Traffic Sign Integrity Analysis Using Deep Learning

Joshua Paolo N. Acilo, Allyana Grace S. Dela Cruz, Michael Kevin L. Kaw, Maynald D. Mabanta, Vemeni Grace G. Pineda, Edison A. Roxas
Department of Electronics Engineering
Faculty of Engineering, University of Santo Tomas
España, Manila, Philippines
joshuaacilo.13@gmail.com, delacruzallyanagrace@gmail.com, michaelkaw123@gmail.com, maynaldmabanta@gmail.com, pinedavem@gmail.com, earoxas@ust.edu.ph

***Abstract*—Standardization and harmonization of high-quantity and low-cost roadway assets such as traffic signs make a compelling case now, especially in the matter of road safety and the advent of autonomous transportation vehicles. As for the standardization and degradation problem in classifying traffic signs, no automatic approach has been developed yet. This paper presents two trained models that can classify the compliance of a traffic sign to a prescribed design standard in accordance to the Vienna Convention on Road Signs and Signals adapted by the Philippines, and its physical degradation status. Due to the limited size of image dataset and a weak computing resource, a deep learning approach using transfer learning was utilized. The process highlights the use of a state-of-the-art pre-trained convolutional neural network - the ResNet-50 which was trained on a subset of the ImageNet database. This network was then fine-tuned on the curated dataset of traffic signs derived on Google Street View with varying design and physical condition. The experiments conducted exhibit test accuracy of 95.70% in the compliance model and 96.20% in the degradation model which shows a great potential in developing a streamlined traffic sign assessment platform.**

Keywords—Traffic Signs, Google Street View, Deep Learning, Convolutional Neural Networks, Transfer Learning, ResNet-50

I. INTRODUCTION

Traffic signs provide important guidance to traffic regulation, varying between warnings, road condition information, traffic regulation and destination information [1]. In this study's context, the word integrity refers to both the design compliance status and degradation of a traffic sign.

According to the United Nations Economic Commission for Europe, "Recognizing that international uniformity of road signs, signals and symbols and of road markings is necessary in order to facilitate international road traffic and to increase road safety". The standardization and strong harmonization of traffic signs is considered as the basis for the improvement of road safety [2]. It has been concluded by some studies that 5,000 lives a year could be saved if damaged traffic signs are occasionally updated [3,4].

The Department of Public Works and Highways (DPWH) incorporated the "Vienna Convention on Road Signs and Signals" and formed the "Highway Safety Design Standard (HSDS) Manual" which serves to establish and maintain a standardized system of signs on all roads of the Philippines [5].

To date, most of the systems involving traffic signs highlight traffic sign detection and recognition which develops intelligent transportation systems and autonomous vehicle driving systems (AVDS). These systems normally consist of two phases – detection and recognition and vary based on distinct approaches upon the properties of traffic signs in color, shape, and inner content [6].

In the traffic sign recognition (TSR) phase, the system identifies the category and content of the signs by using different algorithms. Compared to sign detection where simple image processing is involved, sign classification is a multi-class classification problem that involves machine learning. In recent years, classification (on a limited basis) has become commercially available. However, these systems do not benefit from Google Street View images due to system's need to perform in real-time and high frame rate methods. Currently, emerging sensing technologies such as computer vision, etc. and a wide range of databases such as Google Street have advanced greatly. Consequently, our study aims to develop an automated system which has the ability to classify the degradation and compliance of a traffic sign to the Vienna, which highlights the importance of standardization of traffic signs.

The automated system adapts a type of machine learning called deep learning in which a model learns to perform classification tasks directly from images. It is implemented using a neural network architecture that employs an iterative learning process in which vectors are introduced to the network one at a time, and the weights associated to the input values are adjusted each time [7]. Training a deep neural network from scratch and fine-tuning a pre-trained model to classify images are the two approaches for deep learning. The former produces the most accurate results, but it requires training data with hundreds of thousands of labeled images which is computationally intensive while the latter can be used with new data or task with small to medium size datasets.

By integrating low-level features and classifiers into a unified neural network, deep learning is a state-of-the-art technique that performs end-to-end learning with a considerable degree of accuracy [8]. The "levels" of features can be enhanced by the number of stacked layers (depth). However, a significant problem arises when the network depth is increased. Deeper networks are subjected to degradation

978-1-5386-0389-5/18/$31.00 ©2018 IEEE

problem in which accuracy gets saturated and then degrades rapidly as shown in the study of [9].

