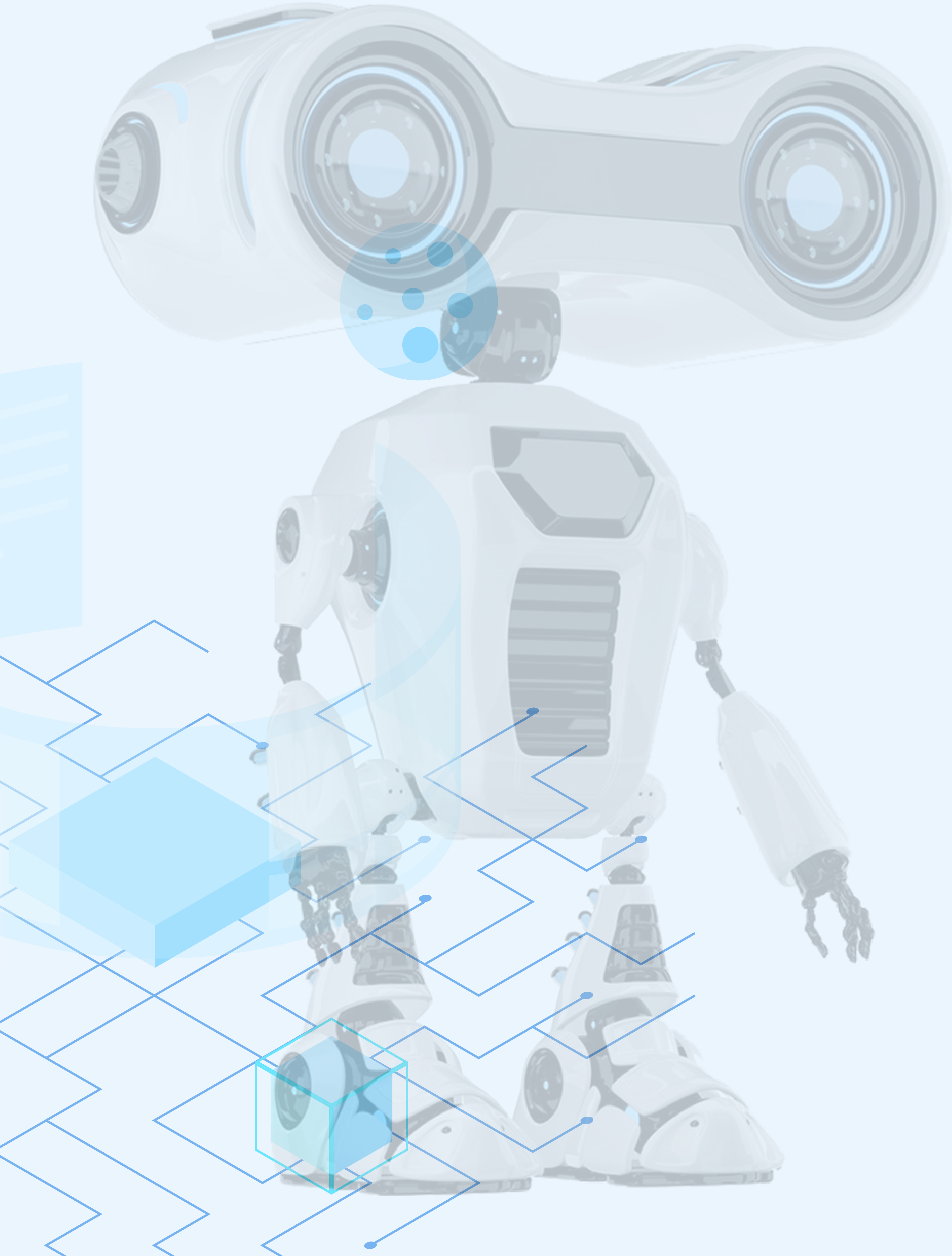


# DEEP LEARNING BASED DRIVER DISTRACTION DETECTION


- MANOHAR REDDY TIPPIREDDY - (110134650)
- KATHAR PATCHA ABDUL RAHIM - (110134610)
- ASIF KASIM LOHAR - (110137509)

# DRIVER ACTIVITY RECOGNITION

- Due to the busy schedules, it becomes very difficult to remain active all the time, so drivers usually do a lot of mistakes while driving.
- One may start feeling drowsy, may fall asleep and got mis-leaded by a phone call.
- These situations may lead to accidents, injuries and even deaths.
- So to understand various kinds of driver behaviors, a driver activities recognition system should be designed which is done in this study.
- It is based on the deep learning approach in which CNN is used.
- Specifically, ten common driving activities are identified which are most frequently considered as distracted driver behaviour

