# A Thorough Evaluation of Kernel Order in CNN Based Traffic Signs Recognition

Lucas Armand Souza Assis de Oliveira[1], Guilherme Lucio Abelha Mota[1], Vitor da Silva Vidal[1]
[1]Post-Graduation Program in Computational Sciences, Rio de Janeiro State University, Brazil

***Abstract*—Convolutional Neural Network is an important deep learning architecture for computer vision. Alongside with its variations, it brought image analysis applications to a new performance level. However, despite its undoubted quality, the evaluation of the performance presented in the literature is mostly restricted to accuracy measurements. So, considering the stochastic characteristics of neural networks training and the impact of the architectures configuration, research is still needed to affirm if such architectures reached the optimal configuration for their focused problems. Statistical significance is a powerful tool for a more accurate experimental evaluation of stochastic processes. This paper is dedicated to perform a thorough evaluation of kernel order influence over convolutional neural networks in the context of traffic signs recognition. Experiments for distinct kernels sizes were performed using the most well accepted database, the so-called German Traffic Sign Recognition Benchmark.**

***Keywords*—autonomous vehicles, CNNs, kernel size, statistical evaluation**

## I. Introduction

Autonomous vehicles is a major trend that will change the paradigm of goods and people transport [1]. Currently, one of the main technological challenges related to autonomous vehicles is the correct perception of external environment [2]. Computer vision is a powerful tool that allows the autonomous system to “understand” the world around. One of the most popular machine learning techniques is Neural Networks [3]. Recently, deep networks have experimented a fast evolvolution and specialization in complex problem-specific designs. In the case of image classification, a particularly effective architecture is the Convolutional Neural Network (CNN) [4].

A problem of particular interest for the autonomous vehicles application in the area of computer vision is classifying traffic signs. We reviewed a wide bibliography on this topic (section II), and found a variety of solutions and architectures proposed for this application. However, very few papers seek to present general conclusions about the best configuration of the proposed neural networks architectures (e.g. amount of filters, number of layers and filter order) for the specific application. Even those who focus on doing multiple tests varying particular architecture have little or no statistical analysis to substantiate the results. For instance, [5] compares the performance of five distinct architectures for traffic signs detection without tuning networks configuration. Authors, indeed, combine any features extractors with distinct base architectures, nevertheless, all components are treated as closed modules without revealing the examination of such “black-boxes”. Another example is [6], which employs genetic algorithms just in the context of obtaining the optimal learning ratio and number of epochs to be used for training a CNN.

One exception is [7], that analyses the outcomes of distinct kernel sizes in a CNN model in the context of traffic sign recognition. However, little or no discussion is made about the character intrinsically stochastic of the learning process and the randomness of the neural network parameters initialization. As usual, comparisons are restricted to the model’s accuracy and lack the statistical significance evaluation of results.

This work proposed a review of Sichkar and Kolyubin paper [7] in the direction of finding the best kernel size in a traffic signs recognition application using a CNN architecture. It aims at validating the results presented by the original paper, while proposing a more in-depth statistical analysis underlain by a broader set of experimental results. The objective of this work is not to advance the state of the art in terms of classification accuracy in the benchmark used, but to point out the need for more robust statistical analysis to reach general conclusions. The results presented here invalidate the conclusions in their original article. This may be explained by an inadequate experimental design.

This paper is organized as follows. In Section 2, a detailed theoretical review of the application of neural networks to the problem of interest is presented. In Section 3, the techniques used and the way the results were produced will be presented. In Section 4, experimental results are evaluated. And finally, section 5 analyses conclusions and perspectives for future work.

## II. Previous Works

Traffic sign detection, tracking and recognition are important issues concerning autonomous and assisted driving, signaling inventory and quality control. This section brings a review of traffic sign detection and recognition approaches as well the most used public benchmarks. Similarly to other computer vision applications, a number of recent researches in these fields are based on deep learning architectures.