Recently, techniques regarding Convolutional Neural Networks (CNN) have been popularized in fields of image processing, classification, and pattern recognition problems, especially since the AlexNet [10] acquired a 15.3% test error rate in ImageNet Large Scale Visual Recognition Competition-2012 (ILSVRC) [11]. CNNs have been useful in a lot of fields. In [12]-[15], deep learning was used in segmentation of brain tumor, liver tumor, anatomical brain, and kidney. In [16]-[17], its detection capability was used in mitosis and glaucoma. In 2014, the authors of [18] used CNNs for object detection through the combination of both CNN and selective search [19]. In [11], the authors generated traffic sign regions for test images, and are then fed into a fine-tuned CNN for classification. Their study attained a test accuracy of above 97% on traffic sign classification.

In deep learning, effectively training different kinds of networks requires substantial resources like great amounts of time and data. A large dataset is convenient, but when faced with a small amount of available data, overfitting becomes a problem. Overfitting is a state of the network wherein it perfectly learns the features from the training data set but does not infer on the testing data set [24]. To solve this, transfer learning is a common solution. The difference of the features of both datasets boosts the performance of the target task [20]. In [21], the authors relied on two types of CNN in order to test the effectiveness of transfer learning for aerial imagery classification, namely; AlexNet and LeNet. In LeNet imagery classification, transfer learning did not produce any improvement versus random initialization. However, an increase of 10.7% in generalization accuracy was observed when transfer learning was used on AlexNet.

In terms of hardware, computational speed is still a limiting factor for deep architectures characterized by many building blocks [7]. Moreover, the limitations in computational resources and the number of training samples constitutes a significant problem. This results to a time-consuming training process which also leads to overfitting. To overcome these problems, transfer learning on pre-trained convolutional neural networks have been proposed as method for the classification of traffic sign design compliance and physical degradation status.

II. Methodology

The methodology is divided into five sections, namely: google street view, road sign design manual, dataset organization, transfer learning, and fine-tuning the convolutional neural network.

A. Google Street View

The authors considered using the GSV web platform in order to develop a traffic sign image database for training, validation, and testing. The image samples of varying design and condition found along the streets and highways of the Philippines were manually cropped by the authors without consideration of the pixel dimension as later pre-processing techniques are to be adopted. The GSV is a network of adjacent 360 degrees high resolution panoramic images, which are divided into quadratic tiles [1]. These massive street imageries are accessed via the Google Maps API, defined by a geolocation input, which can either be numeric (latitudal and longitudinal values) or maybe in the form of a string (specific address). One of GSV's technology is to identify and blur the faces and license plates that is within the Google-contributed imagery. In some cases, text in traffic signs are also blurred out. Table I shows the zoom levels Google Street has to offer.

TABLE I. Zoom Levels Available In The Google Street View

Zoom level	Number of tiles	Resulting image size
0	1 x 1	512 x 256
1	2 x 1	1024 x 512
2	4 x 2	2048 x 1024
3	8 x 4	4096 x 2048
4	16 x 7	8192 x 3584
5	28 x 13	14336 x 6656

B. Road Design Manual

The Highway Safety Design Standards (HSDS) manual is prepared and issued by the Department of Public Works and Highways (DPWH) of the Philippines. This is to guide the local governments and also aid in the installation of standard signs throughout the country. The scope of the manual includes the description of the different types of road signs and pavement marking signs with their prescribed standards and conditions. The manual also states that traffic signs shall be installed upon the Secretary of the DPWH or his delegated authority only, having the necessary jurisdiction. Unofficial, non-standard and non-essential signs are not permitted as well [5]. Fig. 1 shows some examples of compliant traffic signs in the Philippine setting.

Fig. 1. Properly Designed Traffic Signs in the Philippines

In classifying the compliance and degradation of the traffic signs, the HSDS served as the main reference. In the compliance validation phase, traffic signs designed differently from the manual are classified as 'non-compliant', otherwise it is 'compliant'. In the condition validation phase, traffic signs with folds, unnecessary stickers, vandals, and discolorations are automatically considered "bad", otherwise it is 'good'. However, it is important to take note that some of the data collected by the authors from GSV are blurred. This is due to Google's Privacy Policy in which they blur some parts of the images in order to protect the privacy of individuals [22].