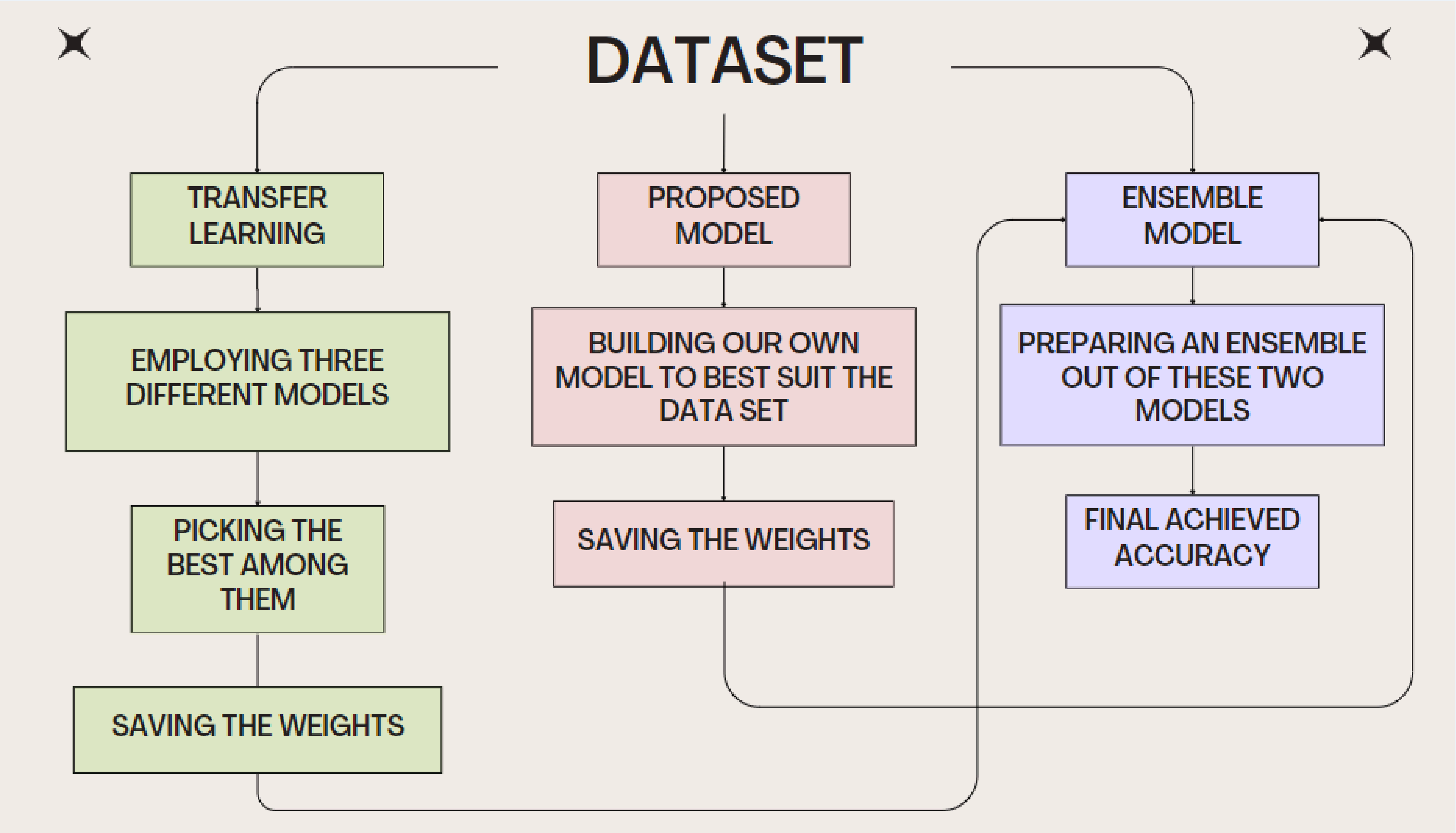
# PROBLEM DEFINITION

- It is also stated that with a precise driver behavior monitoring system, the accident rate can be reduced by 10-20%.
- So the main problem which we have focussed in this work is to recognise the driver activities accurately and to monitor the driver behaviour.
- If we apply this in a real world application, intelligent vehicles can better interact with human drivers.
- And it also helps in making decisions properly and generating human-like driving strategies.
- With real-time tracking of driver behaviour, a particular driver can be classified as worthy or not to takeover the vehicle controls in emergency situations.

# DATASET DETAILS

- We have used the State Farm dataset which has 22,500 (640x480) RGB labelled images
- The dataset has a total of ten driving tasks with one as normal driving behaviour and other as distracted driver behaviours.
- The classes are as following:
  - Normal Driving, Checking Radio
  - Texting with Right Hand, Drinking
  - Calling with Right Hand, Looking Behind
  - Texting with Left Hand, Makeup
  - Calling with Left Hand, Talking with Co- Passenger

# PROPOSED APPROACH



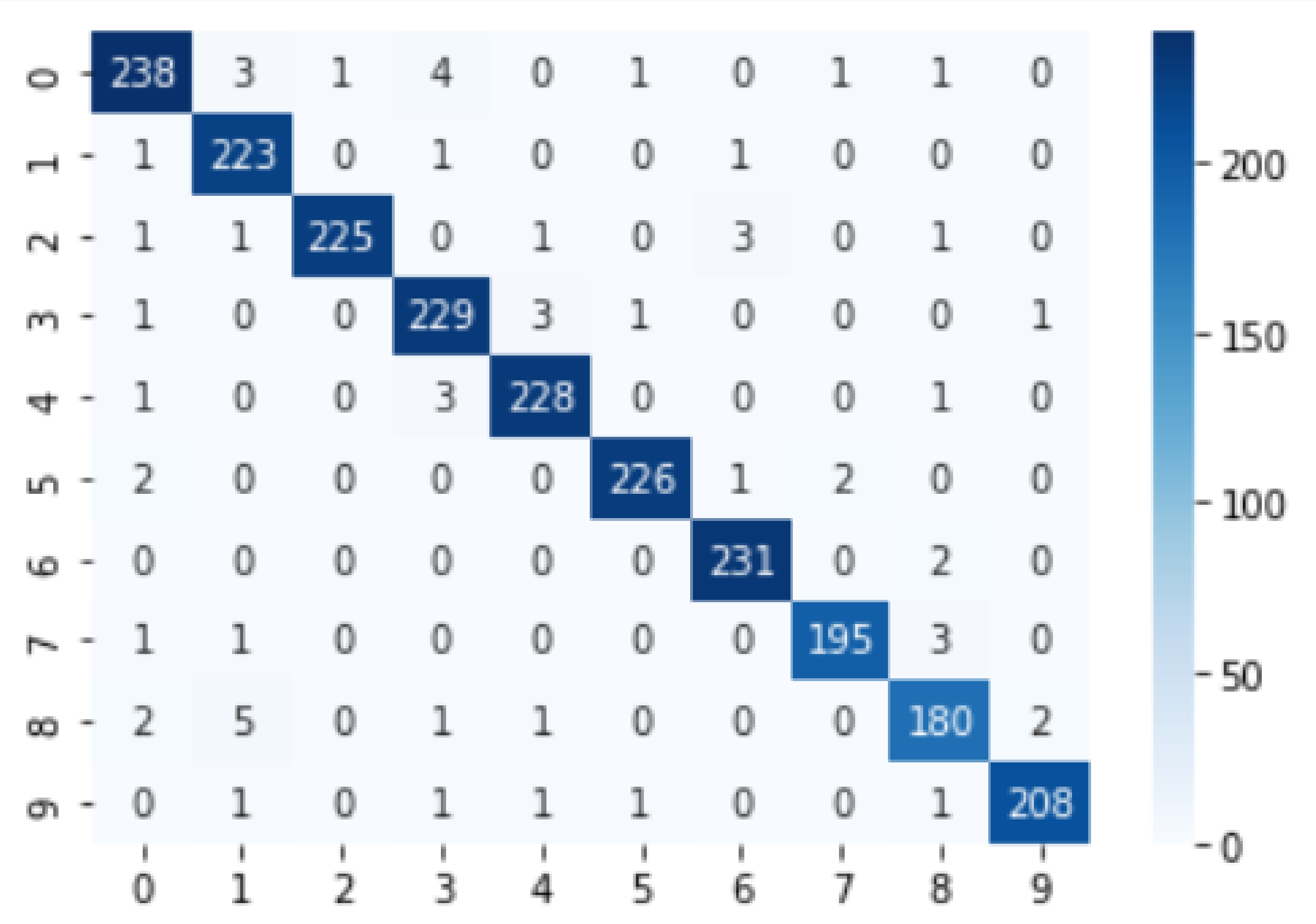
# TRANSFER LEARNING

- Transfer learning is a machine learning technique where a model developed for a particular task is reused as the starting point for a model on a second task.
- Leverages pre-trained models on large datasets to improve performance and reduce training time for new, related tasks.
- We chose to use the following three models due to their strong performance in image classification tasks.
  - DenseNet121
  - VGG16
  - MobileNetV2
- Adapted these models by fine-tuning their weights on the custom dataset specific to distracted driving.
- Trained the models with a tailored dataset of distracted driving images, ensuring relevance and specificity to the task.

