognizes all classes of ideogram-based (non-text) traffic symbols from real footage road signs. Comprehensive comparative results to illustrate the performance of the system are presented.
KEYWORDS: Detection, Recognition, Segmentation, FANN, OpenCV, Traffic Symbol Analysis, Thinning-erode and corrode

1. INTRODUCTION

Automatic traffic sign detection and recognition is an important part of an advanced driver assistance system. Traffic symbols have several distinguishing features that may be used for their detection and identification. They are designed in specific colors and shapes, with the text or symbol in high contrast to the background. Because traffic signs are generally oriented upright and facing the camera, the amount of rotational and geometric distortion is limited. Information about traffic symbols, such as shape and color, can be used to place traffic symbols into specific groups; however, there are several factors that can hinder effective detection and recognition of traffic signs. These factors include variations in perspective, variations in illumination, occlusion of signs and deterioration of signs. Road scenes are also generally much cluttered and contain many strong geometric shapes that could easily be misclassified as road signs. Accuracy is a key consideration, because even onemisclassified or undetected sign could have an adverse impact on the driver.

The proposed method consists of the following two stages:
1) Detection 2) Recognition

The most common approach, quite sensibly, consists of two main stages: Detection and recognition. The detection stage identifies the regions of interest and is mostly performed using color segmentation, followed by some form of shape recognition. Detected candidates are then either identified or rejected during the recognition stage.

1) Detection is performed using a novel application ofmaximally stable extremely regions (MSERs) for this purpose color space and flood fill algorithm of segmentation is used. Before MSERs system applies thinning mechanism with erode and corrode methods for thinning captured frame.

2) Recognition is performed with the help of artificial neural network (ANN). By applying threshold value and frames per second system generate proper training data which is then passed to ANN classifier, it recognizes the symbol type and feeds it as audio notification. At the time of recognition each frame builds neural network, and training data builds train network. Frames neural network and train network is passed to ANN classifier, then it will classify images and from that we get different symbol type. In addition with ANN classifier output image segments are also used to classify different symbol type.

In this paper firstly we outline the methodology used, which includes detection, recognition and the generation of synthetic data then we describe comparative results to illustrate the performance of the system and finally conclusion has drawn.

2. LITERATURE SURVEY

The aim of the system is mainly linked to drivers in safety critical situations rather than to replace them. However in recent years many research advances have been done in this field. In the literature the control of autonomous vehicles is separated in lateral and longitudinal, for this region in the arbitration and control work package of DESERVE project both controllers will be considered [7].

Paper describes howsigns are detected using a set of Haar wavelet features obtained from Ada-Boost training. Classification is performed using Bayesian generative modeling [8].

Another paper says recognition is based on a cascade of support vector machine (SVM) classifiers that were trained using histogram of oriented gradient (HOG) features [1].

3. PROBLEM STATEMENT

The proposed system consists of the following two main stages: detection and recognition. The complete set of road signs used in our training data and recognized by the system is shown in Fig.1.

Fig.1.Sample of road signs used in training the proposedsystem

The traffic symbols (samples shown in fig.1.) from the image are given as input to the system then system finds MSERs from MSERs system genetrates bounding box. Considering those bounding boxes system finds shapes and classifies accurate traffic symbol. Finaly system notifies that symbol to driver with audio.

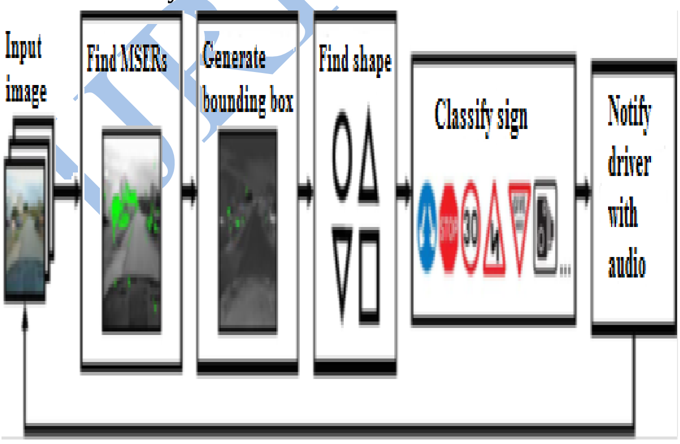

Fig.2. Block diagram of proposed system

PROCESS STAGES

- Frame extraction
- Segmentation
- Training
- Evaluation

4. METHODOLOGY

In this system traffic symbols are detected as MSERs, MSERs are regions that maintain their shape when the image is thresholded at several levels. System extracts frames from video, after applying segmentation on frames bounding box will be created for MSERs. With the help of training data system will get best cases for symbols.

4.1. TRAINING MODE

In this system, training video clipis fed as input. From this video system extracts frames. Performing preprocessing activities and segmentation system generates MSERs with bounding box. By extracting segments and features of frame, system trains neural network.

