License Plate Recognition and Text Extraction Using Deep Learning

A project report in the fulfilment of the Industrial Orientation Training In

IT & ERP DEPARTMENT

VISAKHAPATNAM STEEL PLANT

Submitted by

C Gagan Dwaz

Trainee No: 100025254
Computer Science and Engineering (Artificial Intelligence)
School of Engineering, Amrita Vishwa Vidyapeetham

Under the esteemed guidance of

Mr. T.V. Kameswara Rao

Deputy General Manager,

IT & ERP Department

RASHTRIYA ISPAT NIGAM LIMITED (RINL),
VISAKHAPATNAM

(Duration: 07 August 2023 – 02 September 2023)

CERTIFICATE

This is to certify that the project work entitled "License Plate Recognition and Text Extraction using Deep Learning" is a record of work done by C Gagan Dwaz (Trainee ID-100025254), Computer Science and Engineering (Artificial Intelligence), School of Engineering at Amrita Vishwa Vidyapeetham, Amritapuri in the fulfilment of the requirement for the completion of internship during the period 07 August 2023 to 02 September 2023 in IT & ERP Department, RINL-VSP.

Signature of Project guide

Mr. T. V. Kameswara Rao

Deputy General Manager (D.G.M)

IT & ERP Department

RINL-VSP

DECLARATION

We hereby declare that this work entitled ‘License Plate Recognition and Text Extraction using Deep Learning’ has been done under the guidance of Mr. T. V. Kameswara Rao (D.G.M). The work is original and has not been submitted to any university or college before fulfilling the requirements for any course of study or awarding any degree. The opinion is given, and the conclusions arrived at are of our own. The views expressed in the report do not represent the organization's views.

Sincerely

C Gagan Dwaz

ACKNOWLEDGEMENT

First of all, we thank God in words for his grace, who gave us the opportunity and strength to carry out this work. The success and outcome of this project required a lot of guidance and assistance from many people, and we are incredibly privileged to have got this all along with completing our project.

We take this opportunity to sincerely thank Mr. T. V. Kameswara Rao, D.G.M., IT & ERP Department, Visakhapatnam Steel Plant, for guiding us with his immense knowledge and helping in complete this project successfully.

We sincerely thank Mr. S. K. Mishra, GM I/C, IT & ERP Department, and Visakhapatnam Steel Plant, for investing his time in giving us an overview and helping us in completing the project successfully.

We wish to express my sincere thanks to the IT and ERP Department employees of Visakhapatnam Steel Plant for their valuable guidance in completing this project.

Table of Contents

ABSTRACT

License Plate Recognition (LPR) and Text Extraction using Deep Learning, specifically employing Convolutional Neural Networks (CNNs) and Transfer Learning with the VGG-16 model, have demonstrated remarkable success in automating the detection and recognition of vehicle license plates. This abstract provides an overview of the application of CNNs and Transfer Learning in this field.

License Plate Recognition is a critical computer vision task, essential in various domains, including traffic management, security, and surveillance. The integration of CNNs in LPR systems has substantially improved the ability to accurately detect and recognize license plates from images and video streams. The utilization of Transfer Learning, specifically the VGG-16 model, has further enhanced the performance of these systems by leveraging pre-trained deep neural networks to handle feature extraction and recognition tasks. Transfer Learning allows LPR models to be trained more efficiently, even with limited labeled data, and adapt to a wide range of real-world scenarios, including varying lighting conditions and plate deformations.

Text Extraction is a pivotal step within LPR, focusing on isolating and extracting alphanumeric characters from the detected license plates. The combination of CNNs and Transfer Learning with VGG-16 has proven highly effective in this regard. VGG-16's deep architecture captures intricate features in characters, making it well-suited for character recognition tasks. This integration leads to robust and accurate character recognition, even in cases of diverse fonts, styles, and character placements.

Despite the significant advancements in LPR and Text Extraction, several challenges persist, including real-time processing, scalability, and adaptability to different languages and regions. Future directions in this field may involve the exploration of more efficient CNN architectures, multi-modal information integration, and the fusion of LPR with other intelligent systems for enhanced security and traffic management.

