

A Comparative Approach on Classification of Images with Convolutional Neural Networks



Ravikant Kholwal, Shishir Maurya

Abstract: Image degradation, such as blurring, or various sources of noise are common reasons for distortion happening during image procurement. In this paper, we will study in a systematical manner the efficiency of various Convolutional Neural Networks (CNN) approaches, in respects to the type of architecture and optimization strategies, with two main objectives in mind. Firstly, we examine the CNN performance in classifying clean images, with a dataset containing 8 classes and more than 18,000 images, observing comparatively the obtained results from training on a standard architecture with those obtained from training on a hyper parameters fine-tuned network and lastly, from training on a wider pre fine-tuned network. Secondly, training our model after a degradation function is applied, and after analyzing the results, we propose an approach which will gently balance the efforts stemming from difficult architecture de-sign or adopting the best optimization decisions with obtaining a satisfactory efficiency in a simple manner. We have offered a standard convolution architecture as a solution for classifying images which are distorted, and our results suggest that, departing from a simple design, with possible alterations of hyper parameters and other optimizing routes, the efficiency could massively increase.

Keywords: activation function, Convolution Neural Network, InceptionResNet50, validation accuracy

I. INTRODUCTION

Convolutional Neural Networks (CNN) approaches for image classification are the state of the art and a high percentage of effort in research is presently aimed toward supplementary developing the CNN performance by adjusting model architectures and various techniques in training. Nonetheless, studies trying to explore weaknesses and strengths of already optimised networks are somewhat limited [4]; furthermore, we can encounter the same shortage in literature in respects to studying systematically the efficiency of CNN in classifying images which have been degraded on purpose or are distorted during image

acquisition. And although superior results have been obtained on many tasks in computer vision, [9] strategies to lessen the impact of distortion in image capturing on classification accuracy are not. Numerous and research literature focused on the subject is relatively scarce. In this paper, we concentrate on two major topics: the first one is on finding the best architecture for providing optimal efficiency in image classification, with a few given conditions: implementation should be relatively easy for novices in machine learning, runtime should be quick enough to allow for further exploiting the architecture, by choosing various optimization strategies and the accuracy obtained should be more than 96 per cent. Second point which we will concentrate our efforts consists in applying a degrading function on our set of training images and observe how much will our standard network model suffer a drop in performance, investigating future possible ways to mitigate this loss. The dataset which the model will train on in order to classify images is composed of 8 classes (fruit, flower, aero plane, car, dog, cat, motorbike and person), and contains more than 2000 images in each folder. It was a custom made dataset, departing from a Kaggle collection of images. We initially implemented a standard, simple CNN network, with 4 fully connected layers, with ReLu activation function and max pooling enabled, followed by two fully connected dense layers, a dropout layer and a final dense layer with a softmax activation function. This final layer will output a one-hot list, which represents the model's designated classification of the image. An Adam (Adaptive moment estimation) optimizer is used as it works well in practice and outperforms other adaptive techniques. We have chosen this architecture to be our departing point for future steps of our research, because initial results were satisfactory and we decided to further analyze how our standard architecture will deal with the problems arising from training on degraded images, by adopting a clear path, rather than just guessing the optimal approach. We will provide a comparative evaluation of our standard architecture, a Keras autotuner implementation [6] and an InceptionResnet50 [11] approach, both for clear image classification and degraded image classification. When we refer to degraded images, we mean images which have been purposely distorted, by applying a blur, underwater, or hazy effect, with a degrading function through various filters. Fig. 1 shows examples of such images. Motivation to study the effects of distortion in CNN accuracy performance is given not only by the fact that in creating various datasets, the process may be hindered by

Manuscript received on April 17, 2021.

Revised Manuscript received on April 23, 2021.

Manuscript published on April 30, 2021.

* Correspondence Author

Ravikant Kholwal*, Pursuing Computer Science and Engineering, Indian Institute of Information Technology, Design and Manufacturing, Jabalpur, India.

Shishir Maurya, Pursuing Bachelors in Technology in Computer Science and Engineering at Indian Institute of Information Technology, Design and Manufacturing, Jabalpur, India.

© The Authors. Published by Blue Eyes Intelligence Engineering and Sciences Publication (BEIESP). This is an open access article under the CC BY-NC-ND license (<http://creativecommons.org/licenses/by-nc-nd/4.0/>)

A Comparative Approach on Classification of Images with Convolutional Neural Networks

the low quality images submitted by users, but also, for better understanding of processes in domains like robotics, automated driving, wearable cameras, etc.