In therms of public databases, the most remarkable one is The German Traffic Sign Recognition Benchmark (GTSRB) [8] which contains 51,839 images and 43 classes. GTSRB has a compatriot dedicated to sign detection [9], the German traffic sign detection benchmark (GTSDB), containing 900 images with the corresponding signaling bounding boxes annotations. A more recent database, the The European traffic sign dataset (ETSD) assembles several European public available datasets: from Belgium, the KUL Belgium Traffic Signs dataset [10];

from Croatia, the MASTIF datasets [11]; from France, the Stereopolis dataset [12]; from Germany, the above mentioned GTSRB [8]; from Netherlands, the RUG Traffic Sign Image-Database [13]; and from Sweden, the Swedish Traffic Signs Dataset [14]. ETSD amounts 82,476 images of 164 classes.

### A. Detection

An example of direct detection traffic sign detection is presented in [15]. This approach relies on the Single Shot Detector (SSD) architecture [16], basically, a feed forward CNN, which produces predictions on the position and class of target objects. The predicted bounding box position is then submitted to 2D Pose Prediction which fits the box to the quadrilateral which best adjusts to the traffic sing. The method ends up with a boundary corner estimation process that produces based on the sign class shape an accurate boundary for such occurrence. The presented experiments, in terms of SSD architecture adaptation, is limited to reducing computational complexity in order to accomplish processing time requirements for the application, permitting a low-power mobile platform to reach 7 FPS. Song et al. [17] proposed an efficient CNN which remarkably minimize the redundancy, downsize the parameters set and speed up the networks. So, it reduces its computational cost, achieving 833 ms per frame on a 2048 × 2048 px image.

An region proposal approach is presented in [18]. The proposed deep detection network is composed of four modules. Firstly, CNN layers that compute features. In parallel, the so-called attention network, which makes a rough detection, is a color segmentation module, exploiting intrinsic properties of signs. The third module employs a fully convolutional network to produce the final regions proposals. The last module is an improved Fast Region-based Convolutional Network (Fast R-CNN), functioning as a detector (classifier and regressor) and synthesizing information from the remaining modules. In the experiments, the method is compared with other approaches, without concerning optimizing the internal architecture. In the most successful experiments using a GPU equipped computer, produced a 7.8 FPS for input frames of 1024 × 800 px.

A combination of image analysis and pattern recognition techniques for traffic sign detection dedicated to mobile systems is presented in [19]. The method is based on complementary interest regions extraction approaches relying on color and shape which follow a preprocessing stage which enhance traffic sign regions and fade background. The candidate regions provided by the interest region detectors are then classified as either traffic sign or background by a Support Vector Machine (SVM) using Histograms of Oriented Gradient (HOG) features. Regions claimed as signs are then filtered in order to eliminate false positives.

An adaptive color method for sign detection method based on adaptive color threshold is presented [20]. First an adaptive segmentation threshold is calculated using the cumulative distribution function of the image histogram. Afterwards, an approximate maximum and minimum normalization method is used to suppress the interference of high brightness and background areas. Results are submitted to a shape symmetry detection algorithm based on statistical hypothesis testing. The experimental evaluation on the GTSDB obtained an accuracy which exceeded 94%.

A method for detection and classification of traffic sign is presented in [21]. Roughly speaking, the method can be split up into color based ROIs segmentation and shape classification. While K-means and an area-based filter are exploited for ROIs extraction, shape classification extract pyramids of HOGs which are discriminated by a SVM.

### B. Traffic Signs Recognition

A number of scientific studies in the literature are dedicated to traffic signs recognition. Their performance comparison is easier when they use the GTSRB, the widest spreading traffic signs recognition benchmark. [22] present and evaluate the use of Spatial Transform Network (STN) and CNN. The most successful assemblage was STN-CNN-STN-CNN-STN-CNN consisting of more than 14 million of parameters which achieve an accuracy 99.71%. The deep learning architecture that won the contest in the IJCNN 2011 [23] is presented in [24]. It consists of a committee of 25 CNNs, encompassing approximately 38.5 million of parameters and achieving 99.46% accuracy. Each one of the 25 CNNs parameters are initialized randomly, five well-known image enhancement techniques are presented to the input of five specialized CNNs. Outputs of each CNN relative to each class are democratically averaged producing the outcome of the so-called Multi-Column Deep Neural Network. The use of Multiscale-CNNs was proposed in [25], concerning on a two stages CNNs in which the output of the first stage is also presented, after an additional pooling, to the fully connected layer, conveying a multi-scale feature representation. Authors present some variations of architectures, the most successful consisting in receiving only a gray level image as input which obtained 99.17% accuracy on GTSRB while having 1,437,791 parameters to be trained.

