National College of Ireland

Traffic Sign Recognition & Detection using Transfer learning

MSc Research Project
Data Analytics

Akhil Sharma
Student ID:x17168180

School of Computing
National College of Ireland

Supervisor: Prof. Vladimir Milosavljevic

National College of Ireland

MSc Project Submission Sheet

School of Computing

Student Name: Akhil Sharma

Student ID: X17168180

Programme: MSc Data Analytics **Year:** 2018-19

Module: Research Project

Supervisor: Vladimir Milosavljevic

Submission Due Date: August 12th, 2019

Project Title: Traffic sign recognition and detection using Transfer Learning

Word Count: 7156 **Page Count** 25

I hereby certify that the information contained in this (my submission) is information pertaining to research I conducted for this project. All information other than my own contribution will be fully referenced and listed in the relevant bibliography section at the rear of the project.
ALL internet material must be referenced in the bibliography section. Students are required to use the Referencing Standard specified in the report template. To use other author's written or electronic work is illegal (plagiarism) and may result in disciplinary action.

Signature: Akhil Sharma

Date: August 11th ,2019

PLEASE READ THE FOLLOWING INSTRUCTIONS AND CHECKLIST

Attach a completed copy of this sheet to each project (including multiple copies)	□
Attach a Moodle submission receipt of the online project submission, to each project (including multiple copies).	□
You must ensure that you retain a HARD COPY of the project, both for your own reference and in case a project is lost or mislaid. It is not sufficient to keep a copy on computer.	□

Assignments that are submitted to the Programme Coordinator Office must be placed into the assignment box located outside the office.

Office Use Only	
Signature:	
Date:	
Penalty Applied (if applicable):	

Traffic Sign Recognition & Detection using Transfer learning

Akhil Sharma
x17168180

Abstract

Traffic sign recognition feature is widely employed in industry today by researchers working in artificial intelligence and machine learning fields targeted to create an autonomous driving assistance system. This will help the driver's human eye and manoeuvring capabilities which can't be presumed to be consistent for high accuracy for all weather condition challenges. As the limitation of the human eye perception on a road sign is upsetting the road safety. This research paper illustrates a novel approach for Traffic image classification using Deep learning model to distinguish the different classes of traffic signs only after applying Data augmentation offered by Keras deep learning library for shifting traffic signs images (5%,10%,15%) and applying Gaussian blur technique for blurring (3:3,5:5 & 7:7) images. This will help us re-create a dataset similar to images captured during bad climatic, foggy or windy conditions seen in Ireland or badly captured images due to fast moving cars with limitation in camera angles from roads. We will be introducing these pictures on a pre-trained transfer learning model VGG19 as it has better accuracy and operational speed to create a best fit model. Results of this research work can be used by automobile companies to create enhanced image capturing cameras and classification models to make better autonomous systems and safer roads.

Keywords: Gaussian Blur, Deep Learning, Keras, Transfer learning

1 Introduction

Traffic signs recognition is a critical multi category classification task which involves two steps which include detection of the traffic sign and recognition after classification using machine learning models. As we know "If We Want Machines to Think, We Need to Teach Them to See" Ai.stanford.edu. (2019) hence this becomes a computer vision problem case which is an overlapping field between image processing, Artificial intelligence, Signal processing, Pattern recognition, mathematics and physics. This research project is aiding traffic sign recognition (TSR) systems using these techniques so that we can work on this innovative approach of self-driving or driver less vehicles. Several studies have taken place previously on this subject which are available today and are being employed by major automobile companies. However, the challenges which are faced due to bad weather conditions, Illumination, Rotation, shifting, Occlusion faced by the model increase the complexity range of variation with which a classifier must undergo. In this paper we are working on such challenges by blurring and shifting the images of the dataset, we are using German Traffic Sign Recognition Benchmark (GTSRB) dataset available publicly with

approximately 50 thousand images and 43 classes and taking a subset of 10 classes with 14760 images to create a novel transfer learning model which can withstand the challenges offered to enhance and provide high accuracy for TSR.

1.1 Motivation

With rapid increase in Science & Technology in future Autonomous cars will be widely used and developed which employee sensors and computer vision software's for better safety and control Ramler, R. and Ziebermayr, T. (2017). Innovation is helping to minimize human efforts and enhance use of sophisticated machines for interaction with a clear motive of Automation and achieve better Accuracy of work, Safe precious time and most important safety of human lives. I choose this futuristic topic to work on an approach and area of interest of solving a problem case using computer vision technology for classification of traffic sign with introducing some challenges in the dataset which will aim to be highly productive and a godsend better result. Applications of this project can be directly used in the industry for making better computer vision models for automation.

1.2 Research Question

How can we automate and classify Traffic signs to reduce road accidents in bad weather conditions?

1.3 Research Objective

To achieve this research task and develop an enhanced machine learning model for image classification we have used a pioneering approach used in deep learning called transfer learning. This approach has high accuracy and speed for image classification and works on previously trained dataset. Hence the model doesn't require large dataset for training. We have also tried to recreate bad weather condition dataset by blurring the image using gaussian blur and shifting images for augmentation for real challenges. The results of the project will help us understand the limitations of the model and the level of accuracy using which we can introduce a case study for the automobile companies to showcase improvements on the TSR.

2 Related Work

Computer vision is a scientific field that deals with getting computers an elevated understanding from digital images or videos. This innovative expertise works on principles where we gather, processes and then analysing meaningful information from image or video, digital data from the real-life world has applications like augmented reality, Object detection, Image classification. Recently many researchers have worked on image classification for traffic signs using different datasets and machine learning models to find a high accuracy and suitable methodology. Hence, it's difficult to compare the different working models for

recognition. However, we will try to understand and explain the different research done so far and to find a model best suitable for our experiment.

2.1 Image Recognition

We will try to understand the Image recognition from the research work of author Nguyen, B. et al. (2014) the paper talks about a robust system with two stages which include segmentation and detection of the image processing. The dataset captured for this research is using a VGA camera which showcases a poor quality and bad light condition to create a good dataset for testing the model. The paper also talks about a real time system with high accuracy rate however the segmentation process is only applied for small dataset. In our research we need to find a stable method for large dataset however with a limited input and with high level of operational accuracy to get a benchmark for future.
The next research paper from Changzhen, X. et al. (2016) describes that the existing detection models for traffic sign recognition are just based on limited predefined set of signs. This paper is working on the regional language in china for the discovery of signs using deep convolution neural network (CNN). The dataset is defined into classes, categories and sub-categories for trainings and testing the model. The results show 99% accuracy and uses real time detection in video sequences. The method shared in this report helps us understand vast details which can be incorporated in making a model. We learnt how to categories data for our research project as well.

The author Mahatme, M. and Kuwelkar, M. (2017) yet again describes and new approach for traffic sign classification, the two steps approved in this research is detection and recognition which is like all the related work papers studied. However, the paper describes a new method as layering with an “RGB to red conversation” for getting image red component, filters for noise reduction and edge detection, then later applying a method of thresholding and segmentation after which the image is processed to an artificial neural network for recognition. The images used in this dataset are obtained from Indian traffic signs. The implementation uses MATLAB for the processing which is not a favorable data analytics language. The RGB to red conversation method increases the computational speed and as suggested by the paper the method can be used for parallel processing of hardware like GPU for real time results after implementation which is a promising method.

The paper Tastimur et al. (2016) helps us understand the principle of gaussian blur which is used for tampering and augmentation of images to create a bad condition for the model to test the accuracy level. The author is also trying to solve the gaussian blur inconsistency to get the images using motion blur and defocus blur methods. This research work can help us in creating a blurring image dataset. The model works on calculating the standard deviation of gaussian blur and after calculating this value the image is deblurred which can be introduced into Richardson-Lucy algorithm which will work as a novelty for our model of research work so create blurring dataset. The author Yanjun Fan and Weigong Zhang (2015) is discussing about a novel approach using an algorithm based upon machine vision and machine learning. Again, the author shares the importance of making the algorithm into two steps being detection and recognition. The experiment performed in this paper has very low running time of 40ms and high level of accuracy. Hence the proposed system has good level of performance and is fit for assistance system. The model is best fit for object detection and classification; however, the model works on enhancing the colors of the traffic signs like blue

or red working on chromatic filters which seems like a complex model for bad climatic conditions to work.

Author Liu, C et al. (2016) describes a high accuracy traffic sign recognition model with three step framework which is a novel method discussed in this paper explaining a method for a region of interest (ROI) extraction using a high contrast region of extraction, split-flow cascade tree detector (SFC-tree) and a classification method based called extended sparse representation classification (ESRC).Compared to the previous researcher for extraction of the region of interest this method proposed is designed to keep a good equilibrium in extraction rate and detection rate and is promising to classify a large dataset with high resolution .This method is highly effective to save detection time for the TSR with a level of 8-12 frames of accuracy on the GTSDB. This experiment uses a widely known dataset for the experiment and solution.

2.2 Transfer learning

The next step is to research about the latest technologies to implement TSR as studied several papers transfer learning was discovered as an innovative machine learning model. The research done by author van Opbroek et al. (2015). proposes a transfer learning method as well for image segmentation which will help to overcome the challenges faced by automatic segmentation caused due to using dissimilar scanners or imaging conventions otherwise require labelling of the dataset to get high accuracy. The results of this model are clearly showing that even though the dataset for image recognition is small the accuracy of the model outperforms supervised learning model even with small number of data samples. This highlight transfer learning as a suitable method for the research as this reduces the amount of labelled data required for the model and gives high accuracy.
The author CENGIL, E. and CINAR, A. (2018) in the research paper discusses about a deep learning as a widespread choice for classification specially in healthcare. The implementation of deep learning and tensor flow described in this paper will clarify for our research project as well as it shows ground-breaking results in confusion matrix scores as 3D CNN model is applied even though the images used in the dataset are smaller in size for classification. This research paper helps us in understanding how to implement deep learning model for classification and creation of confusion matrix and CNN model. These steps can be used in architecture and results section of or research

The author Chou, L. et al. (2018) describes about malicious traffic attacks on security as the traditional system are using deep learning model which is applied using TensorFlow for network security. The paper helps us understand the methodology for using deep learning with test and training dataset. The author uses NSL-KDD dataset for training and testing the model. The research done helps us understand how we can divide our dataset into test and train folder and apply the model for machine learning to get results. We can also perform the similar experiment by dividing our traffic signs data images. The research paper by author Choi, S et al. (2019) discusses about artificial intelligence and its impact on autonomous vehicle systems which is the core point of our research. The research work talks about object detection and a real-life scene creation for better safety. The author wants to create a model to prevent accidents. The method of model used is dark flow which is combination of YOLO (You only live once) framework and tensor flow. This makes the object detection easy however the situation detection is little difficult as described in the research work by the author. Performing this experiment with the novelty introduced in our research paper can

become the future work for our research. This will help us create a real time perfect YOLO object detection and classification model which can work in bad weather conditions as well and thus giving an expert level assistance for creating an autonomous driver. This research paper marks a base to understand the TensorFlow and YOLO models.