# TRANSFER LEARNING

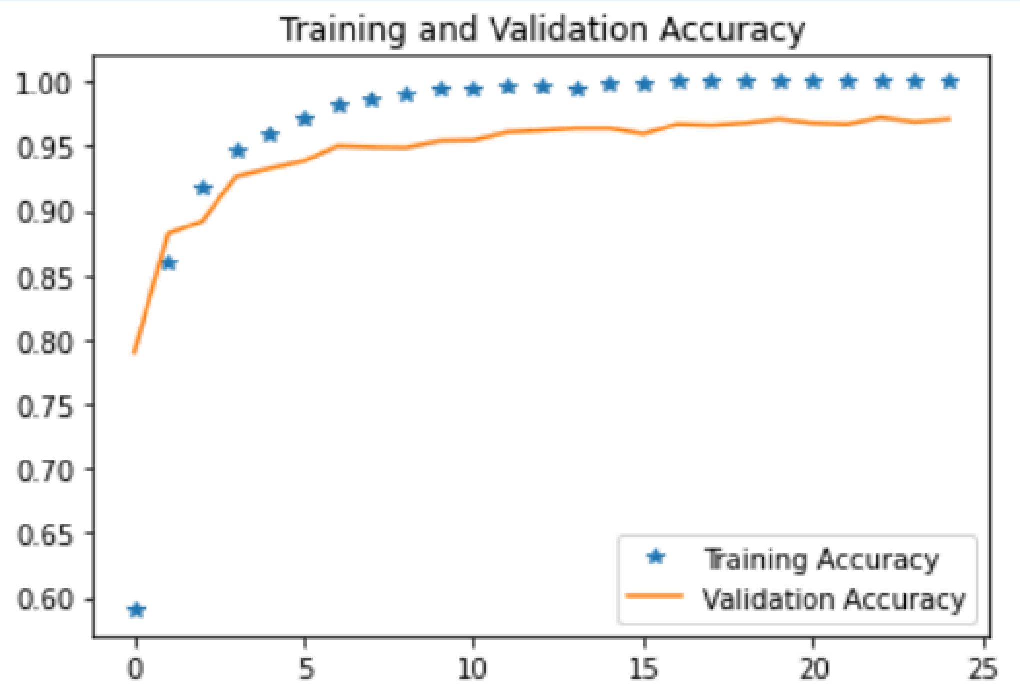
- Learning Rate: The initial learning rate was set to optimize the training process, with schedules and adjustments made during training to enhance convergence.
- Epochs: The number of training iterations was determined through experimentation to balance training time and model performance.
- Batch Size: The optimal batch size was chosen to manage memory usage effectively and ensure stable training.
- Accuracy metrics were obtained for all three models—DenseNet121, VGG16, and MobileNetV2—providing a comprehensive view of their performance.
- For each model, loss and accuracy graphs were plotted to visualize the training and validation progress over epochs.
- Additionally, confusion matrices were generated to analyze the models' classification results in detail, highlighting the true positives, false positives, true negatives, and false negatives.

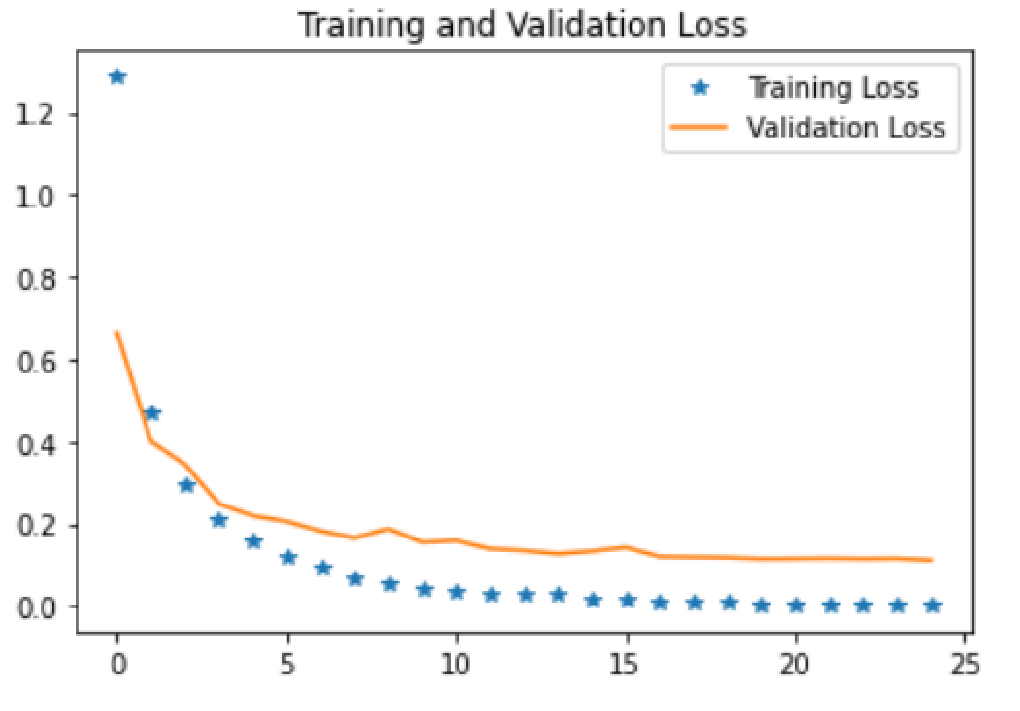
# RESULTS - DENSENET121



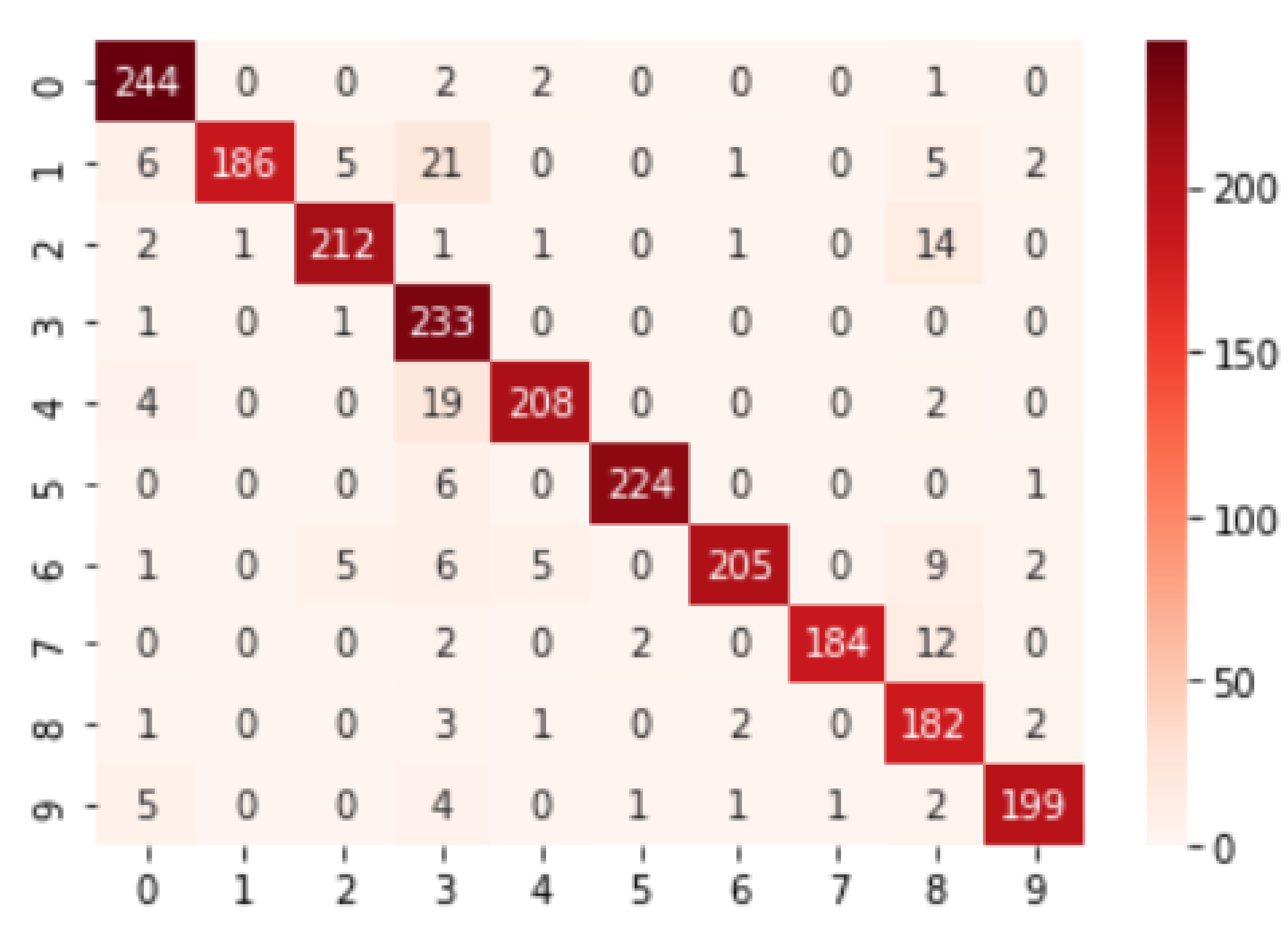

```
# Getting the accuracy of DenseNet model.
test_loss, test_acc = Dense_net.evaluate(test_images, test_labels)
print("Accuracy of the densenet model:",test_acc*100,"%")
```