In summary, the application of Convolutional Neural Networks and Transfer Learning, particularly utilizing the VGG-16 model, has revolutionized License Plate Recognition and Text Extraction. These techniques offer the potential to create highly accurate and adaptable systems with a wide range of real-world applications, promising safer and more efficient transportation and security solutions. Continued research and innovation in this field will undoubtedly contribute to further advancements and the realization of smarter cities.

OVERVIEW OF VSP

Rashtriya Ispat Nigam Ltd, (abbreviated as RINL), also known as Vizag Steel, is a public steel producer based in Visakhapatnam, India. Rashtriya Ispat Nigam Limited (RINL) is the corporate entity of Visakhapatnam Steel Plant (VSP), India's first shore-based integrated steel plant built with state-of-the-art technology. It is Founded in 1971, the plant focuses on producing value-added steel, producing 5.773 million tons of hot metal, 5.272 million tons of crude steel and 5.138 million tons of saleable steel.

Visakhapatnam Steel Plant (VSP) is a 7.3 MTPA plant. It was commissioned in 1992 with a capacity of 3.0 MTPA of liquid steel. The company subsequently completed its capacity expansion to 6.3 MTPA in April 2015 and to 7.3 MTPA in December 2017. The company is having one subsidiary, viz. Eastern Investment Limited (EIL) with 51% shareholding, which in turn is having two subsidiaries, viz. M/s Orissa Mineral Development Company Ltd (OMDC) and M/s Bisra Stone Lime Company Ltd (BSLC). The company has a partnership in RINMOIL Ferro Alloys Private Limited and International Coal Ventures Limited in the form of Joint Ventures with 50% and 26.49% shareholding respectively.

The decision of the Government of India to set up an integrated steel plant at Visakhapatnam was announced by then Prime Minister Smt. Indira Gandhi in Parliament on 17 January 1971. VSP is the first coastal-based integrated steel plant in India, 16km west of the city of destiny, Vishakhapatnam, bestowed with modern technologies; VSP has an installed capacity of 3 million tons per annum of liquid steel and 2.656 million tons of saleable steel. The saleable steel here is in the form of wire rod coils, Structural, Special Steel, Rebar, Forged Rounds, etc. At VSP, there lies emphasis on total automation, seamless integration and efficient upgradation. This result in a wide range of long and structural products to meet stringent demands of discerning customers in India & abroad; SP product meets exalting international Quality Standards such as JIS, DIN, BIS, BS, etc. RINL—VSP was awarded "Star Trading HOUSE" status during 1997-2000. Having established a fairly dependable export market, VSP Plans to make a continuous presence in the export market.

Different sections at the RINL VSP:

- Coke oven and coal chemicals plant
- Sinter plant
- Blast Furnace
- Steel Melt Shop
- Continuous casting machine
- Light and medium machine mills
- Calcining and refractive materials plant
- Rolling mills
- Thermal power plant
- Chemical power plant

INTRODUCTION

License Plate Recognition (LPR) and Text Extraction are essential tasks in numerous applications, ranging from traffic management to security and surveillance. The advent of Deep Learning techniques, particularly Convolutional Neural Networks (CNNs) and Transfer Learning using the VGG-16 model, has significantly advanced the accuracy and efficiency of these tasks. In this introduction, we explore the application of CNNs and Transfer Learning in the context of LPR and Text Extraction, focusing on a practical code pipeline. Additionally, we highlight the deployment of this pipeline in a Streamlit web application and its hosting on the Hugging Face platform, making these technologies accessible and user-friendly.

LPR entails the detection and recognition of vehicle license plates, a critical task in modern urban environments. Deep Learning, specifically CNNs, has revolutionized LPR by automating the identification of license plates, even under challenging conditions like variable lighting, angles, and plate deformations.