Author Ling Shao, Fan Zhu and Xuelong Li (2015) describes in the research paper which helps us understand the issues of over fitting of training data as there is limited accessibility of labeled dataset. As we know the future dataset can't be produced to stop overfitting issues. Transfer learning is used to solve such issues as it works on extracting previous knowledge and transferring them to target .The research done by author discusses various applications of transfer learning such as image recognition with conceptual idea of drifting of images which can be used in our project for creating a bad dataset. Research work is also helped me understand transfer learning approaches for TSR model.
According to the literature survey done ,research work done by author Wei, L., Runge, L. and Xiaolei, L. (2018). we found that steps followed in the paper for images is detection and recognitions of target images features and then feeding the images into SVM or deep learning classifier for image classification. However, model's limitation is the dataset which has overfitting and operational challenges which can be overcome by transfer learning model discussed by the researcher. Also, the author is using the popular GTSDB dataset which can be our dataset for research as it has already bad condition or challenging images for classification. The paper also showcases the position of camera placements and other sensors which can be learning curve for practical application and implementation.

The author in the paper by Li, D., et al. (2018) helps us understand the capacity of deep learning model which is discussed with three step methodology of detection of traffic sign, classification and finally correcting the classification recovering the missed detectable regions. The deep sign model proposed is promising as to remove the false positive and negative signs. The dataset taken for this research was a video clip input which was classified. The model is award winning and had high accuracy and precision as shared by the research paper.

The research paper by Acilo, J., (2018) clearly highlights the importance of road signs for the safety and the need of the hours to make an automated system which is the motivation for our research question. We also learned from the paper that whenever we have limited dataset and weak computing resource, we use a deep learning-based transfer learning approach. As we know applying machine learning algorithms on large dataset required GPU requirements. This paper helps us understand the limitation of our research project and gives an understanding on how to be applying transfer learning model. We also learn the system configuration and requirements of GPU for implementation of a deep learning transfer model

The research work done by Lamouik et al. (2018) is for routing traffic for vehicles with a similar objective of road safety and saving time. Here the classification is done to find the best path for the vehicles . From the research work done we can see that the author is working on implementation of deep learning model for showcasing fast travelling routes which avoid red lights. The research clearly highlights the difference between timings of the actual path and proposed path which are derived after the deep learning model, hence establishing that deep learning technology is a popular machine learning solution for solving complex problems.

3 Research Methodology

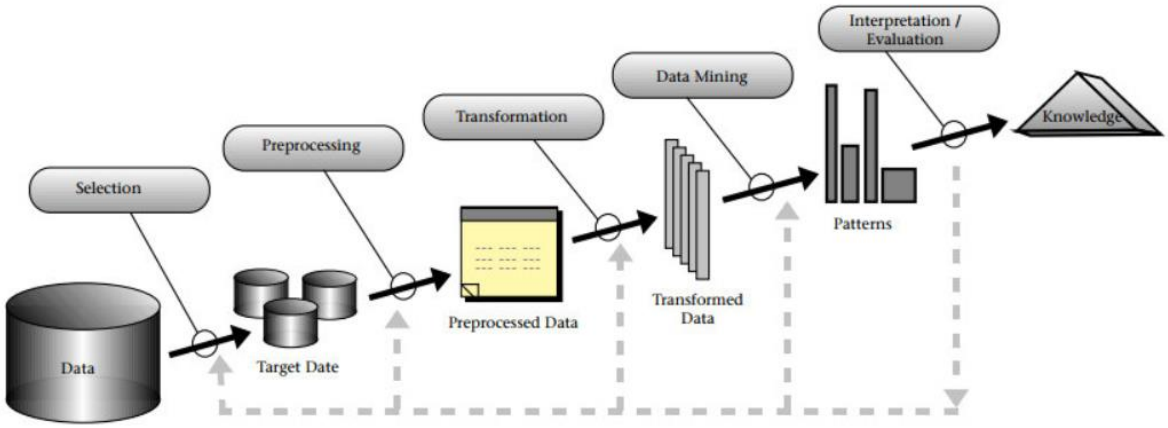


Figure 1. KDD Methodolgy

KDD framework is an abbreviation for Knowledge discovery in database which is a popular project implementation methodology plus helps us provide a structured approach for planning a project (Bruha, 2000). This process helps in stemming useful information called knowledge from the dataset. (Proceedings of the 25th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining, 2019)

KDD works on application of certain principles with the first step being selection of dataset or subset of the data/samples to understand the dataset from where the knowledge must be derived. Second step is integration of multiple dataset if possible, third step involves removing noise or cleaning the data which is also called EDA (Exploratory Data Analysis), fourth step is transformation as showed in the Figure 1 this step helps join and transform the data for the data mining algorithms ,this step is also called reduction as the variables which are relevant for the goal or task are selected so as to make the data ready for the next step where the data mining process are applied to extract the patterns from the dataset. The next principle step which is again an important part of KDD framework is the pattern evaluation or interpretation which helps use business intelligence tools to derive knowledge and can be used to build a presentation to get insights and trends from the dataset. (Gertosio and Dussauchoy, 2004)In this research project we utilize the KDD model as it's the optimum and fits the project to gather the knowledge from the dataset and gain useful information. The source of dataset is the German Traffic Sign Benchmark dataset available from Kaggle which is a public website. After selecting the dataset follows the preprocessing stage, the selected images in the benchmark were already in bad conditions as they were collected from the real-world scenarios and updated in Kaggle .After which we applied blurring to the images (of the images to create bad condition of weather to check the level of accuracy of model.
We also split the image data into three folders train (80% data), validation (10% data) and test (10% data) to transform the data for the data mining algorithms.
Then after we explored the deep learning model and algorithm called transfer learning as researched in the related work section which is state-of-the-art technology appropriate for our project which can be applied with works in limited dataset and showcasing high accuracy for

image classification model. We applied the model for three different levels of blurring and shifted images Final step which is followed is the pattern evaluation where we are using confusion matrix and tableau to gather insights from the derived model to build a user story from the different accuracy levels gathered from the model to understand better the classification for traffic sign recognition model.

4 Design Specification

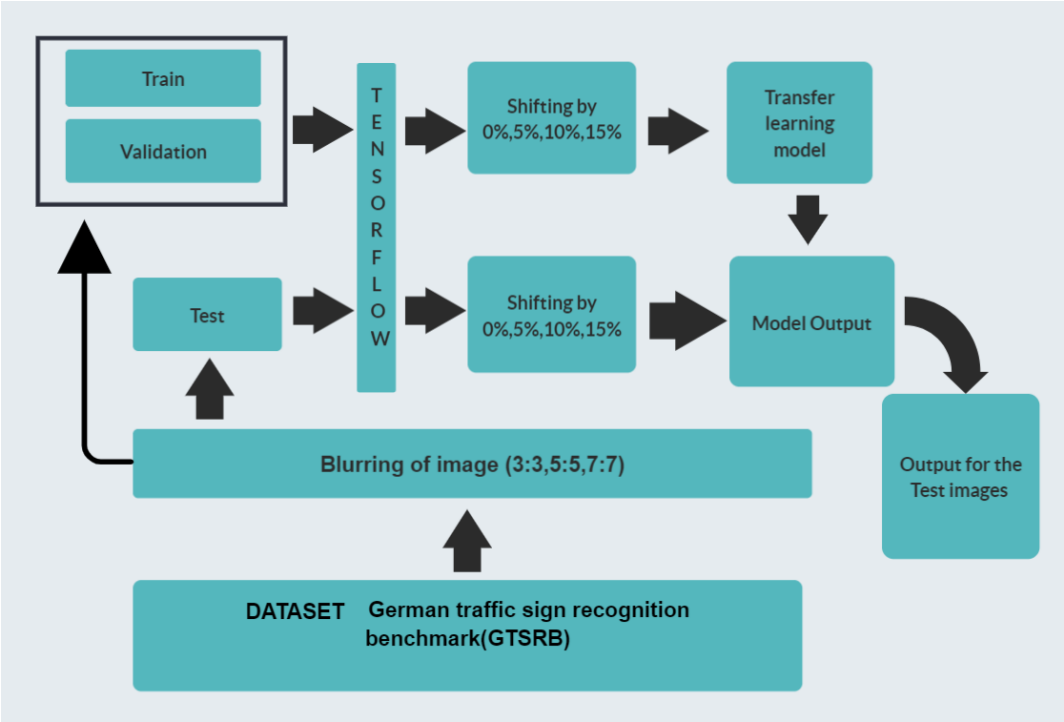


Figure 2. Proposed Architecture diagram for TSR.

In this part of the research paper we explain about the project specification and the architecture as highlighted in the Figure 2. The strategy and specification of the project are listed below following the KDD methodology and detailed explanation of the blocks from the figure.

- Process of our experiment begins with Gathering the dataset from the GTSRB Kaggle website .With selection of 10 classes (Danger, speed 20,30,50,60,70,80,100,120, stop) out of 50 and approximately 1500 images as our dataset to classify in our research project.
- The images are gathered from Kaggle website after selecting 10 classes of the dataset for our project. We are applying Gaussian Blur Mair, B., Wilson, D. and Reti, Z. (1996) for blurring the images as highlighted in the architecture which is done for 3 levels (3:3,5:5 & 7:7) according to gaussian blur technique we use odd series to blur the images for checking the sturdiness of our model and create a dataset which can replicate bad climatic condition in Ireland. (En.wikipedia.org, 2019)

- Next step in our project we are splitting the dataset (GTSRB) which is already labeled and is in a bad condition into Train, Validation and Test folders after blurring them according to the blurring levels. We split the dataset in a format with 80% of the images in the Training folder,10% in the Validation Folder and 10% in the Test folder. This means our model is trained on 90% of the images and tested on 10% of images.
- Next step in the architecture involves activating TensorFlow backend (Seetala, Birdsong and Reddy, 2019) so that the computer software tool is in working conditions for the deep learning model Deeplearning.net. (2019). to take these images as input and identify the patterns for classification.
- Now we perform the second image augmentation step which is shifting the images according to defined levels or percentage (5%,10% & 15%) which creates a real-life challenge for model showcasing tilted images. Using the Keras "imagedatagenrator" library of Python which is used for image augmentation is helping in shifting the images. Custom Image Augmentation (2019).
- Finally, we input these images into the transfer learning model where we are using VGG19 Simonyan, K. and Zisserman, A. (2019) which performs with high accuracy and less operation time to create a model which we can use for testing images to get a final output accuracy of the traffic sign images.
- The output of our model after training on images is tested against the test images to get the final output which is called test accuracy and shows the levels of accuracies which can be used for evaluation and studying the patterns using tableau and confusion matrix for our TRS system.

