Available online at www.sciencedirect.com

ScienceDirect

Procedia Computer Science 165 (2019) 252–258

Procedia Computer Science

www.elsevier.com/locate/procedia

INTERNATIONAL CONFERENCE ON RECENT TRENDS IN ADVANCED COMPUTING 2019, ICRTAC 2019

A Deep Neural Network Framework for Road Side Analysis and Lane Detection

Utkarsh Shukla[a], Ayush Mishra[b], Graceline Jasmine S[c], Vaidehi V[d], Subramaniam Ganesan[e]

[a]*ATLAN, Delhi - 110030*
[b]*DailyCutting, Bangalore - 560044*
[c]*Vellore Institute of Techonology, Chennai - 600127*
[d]*Mother Teresa Women's University, Kodaikanal, Tamil Nadu - 624101*
[e]*Oakland University Rochester, USA*

Abstract

This paper presents a computer vision based framework with the aim of aiding the task of driving. The framework serves the purpose of road analysis. Road analysis is further divided into two sub-tasks. The first task aims at recognition of the different road signs, the second task aims at lane analysis. The task of automatic driving requires humans to multitask and perform many operations in split seconds. The framework is introduced to aid this task of driving if not completely automate it while keeping in mind of using it with a simple hardware and software setup. The effectiveness of the framework lies in its feature of having minimal complexity which enables it to be used in real-time. The results of the pipeline are quantified by first measuring its accuracy in the classification of road signs, second measuring its ability to gather the information about the road (lane analysis and 2 vehicle detection) thirdly by performing the time bench-marking.

Keywords: Computer Vision, Image Processing, Vehicle Detection, Convolution Neural Network,Transfer Learning

1. Introduction

Driving is an amalgamation of different complex tasks and requires un-deviated attention from the driver. Long stretches of driving can become really exhaustive and might cause a lack of attentiveness may lead to accidents. Due to the recent advancement in image processing and deep learning techniques frameworks can be developed with the

[1] *Corresponding Author Tel: +91-790-505-7363
utkarshshukla2912@gmail.com

10.1016/j.procs.2020.01.081

aim to aid in the process of driving. The figure 1 describes a framework capable of road signs recognition and analysis involving lane and vehicle detection. The road sign recognition consists of a deep learning architecture while road analysis is achieved by a combination of different image processing pipelines. For training our model for recognition of road signs we incorporate a huge variety of traffic signs used by different road work departments. This is done to achieve a generalized recognition model. Training models on such a huge variety of data cause a reduction in the accuracy, so the deep learning architecture is build to encounter a variety of traffic signs like performed in [1] [2].The next task aims at the detection of different vehicles that are in motion on the road and analyze the lane of the moving vehicle.

2. Literature Survey

Considerable amount of work has been done towards driver assisting technologies. Some recent work include a framework with the aim of detection, tracking, and recognition of different road signs. For detection of the road sign a hear-cascade based system coupled with adaboost was used [3]. The classification of the detected road signs was performed using a Bayesian classifier The main focus of this paper is a joint modeling of colour and shape within the AdaBoost framework. In the paper *Robust method for road sign detection and recognition* a three fold framework is proposed. The first part aims at detection of road signs by leanings from some prior information of the scenario or colour. The second part focuses on geometric analysis of the detected edges in the first step, which generates candidates to be circular and triangular signs. The last aspect of the framework is a recognition phase that performs validation using cross-correlation methods. [4] The post processing part of the framework consists of time based integration based on Kalman filtering. In [5] Like [3] focus is heavily put on using features based on colour and shape of the traffic signs for detection and identification of the road signs. They used Support vector machines as a classifier for identification of the different road signs. There recognition process can be divided into three parts. First discrimination according to the color of the pixel second traffic-sign detection by shape classification using support vector machines; and the third context understanding based on Gaussian-kernel SVMs. The quoted results puts light on high success rate and a very low amount of false positives in the final recognition stage. Wen-Jia Kuo and Chien-Chung Lin in [6] proposed a two step framework for detection and recognition of the road signs. The first step is a detection step which uses Hough transformation, corner detection, and projection to estimate the location of the road sign in the image under noisy and complicated environment. The recognition task uses tree based approach where convolution, (RBF) deep neural network and K-dtree are implimented to recognise the road signs in two stages.