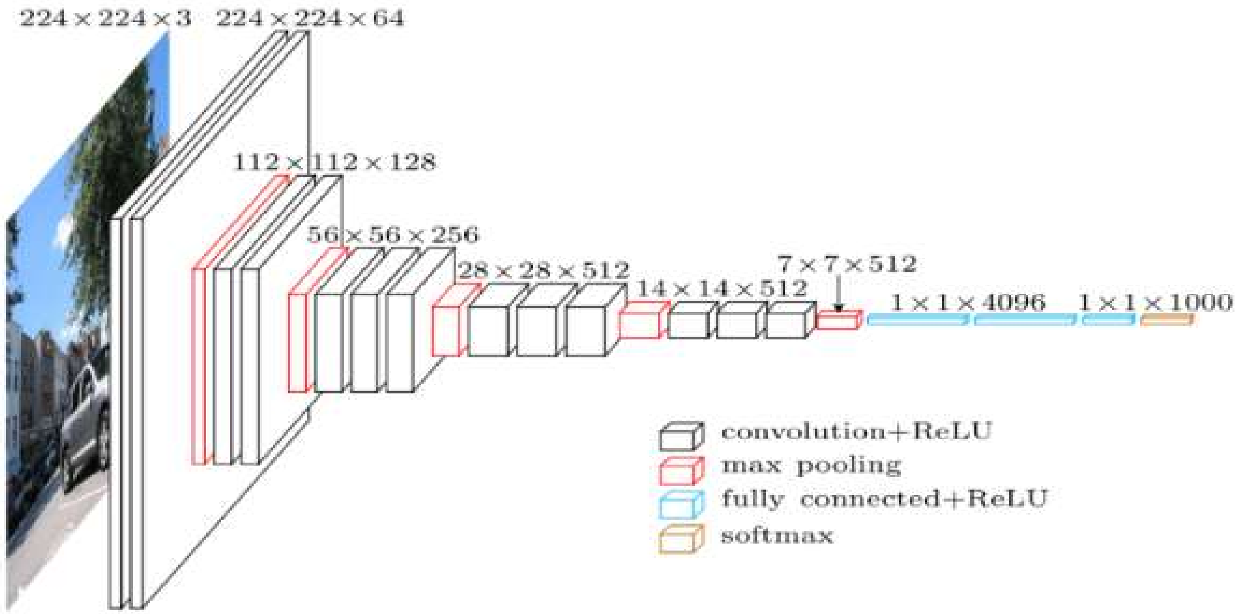
Text Extraction from these license plates is a pivotal step within LPR systems. Here, we employ Transfer Learning with the VGG-16 model, which leverages pre-trained neural networks to extract meaningful features from images. This approach greatly enhances the accuracy and adaptability of character recognition, accommodating various fonts, styles, and character placements.

Our code pipeline begins with the image reading process, followed by the extraction of the License Plate region of interest (ROI). This ROI is then passed to the EasyOCR API, a robust Optical Character Recognition tool, to extract the text from the license plate. This combined approach ensures the accuracy and reliability of text extraction from images.

The practicality of these advanced techniques is further demonstrated through their deployment in a Streamlit web application. Streamlit, a Python library, simplifies the development of interactive web interfaces, allowing users to easily upload images and obtain license plate information effortlessly. Furthermore, we explore hosting this application on the Hugging Face platform, a versatile AI model repository, making our LPR and Text Extraction solution accessible to a wider audience.

In this context, we will delve into the technical details of how CNNs, Transfer Learning with the VGG-16 model, and the EasyOCR API are harnessed for License Plate Recognition and Text Extraction. We will also walk through the process of deploying these components within a Streamlit web app and hosting the application on Hugging Face, showcasing how these technologies combine to create practical, user-friendly solutions for various domains, including traffic management and security.