4.1.1. FANN

FANN - Fast Artificial Neural Network Library is written in ANSI C. The library implements multilayer feed forward ANNs, up to 150 times faster than other libraries. An ANN is normally run in two different modes, a training mode and an execution mode. Although it is possible to do this in the same program, using different programs is recommended.There are several reasons to why it is usually a good idea to write the training and execution in two different programs, but the most obvious is the fact that a typical ANN system is only trained once, while it is executed many times.

ADVANCED USAGE OF FANN

- Adjusting Parameters
- Network Design
- Understanding the Error Value
- Training and Testing
- Avoid Over-Fitting
- Adjusting Parameters DuringTraining

4.1.2. OPENCV

OpenCV (Open Source Computer vision) is a library of programming function mainly aimed at real time computer vision. Originally developed by Intel's research center in Nizhny Novgorod (Russia), it was later supported by Willow Garage and is now maintained by Itseez. The library is cross-platform and free for use under the open source BSD license.

OpenCV is written in C++ and its primary interface is in C++, but it retains a less comprehensive though extensive older C interface. There are bindings in Python, Java and MATLAB/OCTAVE. Wrappers in other languages such as C#, Perl, Ch, Haskell and Ruby have been developed to encourage adoption by a wider audience.

OpenCV runs on variety of platforms. Desktop: Windows, Linux, macOS, FreeBSD, NetBSD, OpenBSD; Mobile: Android, iOS, Maemo, BlackBerry 10. OpenCV uses CMake

4.1.3. SEGMENTATION

In computer vision, image segmentation is the process of partitioning a digital image into multiple segments (set of pixels, also known as super-pixels). The goal of segmentation is to simplify and/or change the representation of an image into something that is more meaningful and easier to analyze. Image segmentation is typically used to locate objects and boundaries (lines, curves, etc.) in images. More precisely, image segmentation is the process of assigning a label to every pixel in an image such that pixels with the same label share certain characteristics. Each of the pixels in a region are similar with respect to some characteristic or computed property, such as color, intensity, texture.

WHY SEGMENTATION?

Segmentation is typically the first step in object identification in an image. It may also be used in compression to compress different areas, segments of an image, at different compression qualities. It also used to remove unwanted objects from frame.

SEGMENTATION ALGORITHMS

FLOODFILL

Flood fill, is also called as Seed fill, is an algorithm that determines the area connected to given node in a multidimensional array. It is used in the "bucket" fill tool of paint programs to fill connected, similarly -colored areas with a different colors. When applied on an image to fill a particular bounded area with color, it is also known as "boundary" fill.

COLOR SPACE

A color space is a useful method for users to understand the color capabilities of a particular digital device or file. It represents what a camera can see, a monitor can display or a printer can print, and etc. There are varieties of color spaces, such as RGB, CMY, HSV, HIS.

RGB COLOR SPACE

RGB (R=Red, G=Green, B=Blue) is a kind of color space which uses red, green and blue to elaborate color model. An RGB color space can be simply interpreted as "all possible colors" which can be made from three colors for red, green and blue. In such conception, each pixel of an image is assigned a range of 0 to 255 intensity values of RGB components.

$$\Omega_{RGB} = \max\left\{ \frac{R}{R+G+B}, \frac{G}{R+G+B}, \frac{B}{R+G+B} \right\} \quad \text{....... Eq 1}$$

4.1.4. THINNING

Thinning is a morphological operation that is used to remove selected foreground pixels from binary images, sum what like erosion or opening. It can be used for several applications, but is particularly useful for skeletonization. In this mode it is commonly used to tidy up the output of the age detectors by reducing all lines to single pixel thickness. Thinning is normally only applied to binary images, and produces another binary image as output. There are two common morphology operators: Dilation and Erosion. For this 'Erode' and 'Dilate' OpenCV functions are used.

DILATE FUNCTION

Dilation consist of convoluting an image with some kernel(), which can have any shape or size, usually a square or circle. The kernel has a defined anchor point, usually being the center of the kernel. As the kernel is scanned over the image, system compute the maximal pixel value overlapped by and replace the image pixel in the anchor point position with that maximal value. This maximizing operation causes bright regions within an image to "grow" (therefore the name dilation). The background (bright) dilates around the black regions of the image.

ERODE FUNCTION

This operation is the sister of dilation. What this does is to compute a local minimum over the area of kernel. As the kernel scanned over the image, we compute the minimal pixel value overlapped by and replace the image pixel under the anchor point with that minimal value. Erosion (usually represented by $\ominus$) is one of two fundamental operations (the other being dilation) in morphological image processing from which all other morphological operations are based. It was originally defined for binary images, later being extended to grayscale images

ALGORITHM OF TRAINING MODE

Input: Video
Output: Trained data

Step 1:
Extract frames from running video as four frames per second

Step 2:
Perform preprocessing operations such as thinning on extracted frames

Step 3:
After preprocessing, segmentation operation is done on frames for next processing

Step 4:
To generate training data system extracts segments which was processed earlier

Step 5:
Here system completely generate trained data

Step 6:
System will extract necessary features from those segments or say trained data

Step 7:
With the help of trained data and extracted features system will train neural network

4.2EXECUTION MODE

At the time of execution system extracts frames from video after that system will perform segmentation algorithms on frames. System uses same segmentation algorithms for training as well as for execution. On applying segmentation on frames system evaluates different symbols and notifies userwith audio.