5 Implementation

Implementation is the most important part of the research project where we will try to follow the methodology and architecture design formed from the studies done in the literature survey research papers in related work section to perform the experiment.

5.1 Environment setup

We used several tools in the research project depending on the stages of implementation. The project involves the use of python 3.7.4 version for coding in jupyter notebook which is installed for windows, TensorFlow library for implementation of deep learning model installed using anaconda prompt along with Keras which is an API on top of TensorFlow backend widely used library for image recognition and finally tableau desktop is made use for graphical evaluation and results (All Installation are explained in configuration manual).

5.2 Dataset pre-processing

The image data collected from the GTSRB Kaggle website is processed so that the model has a uniformity of the input image data for the deep learning classification model achieves best performance.

- First step in pre-processing, we used Image.ANTIALIAS functions from python PIL (python imaging library) is used for resizing the images and setting the width and height of all the downloaded images to 75*75 in JPEG format with image quality set to 2000 to standardize all images. Opensource.com. (2019).
- Next step is to blur the downloaded images using gaussian blurring on various levels 3*3,5*5 and 7*7 as we need to test the images in a bad condition and challenges using the gaussian function Pillow.readthedocs.io. (2019).

$$G(x) = \frac{1}{\sqrt{2\pi\sigma^2}} e^{-\frac{x^2}{2\sigma^2}}$$

- Gaussian blur is an image softening effect which creates a slightly blurred variety of the original image so that these images can be used by machine learning algorithms. Gaussian kernel weights are used for blurring images which is a 2D square array with the pixel values highlighting in a gaussian curve.

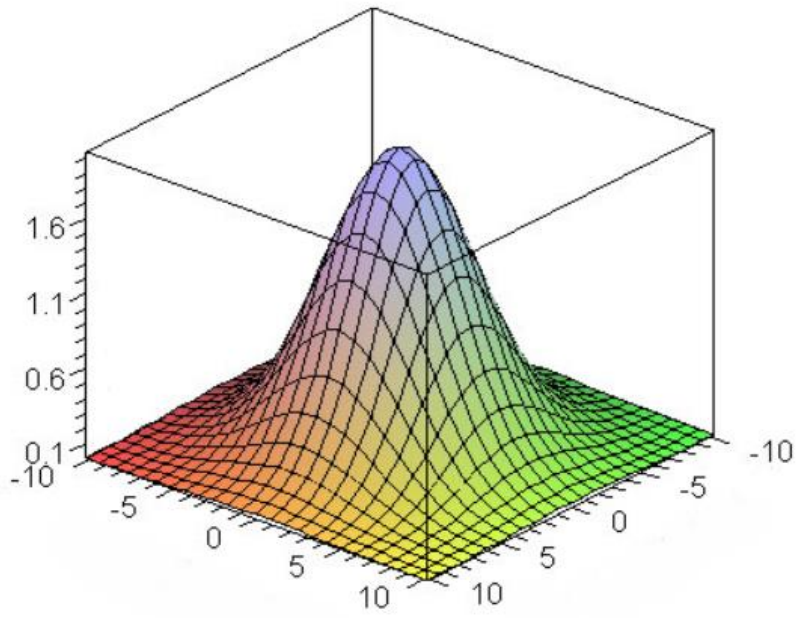


Figure 3. Graphical representation of 2D Gaussian function

$$\frac{1}{256} \cdot \begin{bmatrix} 1 & 4 & 6 & 4 & 1 \\ 4 & 16 & 24 & 16 & 4 \\ 6 & 24 & 36 & 24 & 6 \\ 4 & 16 & 24 & 16 & 4 \\ 1 & 4 & 6 & 4 & 1 \end{bmatrix} = \frac{1}{256} \cdot \begin{bmatrix} 1 \\ 4 \\ 6 \\ 4 \\ 1 \end{bmatrix} \cdot \begin{bmatrix} 1 & 4 & 6 & 4 & 1 \end{bmatrix}$$

- The images are multiplied by gaussian kernel to get the output image. The center pixel of kernel and the input images are multiplied which are overlapping. The output values after multiplication and adding up and form the destination image pixel.
- In the example figure below (0,0) is multiplied by (i), (1,0) is multiplied by (h) , (2,0) is multiplied by (g) and so on which after addition will for the resulting pixel value (1,1) as on the output image. This phenomenon working of gaussian kernel will take more time if the blurring image or region is large. Rastergrid.com. (2019).

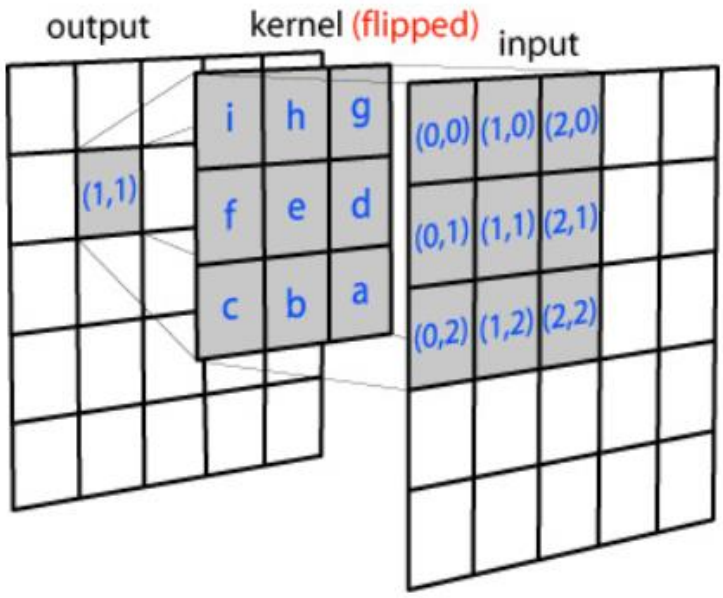


Figure 4. Gaussian blur working

- Using this phenomenon all the images are blurred and saved respectively in Test, Train and validation folders.
- As explained above the principle of gaussian blur is applied and cv2.gaussianblur() function from the OpenCV is used to apply smoothing on the input images. The implementation is highlighted in the website. Tutorial Kart. (2019) The test folder has 1476 images which are all blurred after setting the gaussian kernel size to 3,5, and 7 and saved in the test folder with respective classes names.
- The number of images in the Train folder are 11808 which are appended in their respective classes folder and saved in a list. We use a random selecting strategy to blur all the images where we are blurring every 11th image by 3*3 ,10th image by 5*5 and 9th image by 7*7 Gaussian kernel size and saving them using cv2.imwrite() function in the folders. As we can see from the Figure below the original dataset is already in bad condition with images captured in dark, blurry condition which undergo smoothing to create a challenging dataset for the deep learning model for our project. Gomes, D. et al. (2015).

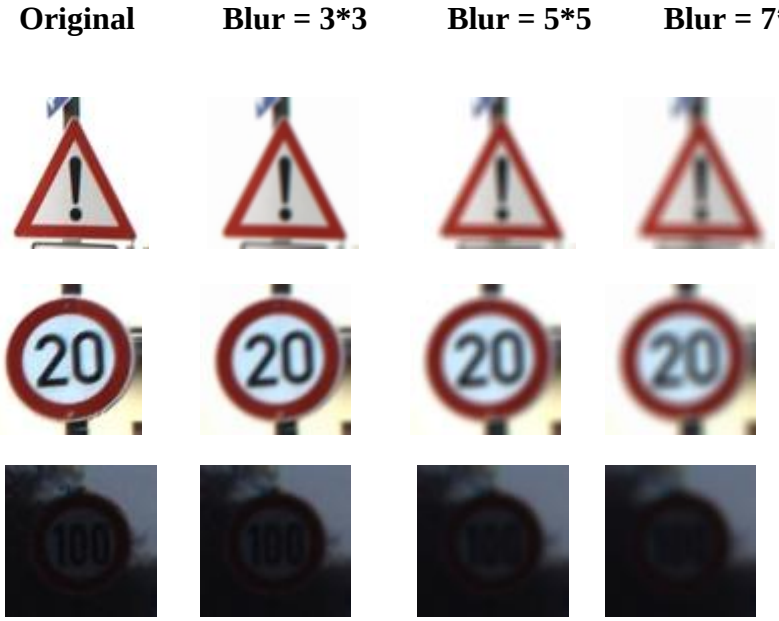


Figure 5. Blurring of images from GTSRB

- The Validation folder images are1476 which is 10% of the total dataset just like the test folder and the same strategy of smoothing the 11^{th} ,10^{th} and 9^{th} image is followed from the dataset after setting the Gaussian kernel level to 3*3,5*5 and 7*7 is done.

5.3 TensorFlow

As shared in the design specification we are working on a computer vision and a deep learning problem case which requires the use of Keras, a high-level library for transfer learning which works best on a framework of TensorFlow which is an engine developed by google team. TensorFlow. (2019).

Tensors are multi-dimensional arrays which defines, optimizes and calculates mathematical expressions being a programming support for the machine learning and deep neural network techniques employing multiple GPU and CPU computing power. (For installation of TensorFlow see configuration manual). We will be using Keras which is just a library wrapping around the google based engine TensorFlow.
Tensor flow 1.14.0 version is installed for the model implementation. InfoWorld (2019) As we know TensorFlow creates a dataflow graphs which describes the structures and defines how the data flows in a graphical manner and it's vital for deep learning model application. The benefits of abstraction offered by tensor flow will help our project with implementation of the TSR system using deep neural networking model.