A traffic sign recognition approach based on a combination of complementary and discriminant feature sets containing HOG, Gabor features and Compound local binary pattern is proposed in [26]. The method used a extreme learning machine (ELM) network as classifier. The results of the experimental work concerning the GTSRB reached 99.10% of accuracy. A similar approach using SVM [27] achieved 97.04%. An approach based on robust traffic sign image descriptor, consisting on a variant of HOG, and sparse classifiers is presented in [28]. The method provided 98.17% of accuracy on GTSRB.

## III. Methodology

As previously introduced, in this work will be made a review of [7], so we implement the same architecture of CNN, but we plan our experiments to enable a more thorough statistical analysis of the outcomes.

### A. Convolutional neural network architecture

The herein presented CNN is composed of a convolutional layer, a layer of dimensionality reduction (pooling), one hidden layer and the output layer. A $3 \times 3 \times 3$ version of the standard

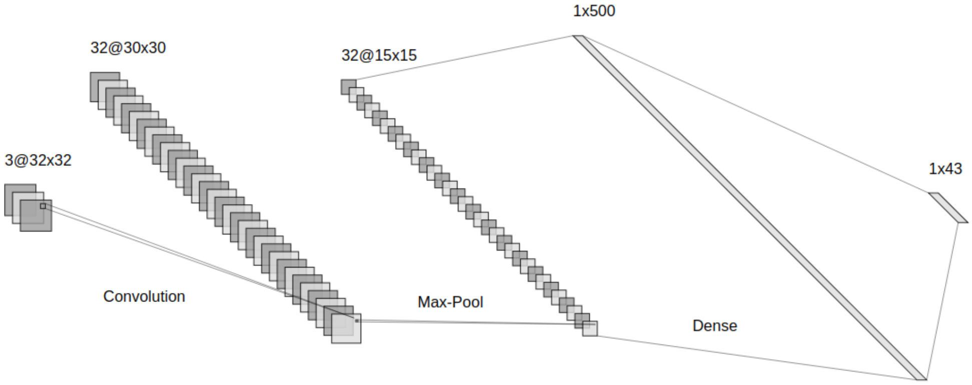

Fig. 1. CNN architecture example with a 3x3 kernels' size.

TABLE I
CNN SPECIFICATIONS

| Parameters | Description |
|---|---|
| Weights Initialization | HE Normal |
| Weights Update | Policy Adam |
| Activation Function | ReLU |
| Pooling | 2 x 2 Max |
| Loss Function | Negative log-likelihood |
| Cost Function | Average of Loss Functions |
| Stride for Convolution Layer | 1 |
| Stride for Pooling Layer | 2 |

convolutional neural network applied for the problem of traffic signs recognition is presented in Fig. 1. The architecture receives a $32 \times 32$ RGB input image which is submitted to by 32 $N \times N \times 3$ filters, where distinct values for $N$ are to be evaluated in the experiments. Filtered maps are, then, processed by a rectified linear unity activation function followed by a $2 \times 2$ max pooling. Remaining maps are fully connected to a hidden layer with 500 neurons which in turn are connected to the 43 neurons on the output layer, accordingly to the number of classes in the dataset. Table I presents the functions and some other specific characteristics used in such CNN implementation.

## *B. Statistical analysis*

In order to compare multiple models in machine learning Pizarro [29] proposes two approaches: Parametric Analysis and non-parametric analysis. However, as [30] points out, parametric analyzes (e.g. ANOVA) is based on assumptions that the samples are drawn from normal distributions and, in general, there is no guarantee for normality of classification accuracy distributions across a set of problems. Therefore, in this work, a non-parametric analysis of the accuracy of the models will be made.

Dietterich [31] proposes tests based on $5 \times 2$ cross validation as a strategy that counterbalances the need for multiple runs, while avoiding overlapping test sets for each round (which inflates the hypothesis of independence between runs). Otherwise [29] proposes thirty rounds of execution with re-shredding of the data, however with multiple executions every time to deal with outliers.

The nonparametric approach consists of transforming each round of execution, in relatively ranked results. So the best result (highest accuracy and / or lowest error rate) is "the first", that is, receive rank 1. Similarly, rank two, three, four, etc. and so on to the other results are assigned for each of the thirty repetitions.