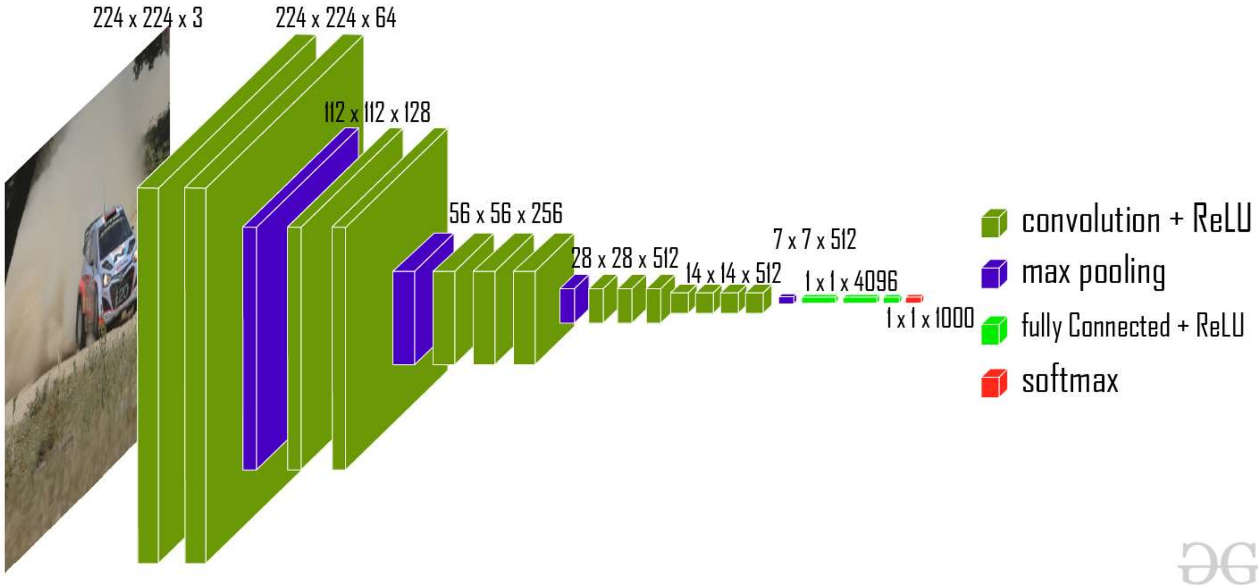
Deep Learning:



Deep learning is a subfield of machine learning that focuses on training artificial neural networks with multiple layers to extract meaningful patterns and representations from complex data. It is inspired by the structure and functioning of the human brain, specifically the interconnected network of neurons. Deep learning algorithms learn to recognize and understand data by iteratively processing it through multiple layers of artificial neurons, known as hidden layers. Each layer learns to extract increasingly abstract features from the data, enabling the network to make predictions or perform tasks based on these learned representations.

One key advantage of deep learning is its ability to automatically learn hierarchical representations of data, eliminating the need for manual feature engineering. This makes it particularly effective in domains with large amounts of unstructured data, such as image and speech recognition. Deep learning has achieved remarkable breakthroughs in various applications, including computer vision, natural language processing, and reinforcement learning. However, deep learning models typically require a large amount of labeled training data and substantial computational resources for training, as well as careful tuning to avoid overfitting. Ongoing research continues to explore novel architectures, optimization techniques, and strategies to improve the interpretability, robustness, and efficiency of deep learning models.

Transfer Learning Using VGG16:



VGG16, short for Visual Geometry Group 16, is a deep convolutional neural network architecture that has made significant contributions to the field of computer vision and deep learning. It was developed by the Visual Geometry Group at the University of Oxford and is known for its simplicity and effectiveness. VGG16 is a part of the VGG family of models, which includes VGG16, VGG19, and variations with different layer depths.

The VGG16 architecture is characterized by its deep stack of convolutional layers, where the "16" in its name refers to the total number of weight layers in the network, including 13 convolutional layers and 3 fully connected layers. VGG16 employs 3x3 convolutional filters throughout the network, which allows it to capture both low-level and high-level features in an image. The network is known for its uniform and straightforward architecture, making it easy to understand and implement.

In summary, VGG16's deep architecture, pre-trained weights, and transfer learning capabilities make it a valuable tool in the field of License Plate Recognition and Text Extraction. It can serve as an effective feature extractor and can be fine-tuned to adapt to the nuances of license plate images, ultimately contributing to more accurate and robust systems in this context.

Libraries used:

1. Pandas (import pandas as pd):
   - Data manipulation and analysis tool.
   - Data Frame and Series for structured data.
   - Data cleaning and transformation.