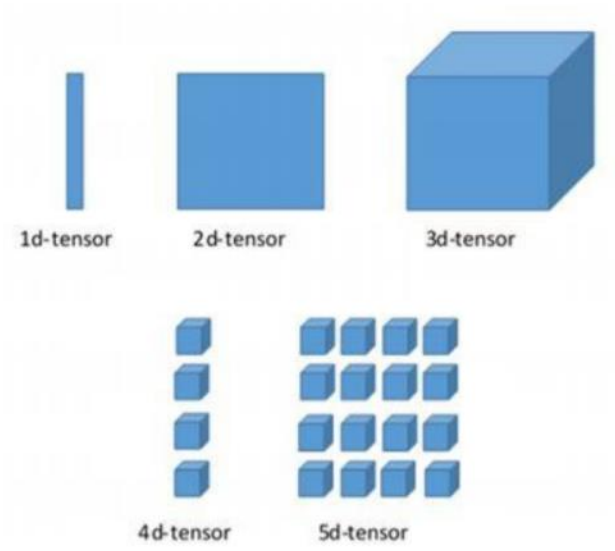


Figure 6. Different dimensional matrix of tensors

5.4 Loading the images

The Dataset contains 14760 of total images with 10 different classes. This dataset is spilt into Training dataset with 11808 images which is 80% of the entire dataset, Validation images which is 10% of total images being 1476 and similarly testing images is also 10% of total being 1476.
Also, when working with TensorFlow as backend for Keras we require a 4D array. We will input the string of images to the function which will return a 4D tensor for the Keras by first dividing the pixels by 75 times 75 for resizing the pixels. Then we will convert the images into an array which is finally resized to 4D Tensors.
We will future rescale the images as well by diving every pixel in all the images by 255.These steps are standard procedures followed for images recognition model for Keras library.

5.5 Shifting of images

Before we input the images into the Transfer learning model, we will perform the second novelty challenge as discussed earlier which is shifting the images by 3 different levels of height and width levels 5%,10% and 15% to test our deep learning models accuracy. Suki Lau (2017). We are making use of "ImageDataGenerator" to shift the images randomly from the dataset.

The traffic sign for speed 50 showcases how the images are randomly shifted from the original image just giving a challenge for the data mining model. Image Augmentation for Deep Learning with Keras (2019)

Figure 7. Shifting of speed 50 images

5.6 Using Transfer learning model

Before we explain the working of VGG19 we need to understand the difference between traditional machine learning and transfer learning models as shown in the figure the traditional machine learning methods the knowledge gained from learning from the dataset 1 and dataset 2 is kept separately and independent.
However, in case of a transfer learning model the knowledge gained from the learning of dataset 1 is utilized as knowledge while learning from dataset 2 even though the dataset 2 has limited training dataset this is a groundbreaking technological advancement and giving faster results and saves time. Beloglazov, A. and Buyya, R. (2015).

This phenomenon is best for getting Realtime results and computational accuracy for the traffic sign recognition model which is proposed in our research project.

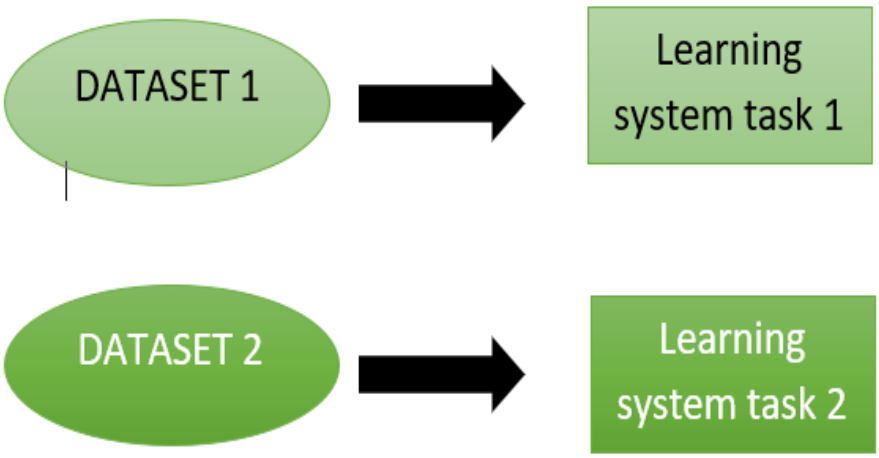


Figure 8. Traditional Machine learning model

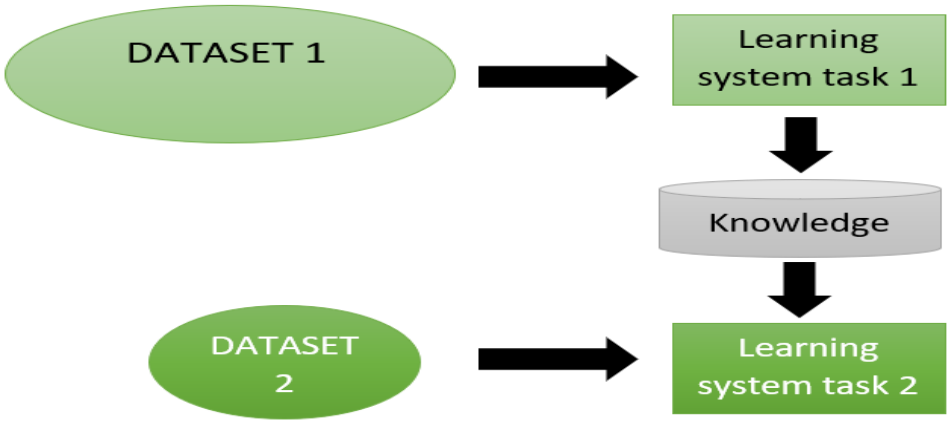


Figure 9. Transfer learning model

This explains the need and use of transfer learning model in our research project. Next step is to use this transfer learning and CNN to create a model which will identify the traffic signs from our dataset.

The model will use VGG19 which is a feature extractor. As shared in the diagram a pre-trained VGG19 model has high level of accuracy and operational time which is best suitable for our research project from the package of Keras libraries. Feng, G. and Buyya, R. (2016).

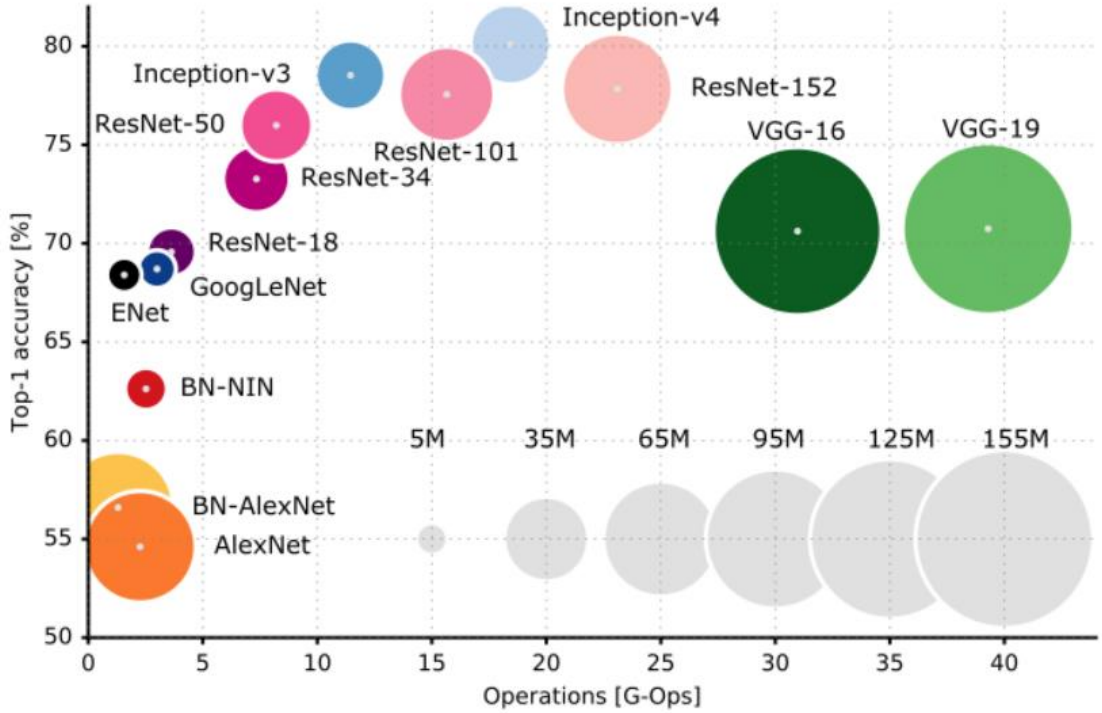


Figure 10. Comparison graph of models. Datumbox.com. (2018).

5.7 Applying VGG19 model

As the name visual geometry group, VGG19 model developed has19 weighted layers as shown in the figure which highlights the different layers of the model itself and the different

levels of the architecture. The VGG19 architecture was introduced by Simonyan and Zisserman in their 2014 paper Simonyan, K. and Zisserman, A. (2014). which is highlighted in the figure and shows the general architecture of VGG model. Zheng, Y., Yang, C. and Merkulov, A. (2018)

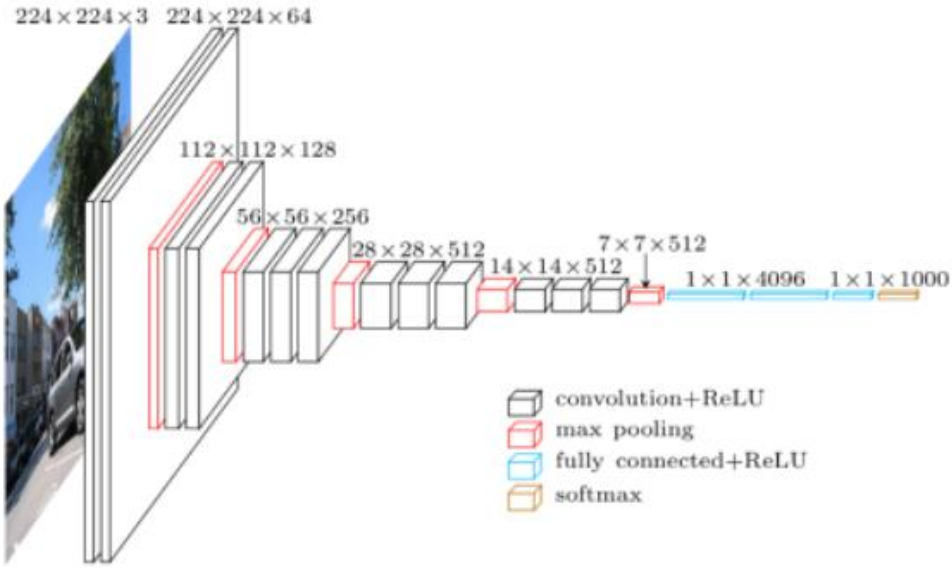


Figure 11.VGG layering architecture. Davi Frossard (2016).

- As shown in the diagram the VGG network uses input images 224 * 224 RBG format which are already converted in the pre-processing steps
- VGG model uses filter which are small for classification
- The use of max pooling and repetitive use of convolution pooling makes it a better deep learning model.
- Apart from a layer of VGG19 we are adding a dense layer in our architecture for the model.