Initially we tested the hypothesis that all the algorithms were equivalent and that the difference between results in each round is due to nothing more than luck. If this is true, no algorithm should perform better than another consistently, that is, if this hypothesis is true, even if in some cycle, one of the algorithms is better than another, in general, the average rank of all of them must be the same. For this we use the Friedman's [32] test.

In Equation 1, be $r_{ij}$ the rank of the j-th of k algorithms on the i-th of N data sets. The Friedman test compares the average ranks of algorithms $\frac{1}{n}\sum r_{ij}$, about the null hypothesis, which states that all the algorithms are equivalent and so their average ranks should be equal. Being $k$ the number of models and $n$ the total number of rounds of execution of the models.

$$\chi_F^2 = \frac{12n}{k(k+1)} \left[ \sum_{j=1}^{k} \left( \frac{1}{n} \sum_{i=1}^{n} r_{ij} \right)^2 \right] - 3n(k+1) \quad (1)$$

If it is possible to falsify the hypothesis that all algorithms have an equivalent accuracy, the question arises which algorithm is better and how they can be compared with each other. In this problem [30] proposes the use of tests that avoid two-by-two comparisons, for this purpose we use the Nemenyi Test [33] to calculate the difference between the averages of rank and compare them with a "critical difference" (CD). The CD tests if there is statistical significance to affirm that there is a difference between the accuracy of the methods. Equation 2 presents the calculation of the critical difference by the Nemenyi method:

$$CD_F = z_{adj} \sqrt{\frac{nk(k+1)}{6}}, \quad (2)$$

where $n$ and $k$ are the same as in Eq. 1 and the value of $z_{adj}$ is obtained from the table of the Normal Distribution [34] and will be a function of Type I error rate that will be tolerated by researches.

## IV. Experiments

### A. Experiments design

The dataset used for training and evaluating the CNN performance in this work is [35] which is the same employed in [7]. Broadly speaking, it is a pre-processed derivation of the GTSRB [8], with insertion of artificially generated data to balance the number of available elements of all classes. In the following experiments, the dataset was divided between training, validation and test data, being respectively 86,989, 4,410 and 12,630, as in the reference article. However, differently from [7], each network configuration was trained from the scratch for 30 times.

The purpose of the experiments is evaluating the influence of a specific CNN parameter value (in this case, the kernel size of the convolutional layer) on the accuracy of the network. The choice of accuracy as an analysis parameter was to allow comparisons with the reference article [7]. The analyzed models have kernel sizes $3 \times 3$, $5 \times 5$, $9 \times 9$, $13 \times 13$, $15 \times 15$, $19 \times 19$, $23 \times 23$, $25 \times 25$ and $31 \times 31$. As the database images are RGB $32 \times 32$ px, these kernel sizes were varied from the smallest possible $3 \times 3$ to close to the maximum possible $31 \times 31$ and are in accordance with the reference article.

### B. Results and Analysis

Fig. 2 shows the results of the thirty executions' accuracy for each model. The boxplot show the mean, standard deviation, minimum and maximum acuracy.

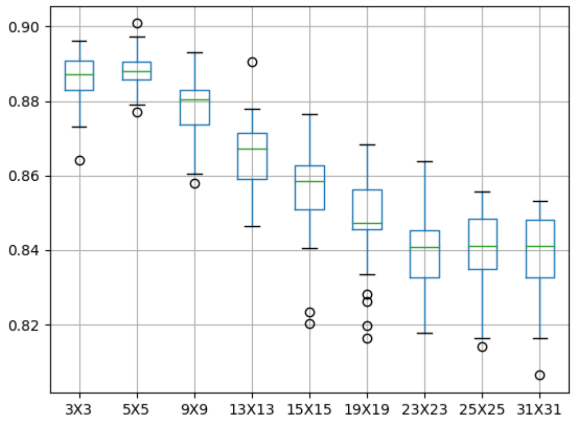

Fig. 2. BoxPlot of Model's Accuracy.

Is already possible to notice that there is a great variation between the results of the models. Models with better kernel size obtain better results, especially models with kernel size $3 \times 3$ and $5 \times 5$, with the latter having the highest mean accuracy.