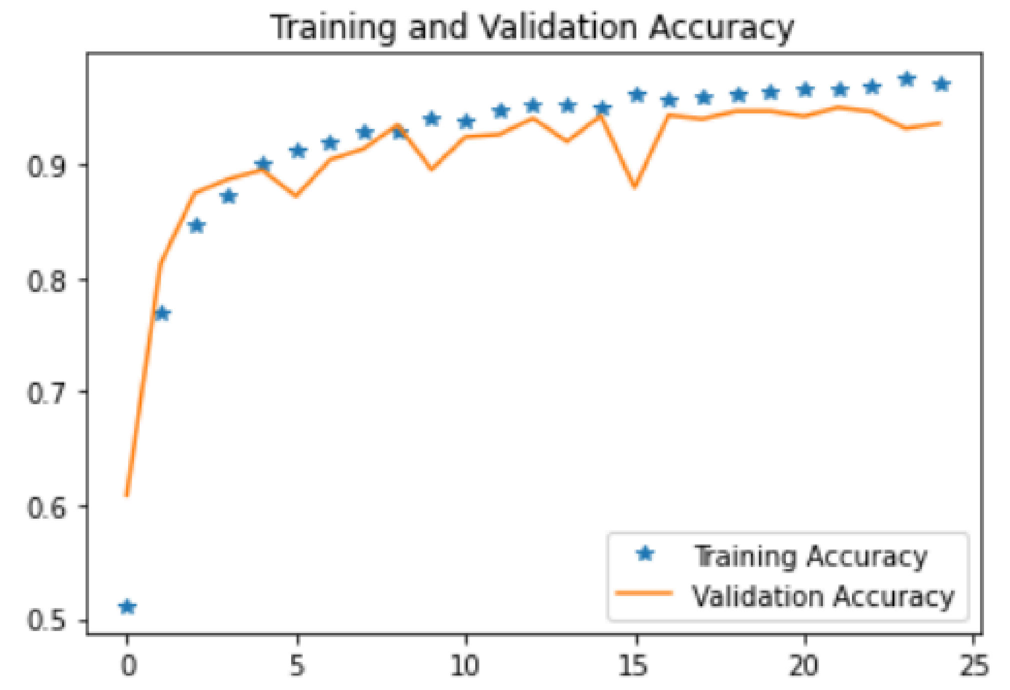
```
71/71 [==============================] - 6s 62ms/step - loss: 0.0950 - ac
Accuracy of the densenet model: 97.2804307937622 %
```