In [7] a new traffic sign detection system was proposed that simultaneously estimates the location and precise boundary of traffic signs using convolutional neural network (CNN). It solves the problem of most of the method that only provide bounding boxes of traffic signs as output, and hence requires processes such as contour estimation or image segmentation to obtain the precise boundary of signs. In [8] Songwen Pei1, Fuwu Tang1, Yanfei Ji1, Jing Fan1 and Zhong Ning proposed the use of Multiscale Deconvolution Networks to solve the problem of the huge amount of time taken for preprocessing the images and applying complicated algorithms for improving and finding blurred and subpixel images of the signs. Multi-Scale Deconvolution Networks (MDN), smoothly conflates multi-scale conv nets with deconvolution network, resulting in an efficient and robust localized traffic sign recognition model training. In [9] is presented a road signs recognition and classification system focused on a three-step algorithm consisting of color segmentation, shape identification, and a deep neural network architecture. The ultimate aim of the algorithm is to recognise and distinguish varied road signs present along Italian roads. The system is proposed to achieve real time application. The shape detection is achieved by using two different model of pattern matching and the other using edge detection and geometrical cues. Radu Timofte Karel Zimmermann Luc Van Gool in [10] proposed a multiview traffic sign detection which uses a multidimentional algorithm to augment results beyond the state-of-the-art. The focus was to shift the detection method which still focuses on single view detection. A speedup in the process is achieved through a novel bounded evaluation of ensemble AdaBoost detectors. The 2D detection in multiple views are combined to estimte a 3D hypotheses.

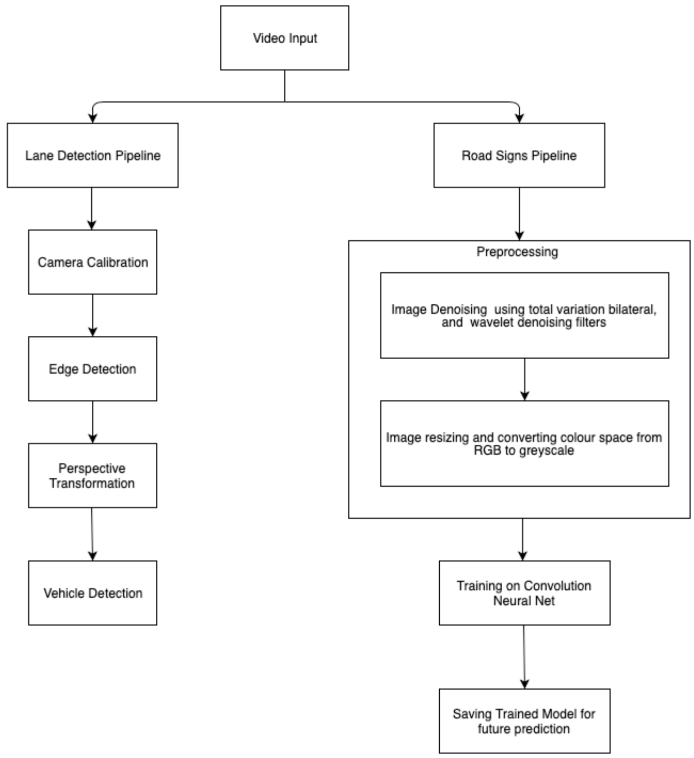


Figure 1: Framework capable of road signs recognition and analysis involving lane and vehicle detection

3. Data Descriptions

For the purpose of detection of signals from the roadside, the model is trained over standard images of different Traffic signals. The dataset used is the German Traffic Sign Benchmark used in the paper . The dataset has the following features

- The dataset contains in total 43 different categories of road sign which are all used for recognition
- More than 50,000 images in total, depicting ground-truth data.
- Physical traffic sign instances are unique within the dataset (each real-world traffic sign only occurs once).

For road analysis, KITTI Vision benchmark dataset is used. The dataset was collected by equipping a standard station wagon with two high-resolution color and gray scale video cameras. Dataset was collectied by driving around the mid-size city of Karlsruhe, in rural areas and on highways.