TABLE II
Average Accuracy Table

| Average Accuracy | | | | |
|---|---|---|---|---|
| **3X3** | **5X5** | **9X9** | **13X13** | **15X15** |
| 0.886685 | 0.888390 | 0.878694 | 0.866030 | 0.856490 |

| Average Accuracy | | | |
|---|---|---|---|
| **19X19** | **23X23** | **25X25** | **31X31** |
| 0.847786 | 0.840878 | 0.839751 | 0.839692 |

Table II shows the average accuracy in each model. Comparing these results with those presented in the reference paper [7] it is possible to notice that, with the exception of the accuracy for 3x3 and 5x5, the average accuracy presented by Sichkar and Kolyubin paper fit inside of the distance of two standard deviations from the mean accuracy obtained in our experiments.

To perform the non-parametric test of the null-hypothesis that all the models are equivalent, we must rewrite the results in terms of the relative rank they obtained in each round [36]. In this way, the best result (the most accurate) receives rank 1, the second highest accuracy receives rank 2 and so on until all methods are ranked in each round for all rounds. For all of the thirty rounds, each method will receive a rank between 1 and 9.

Fig. 3 presents the histogram of the ranking results as discussed previously.

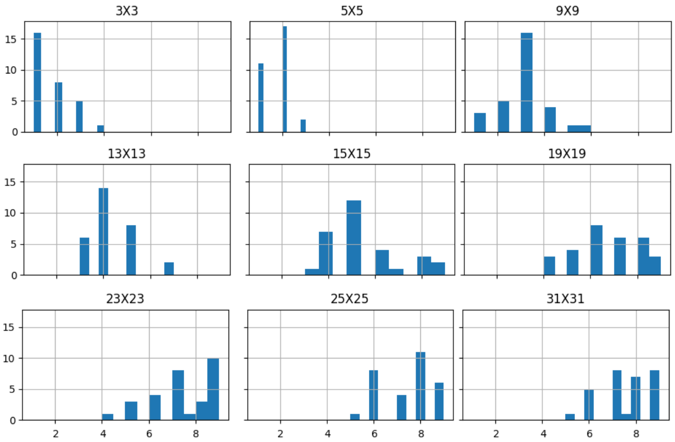

Fig. 3. Histogram of Rank of Models.

It is possible to notice that some methods were ranked with non-integer values. This occurs when two methods achieved exactly the same result in one run, so they were given the average between the ranks (e.g., instead of both being classified as seventh place, or both being classified as eighth place, the two received rank "7.5" and the other methods are classified regardless of what happened).

When comparing the results of the ranks' histograms with the boxplot, it is noteworthy that the model with a $3 \times 3$ kernel seems to have better rank results than the $5 \times 5$ model, even though the second has a higher average and a smaller variance

TABLE III
AVERAGE RANK

| Average Rank | | | | |
|---|---|---|---|---|
| **3X3** | **5X5** | **9X9** | **13X13** | **15X15** |
| 1.7 | 1.7 | 2.3 | 4.27 | 5.47 |

| Average Rank | | | |
|---|---|---|---|
| **19X19** | **23X23** | **25X25** | **31X31** |
| 6.57 | 7.35 | 7.43 | 7.55 |

around the average. This can be explained because the rankings depend not only on the accuracy of the model in each round, but also on how this accuracy is compared to the other models in the same round. So, when we represent the results in terms of rank, part of the correlation between the accuracy of the models becomes explicit, which is not possible to notice when we look only at the accuracy distribution of each model individually.

Table III shows the average rank of each model.

From the ranks averages we can have a good understanding of how the models perform in relation to others.

To test the null hypothesis (what explains the variation between the data is luck) we will use the chi-square [34] (Table A4). Table of the Chi-Square Distribution for p-value of 0.99 (with 8 grades of freedom) we have $\chi^2_{0.99} = 20.09$. Equation 1 presents the chi-square estimation for our problem. Calculating for $n = 30$ and $k = 9$ we get $\chi^2_F = 187.49$, that is, the result of this calculation shows us that the null hypothesis can be rejected.