C. Dataset Organization

The authors considered traffic sign classes frequently appearing in the GSV. A total of 358 traffic sign images in 13 classes were collected in the study. These traffic sign samples are forwarded to the DPWH Road Safety Audit (RSA) team to label the images acquired, in terms of compliance and in terms of degradation, thus serving as the ground truth in the study. Fig. 2 shows a montage of impaired traffic signs in the acquired image database, while Fig. 3 shows the frequency distribution of the selected 13 traffic sign classes.

Fig. 2. Examples of Impaired Traffic Signs from GSV

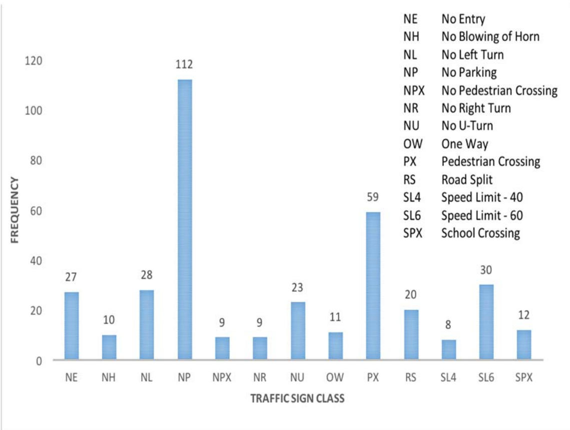


Fig. 3. Frequency Histogram of Acquired Database

In order to balance the acquired dataset, the authors augmented the data twice, one for classification of good and bad, and another for compliant and non-compliant. A total of 780 images were derived after the dataset augmentation was performed. This was done by automating a script that resizes the pixel dimension of each of the traffic sign classes. In this way, the number of images in the dataset is artificially increased before it is fed for training in the network. Two labeled image datasets are presented in the study, one for design compliance, and another for physical degradation.

D. Transfer Learning

The system was implemented using convolutional neural network (CNN) algorithm since it exhibits impressive performance in image classification [9]. The network contains layered architecture that combines multiple nonlinear processing layers, using simple elements operating in parallel. It assumes fully connected layers (FC) which consists of an input layer, hidden layers, and an output layer. Each of these layers has certain number of nodes which are fully connected with the next layer. The nodes perform dot product of weight and input, and feed forwards the output to the next layer. Each hidden layer uses the output of the previous layer as its input.

The approach of the study relies upon using a pre-trained network and fine-tuning it to perform new recognition task. To accomplish this, a technique called transfer learning was used. It relies on the idea that knowledge gained from the pre-trained network can be utilized to improve the training process of a new network with similar task. In this case, it takes the process of fine-tuning the weights of each FC layer in the network.

The pre-trained convolutional neural network named ResNet-50 was utilized by the system. This network was trained on a subset of the ImageNet database, which won the 1st place in the ImageNet Large-Scale Visual Recognition Challenge (ILSVRC) 2015 classification competition [9]. The network's layers were arranged as a directed acyclic graph (DAG) involving more complex architecture that contains multiple layers.

The pre-trained network [9] used a residual learning framework to alleviate the training of very deep networks. Every few stacked layers were fitted to a residual mapping instead of hoping every few stacked layers directly fit the desired underlying mapping. This was assumed to be easier for optimization. Denoting the desired underlying mapping as H(x) and the input as x, the residual mapping is defined as F(x)=H(x)-x. The original mapping is then remodeled into F(x)+x. This can be realized by feedforward neural networks with shortcut connections. The shortcut connections simply perform identity mapping. Its outputs were added to the outputs of the stacked layers. Identity shortcut connections add neither extra parameter nor computational complexity. The residual units are stacked in chain mode as shown in Fig. 4.

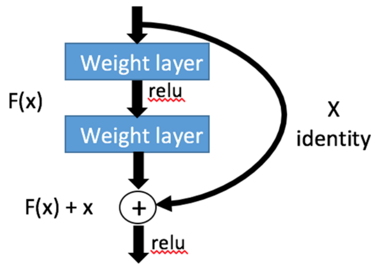


Fig. 4. Residual Learning Framework

E. Fine Tuning the Convolutional Neural Network (CNN)

In MATLAB R2017b, the pre-trained CNN-resnet50 is readily available in the neural network toolbox. In order to implement the transfer learning technique, the approach in Fig. 5 is followed. The last three layers, which contain the information of how to combine the features that the network extracts into class probabilities and labels, were replaced. A fully connected layer, a softmax layer, and a classification output layer were added to the layer graph. These layers were then connected to the average pool layer as shown in Fig. 6.