4. Proposed Method

The framework has been divided into two objectives first of recognizing road signs, the second part is the analysis of the road. For meeting the first objective we dive into Deep Learning approach of Convolution Neural Network. We first compute our results over different architectures that are popularly used and then finalized over using transfer learning for this purpose. The concept of transfer learning is been implemented in the domain of deep learning where models trained for a task is re-purposed on a second related task. We use ResNet50 architecture present on Keras as our base model for final implementation purpose. We customize this model by hyper-parameters optimization and adding custom layers to use it with our use-case. The model displayed is just the customized architecture attached to the pre-trained Resnet model. The second objective has to first deal with the main task of road analysis as it plays a vital role in roadside vehicle detection while traveling. This process involves two sub-processes of:

- Analysing different lanes of roads and identifying the current lane so it can be used for determining future maneuver
- Tracking the vehicles and objects around the car.

At first, the focus is on finding the track on which our vehicle is running or the track which is vacant that is having no vehicles in front. Then we aim at detecting the lanes in a video frame with cars present on them.

Layer (type)	Output Shape	Param #
batch_normalization_1 (Batch	(None, 50, 50, 1)	4
conv2d_1 (Conv2D)	(None, 48, 48, 16)	160
max_pooling2d_1 (MaxPooling2	(None, 24, 24, 16)	0
batch_normalization_2 (Batch	(None, 24, 24, 16)	64
dropout_1 (Dropout)	(None, 24, 24, 16)	0
conv2d_2 (Conv2D)	(None, 22, 22, 16)	2320
max_pooling2d_2 (MaxPooling2	(None, 11, 11, 16)	0
batch_normalization_3 (Batch	(None, 11, 11, 16)	64
dropout_2 (Dropout)	(None, 11, 11, 16)	0
conv2d_3 (Conv2D)	(None, 9, 9, 16)	2320
max_pooling2d_3 (MaxPooling2	(None, 4, 4, 16)	0
batch_normalization_4 (Batch	(None, 4, 4, 16)	64
dropout_3 (Dropout)	(None, 4, 4, 16)	0
global_average_pooling2d_1 (	(None, 16)	0
dense_1 (Dense)	(None, 43)	731

Total params: 5,727
Trainable params: 5,629
Non-trainable params: 98

Figure 2: Convolution Neural Network Architecture

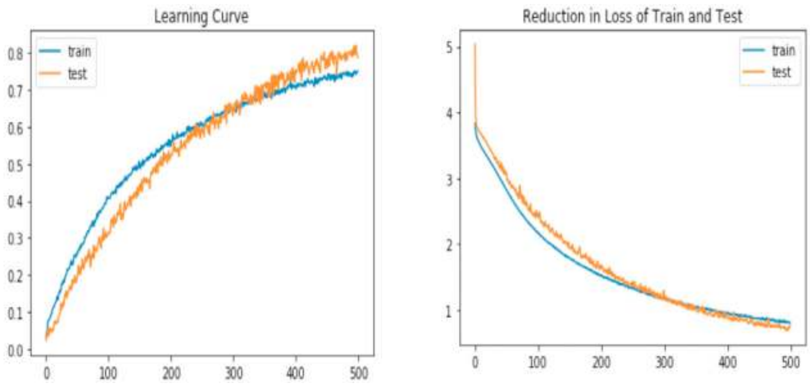
5. Experimental Setup

This section gives a detailed information about the experimental setup involved in the

5.1. Road Sign Detection

5.1.1. Data Prepossessing

Image Data before it can be used by the neural network are pre-processed. The efficiency of a deep learning models depends vastly on the kind of the data used which in turn determines the effectiveness of feature representation leaned by the model. First of all the images is cleaned by denoising the image samples using the total variation filter combined by wavelet denoising and bilateral filters. Denoising images result in *posterized* images with flat domains separated by sharp edges. Degree of posterization is controlled by considering the trade off between denoising and faithfulness to the original image. The images are then converted into grey-scale which is then resized into [50 X 50] pixel values. Even though the image size is reduced it contains good representation of the data, the resizing is necessary as lower the size less computationally expense also by converting RGB representation of the images to grey scale reduces the depth of image which again reduces the expense.