Steps undertaken for working of the VGG19 model

1. Getting the images in Keras form the various folders in a readable format as defined by the Keras library
2. Loading the VGG model in Keras using the *load_img()* as the network expects the Prepared images in the 4D format we have converted them into 4D tensors and loaded the images.
3. Making a prediction using *predict()* which will run on all the input images.
4. Interpreting the prediction results using Keras library which returns a probability percentage using the *decopredictions (ns()* function .

5.8 Model Output

The trained model after transfer learning phenomenon and applying VGG19 model is used to predict the input images from the test folder to get an output confusion matrix result. These results show the performance of our model. As we test the model against the test images from the test dataset, we received the test accuracy. Kune, R et al. (2016).
The output of the probabilities is highlighted in the results section in form of confusion matrix.

6 Evaluation

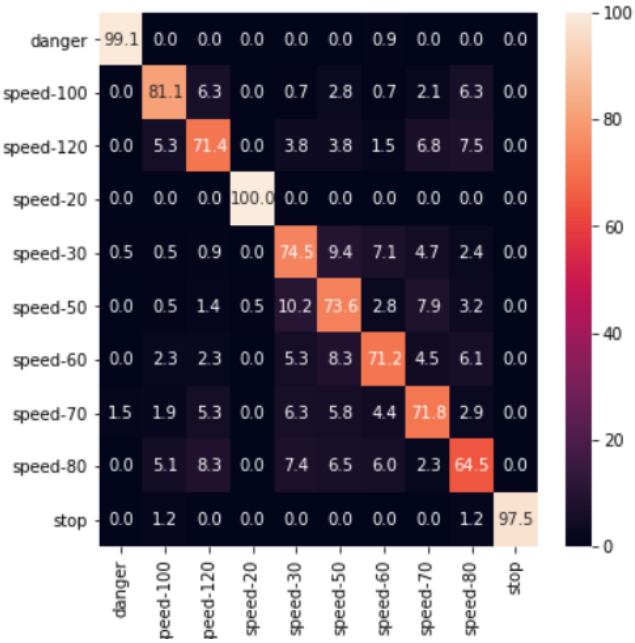
The results and Evaluation of our research is done using confusion matrix and graphical representation using Python and tableau. As we performed this experiment for different levels of shifts in images (0%,5%,10% and 15%) we received different levels of accuracy for the different blurring (0,3*3,5*5 and 7*7) of images. As we performed the experiment four times and for four levels of blur, we have formed 16 confusion matrixes as results. Image Shift level 0%

Blur level 0

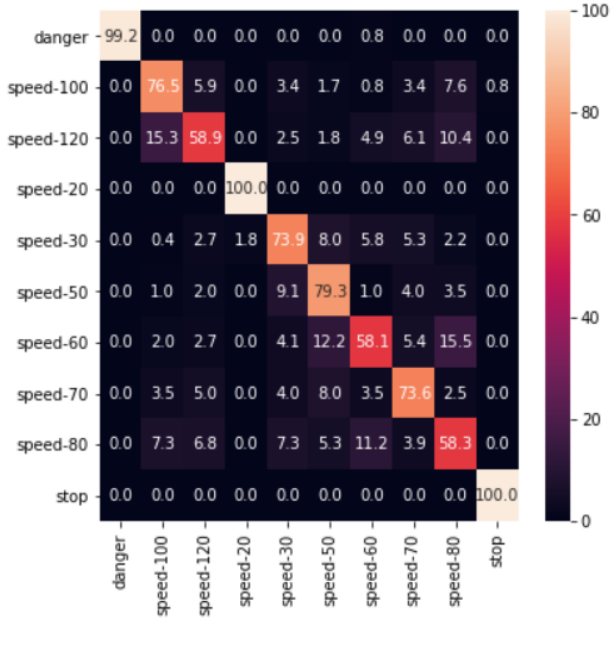
The Image shift of 0% shows 76.15% accuracy for blur level of zero which is for our GTSRB dataset in normal condition as the images are already in a bad condition.
The confusion matrix shows that the speed-20 class has 100% image recognition and accuracy for normal blur level dataset and 3*3 blur level beyond which the accuracy decreases.

Confusion matrix

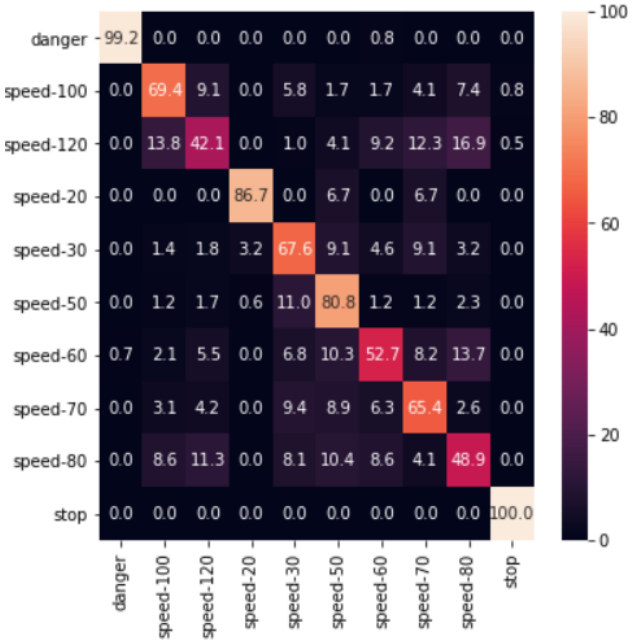
```
Test accuracy on normal
test set: 76.1518%
```



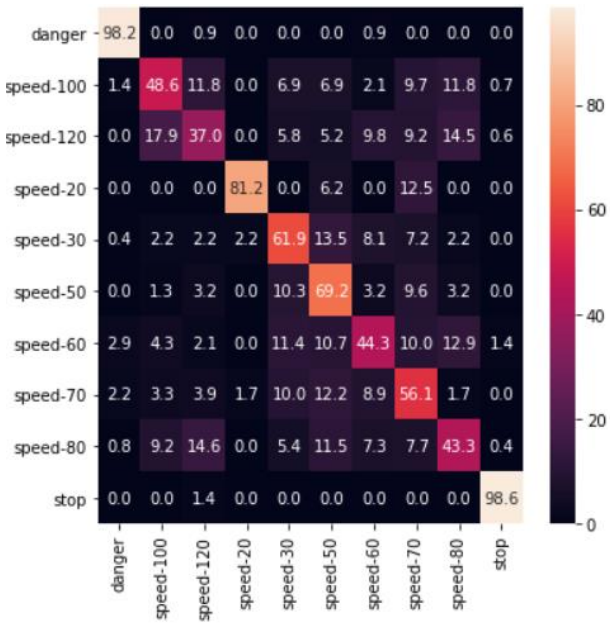

```
Test accuracy with
blurring of 3x3: 73.1030%
```



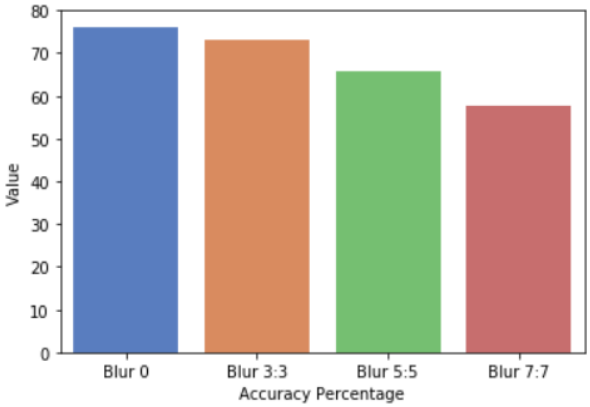
Test accuracy with blurring of 5x5: 65.7859%



Test accuracy with blurring of 7x7: 57.5203%



The below graphs show the cumulative accuracy scores of all the blur levels for the shift of level 0% on the traffic images.



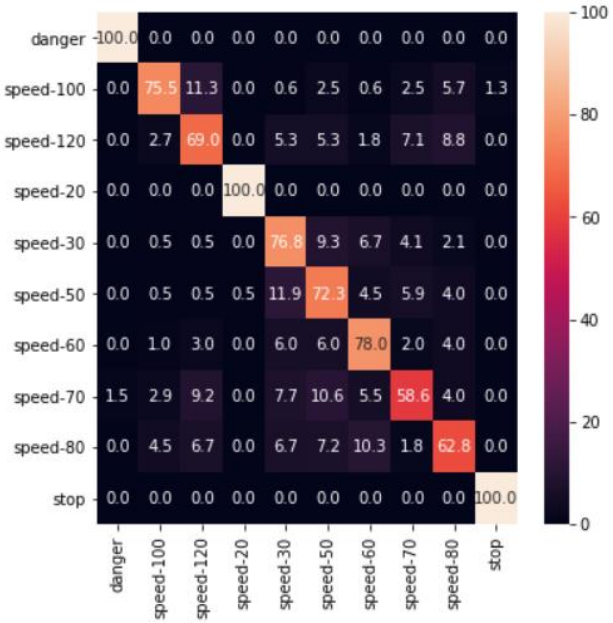
	Accuracy Percentage	Value
0	Blur 0	76.151762
1	Blur 3:3	73.102981
2	Blur 5:5	65.785908
3	Blur 7:7	57.520325

6.1 Image Shift level 5%

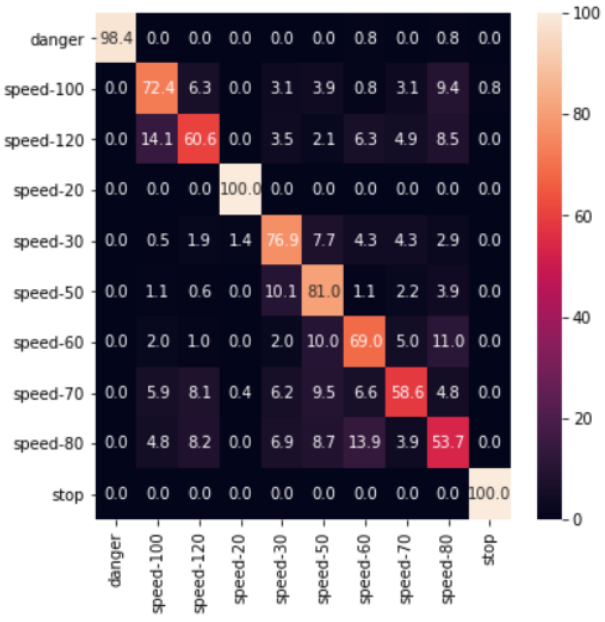
From the below confusion matrix of different classes, we can see our model works best in recognising the danger and stop signs for all levels of blur even after shift level is 5 % which shows a high level of accuracy for critical signs.