Fig.1. Degraded Images

Generally, our findings reinforce the fact that our original convolutional neural network acts in a volatile manner when confronted with strange input, and that is quite efficient in computing output, without any changes in architecture. Furthermore, our experiments bring more observation for configuring systems where it is necessary to achieve dependable predictions, in practical real-life scenarios, by interpreting images which may be degraded by unknown sources of noise.

II. RELATED WORKS

Acquiring capabilities similar to those of humans, Convolutional Neural Networks emerged as a successful solution in eliminating human labour in the process of image-understanding tasks. Even though in some situations, CNN's perform with very little amount of error, studies have showed that some factors in image built may deeply affect a neural network, irrespectively of techniques used in order to increase efficiency. For example, Dodge and Karam [2] studied the impact of various image quality variations on the performance of deep CNN architectures, concluding that blur and noise are the most damaging elements. Studying face recognition against image degradations, Klemen et al [4] gave a comprehensive overview of strengths and weaknesses of various deep learning models, observing that all 4 models studied highly degrade in performance. While blur and noise are considered to be significantly decreasing the performance in deep CNN models as well, Karahan et al [7], also observed that studied networks generally tend to perform well with colour distortions, or changes in colour balance, providing a benchmark in face recognition algorithms' performance when confronted with various image degradation. While previously mentioned literature adopts a tone of pessimism regarding the detrimental influence over performance given by degradations in input, other sources provide a more optimistic note. For example, Zhou et al [10] focused their research on analysing efficiency of deep convolutional neural networks, by proposing a solution which will assuage the effects of distortion, by re-training the model and fine-tuning, and taking into consideration that in order to succeed in obtaining a higher performance, a balance between the size of the

dataset and the complexity of the model has to be maintained, and decision whether to re-train or fine-tune depends on the distortion level. A similar analysis is presented by Roy et al [8], who proposed a new capsule network which achieves very high accuracy in the case of training with degraded images, concluding that to their surprise, more shallow network perform better with degradation, and it is advisable to avoid increasing the depth of the network in this kind of task, because the conventional way of going deeper into the network will highly affect the robustness in training with degraded images. This last point of view is also our departing point, on which we are building our exploration efforts. We are analysing how our standard, not fine-tuned network maintains its robustness in classifying clear images at first, and how affected it will be when met with noisy input and lastly, its performance output after fine-tuning it.

III. EXPERIMENTS AND RESULTS

To understand how the proposed architecture performs in the task of image classification, we have started by initially training the model on 4 classes: aeroplane, car, cat, and dog and obtained an accuracy of 96 percent with a very small variance between training and validation accuracy and very small value of loss.