(a) Learning Curve (b) Loss Curve

Figure 3: Model Training Report

5.1.2. Model Construction

The neural network is a derivative of the Resnet50, initialized with the pertained weights from Image-Net. The last fully connected layer is removed and combine it with three convolution block which is defined by a combination of single convolution layer, batch normalization max pooling and then a dropout layer as shown in figure 4. The figure 2 shows the proposed architecture and the arrangement of the convolution blocks with the aim of learning the proper representation of the training images. Batch normalization reduces the amount by which the hidden unit values shift during training. Maxpooling layer is simply defined as trans-versing of a moving window on a 2D input space, where the max magnitude within that window is the output. It reduces the dimension of the 2D array space, the reduction is dependent on the size of the window that we choose to convolve over the 2D input matrix. Dropout is a technique used to reduce over-fit on neural networks. While training a certain amount of neurons on a particular layers are deactivated. This is done to introduce generalization because it forces the layer to learn with different neurons.

5.1.3. Training and Testing of the Model

The dataset is been divided into training testing and validation with the ratio of 70:20:10 respectively. Validation data is used to alter the hyperparameters used by the convolution network. While our final results are cited over the testing data. Each split of the data of training, testing and validation have equal representation of the different classes this is done by a custom splitter. The model is trained for 500 epochs with a batch size of 128 images. The metric for which the neural network optimizes the weights is *categorical accuracy* which checks if the index of the maximal true value is equal to the index of the maximal predicted value. The optimizer used is *Adagrad* with the default parameters provided with Keras it vriest its weight based on the variation of the model parameter, The learning rate changes based on how frequently the weights gets changed during training. More the variation and frequency of updates of a parameter, the smaller the updates by the optimizer. In Figure 3 we show the learning curve and the error rate curve for our model. The blue line denotes the performance while training while the orange shows for the validation set, from the graph it's clearly evident that the model doesn't overfit on the data and the learning is pretty generalized.

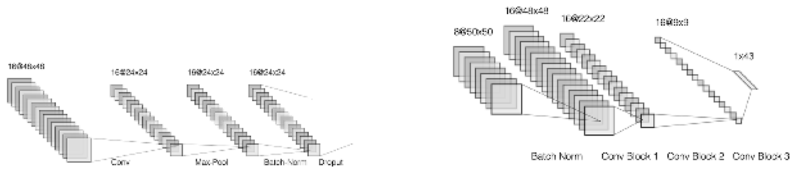
5.2. Lane and Vehicle Detection

5.2.1. Camera Calibration

The feed from the cameras mounted over the vehicle is passed to the second pipeline where geometrical analysis computes the parameters of an image sensor and lenses of the video camera. The parameters are then used to estimate lens distortion, calibrate the object size, or narrow down the location of the camera in the scene.

5.2.2. Edge Detection

Canny Edge detection is the most frequently used edge detection algorithm where we determine the intensity gradient of the Image and also determine the magnitude and direction of the gradient, a compete analysis of an data is



(a) Convolution Block (b) Combined Convolution Block

Figure 4: Neural Architecture with Convolution Block

performed to eliminate any noisy pixels which does not constitute the edge. This is used to highlight the edges of the different lane of roads

5.2.3. Perspective Transformation

Is used to perform varied geometrical transformations on two dimensional space. The aim is to prevent any alteration of the image content but to ac hive deformation of the pixel grid and map this deformed grid to the deformed image. An approximate road curvature equation is defined below which A and B are constants and y is a variable

$$F(y) = Ay^2 + By + C. \tag{1}$$

The above equation is for finding the radius of curvature from any arbitrary point on road. The equation for finding the radius of curvature used is.

$$R_{curve} = \frac{(1 + (\frac{d_x}{d_y})^2)^{\frac{3}{2}}}{2A} \tag{2}$$

This helps in determining the angle at which the car has to turn and at what distance. Therefore if there is no turn then the radius of curvature will have a very large value or if having a very steep turn, this value will be very small. The equations given above helps in determining the turn in the road and the distance of vehicle from that turn.