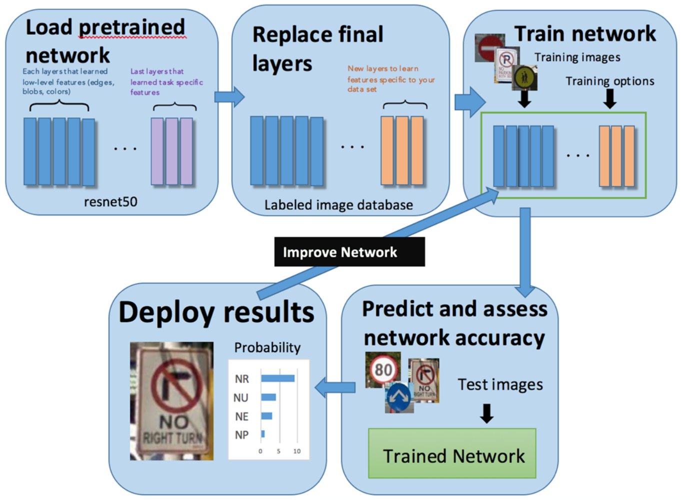


Fig. 5. Re-using a pre-trained network

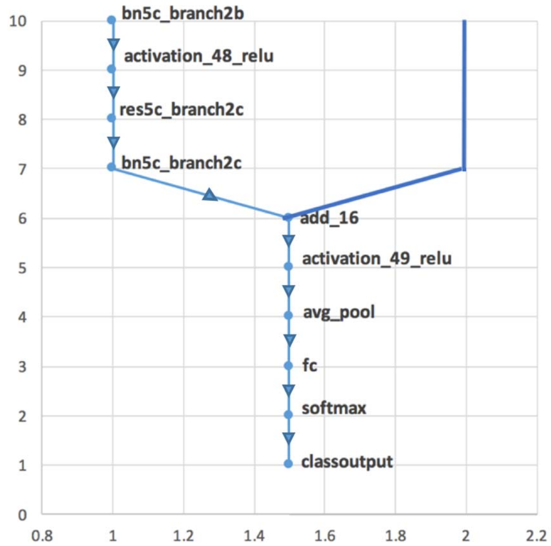


Fig. 6. Layer Graph of the Proposed Final Layers of ResNet-50

The images from the curated dataset are pre-processed prior to training to 224x224x3, pertaining to the pixel length, pixel width, and to the number of channels in the image (3 is RGB) by applying the imresize function to the image datastore generated in MATLAB. This was done to accommodate the size of the image input layer of the ResNet-50 architecture.

The network was trained using the Stochastic Gradient Descent with Momentum (SGDM) optimization algorithm in MATLAB. This technique updates the parameters (weights and biases) to minimize the error function by taking small steps in the direction of the negative gradient of the loss function [25]. By default, the program shuffles the data once before training.

$$\theta_{\ell+1} = \theta_\ell - \alpha \nabla E(\theta_\ell) + \gamma(\theta_\ell - \theta_{\ell-1}) \qquad (1)$$

where, ℓ: *iteration number*
α: *learning rate*
θ: *parameter vector*
$E(\theta)$: *loss function*

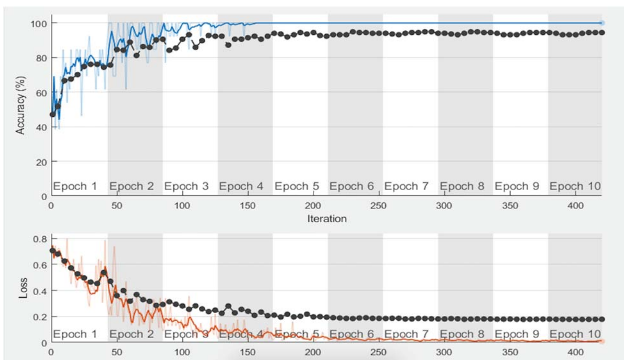
Aside from the training algorithm used, there are training options utilized in the program and is presented in Table II. These specifications were tested manually using the aforementioned SGDM function and the presented parameters in the Table II yielded the best results.