Confusion matrix

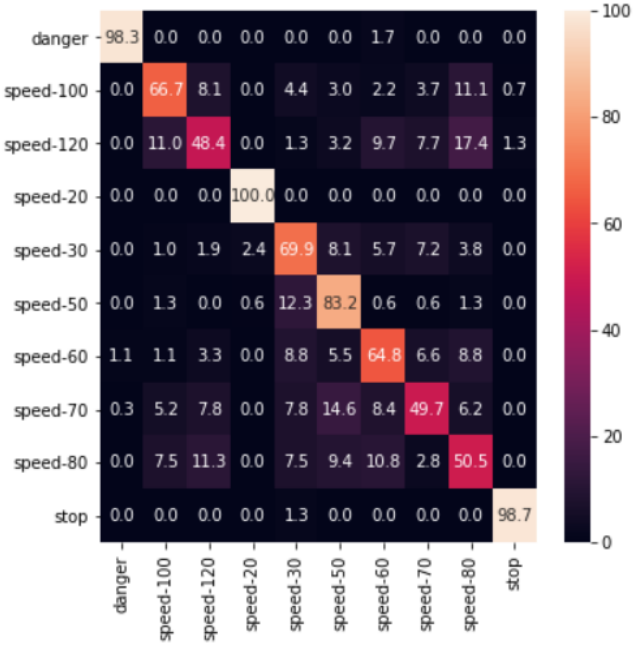
Test accuracy on normal test set: 73.3740%



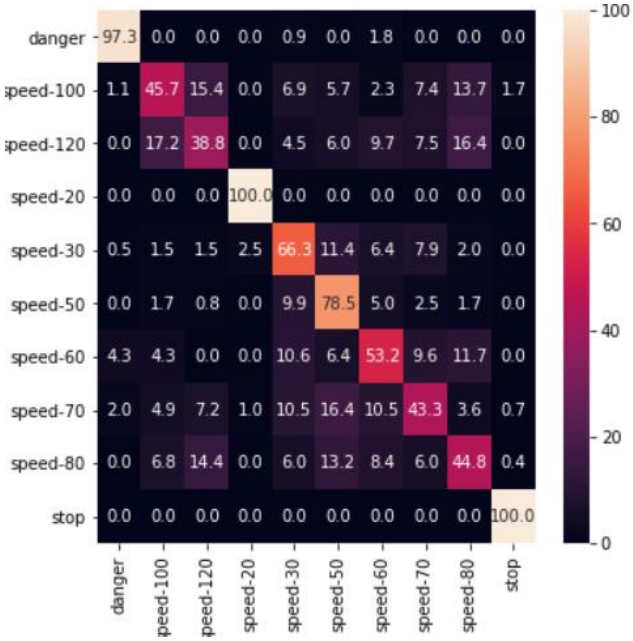
Test accuracy with blurring of 3x3: 71.1382%



Test accuracy with blurring of 5x5: 65.5149%

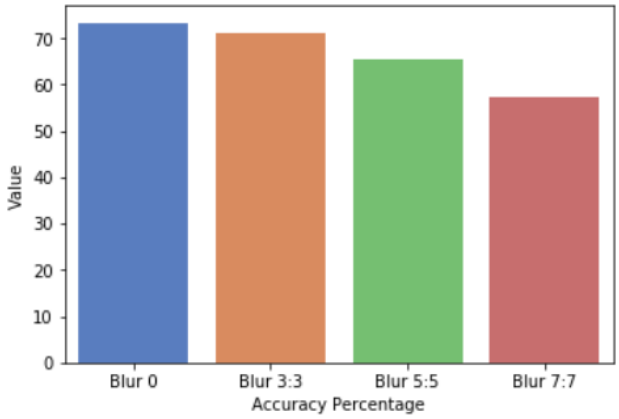


Test accuracy with blurring of7x7: 57.3848%



The below graphs show a cumulative accuracy score for the blur level 5% which is decreasing with the increase in blur level as seen the graphical representation and table.

	Accuracy Percentage	Value
0	Blur 0	73.373984
1	Blur 3:3	71.138211
2	Blur 5:5	65.514905
3	Blur 7:7	57.384824

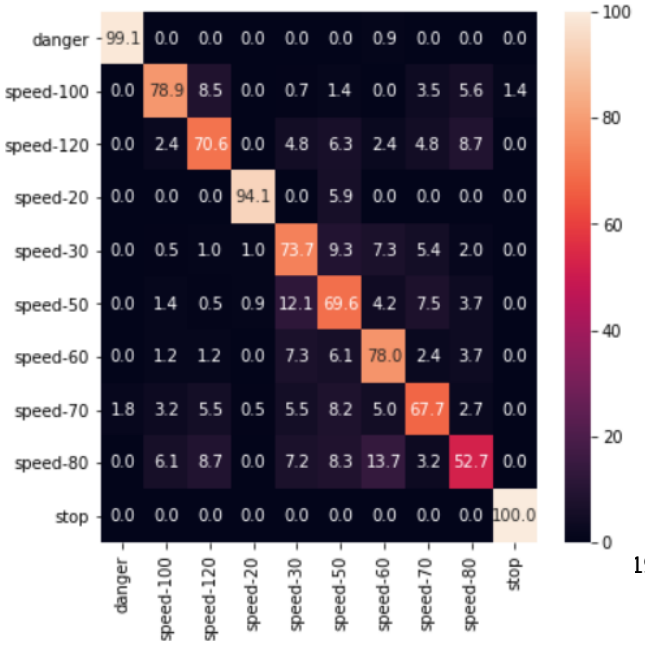


6.2 Image Shift level 10%

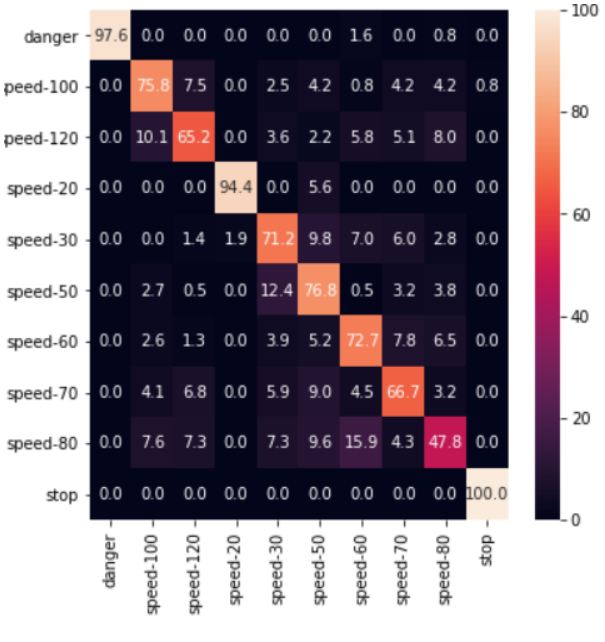
Different levels of blur and the accuracy score of the 10 classes are highlighted by the confusion matrix in for shift level 10%

Confusion matrix

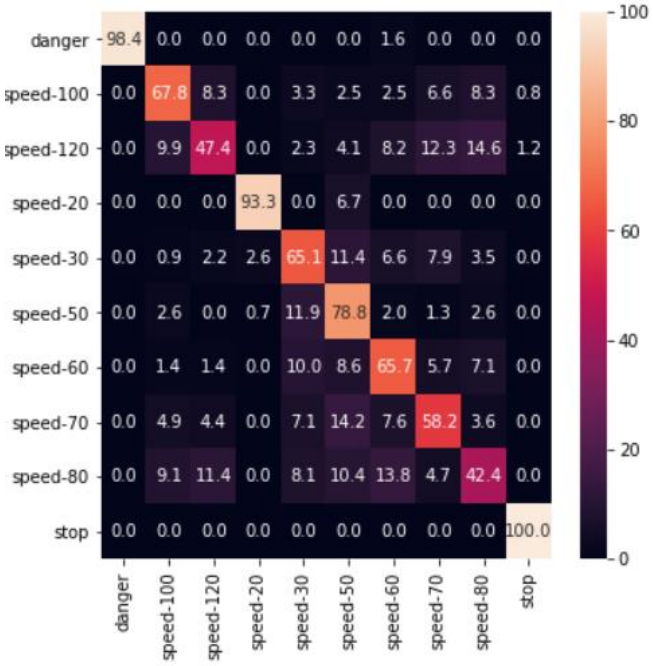
Test accuracy on normal test set: 72.3577%



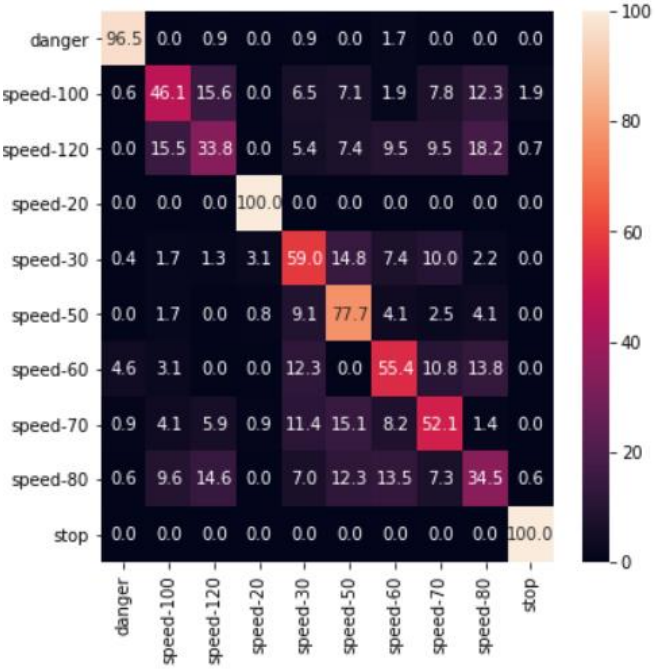
Test accuracy with blurring of 3x3: 70.3252