"Raspberry Pi" hardware is used for final execution. Raspbian Jessie operating system is installed on raspberry pi for processing. Raspbian is a Debian-based computer operating system for Raspberry Pi, it uses PIXEL, Pi Improved Xwindows Environment, Lightweight as its main desktop environment as of the latest update.

Fig.3. Road sign image acquisition for detection and recognition system

ALGORITHM OF EXECUTION MODE

Input: Video

Output: Audio notification

Step 1:
Extract frames from fetched video

Step 2:
Segmentation operation is done on frames

Step 3:
With the help of trained data neural network builds

Step 4:
By comparing segmented frames with trained data FANN classifier classify images

Step 5:
After that segment filtration is done on frames

Step 6:
Finallysystem gives audio notification to the user

In execution mode, segmentation, building neural network, FANN classification and filtration are important steps. Same segmentation algorithms are used for execution, which was used for training mode. Neural network is build using training data and trained network. FANN classifier classifies images with the help of neural network and segmented images. In filtration two filtering techniques are used viz. threshold filter and erode filter.

5. RESULTS AND DISCUSSION

The proposed system can operate at a range of vehicle speeds and was tested under a variety of lighting and weather conditions. A considerable increase in speed was gained by implementing the algorithm in parallel as a pipeline to around20 frames per second, running on 2.30-GHz Intel Core i3central processing unit under OpenCV. However, the system retained a latencyof around 200 ms.

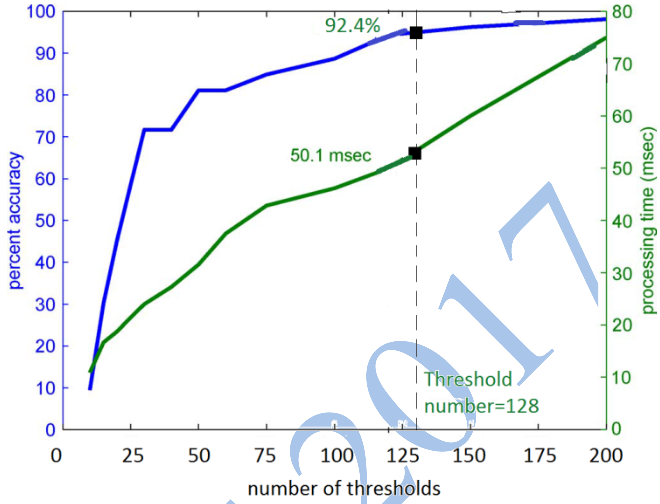

Fig.4.Chart that shows the number of thresholds used for MSER plottedagainst accuracy of detection and processing time.

The classifier that was trained on realdata gave an accuracy of 85.9%.To show that the features learned by the classifier relateonly to the road signs and not to background information,the classifier was also tested using a data set that comprises real data generated images, but with different backgroundsfrom that in the training set. The accuracy achieved for thisexperiment was 92.4%, which verified the claim.

To more thoroughly validate the system, another classifierwas trained, with a data set that contains real images andsynthetically generated interpolations, created using randomlydistorted version of the real images. The total number of imagesin this data set was 43 509. This classifier had an overallaccuracy of 89.2%, which was greater than either the fullysynthetic or the fully real data set.

6. FUTURE ENHANCEMENT

We can improve this system by adding extra night vision using local contrast enhancement and some other factors. With the help of multiple disciplinary people we can increase the performance rate of system. As per this system instead of only notifying driver we can control the vehicle using this system like to slow down vehicle when there is sign like 30 speed limit.

CONCLUSION

We have proposed a real-time system for the automatic detection and recognition of traffic symbols. Candidate regions are detected as MSERs. This detection method is significantly insensitive to variations in illumination and lighting conditions. To get MSERs major factor in this system is segmentation. Traffic symbols are recognized using FANN classifiers. A method for the synthetic generation of training data has been proposed, which allows large data sets to be generated from real footage images. Our system can identify signs from the whole range of ideographic traffic symbols, which form the basis of our training data. The system retains a high accuracy at a variety of vehicle speeds.

ACKNOWLEGMENT

We are extremely grateful to our guide Prof. Anil Kannur, CSE department, AGPIT, Solapur for his support and guidance and would also like to thank all the staff members of CSE department, AGPIT.

REFERENCES

1) IEEE transactions on intelligent transportation systems, vol. 13, no. 4, December 2012Jack Greenhalgh and MajidMirmehdi, Senior Member, IEEE
2) "The OpenCV Reference manual Release 2.4.13.1", September 22, 2016
3) http://en.wikipedia.org/Raspberry pi
4) http://www.stackoverflow.com/Question-Answer
5) "FANN User Guide" Steffen Nissen 2003-11-07
6) "OpenCV 2.4 Cheat Sheet (C++)"
7) P. Paclik and J. Novovicová, "Road sign classification without color information"in Proc. ASIC, Lommel, Belgium, 2000.
8) 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. IVS, 2005, pp. 255–260.