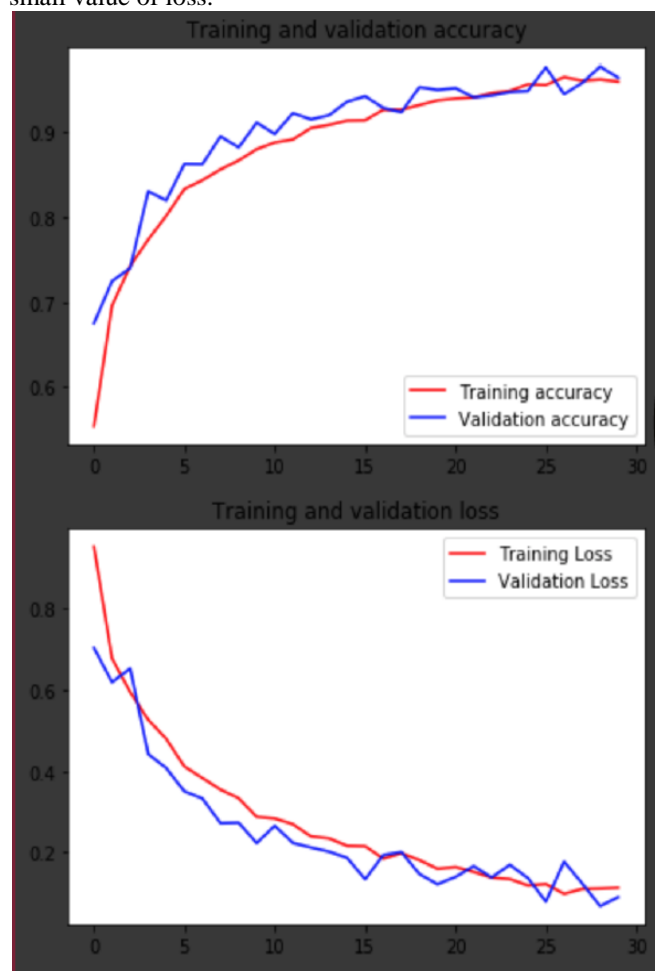


Fig.2. accuracy and loss of the original CNN Model

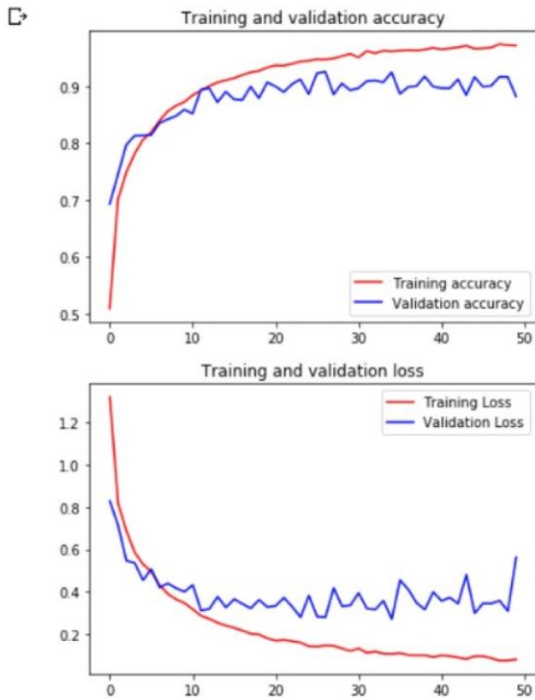


Fig.3. Accuracy and loss of CNN model with increased dataset

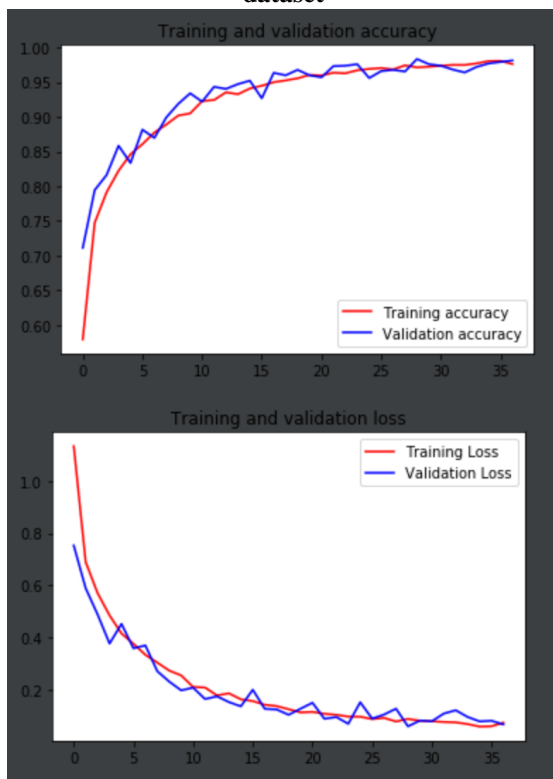


Fig.4. Accuracy and loss of improved CNN Model

Fig.2 shows that with running the model on 30 epochs and a batch size of 16, the loss associated with the training is less than 1 percent, which further encouraged us to train the model on 8 classes, adding flower, fruit, motorbike and person to the 4 classes already existent. From Fig. 3, we can see that by keeping the layers with the original parameters, the accuracy drops at only 88 per cent and the variance between validation-training increases. We decided to optimize this architecture, by changing the dropout rate to 0.15 as opposed to the original 0.10 and by increasing the batch size to 64 and epochs number to 37, accuracy obtained is 98 per cent and

validation-training accuracy / losses, are almost converging throughout the learning process (Fig. 4).