TABLE II. NEURAL NETWORK TRAINING PARAMETERS

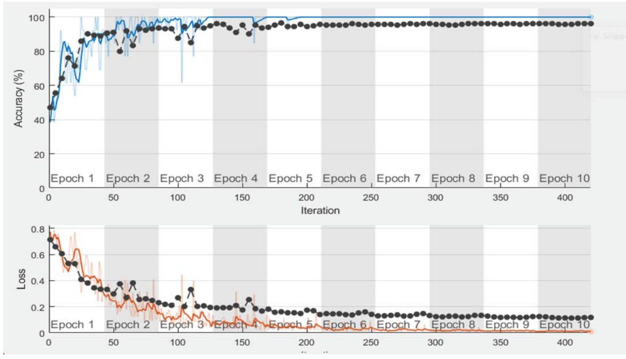
SGDM Property	Definition	Value
Momentum	The contribution of the gradient step from the previous iteration to the current iteration of the training. A scalar value from 0 to 1.	0.9
Initial Learning Rate	If the learning rate is too low, training takes a long time, if it is too high, the training might reach suboptimal result. A positive scalar value.	0.0001
L2 Regularization	The factor for L2 regularizer (weight decay). A non-negative scalar value.	0.0001
Max Epochs	Maximum number of epochs for training. A positive integer.	10
Mini Batch Size	Size of the mini-batch used per iteration A positive integer.	13
Validation Frequency	The frequency of network validation in number of iterations. A positive integer.	5

III. DATA AND RESULTS

The training was performed on MATLAB using a CUDA Device of GeForce 940m with a compute capability score of ‘5.0’, which is much higher than the least recommended ‘3.0’ score by the software for training a CNN. The training performance of the network in the two databases presented are shown in Fig. 8. The SGDM parameters and the number of images was not changed in the training phase. The plots presented show that the validation accuracy was higher and the validation loss was lower in the first 3 epochs in Fig. 7 (a) as compared to Fig. 7 (b). The accuracy started to stabilize in epoch 5 for the two networks presented. However, Fig. 7 (b) presented a higher final validation accuracy of 96.15% as compared to Fig. 7 (a) garnering a score of 94.44%.



(a)



(b)

Fig. 7. (a) Training Performance of Compliance Model; (b) Training Performance of Degradation Model; (Blue - Accuracy; Orange - Loss; Black – Validation)

Prior to training, each of the two image datasets were split into 70-30 percent, for the training set and the testing set, respectively. After the network was trained, it is tested on images not present in the training, which is called the testing set. The performance of the two networks is shown in Fig. 8 and Fig. 9 using confusion matrices. In the compliance dataset, shown in Fig. 8, 46.2% of the traffic signs are correctly classified as compliant while 49.6% are correctly classified as non-compliant. Whereas 0.4% are incorrectly classified as compliant, and 3.8% are incorrectly classified as non-compliant. Overall, 95.7% of the predictions in the compliance test set are correct while the remaining 4.3% are wrong classifications.

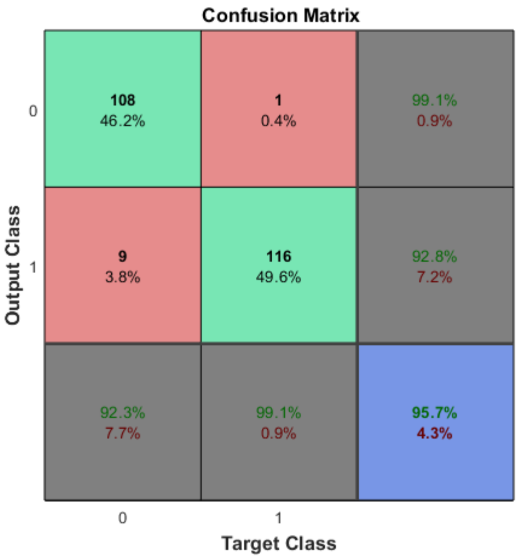


Fig. 8. Confusion Matrix for Compliance Model
0: Compliant, 1: Non-compliant

In the degradation dataset, shown in Fig. 9, 47.0% of the traffic signs are correctly classified as good while 49.1% are correctly classified as bad. Whereas 0.9% are incorrectly classified as good, and 3.0% are incorrectly classified as bad. Overall, 96.2% of the predictions in the degradation test set are correct while the remaining 3.8% are wrong classifications.