Test accuracy with blurring of 5x5: 63.8889%



Test accuracy with blurring of 7x7: 55.0136%

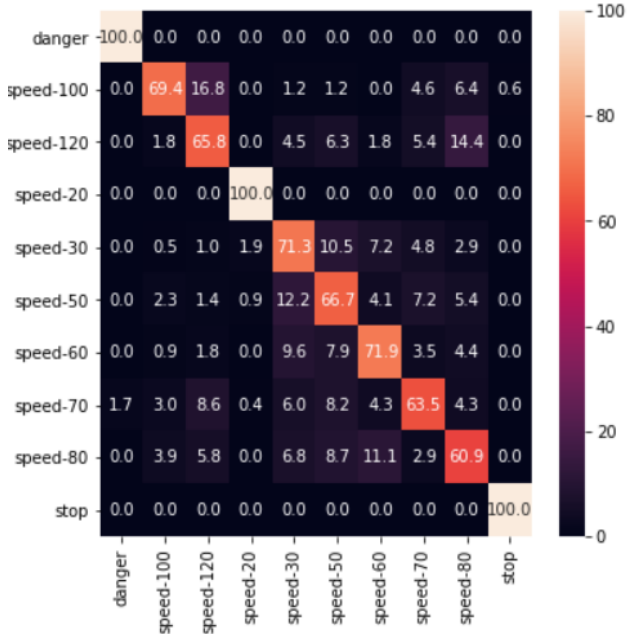


6.3 Image Shift level 15%

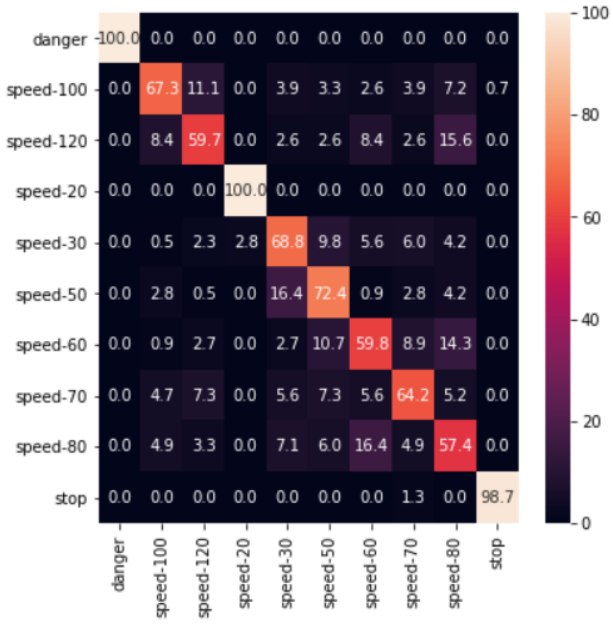
Different levels of blur and the accuracy score of the 10 classes as highlighted by the confusion matrix below for shift level 15%

Confusion matrix

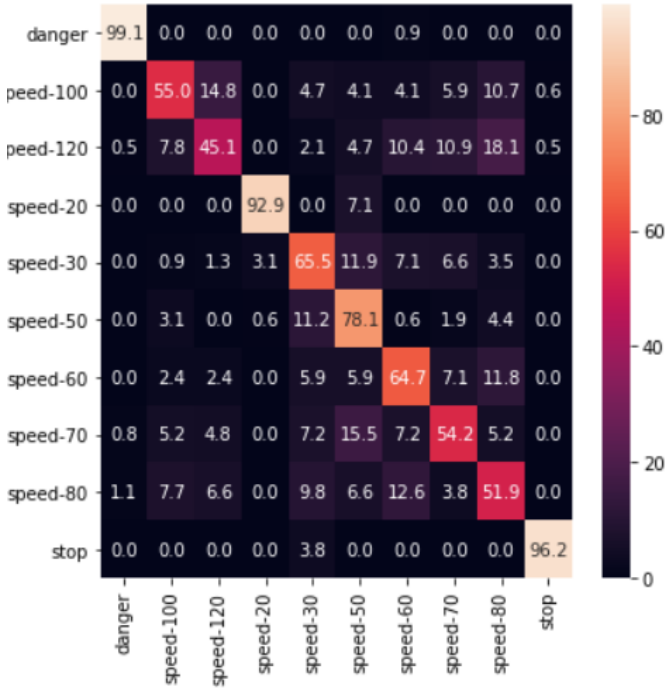
Test accuracy on normal test set: 71.3415%



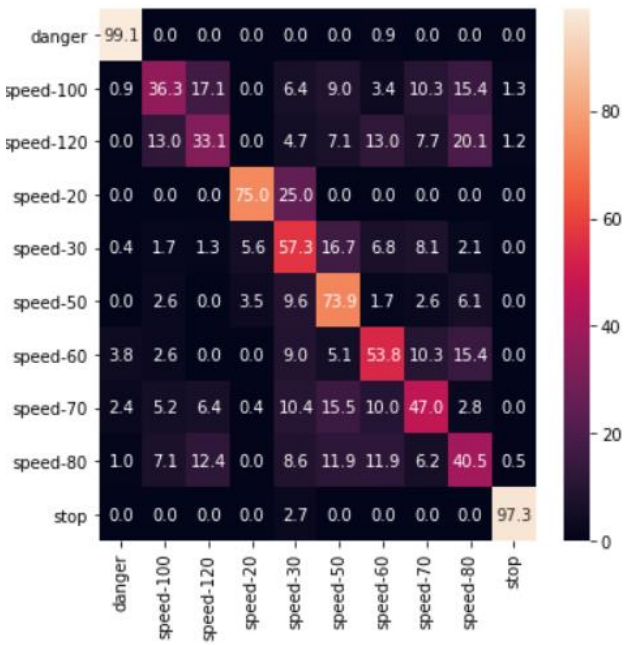
Test accuracy with blurring of 3x3: 69.8509%



Test accuracy with blurring of 5x5: 63.8889%



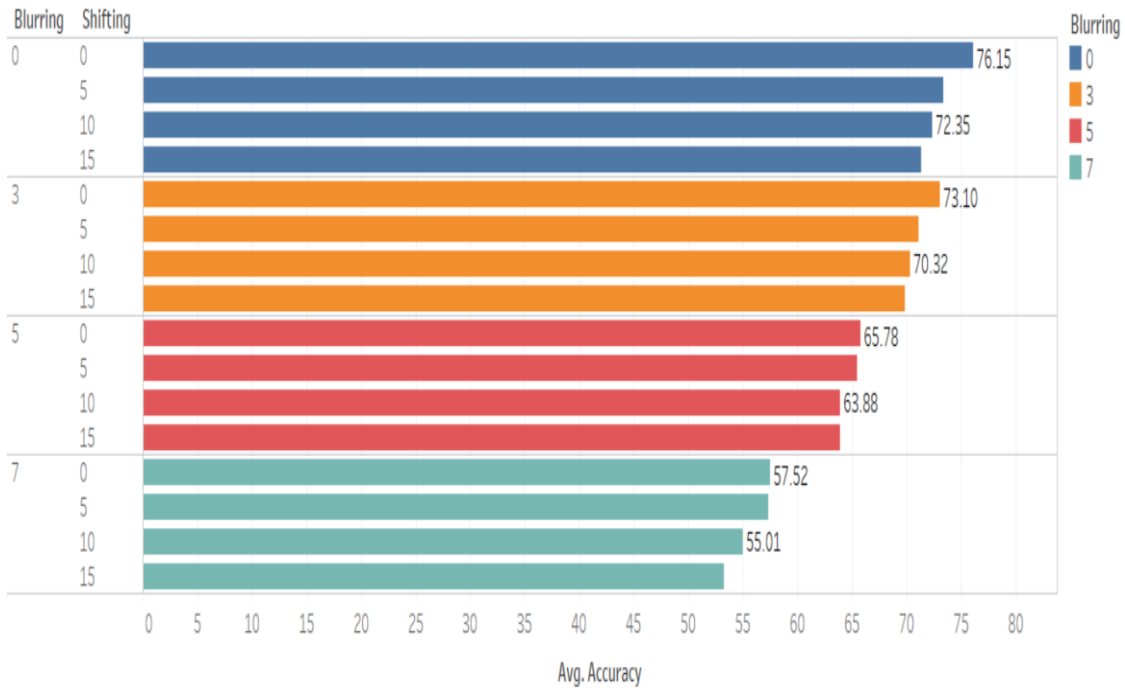
Test accuracy with blurring of 7x7: 53.2520%



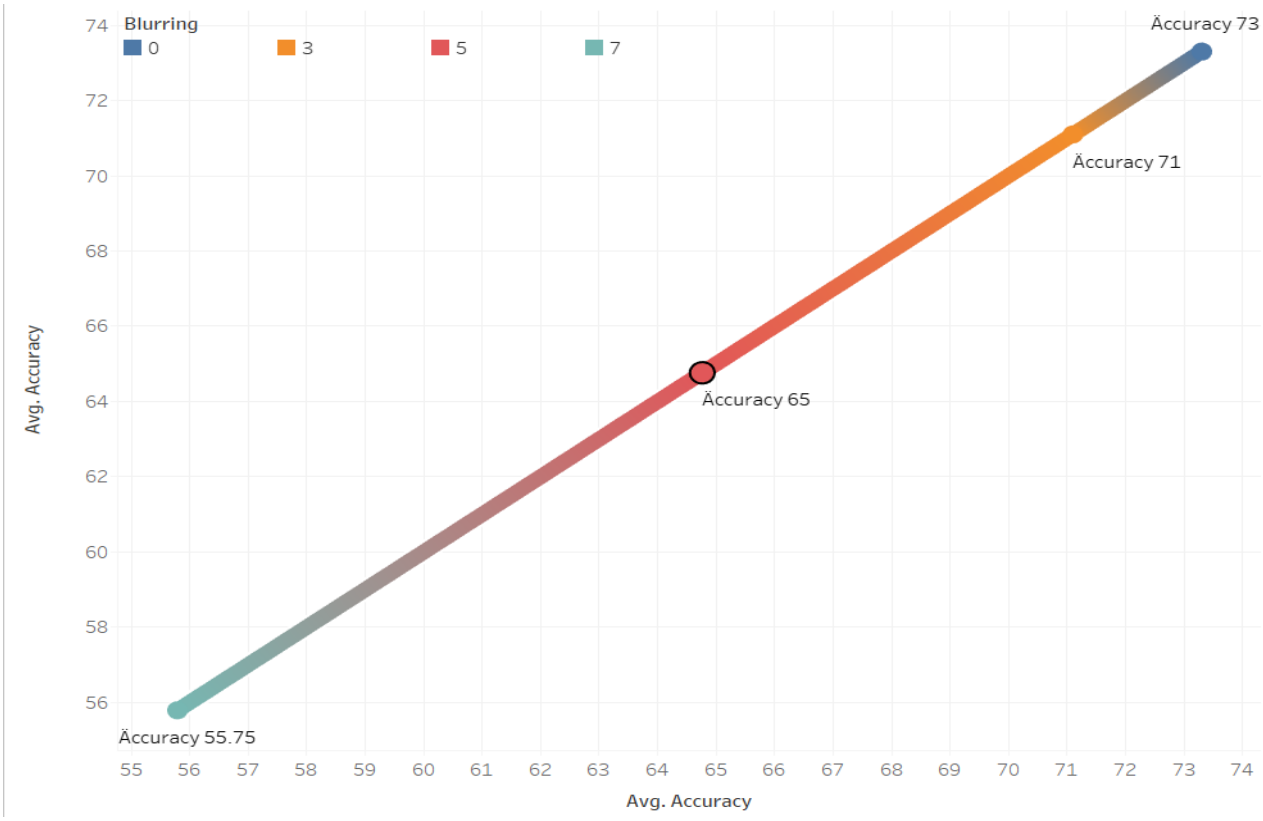
6.4 Graphical Interpretations

We will be using tableau to show relationships between accuracy, shift and blur levels.