Since our goal was to comparatively observe the results of experiments done on various approaches, we applied a Keras auto-tuner to our original architecture. The motivation in choosing an automated tool in fine-tuning stems from various reasons. First of all, reiterating Ian Goodfellow’s idea of strategically planning the selection of a specific algorithm to be implemented [1], it was obvious that “wild guessing” an approach would be unfruitful for the scope of this research. With Keras Tuner implementation, we obtained an accuracy of 96 per cent, an insignificant variance between training and validation, and even though this result was less satisfactory than what we have previously obtained, it provided a good way of having a pre fine-tuned network , potentially useful for further research. Fig. 5 shows the accuracy summary report.

	precision	recall	f1-score	support
Flower	0.95	0.94	0.95	152
aeroplane	0.93	0.98	0.95	104
car	0.97	0.99	0.98	90
cat	0.92	0.93	0.93	90
dog	0.94	0.90	0.92	117
fruit	0.97	0.97	0.97	205
motorbike	0.99	0.99	0.99	91
person	1.00	0.98	0.99	96
accuracy			0.96	945
macro avg	0.96	0.96	0.96	945
weighted avg	0.96	0.96	0.96	945

Fig.5. Accuracy and loss of Model after implementing Keras tuner

In addition to exploring how our standard architecture performs with clean image classification, we trained our model on a pre-built deeper architecture, InceptionResnet50. Unsuccessfully, for our dataset, Inception Resnet50, as shown in Fig. 6 obtained only 88 percent accuracy, and a high variance between training and validation accuracy.

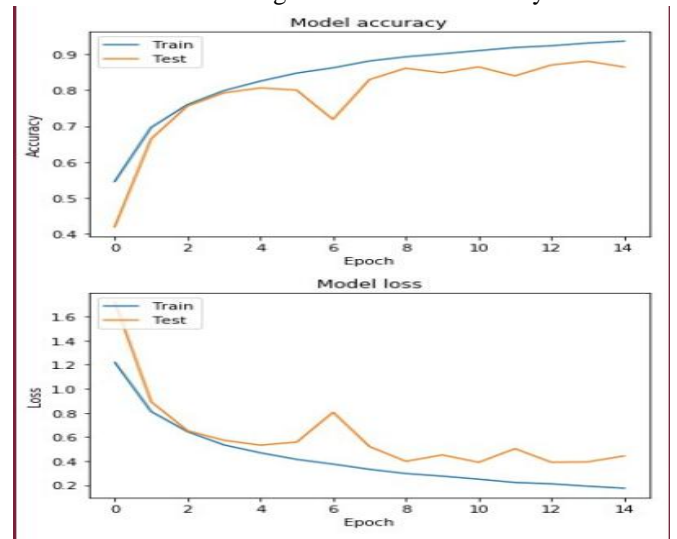


Fig.6. Accuracy and loss of InceptionResnet50 Model

After exploring these various approaches, we implemented Roy et al [8] degrading function in our standard CNN model, and after applying randomly, a number of 6 filters, we reached an accuracy of 80.26 percent, as shown in Fig.7 .

A Comparative Approach on Classification of Images with Convolutional Neural Networks

The model is running on only 14 epochs, with a batch size of 64. The filters applied are similar to those applied by Roy et al [8], with similar filter ratios, but to 100 percent of the training data, and images obtained can be seen in Fig.8. By taking the matter further, we implemented this degrading function in a Keras fine-tuned network and accuracy increased at 87 percent