At that point we initiate the post hoc analysis. Equation 2 presents the calculation of the critical difference by Nemenyi's test [33]. Using values proposed by [34], assuming the per comparison Type I error rate ($\alpha_{PC}$) of 0.05, we will use a $z_{adj} = 2.39$. The Equation 2 result $CD = 50.7$. If we normalize the CD by the number of replications we can directly compare the value with the average rank of each model [30]. So our normalized critical difference is $CD/n = CD_n = 1.69$.

The idea behind Neyemin's test is that when performing multiple independent 2-by-2 tests the probability that at last one of them, by chance, results in a false positive increases exponentially with the number of models. The critical difference is a factor that already considers the number of models to be compared and, instead of conducting all paired t-tests (e.g., in our case $9 \times 8/2 = 36$ comparisons), we can compare all the differences between ranks models with CD to determine if the difference between models' results has statistical significance.

Fig. 4 presents the critical difference as a distance, placing all average ranks in a "ruler" for comparison. If the size of the difference between the average rank of two models is greater than CD, then the hypothesis that they are equivalent can be rejected, otherwise, there is no statistical significance in the difference between the results, so this test says nothing about these models.

The image shows a spatial perspective about the relationship between the average rank of models and the calculated CD.

Finally, a Table IV with the final results of the comparisons between the models. We can see that the statistical analysis indicates which are the best models with $3 \times 3$, $5 \times 5$ and $9 \times 9$ kernel sizes and that we cannot show that there is a statistical difference between them. Our conclusions differ from the reference article [7] since our results indicates smaller kernels provides a greater accuracy for a CNN with this architecture.

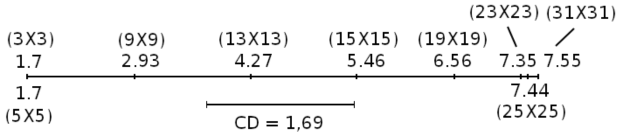

Fig. 4. Average Ranks dispose in a ruler, cooperation with CD.

## V. CONCLUSION

This paper has presented the use of standard convolutional neural networks for the problem of traffic signs recognition. The architecture recieves a $32 \times 32$ RGB input image which is convolved by $32\ N \times N \times 3$ filters. Filtered maps are, then, submitted to a rectified linear unity activation function followed by a $2 \times 2$ max pooling. Remaining maps are fully connected to a hidden layer with 500 neurons which in turn are connected to the 43 neurons on the output layer, accordingly to the number of classes in the dataset, which was derived from the German traffic sign recognition benchmark. Nine distinct values for the $N$ parameters were evaluated, each of them was trained from the scratch for 30 times.

The statistical analysis herein presented indicates that the best results where provided by convolutional layers of $3 \times 3$, $5 \times 5$ and $9 \times 9$ which did not produced significant statistic difference. This conclusion is somehow different to the one presented by Sichkar and Kolyubin in [7] which pointed out $9 \times 9$ and $19 \times 19$ as the ones which produced the best accuracies. The reason for that discrepancy is probably due to the stochastic characteristic of the network training that was not so carefully taken into consideration in [7].

Future work could explore statistic analysis with multiple CNN architectures and multiple data sets of traffic sign. At same time, focus in more robust indices to determine the quality of neural networks than accuracy (ROC, AUC, ...).

## REFERENCES

[1] M. G. Speranza, "Trends in transportation and logistics," *European Journal of Operational Research*, vol. 264, no. 3, pp. 830 – 836, 2018. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S0377221716306713

[2] F. Favarò, S. O. Eurich, and N. Nader, "Autonomous vehicles' disengagements: Trends, triggers, and regulatory limitations." *Accident; analysis and prevention*, vol. 110, pp. 136–148, 2018.

[3] O. Abiodun, A. Jantan, O. Omolara, K. Dada, N. Mohamed, and H. Arshed, "State-of-the-art in artificial neural network applications: A survey," *Heliyon*, vol. 4, p. e00938, 11 2018.

[4] Y. LeCun, P. Haffner, L. Bottou, and Y. Bengio, *Object Recognition with Gradient-Based Learning*. Berlin, Heidelberg: Springer Berlin Heidelberg, 1999, pp. 319–345.

[5] Q. Tang and K. Jo, "Analysis of various traffic sign detectors based on deep convolution network," in *2019 IEEE/SICE International Symposium on System Integration (SII)*, Jan 2019, pp. 507–511.