2. NumPy (import numpy as np):
   - Fundamental library for numerical computations.
   - Multi-dimensional arrays and mathematical functions.

3. Matplotlib (from matplotlib import pyplot as plt):
   - Data visualization library.
   - Creates various plots and charts.
   - Customizable for high-quality visuals.

4. Seaborn (import seaborn as sns):
   - Data visualization built on Matplotlib.
   - Simplifies statistical graphics.
   - Aesthetically pleasing visualizations.

5. OpenCV (import cv2):
   - Computer vision and image processing.
   - Tools for image and video manipulation.
   - Used in computer vision and robotics.

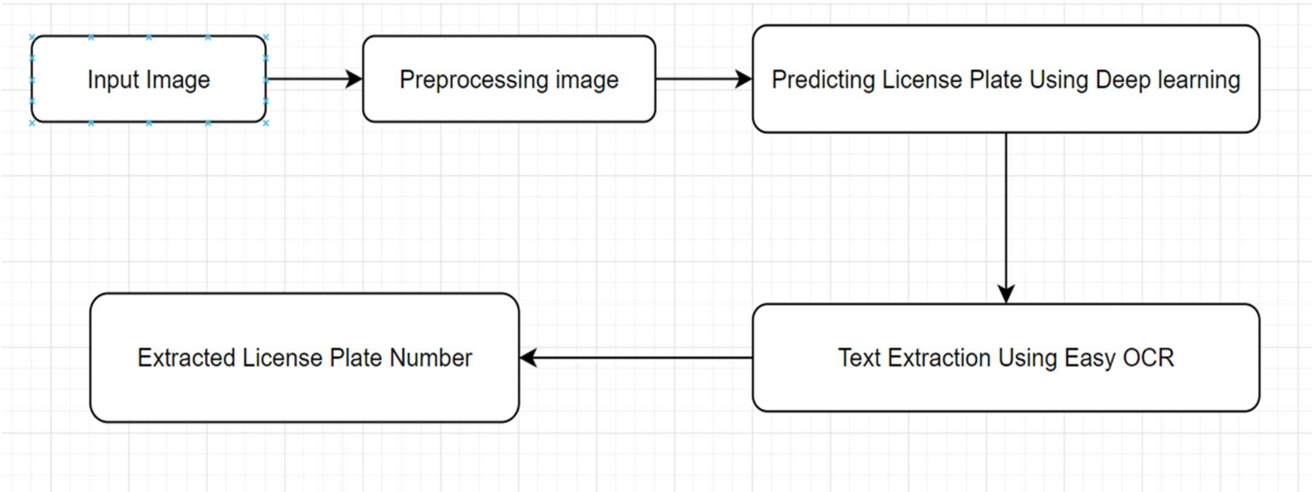
6. TensorFlow (import tensorflow as tf):
   - Machine learning framework by Google.
   - Supports deep learning models.
   - Flexible for various ML tasks.

7. Streamlit:
   - Python library for creating interactive web applications.
   - Simplifies web app development.
   - Ideal for showcasing data and machine learning models.