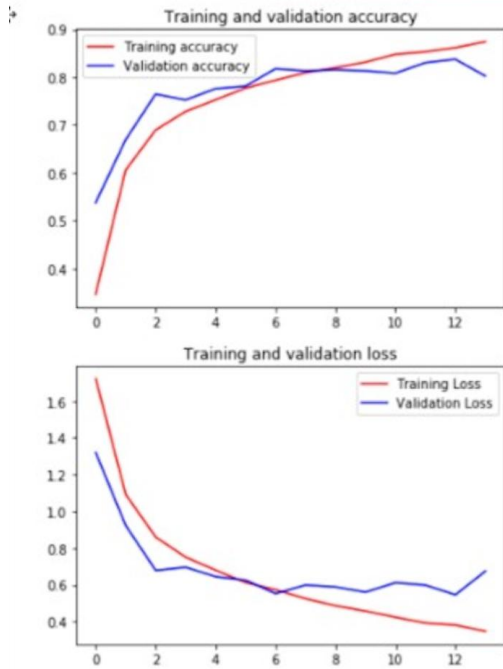


Fig.7. Accuracy and Loss of CNN Model after applying degrading function

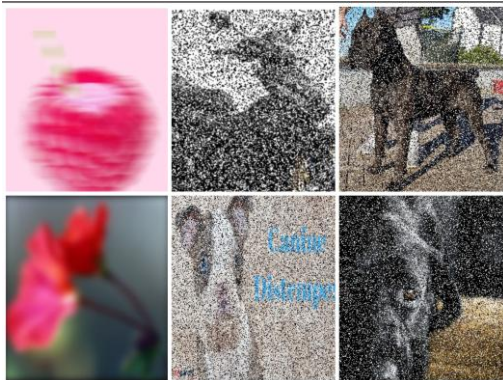


Fig 8. Resultant Images after applying degrading function

	precision	recall	f1-score	support
Flower	0.84	0.87	0.86	130
aeroplane	0.90	0.88	0.89	123
car	0.92	0.90	0.91	91
cat	0.70	0.70	0.70	90
dog	0.76	0.74	0.75	116
fruit	0.85	0.91	0.88	178
motorbike	0.98	0.96	0.97	94
person	0.99	0.93	0.96	123
accuracy			0.87	945
macro avg	0.87	0.86	0.86	945
weighted avg	0.87	0.87	0.87	945

Fig.9. Accuracy summary Report after applying degrading function

but variance between training and validation accuracy still remains high, around 20 percent difference (Fig. 9)

IV. DISCUSSION

On applying the various approaches for classifying clean and

degraded images, we reached some observations, essential to take into account for the scope of this research. First of all, the implementation of Keras tuner was an extremely useful tool into fine-tuning our network; on account of overfitting being a major problem in fine-tuning, there has to be a clear methodology to perform tuning of hyper parameters which would not discourage the efforts to make the network more robust.

Keras Tuner provides a random search, with 10 executions per trial, giving as output the optimal model, and by self-adjusting the learning rate depending on the value of sparse weights, prevents major overfitting from happening. In addition, Keras Tuner allowed a certain degree of flexibility to adjust the process of obtaining the best model. Adding batch normalization [5], with a momentum of 0.5, a very small epsilon, a gamma initializer and training with all 6 filters, at once and randomly, we obtained hazy, underwater, motion blurred and fish-eye images and the accuracy in predictions from training on our standard model after implementing hyper parameters is comparable to the accuracy obtained by Roy et al [8] in similar conditions.

Figure axis labels are often a source of confusion. Use words rather than symbols. Images obtained in Fig. 8 were distorted with: a filter sigma of 0.3 (Gaussian white and colour noise), a filter density of 0.3 (salt and pepper effect), a filter motion blur of 30, a filter of Gaussian blur of 25 and jpeg compression filter of 10. A momentum of 0.5 was used in order to avoid getting stuck in a local minima and ensuring the right balance between fast convergence and a small self-adjustable learning rate, avoiding skipping the local minima with very high steps. In comparison, by applying the degrading function in Inception-Resnet50, we obtained an accuracy of only 76 percent. One point worth mentioning is that when comparing the general robustness of our implementation with a network such as Inception is that by measuring the change in the filter output of the last convolutional layer in the two networks when introducing noise in the input, we can observe that the features extracted by our basic network change much slower than the features extracted by inception. Without going deep into architecture design alterations and by keeping a rather simple flow of convolutions, we can observe that the proposed approach has kept its robustness during learning and one contributing factor is that since our CNN model will not go very deep [3], it maintains being quite easy to train for a larger and varied dataset, in line with the analysis done by Roy et al [8]. We can understand from the accuracy summary, that the precision of prediction has stabilized at a very high percentage in the person class, and it is quite easy to understand why, since it contains the largest number of qualitative images collected and feature extraction happen with more accuracy for this category, suggesting that the more clear the features, the better chances for learning, even with degradation applied [12].

V. CONCLUSION

A conclusion section is not required. Although a conclusion may review the main points of the paper, do not replicate the abstract as the conclusion. In this research, we validate the efficiency of a shallower network in image classification, applicable datasets consisting of highly varied data and observe the effects of applying a degradation function on CNN models. It is obvious that all CNN architectures suffer drops in efficiency when training happens on distorted input. It is compelling to discover that a standard simple approach is more buoyant to degradation, achieving more firmness than deeper high efficiency promising architectures, like InceptionResnet50. We proposed a straightforward architecture, which achieves the highest accuracy among the CNN concepts discussed here, and by having a small number of layers, and not going very deep in learning, our model proves to be volatile to degradation. Image degradation classification is of major importance, especially in an age of technological advance, where image acquisition may happen under many circumstances and many scenarios and may be susceptible to a low quality in the process. Therefore, it is important, for future work, to take into account that deepening the network will not produce satisfactory results in distorted image classification, and if new networks are to be further developed, other strategies and techniques for increasing prediction performance are to be implemented with care in selecting the algorithm, in order to equip CNN models with robustness and stability against image degradation, while maintaining high accuracy.