TABLE IV
Final cooperation between models (p-value x)

| Kernel's Size | Models not significantly different | Model significantly worse accuracy | Model significantly better accuracy |
|---|---|---|---|
| **3X3** | 5x5 e 9x9 | 13x13,15x15,19x19,23x23,25x25 e 31x31 | - |
| **5X5** | 3x3 e 9x9 | 13x13,15x15,19x19,23x23,25x25 e 31x31 | - |
| **9X9** | 3x3, 5x5 e 13x13 | 15x15,19x19,23x23,25x25 e 31x31 | - |
| **13X13** | 9x9 e 15x15 | 19x19, 23x23, 25x25 e 31x31 | 3x3 e 5x5 |
| **15X15** | 13x13 e 19x19 | 23x23,25x25 e 31x31 | 3x3, 5x5 e 9x9 |
| **19X19** | 15x15, 23x23, 25x25 e 31x31 | - | 3x3, 5x5, 9x9 e 13x13 |
| **23X23** | 19x19, 25x25 e 31x31 | - | 3x3, 5x5, 9x9, 13x13 e 15x15 |
| **25X25** | 19x19, 23x23 e 31x31 | - | 3x3, 5x5, 9x9, 13x13 e 15x15 |
| **31X31** | 19x19, 23x23 e 25x25 | - | 3x3, 5x5, 9x9, 13x13 e 15x15 |

[6] A. Jain, A. Mishra, A. Shukla, and R. Tiwari, "A novel genetically optimized convolutional neural network for traffic sign recognition: A new benchmark on belgium and chinese traffic sign datasets," *Neural Processing Letters*, vol. 50, no. 3, pp. 3019–3043, 02 2019.

[7] V. Sichkar and S. Kolyubin, "Effect of various dimension convolutional layer filters on traffic sign classification accuracy," *Scientific and Technical Journal of Information Technologies, Mechanics and Optics*, vol. 19, pp. 546–552, 06 2019.

[8] J. Stallkamp, M. Schlipsing, J. Salmen, and C. Igel, "The german traffic sign recognition benchmark: A multi-class classification competition," in *The 2011 International Joint Conference on Neural Networks, IJCNN 2011, San Jose, California, USA, July 31 - August 5, 2011*. IEEE, 2011, pp. 1453–1460.

[9] S. Houben, J. Stallkamp, J. Salmen, M. Schlipsing, and C. Igel, "Detection of traffic signs in real-world images: The german traffic sign detection benchmark," in *The 2013 International Joint Conference on Neural Networks (IJCNN)*. IEEE, aug 2013.

[10] R. Timofte, K. Zimmermann, and L. Van Gool, "Multi-view traffic sign detection, recognition, and 3d localisation," *Mach. Vision Appl.*, vol. 25, no. 3, p. 633–647, Apr. 2014. [Online]. Available: https://doi.org/10.1007/s00138-011-0391-3

[11] S. Šegvić, K. Brkić, Z. Kalafatić, V. Stanisavljević, M. Ševrović, D. Budimir, and I. Dadić, "A computer vision assisted geoinformation inventory for traffic infrastructure," in *13th International IEEE Conference on Intelligent Transportation Systems*, Sep. 2010, pp. 66–73.

[12] N. Paparoditis, J.-P. Papelard, B. Cannelle, A. Devaux, B. Soheilian, N. David, and E. Houzay, "Stereopolis ii: A multi-purpose and multi-sensor 3d mobile mapping system for street visualisation and 3d metrology," *Revue française de photogrammétrie et de télédétection*, vol. 200, no. 1, pp. 69–79, 2012.

[13] C. Grigorescu and N. Petkov, "Distance sets for shape filters and shape recognition," *IEEE Transactions on Image Processing*, vol. 12, no. 10, pp. 1274–1286, Oct 2003.

[14] F. Larsson and M. Felsberg, "Using fourier descriptors and spatial models for traffic sign recognition," in *Scandinavian conference on image analysis*. Springer, 2011, pp. 238–249.

[15] 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, may 2018.

[16] W. Liu, D. Anguelov, D. Erhan, C. Szegedy, S. Reed, C.-Y. Fu, and A. C. Berg, "Ssd: Single shot multibox detector," in *Computer Vision – ECCV 2016*, B. Leibe, J. Matas, N. Sebe, and M. Welling, Eds. Cham: Springer International Publishing, 2016, pp. 21–37.