# RESULTS - VGG16



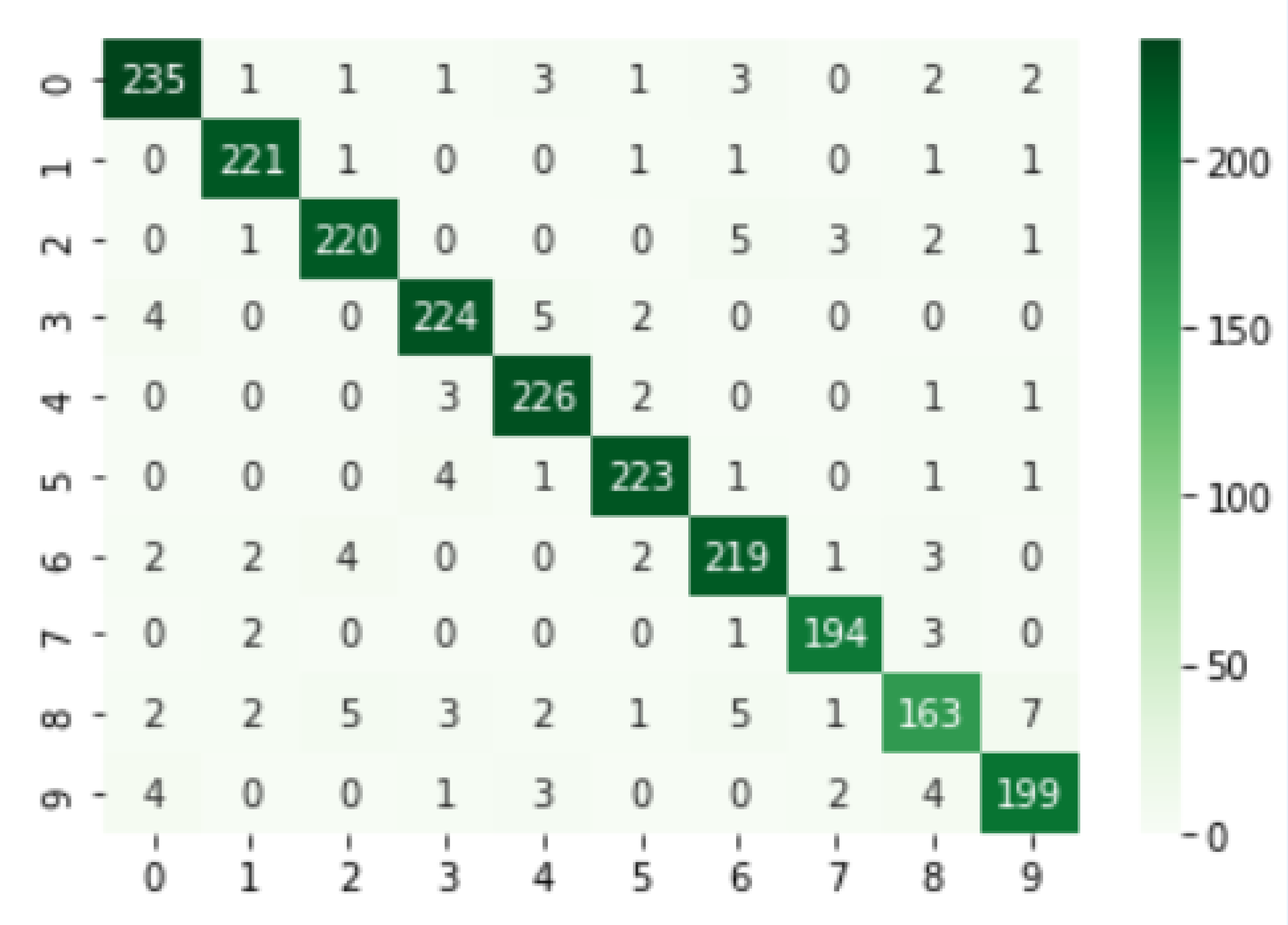



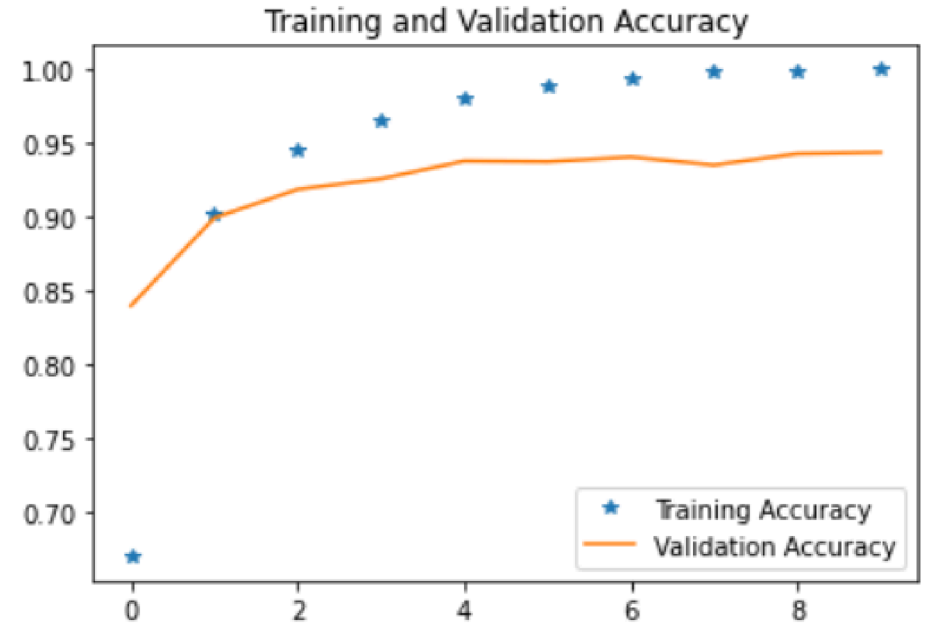


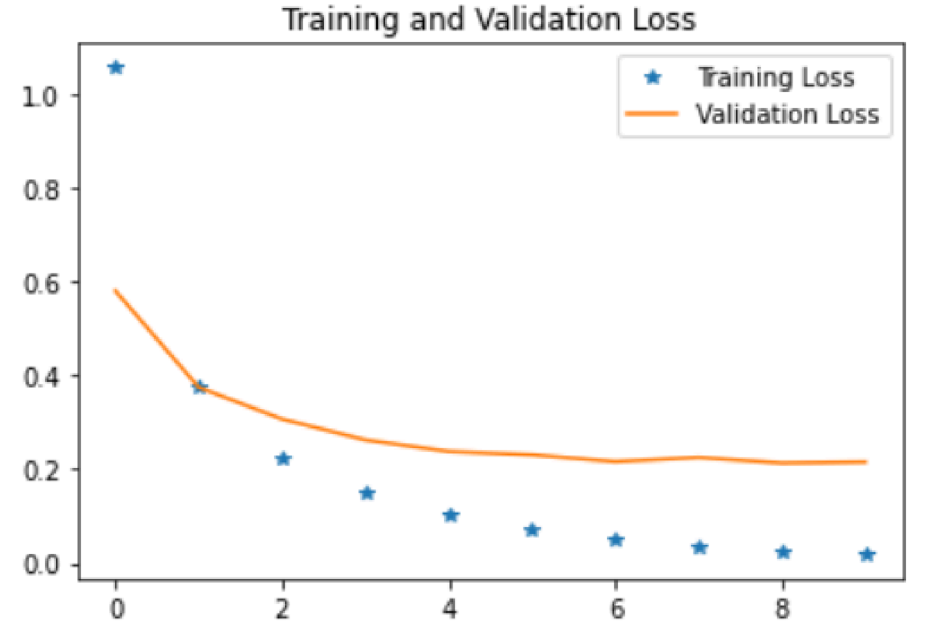
```
# Getting the accuracy of VGG16 model.
test_loss, test_acc = vgg_net.evaluate(test_images, test_labels)
print("Accuracy of the vgg16 model:",test_acc*100,"%")
```

```
71/71 [==============================] - 2s 26ms/step - loss: 0.2703 - ac
Accuracy of the vgg16 model: 92.5991952419281 %
```

# RESULTS - MOBILENETV2



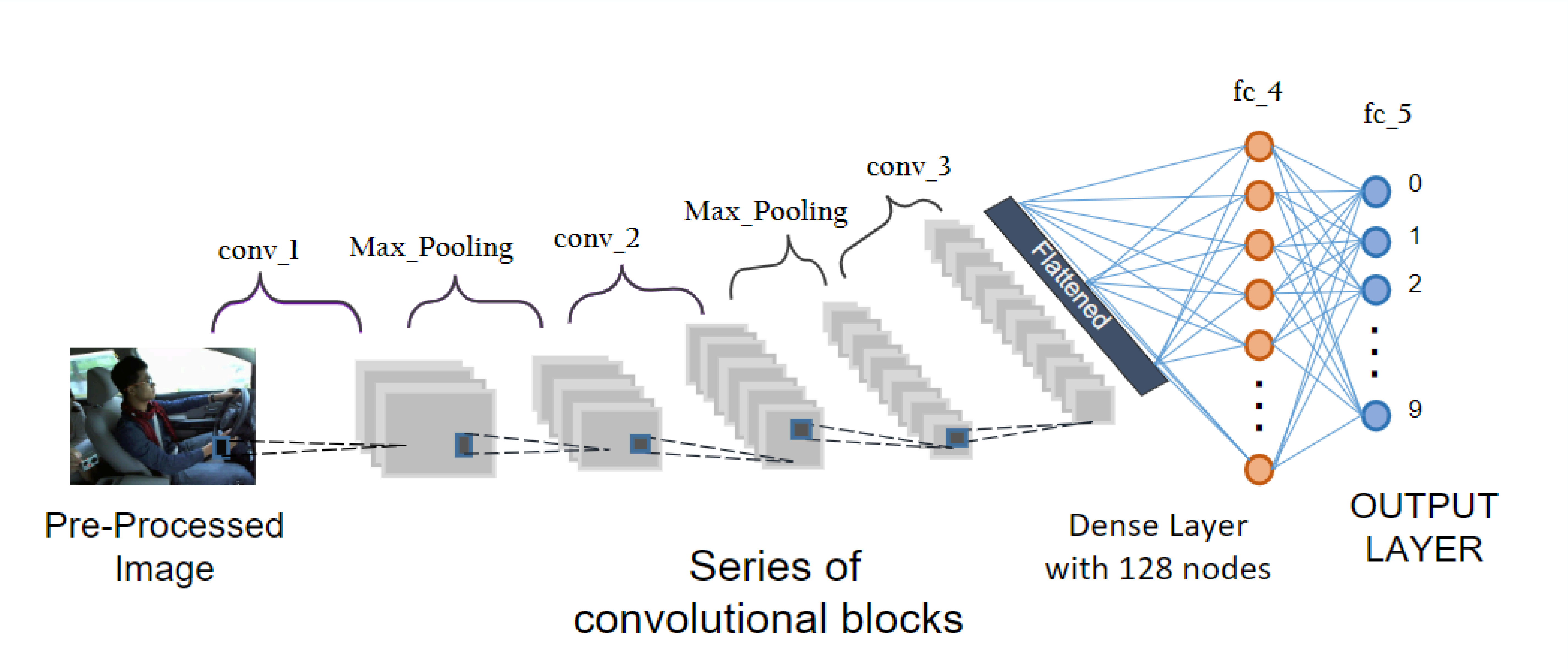

```
# Getting the accuracy of resnet model.
test_loss, test_acc = mb_net.evaluate(test_images, test_labels)
print("Accuracy of the MobileNetV2 model:",test_acc*100,"%")
```

```
71/71 [==============================] - 2s 24ms/step - loss: 0.1860 -
Accuracy of the MobileNetV2 model: 94.69460248947144 %
```