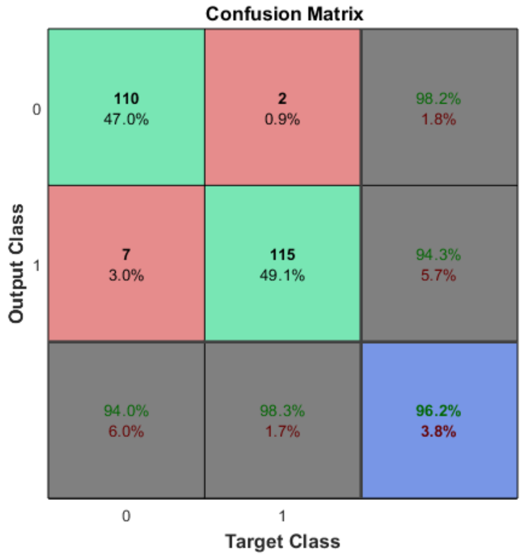


Fig. 9. Confusion Matrix for Degradation Model
0: Good, 1: Bad

IV. Discussion of Results

For the training phase, the two trained models' validation accuracy differ only by 1.71% in favor of the trained degradation classifier. Both are trained in under 50 minutes using a single GPU and a constant learning rate of 0.0001. The two models suffer in terms of accuracy as shown in the early epochs of the training, but the two started to reach the 90% mark as early as the 3rd epoch. The fluctuation in the accuracy and loss started to diminish by the 5th epoch, as both of the network's accuracy started to stabilize.

In the testing phase, the two trained models' testing accuracy - the network's performance on unseen data, differs only by 0.5%, also in favor of the degradation classifier. Overall, both of the trained models show good results and thus can be considered in real world application of the assessment of the integrity of traffic sign imagery.

This study has several limitations. First, the scope of all the traffic sign classes present in the Philippines were not covered as some of the missed classes are quite few in the GSV platform. Second, the classification module is only a part of a full traffic sign recognition system, so the model can only predict signs fed to it as long as it is cropped, hence it cannot detect or isolate traffic signs from a given background.

V. Conclusion and Future Work

The main focus of this paper is to validate the feasibility of developing a model that can classify a traffic sign in terms of its design compliance to a set standard and to classify its physical degradation status. The goal of this research is two-fold. First, is to construct a traffic sign image database of varying design and degradation status specific to the Philippine setting. Second, is to train two models that can distinguish a compliant to a non-compliant traffic sign, and a good to a bad traffic sign, both spanning multiple classes. The latter was carried out by fine-tuning a pre-existing convolutional neural network – the ResNet50, specific to the developed dataset through a process called transfer learning. The researchers achieve a test accuracy of 95.70% in the compliance model and 96.20% in the degradation model using only a small database of traffic sign images.

The future of this research will be to develop a full traffic sign recognition system, capable of both detection and classification using the publicly available Google Street View imagery, in order to develop a more streamlined approach in traffic sign assessment. The realization of this work can be used to map non-compliant and bad traffic signs, as well as compliant and good, in a wide scale for a considerable amount of time using only a computer. In addition, the scope of the traffic sign classes can be increased to cover all possible categories as the researchers limited the study to 13 classes.

ACKNOWLEDGMENT

The authors would like to thank the Department of Public Works and Highways for providing significant resources. Also, special thanks to the Electronics Engineering department of the University of Santo Tomas for their technical support and continuous guidance.

REFERENCES

[1] C. Ai, "A Sensing Methodology for an Intelligent Traffic Sign Inventory and Condition Assessment Using GPS/GIS, Computer Vision and Mobile LiDAR Technologies", 2013.

[2] “White paper – ‘European transport policy for 2010: time to decide’ - Mobility and Transport - European Commission", Mobility and Transport, 2001. [Online]. Available: https://ec.europa.eu/transport/themes/strategies/2001_white_paper_en. [Accessed: 04- May- 2017].

[3] European Commission (Directorate General Energy and Transport), "Harmonisation of road signs and road marking on the TERN from a safety point of view", IMPROVER (Impact Assessment of Road Safety Measures for Vehicles and Road Equipment), 2006.

[4] "Halving the number of road accident victims in the European Union by 2010: a shared responsibility.", Eur-lex.europa.eu, 2008. [Online]. Available: http://eur- lex.europa.eu/legal-content/EN/ALL/?uri=celex%3A52008SC0350. [Accessed: 04- May-2017].

[5] Road Signs and Pavement Markings Manual. (2012). 2nd ed. Manila, Philippines.