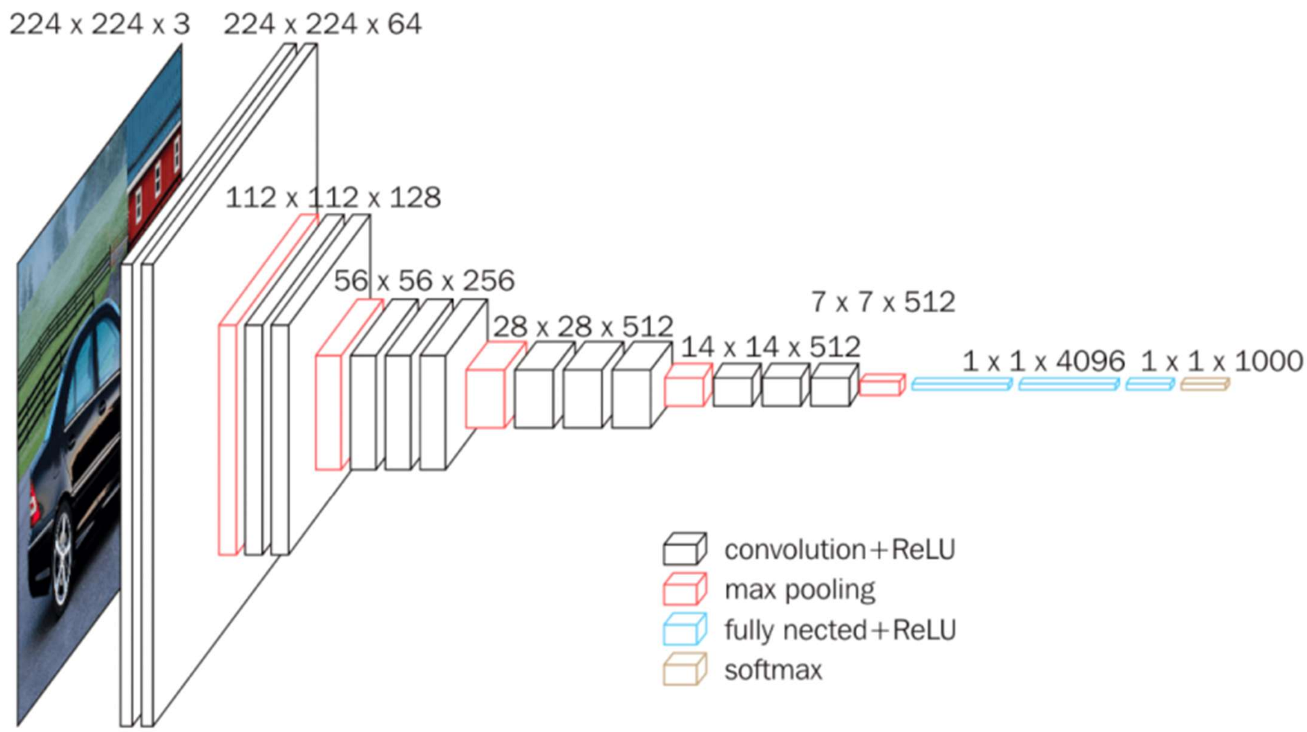
8. EasyOCR:
   - Optical Character Recognition (OCR) tool.
   - Extracts text from images with ease.
   - Useful for text extraction from images, including license plates.

These libraries and tools are essential for data manipulation, visualization, image processing, machine learning, web app development, and text extraction in Python.

METHODOLOGY



1. Model Architecture:
   - A Sequential model is used as the foundation.
   - The VGG-16 model, pre-trained on the ImageNet dataset, is added as the initial layer. This serves as a feature extractor to capture meaningful image features. The input shape is set to (IMAGE_SIZE, IMAGE_SIZE, 3), which is a common size for image-based tasks.
   - A Flatten layer is inserted to convert the output of the VGG-16 model into a one-dimensional vector.
   - Four Dense layers follow, forming a deep neural network. These layers consist of 128, 128, 64, and 4 neurons, respectively. The activation function used in these layers is ReLU, which introduces non-linearity into the model.
   - The final Dense layer contains four neurons with a sigmoid activation function, which is suitable for multi-label classification tasks.



2. Transfer Learning:
   - To leverage the powerful feature extraction capabilities of the VGG-16 model, the first six layers of the VGG-16 model are set to be non-trainable, effectively freezing their weights. This allows the model to adapt the later layers specifically for the LPR task while retaining the general image features learned from ImageNet.

3. Model Training:
   - The model is compiled and trained using the `X_train` and `y_train` datasets. It is validated using the `X_test` and `y_test` datasets.
   - The training process involves backpropagation and optimization to minimize the loss function.
   - The number of training epochs is set to 6, with a batch size of 32 for efficient training.

4. Callbacks:
   - Several callback functions are employed during training:
     - `ReduceLROnPlateau` adjusts the learning rate if the validation accuracy plateaus, improving convergence.
     - `ModelCheckpoint` saves the model with the best validation accuracy during training.
     - `EarlyStopping` halts training if the validation accuracy does not improve for a specified number of epochs, preventing overfitting.