- The graph shows a decreasing accuracy level with the increase in the blur and shift level.



- A linear relationship graphs show between the average accuracy and blur levels.



7. Conclusion and Future Work

In this research project we implemented transfer learning based VGG 19 model for traffic sign detection and classification based on the 10 classes of traffic signs. The dataset used from GTSRB already used was already in a bad condition as the images were taken from real world examples showcasing a test accuracy of 76% without any introduction of novelty of shifting and blurring of images. After blurring the traffic images to 15% and blur level 7*7 we achieved a lowest accuracy of 53% which is still a decent accuracy for poor condition of images. When we shift the images by 5%,10% or 15% respectively we saw a logical decrease in the accuracy levels which is understandable as by decreasing image quality it gets difficult for the model to predict the traffic image classes. We perform the experiment on 60 epochs and 5 batch size to save experimental time. However, if we increase the number epochs to 100, we can achieve better trained model for prediction. We also learnt that transfer learning model is highly robust and accurate even after training on a bad dataset. For future work of this project we will like to work on YOLO object detection with bad and challenging video to get Realtime prediction on video captured in extremely bad weather conditions and suggestions for safety on roads.

References

Stanford Artificial Intelligence Laboratory. [online] Available at: http://ai.stanford.edu/ [Accessed 1 Aug. 2019].

Ramler, R. and Ziebermayr, T. (2017). What You See Is What You Test - Augmenting Software Testing with Computer Vision. 2017 IEEE International Conference on Software Testing, Verification and Validation Workshops (ICSTW).

Nguyen, B. T., Ryong, S. J., & Kyu, K. J. (2014). Fast traffic sign detection under challenging conditions. 2014 International Conference on Audio, Language and Image Processing.doi:10.1109/icalip.2014.7009895

Changzhen, X., Cong, W., Weixin, M. and Yanmei, S. (2016). A traffic sign detection algorithm based on deep convolutional neural network. *2016 IEEE International Conference on Signal and Image Processing (ICSIP)*.

Mahatme, M. and Kuwelkar, M. (2017). Detection and recognition of traffic signs based on RGB to red conversion. *2017 International Conference on Computing Methodologies and Communication (ICCMC)*.

Tastimur, C., Karakose, M., Celik, Y. and Akin, E. (2016). Image processing-based traffic sign detection and recognition with fuzzy integral. *2016 International Conference on Systems, Signals and Image Processing (IWSSIP)*.

Yanjun Fan and Weigong Zhang (2015). Traffic sign detection and classification for Advanced Driver Assistant Systems. *2015 12th International Conference on Fuzzy Systems and Knowledge Discovery (FSKD)*.

Liu, C., Chang, F., Chen, Z. and Liu, D. (2016). Fast Traffic Sign Recognition via High-Contrast Region Extraction and Extended Sparse Representation. *IEEE Transactions on Intelligent Transportation Systems*, 17(1), pp.79-92.

van Opbroek, A., Ikram, M., Vernooij, M. and de Bruijne, M. (2015). Transfer Learning Improves Supervised Image Segmentation Across Imaging Protocols. *IEEE Transactions on Medical Imaging*, 34(5), pp.1018-1030.

CENGIL, E. and CINAR, A. (2018). A Deep Learning Based Approach to Lung Cancer Identification. *2018 International Conference on Artificial Intelligence and Data Processing (IDAP)*.

Chou, L., Tseng, C., Lai, M., Chen, W., Chen, K., Yen, C., Ou, T., Tsai, W. and Chiu, Y. (2018). Classification of Malicious Traffic Using TensorFlow Machine Learning. *2018 International Conference on Information and Communication Technology Convergence (ICTC)*.

Choi, S., Jeong, H., Park, K. and Ha, Y. (2019). Efficient Driving Scene Image Creation Using Deep Neural Network. *2019 IEEE International Conference on Big Data and Smart Computing (BigComp)*.

Ling Shao, Fan Zhu and Xuelong Li (2015). Transfer Learning for Visual Categorization: A Survey. *IEEE Transactions on Neural Networks and Learning Systems*, 26(5), pp.1019-1034.

Wei, L., Runge, L. and Xiaolei, L. (2018). Traffic sign detection and recognition via transfer learning. *2018 Chinese Control And Decision Conference (CCDC)*.

Bruha, I. (2000). From machine learning to knowledge discovery: Survey of preprocessing and postprocessing. *Intelligent Data Analysis*, 4(3-4), pp.363-374

Proceedings of the 25th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining. (2019). In: *KDD '19 Proceedings of the 25th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*. New York, NY, USA: ACM New York, NY, USA ©2019, p.3253.

Gertosio, C. and Dussauchoy, A. (2004). Knowledge discovery from industrial databases. *Journal of Intelligent Manufacturing*, 15(1), pp.29-37.

Mair, B., Wilson, D. and Reti, Z. (1996). Deblurring the discrete Gaussian blur. *Proceedings of the Workshop on Mathematical Methods in Biomedical Image Analysis*.

En.wikipedia.org. (2019). *Gaussian blur*. [online] Available at: https://en.wikipedia.org/wiki/Gaussian_blur [Accessed 9 Aug. 2019].

Seetala, K., Birdsong, W. and Reddy, Y. (2019). Image Classification Using TensorFlow. *16th International Conference on Information Technology-New Generations (ITNG 2019)*, pp.485-488.

Deep Learning. [online] Available at: http://deeplearning.net/ [Accessed 9 Aug. 2019].

Medium. (2019). *Custom Image Augmentation*. [online] Available at: https://towardsdatascience.com/image-augmentation-14a0aafd0498 [Accessed 9 Aug. 2019].

Very Deep Convolutional Networks for Large-Scale Image Recognition. [online] arXiv.org. Available at: https://arxiv.org/abs/1409.1556 [Accessed 9 Aug. 2019]

[gaussina blur]Resize images using Python. [online] Available at:https://opensource.com/life/15/2/resize-images-python

Pillow (PIL Fork) 3.0.0 documentation. [online] Available at: https://pillow.readthedocs.io/en/3.0.x/releasenotes/2.7.0.html

Efficient Gaussian blur with linear sampling – RasterGrid Blogosphere. [online] Available at: http://rastergrid.com/blog/2010/09/efficient-gaussian-blur-with-linear-sampling/ [Accessed 10 Aug. 2019].

OpenCV Gaussian Blur - cv2.gaussianblur() - Example. [online] Available at: https://www.tutorialkart.com/opencv/python/opencv-python-gaussian-image-smoothing/ [Accessed 10 Aug. 2019].

TensorFlow. (2019). *TensorFlow Core | TensorFlow*. [online] Available at: https://www.tensorflow.org/tutorials/ [Accessed 2 Aug. 2019].

Serdar Yegulalp (2019). What is TensorFlow? The machine learning library explained. [online] InfoWorld. Available at: https://www.infoworld.com/article/3278008/what-is-tensorflow-the-machine-learning-library-explained.html [Accessed 1 Aug. 2019].

Datumbox.com. (2018). The Batch Normalization layer of Keras is broken | Datumbox. [online] Available at: https://blog.datumbox.com/the-batch-normalization-layer-of-keras-is-broken/ [Accessed 12 Aug. 2019].
Suki Lau (2017). Image Augmentation for Deep Learning. [online] Medium. Available at: https://towardsdatascience.com/image-augmentation-for-deep-learning-histogram-equalization-a71387f609b2 [Accessed 2 Aug. 2019].

Image Augmentation for Deep Learning With Keras. [online] Machine Learning Mastery. Available at: https://machinelearningmastery.com/image-augmentation-deep-learning-keras/ [Accessed 2 Aug. 2019].

Simonyan, K. and Zisserman, A. (2014). Very Deep Convolutional Networks for Large-Scale Image Recognition. [online] arXiv.org. Available at: https://arxiv.org/abs/1409.1556.

Zheng, Y., Yang, C. and Merkulov, A. (2018). Breast cancer screening using convolutional neural network and follow-up digital mammography. Computational Imaging III.

Li, D., Zhao, D., Chen, Y. and Zhang, Q. (2018). DeepSign: Deep Learning based Traffic Sign Recognition. *2018 International Joint Conference on Neural Networks (IJCNN)*.

Acilo, J., Dela Cruz, A., Kaw, M., Mabanta, M., Pineda, V. and Roxas, E. (2018). Traffic sign integrity analysis using deep learning. *2018 IEEE 14th International Colloquium on Signal Processing & Its Applications (CSPA)*.

Lamouik, I., Yahyaouy, A. and Sabri, M. (2018). Deep neural network dynamic traffic routing system for vehicles. *2018 International Conference on Intelligent Systems and Computer Vision (ISCV)*.

Davi Frossard (2016). VGG in TensorFlow · Davi Frossard. [online] Toronto.edu. Available at: https://www.cs.toronto.edu/~frossard/post/vgg16/ [Accessed 12 Aug. 2019].

Beloglazov, A. and Buyya, R. (2015). Openstack neat: a framework for dynamic and energy-efficient consolidation of virtual machines in openstack clouds, *Concurrency and Computation: Practice and Experience* 27(5): 1310–1333.

Feng, G. and Buyya, R. (2016). Maximum revenue-oriented resource allocation in cloud, *IJGUC* 7(1): 12–21.

Gomes, D. G., Calheiros, R. N. and Tolosana-Calasanz, R. (2015). Introduction to the special issue on cloud computing: Recent developments and challenging issues, *Computers & Electrical Engineering* 42: 31–32.

Kune, R., Konugurthi, P., Agarwal, A., Rao, C. R. and Buyya, R. (2016). The anatomy of big data computing, *Softw., Pract. Exper.* 46(1): 79–105.