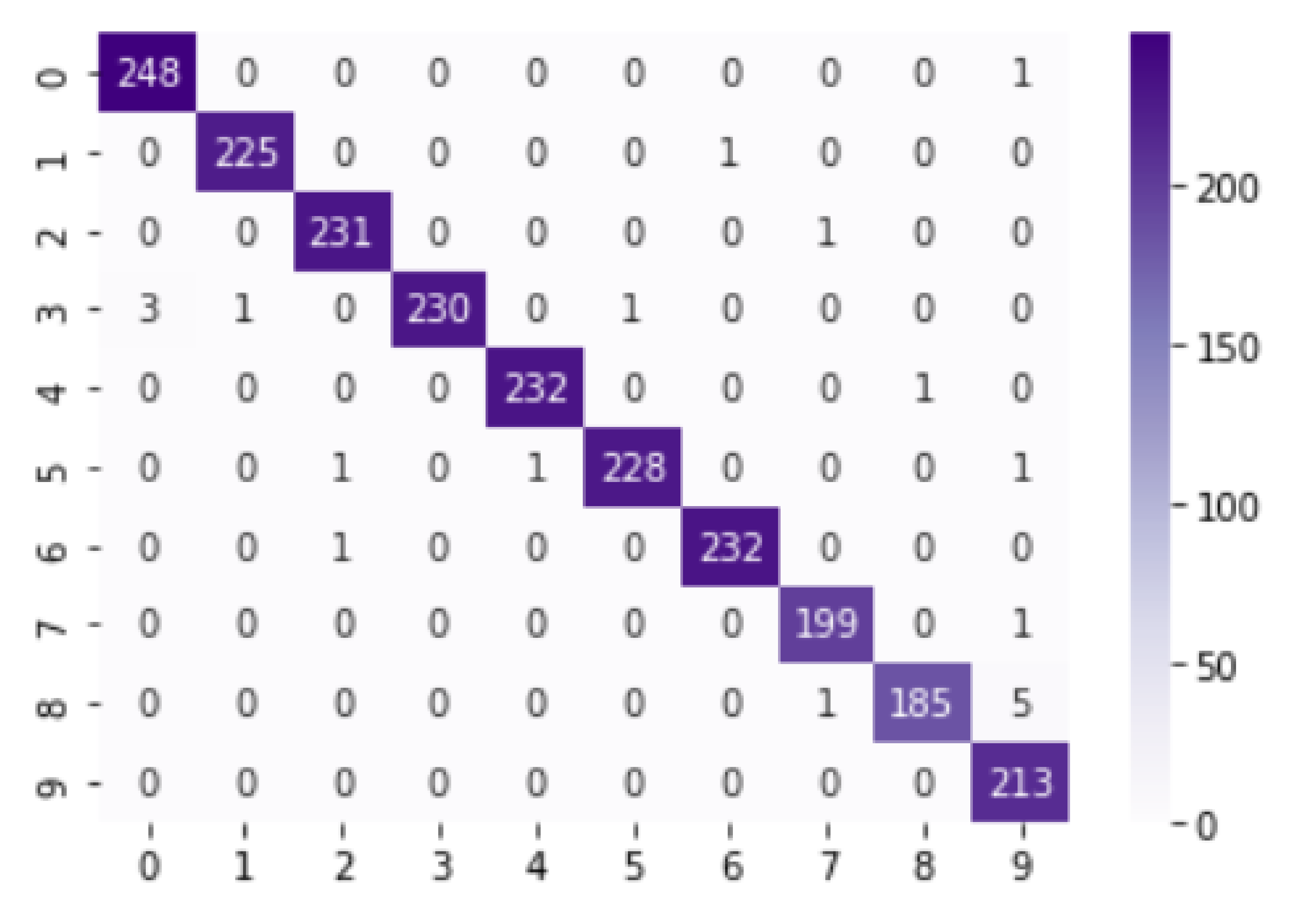
# PROPOSED CNN MODEL - DARNET

- The custom CNN architecture starts with a convolutional layer of 64 filters (3x3) to capture essential features from input images.
- Followed by MaxPooling2D layers to reduce spatial dimensions and computation while preserving key features.
- Additional convolutional layers with 32 filters continue feature extraction.
- The Flatten layer then transforms the 2D feature maps into a 1D vector for the Dense layer with 128 neurons, which learns complex patterns.
- The final Dense layer with 10 neurons and a softmax activation function classifies the images into 10 categories, making the network well-suited for detecting and classifying distracted driving behaviors according to the State Farm Dataset that we have used.