5.2.4. Vehicles Detection

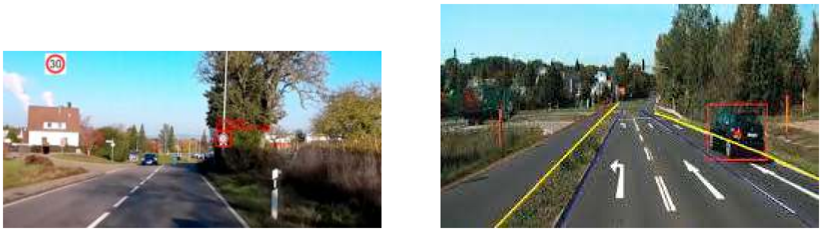
For this purpose we use (HOG) which is a descriptor used in image processing in order to achieve object localization. The algorithm enumerates occurrences of gradient orientation in localized portions of an image. It is a scale Invariant feature. The algorithm is provided with a huge set of training data which contains images of vehicles and non vehicles. The idea behind doing this is to determine a set of feature that distinctively represents the training sample and can be used by machine learning algorithms for distinguishing the two classes. As a feature set we combine HOG features with Spatial Binning features and Histograms of Color features. These extracted features are then passed to a SVM with a RBF kernel for training so that it can now distinguish between vehicles and non-vehicle frames. This process is a subset of the lane identification task as it helps the algorithm to get aware of the vehicles surroundings.

6. Results and Conclusions

After training the proposed deep learning architecture for road sign detection for 500 epochs as shown in 3 we get an accuracy of 98.21% with a precision of 93.94%. The lane detection workflow was able to detect lanes, objects and vehicles and in both still images and moving video frame figure 5 with total time computation of 64 frames per second with a complexity of

$$O(n^2) \tag{3}$$

The model after it has been trained is predicts a batch of 128 images in 3secs. The model size ranges from 56.8 MB to 57.6 MB based on the amount of data used to train it. Since the size of the model is pretty less it is easily deployable



(a) Road Sign Detection

(b) Lane Detection

Figure 5: Results

and the lane detection uses only matrix operations which are heavily optimized to perform on low level hardware. If GPUs can be used on the vehicles algorithms like MAGMA can hugely reduce the time and computational expenses for lane and object detection.

7. Future Work

The future of this work consists of adding modularity to the implementation of the different objectives, setting of better hardware compatible coding standards. Also we have only proposed the algorithm for multi-lane roads which suggests the driver to be on the same lane. There are many roads which are single lanes and the traffic is from both the sides on the same lane, specially in India, therefore an improvement on single lane roads is required. This task may involve various new complexities. Also we can extend this work to bad weather conditions where its hard to predict the lane while driving.

References

[1] B. Huval, T. Wang, S. Tandon, J. Kiske, W. Song, J. Pazhayampallil, M. Andriluka, P. Rajpurkar, T. Migimatsu, R. Cheng-Yue, *et al.*, “An empirical evaluation of deep learning on highway driving,” *arXiv preprint arXiv:1504.01716*, 2015.

[2] C. Chen, A. Seff, A. Kornhauser, and J. Xiao, “Deepdriving: Learning affordance for direct perception in autonomous driving,” in *Proceedings of the IEEE International Conference on Computer Vision*, pp. 2722–2730, 2015.

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

[4] 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, 1996.

[5] S. Maldonado-Bascón, S. Lafuente-Arroyo, P. Gil-Jimenez, H. Gómez-Moreno, and F. López-Ferreras, “Road-sign detection and recognition based on support vector machines,” *IEEE transactions on intelligent transportation systems*, vol. 8, no. 2, pp. 264–278, 2007.

[6] W.-J. Kuo and C.-C. Lin, “Two-stage road sign detection and recognition,” in *2007 IEEE international conference on multimedia and expo*, pp. 1427–1430, IEEE, 2007.

[7] H. S. Lee and K. Kim, “Simultaneous traffic sign detection and boundary estimation using convolutional neural network,” *IEEE Transactions on Intelligent Transportation Systems*, vol. 19, no. 5, pp. 1652–1663, 2018.

[8] S. Pei, F. Tang, Y. Ji, J. Fan, and Z. Ning, “Localized traffic sign detection with multi-scale deconvolution networks,” in *2018 IEEE 42nd Annual Computer Software and Applications Conference (COMPSAC)*, vol. 1, pp. 355–360, IEEE, 2018.

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

[10] R. Timofte, K. Zimmermann, and L. Van Gool, “Multi-view traffic sign detection, recognition, and 3d localisation,” *Machine vision and applications*, vol. 25, no. 3, pp. 633–647, 2014.