[17] S. Song, Z. Que, J. Hou, S. Du, and Y. Song, "An efficient convolutional neural network for small traffic sign detection," *Journal of Systems Architecture*, jan 2019.

[18] T. Yang, X. Long, A. K. Sangaiah, Z. Zheng, and C. Tong, "Deep detection network for real-life traffic sign in vehicular networks," *Computer Networks*, vol. 136, pp. 95–104, may 2018.

[20] X. Xu, J. Jin, S. Zhang, L. Zhang, S. Pu, and Z. Chen, "Smart data driven traffic sign detection method based on adaptive color threshold and shape symmetry," *Future Generation Computer Systems*, vol. 94, pp. 381–391, may 2019.

[19] S. Salti, A. Petrelli, F. Tombari, N. Fioraio, and L. D. Stefano, "Traffic sign detection via interest region extraction," *Pattern Recognition*, vol. 48, no. 4, pp. 1039–1049, apr 2015.

[21] H. Li, F. Sun, L. Liu, and L. Wang, "A novel traffic sign detection method via color segmentation and robust shape matching," *Neurocomputing*, vol. 169, pp. 77–88, dec 2015.

[22] Á. Arcos-García, J. A. Álvarez-García, and L. M. Soria-Morillo, "Deep neural network for traffic sign recognition systems: An analysis of spatial transformers and stochastic optimisation methods," *Neural Networks*, vol. 99, pp. 158–165, mar 2018.

[23] A. A. Minai, "2011 international joint conference on neural networks (ijcnn 2011) [conference reports]," *IEEE Computational Intelligence Magazine*, vol. 7, no. 1, pp. 13–15, Feb 2012.

[24] 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, aug 2012.

[25] P. Sermanet and Y. LeCun, "Traffic sign recognition with multi-scale convolutional networks," in *The 2011 International Joint Conference on Neural Networks*, 2011, pp. 2809–2813.

[26] S. Aziz, E. A. Mohamed, and F. Youssef, "Traffic sign recognition based on multi-feature fusion and ELM classifier," *Procedia Computer Science*, vol. 127, pp. 146–153, 2018.

[27] S. Kaplan Berkaya, H. Gunduz, O. Ozsen, C. Akinlar, and S. Gunal, "On circular traffic sign detection and recognition," *Expert Systems with Applications*, vol. 48, 12 2015.

[28] P. H. Kassani and A. B. J. Teoh, "A new sparse model for traffic sign classification using soft histogram of oriented gradients," *Applied Soft Computing*, vol. 52, pp. 231–246, mar 2017.

[29] J. Pizarro, E. Guerrero, and P. Galindo, "Multiple comparison procedures applied to model selection," *Neurocomputing*, vol. 48, pp. 155–173, 10 2002.

[30] J. Demšar, "Statistical comparisons of classifiers over multiple data sets," *The Journal of Machine Learning Research*, vol. 7, pp. 1–30, 2006.

[31] T. G. Dietterich, "Approximate statistical tests for comparing supervised classification learning algorithms," *Neural computation*, vol. 10, no. 7, pp. 1895–1923, 1998.

[32] M. Friedman, "The use of ranks to avoid the assumption of normality implicit in the analysis of variance," *Journal of the American Statistical Association*, vol. 32, no. 200, pp. 675–701, 1937. [Online]. Available: http://www.jstor.org/stable/2279372

[33] P. Nemenyi, *Distribution-free Multiple Comparisons*. Princeton University, 1963. [Online]. Available: https://books.google.com.br/books?id=nhDMtgAACAAJ

[34] D. J. Sheskin, *Handbook of Parametric and Nonparametric Statistical Procedures*, 4th ed. Chapman & Hall/CRC, 2007.

[35] K. Sichkar V. N. (2019) Traffic signs preprocessed data. [Online]. Available: https://www.kaggle.com/valentynsichkar/traffic-signs-preprocessed

[36] M. Friedman, "A comparison of alternative tests of significance for the problem of m rankings," *The Annals of Mathematical Statistics*, vol. 11, no. 1, pp. 86–92, 1940. [Online]. Available: http://www.jstor.org/stable/2235971