# MODEL ARCHITECTURE - DARNET

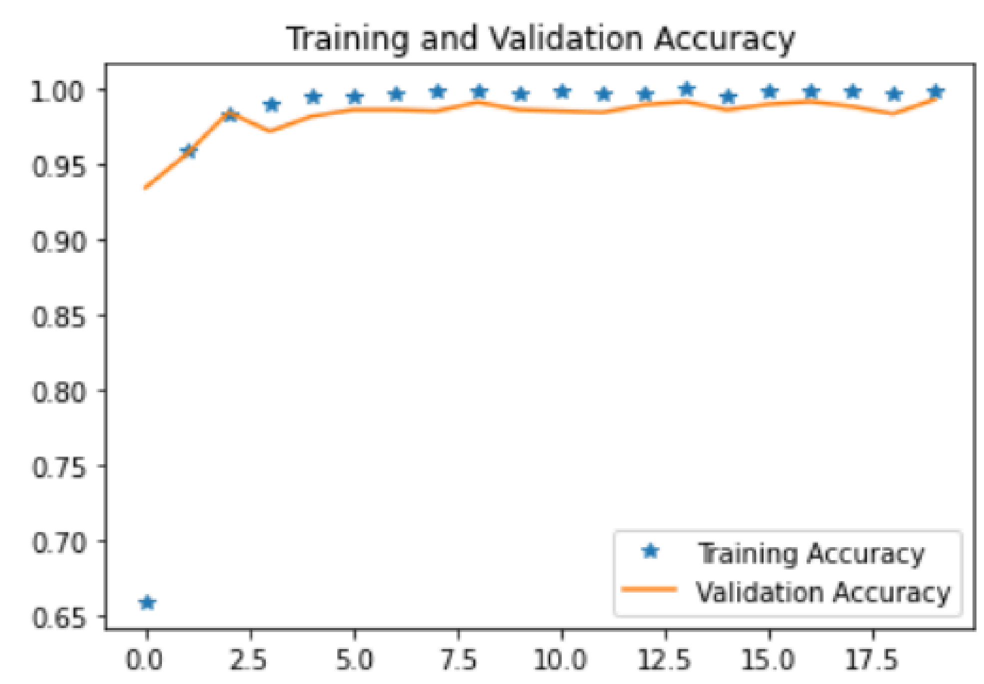


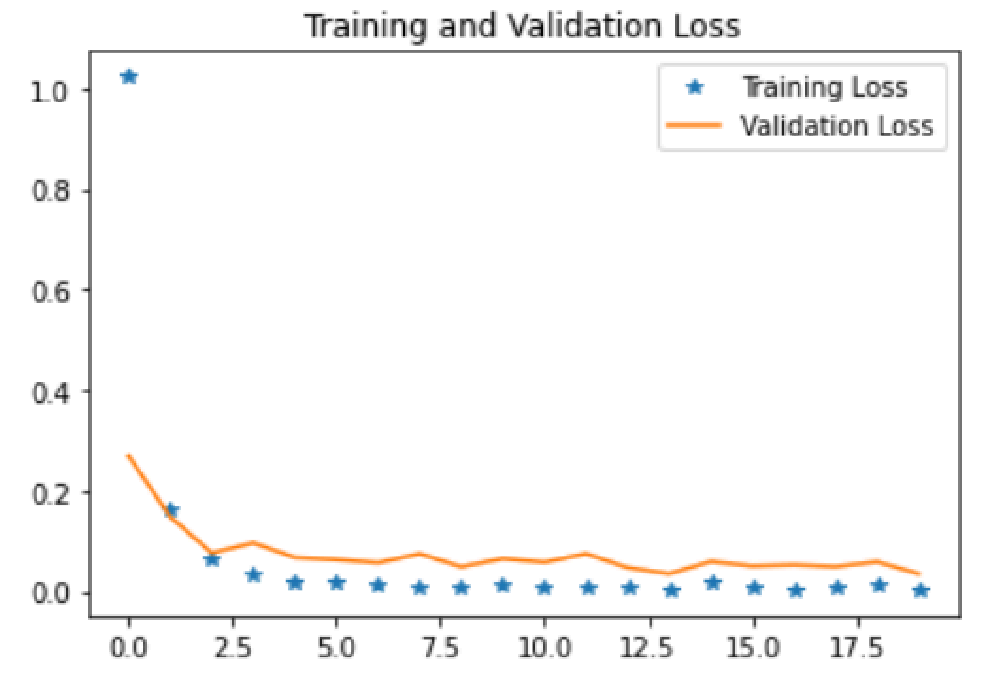
# RESULTS - DARNET




```
# Getting the accuracy of our model.
test_loss, test_acc = network_cnn.evaluate(test_images, test_labels)
print("Accuracy of the proposed model:",test_acc*100,"%")
```

```
71/71 [==============================] - 1s 7ms/step - loss: 0.0390 - ac
Accuracy of the proposed model: 98.60833835601807 %
```



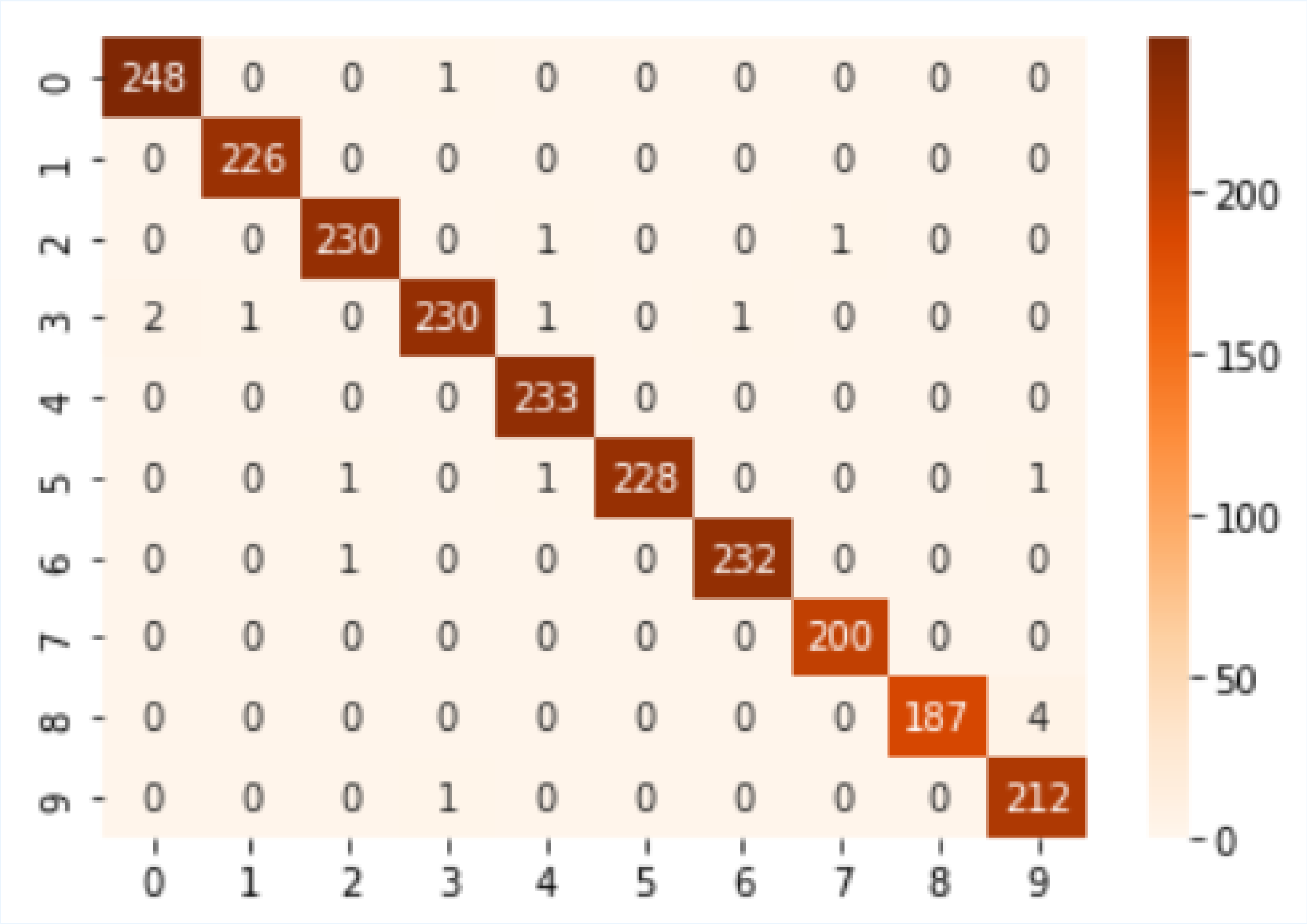
# ENSEMBLING TECHNIQUE

- Ensembling is a technique that combines the predictions from multiple models to improve overall performance and accuracy.
- By aggregating the strengths of different models, ensembling can reduce errors and enhance generalization, leading to more robust and reliable predictions.
- Enhanced Generalization: The ensemble approach can better capture various aspects of the data, improving the model's ability to generalize to new, unseen examples.
- Improved Accuracy: By averaging the predictions, the ensemble can mitigate the weaknesses of individual models, leading to improved overall accuracy and robustness.
- Reduced Overfitting: Combining multiple models helps to reduce overfitting by leveraging diverse learned features and reducing the impact of any single model's errors.