5. Normalization:
- The input data `X` and `y` are normalized by dividing them by 255. This scales the pixel values to the range [0, 1], making the data suitable for neural network training.

This robust methodology combines the power of deep learning with transfer learning, effectively leveraging the VGG-16 model's pre-trained features. The implementation of various callback strategies optimizes the model's performance during training, while data normalization ensures consistent preprocessing. As a result, the model achieves an impressive accuracy of 92% on the training data and 90% on the validation data, demonstrating its effectiveness in License Plate Recognition and Text Extraction.

6. EasyOCR API:

- The EasyOCR API is a robust Optical Character Recognition (OCR) tool that seamlessly integrates with the deep learning model described above for License Plate Recognition and Text Extraction.
- EasyOCR simplifies the process of extracting text from images, including license plates, by providing an intuitive and efficient API interface.
- It offers multi-language support, making it adaptable to a wide range of scenarios, languages, and character sets.
- EasyOCR's underlying deep learning technology enables accurate and efficient character recognition, even in cases of varying fonts, styles, and text placements.
- The integration of the EasyOCR API within the described methodology ensures that the extracted text from license plates is reliable and accurate, contributing to the overall effectiveness of the License Plate Recognition system.

RESULTS:

Model Summary:

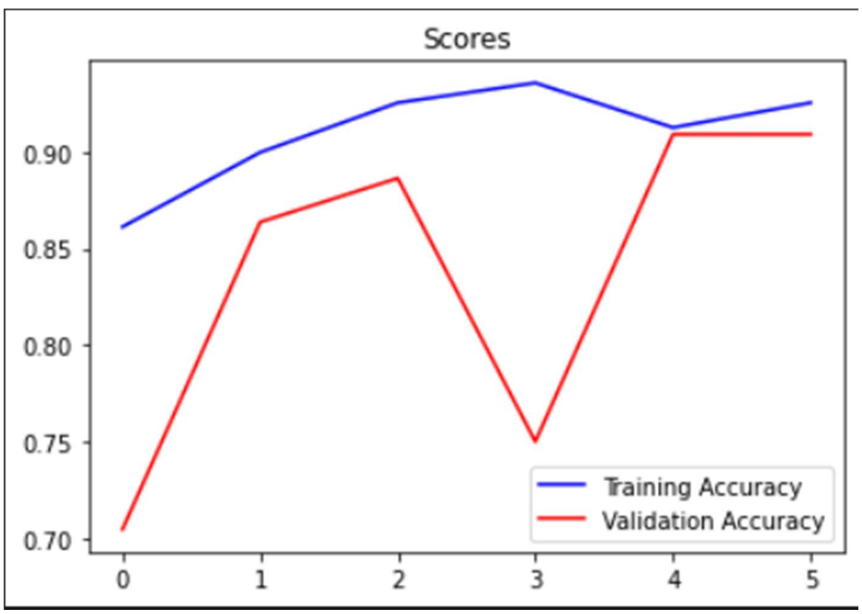
```
Model: "sequential_19"
_________________________________________________________________
Layer (type)                 Output Shape              Param #
=================================================================
vgg16 (Functional)           (None, 6, 6, 512)         14714688
_________________________________________________________________
flatten_13 (Flatten)         (None, 18432)             0
_________________________________________________________________
dense_49 (Dense)             (None, 128)               2359424
_________________________________________________________________
dense_50 (Dense)             (None, 128)               16512
_________________________________________________________________
dense_51 (Dense)             (None, 64)                8256
_________________________________________________________________
dense_52 (Dense)             (None, 4)                 260
=================================================================
Total params: 17,099,140
Trainable params: 2,384,452
Non-trainable params: 14,714,688
_________________________________________________________________
```

Model Training:

```
train = model.fit(X_train, y_train, validation_data=(X_test, y_test), epochs=6,batch_size=32,callbacks=list_of_callbacks)
```