REFERENCES

1. Yoshua Bengio, Ian Goodfellow, and Aaron Courville, *Deep learning*, volume 1, Citeseer, 2017.
2. Samuel Dodge and Lina Karam, "Understanding how image quality affects deep neural networks", in 2016 eighth international conference on quality of multimedia experience (QoMEX), pp. 1-6. IEEE, (2016).
3. Sanjukta Ghosh, Rohan Shet, Peter Amon, Andreas Hutter, and Andr'e Kaup, "Robustness of deep convolutional neural networks for image degradations", in 2018 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), pp. 2916-2920. IEEE, (2018).
4. Klemen Grm, Vitomir S'truc, Anais Artiges, Matthieu Caron, and Hazim K Ekenel, "Strengths and weaknesses of deep learning models for face recognition against image degradations", IET Biometrics, 7(1), 81-89, (2017).
5. Sergey Ioffe and Christian Szegedy, "Batch normalization: Accelerating deep network training by reducing internal covariate shift", arXiv preprint arXiv: 1502.03167, (2015).
6. Haifeng Jin, Qingquan Song, and Xia Hu, "Auto-keras: An efficient neural architecture search system", in Proceedings of the 25th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining, pp. 1946-1956. ACM, (2019).
7. Samil Karahan, Merve Kilinc Yildirim, Kadir Kirtac, Ferhat Sukru Rende, Gultekin Butun, and Hazim Kemal Ekenel, "How image degradations affect deep cnn-based face recognition?", in 2016 International Conference of the Biometrics Special Interest Group (BIOSIG), pp. 1-5. IEEE, (2016).
8. Prasun Roy, Subhankar Ghosh, Saumik Bhattacharya, and Umapada Pal, "Effects of degradations on deep neural network architectures", arXiv preprint arXiv: 1807.10108, (2018).
9. Ali Sharif Razavian, Hossein Azizpour, Josephine Sullivan, and Stefan Carlsson, "Cnn features off-the-shelf: an astounding baseline for recognition", in Proceedings of the IEEE conference on computer vision and pattern recognition workshops, pp. 806-813, (2014).
10. Yiren Zhou, Sibong Song, Ngai-Man Cheung, "On Classification of Distorted Images with Deep Convolutional Neural Networks", in 2017

11. IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), p. 1213-1217. IEEE, (2017).
12. Christian Szegedy, Vincent Vanhoucke, Sergey Ioffe, Jon Shlens, and Zbigniew Wojna, "Rethinking the inception architecture for computer vision", in Proceedings of the IEEE conference on computer vision and pattern recognition, pp. 2818-2826, (2016).
13. Jonghwa Yim and Kyung-Ah Sohn, "Enhancing the performance of convolutional neural networks on quality degraded datasets", in 2017 International Conference on Digital Image Computing: Techniques and Applications (DICTA), pp. 1-8.

AUTHORS PROFILE



Ravikant Kholwal, is a final year undergraduate student pursuing Computer Science and Engineering at Indian Institute of Information Technology, Design and Manufacturing, Jabalpur. He has developed several projects in the field of Android Development. One of his projects is Smart Parking. In this app, Users register their account and can book parking slots before reaching their respective destination. They can also pay for their Parking slot using UPI. The front end part is developed using XML and Java, and in the backend Firebase is implemented. When the car comes to the parking slot, the sensor sends information to the NodeMCU, which in turn sends information to the Firebase database.



Shishir Maurya, is a programming, Full-Stack and Machine Learning Enthusiast. He is currently pursuing Bachelors in Technology in Computer Science and Engineering at Indian Institute of Information Technology, Design and Manufacturing (IIITDMJ) as a Final Year Student. He believes in working hard and staying positive and focused to achieve goals. He is also an active member of IIITDMJ Programming and has been actively participating in several hackathons. He has also done worked at gNAP Services Pvt Limited as a Software Development Intern. I have a deep interest for mathematics, programming and tasks involving problem solving and complex computations. He has also developed ERP Project in college using Django Framework for Backend and Semantics UI for Frontend work.