[6] Zakir, U., I. Zafar, and E. A. Edirisinghe, 2011. Road Sign Detection and Recognition by Using Local Energy based Shape Histogram (LESH), International Journal of Image Processing (JIIP), vol. 4, no. 6, pp. 567-583.

[7] D. Cireşan, U. Meier, J. Masci and J. Schmidhuber, "Multi-column deep neural network for traffic sign classification", Neural Networks, vol. 32, pp. 333-338, 2012.

[8] MathWorks Inc., Introducing Deep Learning with MatLab, 1st ed. .

[9] K. He, X. Zhang, S. Ren and J. Sun, "Deep Residual Learning for Image Recognition", 2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2016.

[10] Zakir, U., I. Zafar, and E. A. Edirisinghe, 2011. Road Sign Detection and Recognition by Using Local Energy based Shape Histogram (LESH), International Journal of Image Processing (JIIP), vol. 4, no. 6, pp. 567-583.

[11] A. Krizhevsky, I. Sutskever and G. Hinton, "ImageNet classification with deep convolutional neural networks", Communications of the ACM, vol. 60, no. 6, pp. 84-90, 2017.

[12] Y. Li, A. Mogelmose and M. Trivedi, "Pushing the “Speed Limit”: High-Accuracy US Traffic Sign Recognition With Convolutional Neural Networks", IEEE Transactions on Intelligent Vehicles, vol. 1, no. 2, pp. 167-176, 2016.

[13] W. Li, F. Jia and Q. Hu, "Automatic Segmentation of Liver Tumor in CT Images with Deep Convolutional Neural Networks", Journal of Computer and Communications, vol. 03, no. 11, pp. 146-151, 2015.

[14] A. de Brebisson and G. Montana, "Deep neural networks for anatomical brain segmentation", 2015 IEEE Conference on Computer Vision and Pattern Recognition Workshops (CVPRW), 2015.

[15] S. Pereira, A. Pinto, V. Alves and C. Silva, "Brain Tumor Segmentation Using Convolutional Neural Networks in MRI Images", IEEE Transactions on Medical Imaging, vol. 35, no. 5, pp. 1240-1251, 2016.

[16] W. Thong, S. Kadoury, N. Piché and C. Pal, "Convolutional networks for kidney segmentation in contrast-enhanced CT scans", Computer Methods in Biomechanics and Biomedical Engineering: Imaging & Visualization, pp. 1-6, 2016.

[17] S. Albarqouni, C. Baur, F. Achilles, V. Belagiannis, S. Demirci and N. Navab, "AggNet: Deep Learning From Crowds for Mitosis Detection in Breast Cancer Histology Images", IEEE Transactions on Medical Imaging, vol. 35, no. 5, pp. 1313-1321, 2016.

[18] X. Chen, Y. Xu, D. Wong, T. Wong and J. Liu, "Glaucoma detection based on deep convolutional neural network", 2015 37th Annual International Conference of the IEEE Engineering in Medicine and Biology Society (EMBC), 2015.

[19] R. Girshick, J. Donahue, T. Darrell and J. Malik, "Rich Feature Hierarchies for Accurate Object Detection and Semantic Segmentation", 2014 IEEE Conference on Computer Vision and Pattern Recognition, 2014.

[20] J. Uijlings, K. van de Sande, T. Gevers and A. Smeulders, "Selective Search for Object Recognition", International Journal of Computer Vision, vol. 104, no. 2, pp. 154-171, 2013.

[21] S. Pan and Q. Yang, "A Survey on Transfer Learning", IEEE Transactions on Knowledge and Data Engineering, vol. 22, no. 10, pp. 1345-1359, 2010.

[22] D. Cisek, M. Mahajan, J. Dale, S. Pepper, Y. Lin and S. Yoo, "A transfer learning approach to parking lot classification in aerial imagery", 2017 New York Scientific Data Summit (NYSDS), 2017.

[23] "Google Street View – Image Acceptance & Privacy Policies", Google Street View. [Online]. Available: https://www.google.com/streetview/privacy/#privacy-and-blurring. [Accessed: 20- Nov- 2017].

[24] S. Prajapati, R. Nagaraj and S. Mitra, "Classification of dental diseases using CNN and transfer learning", 2017 5th International Symposium on Computational and Business Intelligence (ISCBI), 2017.

[25] C. Bishop, Pattern recognition and machine learning. New York [u.a.]: Springer, 2013.