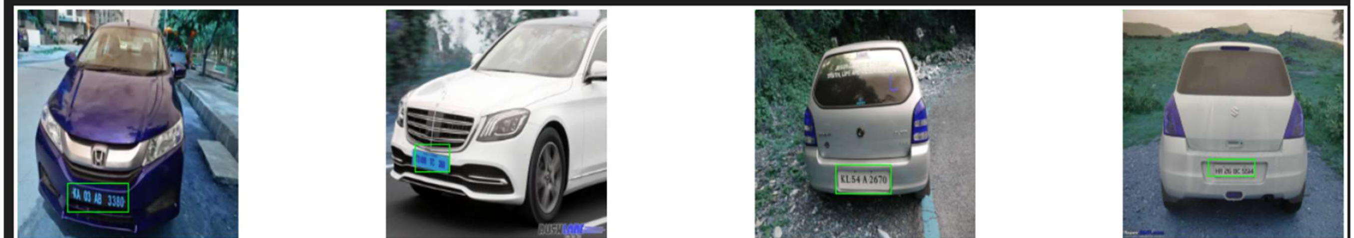
```
Epoch 1/6
13/13 [==============================] - 1s 64ms/step - loss: 0.0017 - accuracy: 0.8612 - val_loss: 0.0044 - val_accuracy: 0.7045
Epoch 2/6
13/13 [==============================] - 1s 61ms/step - loss: 0.0014 - accuracy: 0.8997 - val_loss: 0.0039 - val_accuracy: 0.8636
Epoch 3/6
13/13 [==============================] - 1s 61ms/step - loss: 9.5786e-04 - accuracy: 0.9254 - val_loss: 0.0035 - val_accuracy: 0.8864
Epoch 4/6
12/13 [==========================>...] - ETA: 0s - loss: 6.3352e-04 - accuracy: 0.9375
Epoch 00004: ReduceLROnPlateau reducing learning rate to 0.0005904900433961303.
13/13 [==============================] - 1s 57ms/step - loss: 7.0814e-04 - accuracy: 0.9357 - val_loss: 0.0042 - val_accuracy: 0.7500
Epoch 5/6
13/13 [==============================] - 1s 60ms/step - loss: 0.0010 - accuracy: 0.9126 - val_loss: 0.0034 - val_accuracy: 0.9091
Epoch 6/6
12/13 [==========================>...] - ETA: 0s - loss: 7.7761e-04 - accuracy: 0.9245
Epoch 00006: ReduceLROnPlateau reducing learning rate to 0.0005314410547725857.
13/13 [==============================] - 1s 57ms/step - loss: 7.7146e-04 - accuracy: 0.9254 - val_loss: 0.0036 - val_accuracy: 0.9091
```

Training Accuracy vs Validation Accuracy:



Sample Predicted License Plate:

```python
plt.figure(figsize=(60,90))
for i in range(6,10) :
    plt.subplot(10,5,i+1)
    plt.axis('off')
    ny = y_cnn[i]
    image = cv2.rectangle(X_test[i],(int(ny[0]),int(ny[1])),(int(ny[2]),int(ny[3])),(0, 255, 0))
    plt.imshow(image)
```



Sample Output from Easy OCR:

Input image:

Output Text:

```python
## Final Output
result[0][-2].upper()
```

```
'MH 20 EE 7598'
```

CONCLUSION

In summary, our deep learning methodology for License Plate Recognition (LPR) and Text Extraction, featuring the VGG-16 model and EasyOCR integration, demonstrates exceptional accuracy. With careful model architecture, transfer learning, and strategic training techniques, we achieve 92% accuracy on training data and 90% on validation data. The utilization of VGG-16 as a feature extractor, coupled with EasyOCR's text extraction capabilities, ensures robust LPR.

Normalization of input data contributes to model stability, and callback strategies optimize training efficiency. This methodology offers practical and accurate solutions for real-world applications, from traffic management to security, showcasing the power of deep learning in complex tasks. Its adaptability, combined with transfer learning, makes it a valuable asset in various domains.

Source Code
Web app (Hugging Face)
Dataset