# ENSEMBLE MODEL

- There are various techniques through which ensembling can be employed in our projects such as bagging, boosting, voting, stacking and averaging.
- Among these, we have employed average ensembling technique in our project.
- In this project, we have picked the best-performing pretrained model which is to be DenseNet121 in our case.
- And our custom CNN architecture (DARNET) and combined them using average ensembling by extracting the predicted probabilities from the final dense layer of each model and then averaging them to make a final prediction.
- This technique leverages the complementary strengths of both models: DenseNet121's deep feature extraction capabilities and the custom CNN's specific architectural design.

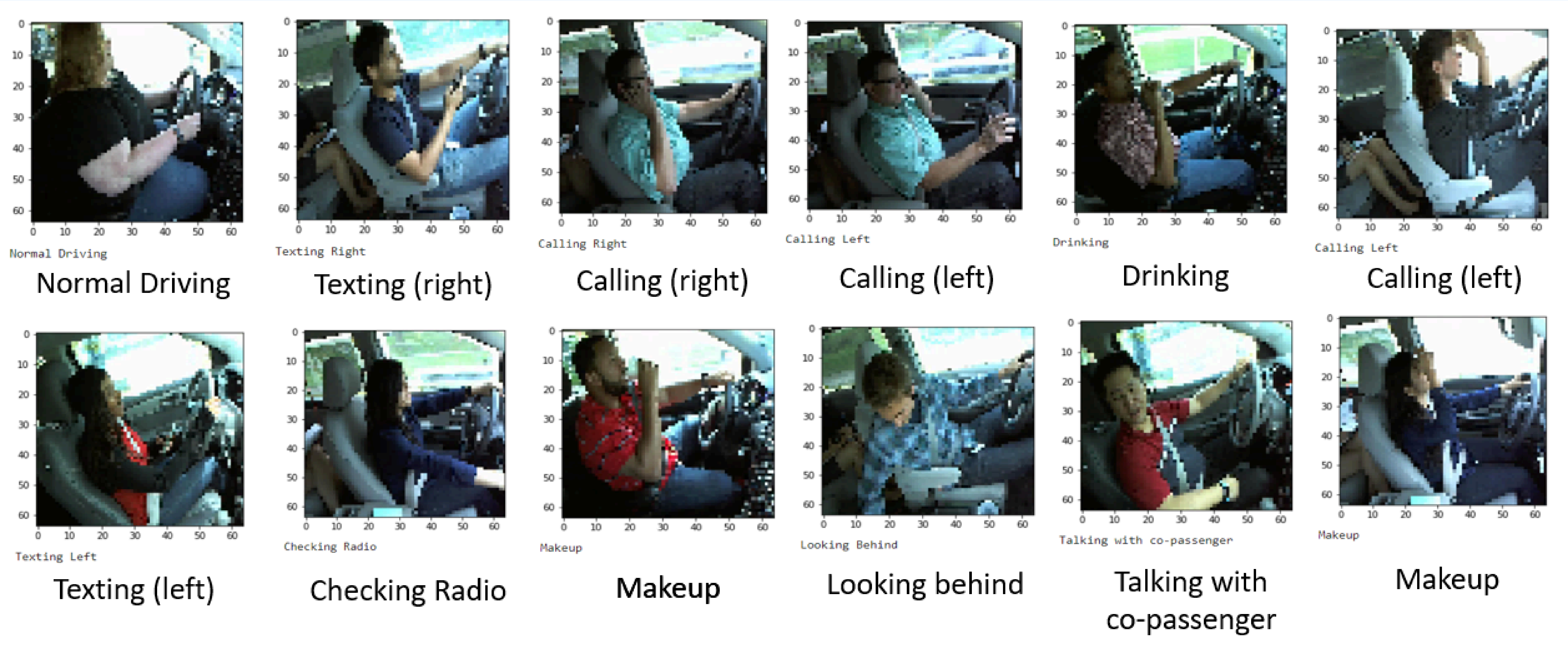
# RESULTS - ENSEMBLE MODEL




```python
# Accuracy
correct_classifications=0
for i in range(len(test_images)):
    if pred_vals[i]==test_vals[i]:
        correct_classifications=correct_classifications+1
print("\n\nAccuracy of the ensemble model:",(correct_classifications/len(test_images))*100)
```

```
Accuracy of the ensemble model: 99.2420864913063
```

# SAMPLE RESULTS



# DEMO

# COMPARATIVE ANALYSIS

## TESTING ACCURACY COMPARISION

| Model Name | Training Loss | Training Accuracy | Validation Loss | Validation Accuracy | Testing Loss | Testing Accuracy |
|---|---|---|---|---|---|---|
| DARNET | 0.0056 | 99.81% | 0.0357 | 99.24% | 0.0390 | 98.60% |
| MobileNetV2 | 0.0196 | 99.94% | 0.2153 | 94.34% | 0.1860 | 94.69% |
| DenseNet121 | 0.0027 | 100.00% | 0.1329 | 96.65% | 0.0950 | 97.28% |
| VGG16 | 0.0898 | 96.97% | 0.2455 | 93.58% | 0.2703 | 92.60% |
| Ensemble Model | --- | --- | --- | --- | --- | 99.24% |

# COMPARATIVE ANALYSIS

## CLASS-WISE ACCURACY COMPARISION

| Class Label | Class Name | DARNET | VGG16 | MobileNetV2 | DenseNet121 | Ensemble |
|---|---|---|---|---|---|---|
| C : 0 | Safe driving | 99.59 | 93.17 | 95.18 | 97.18 | 99.20 |
| C : 1 | Texting (right) | 100.00 | 95.57 | 97.78 | 99.55 | 99.55 |
| C : 2 | Cell-phone talking (right) | 100.00 | 96.55 | 96.12 | 97.41 | 99.13 |
| C : 3 | Texting (left) | 98.72 | 94.46 | 94.89 | 96.17 | 99.13 |
| C : 4 | Cell-phone talking (left) | 99.57 | 93.56 | 96.56 | 97.85 | 98.72 |
| C : 5 | Operating the radio | 98.70 | 95.67 | 96.96 | 98.70 | 100.00 |
| C : 6 | Drinking | 99.57 | 89.47 | 92.70 | 99.42 | 99.57 |
| C : 7 | Reaching Behind | 100.00 | 94.00 | 97.50 | 98.50 | 99.50 |
| C : 8 | Hair and makeup | 98.42 | 81.67 | 89.52 | 91.62 | 100.00 |
| C : 9 | Talking to passengers | 99.53 | 90.61 | 94.33 | 97.18 | 97.69 |
| **Overall Accuracy** | | **98.60** | **92.59** | **94.69** | **97.28** | **99.24** |

# CONCLUSION

- In this work, we have significantly improved the Distracted Driver Behavior Monitoring System to precisely recognize various driver activities through Transfer Learning through three stages.
- A new model is proposed and has been trained by fine tuning the parameters to better suit the State Farm Dataset.
- Applied transfer learning using various popular deep learning architectures like MobileNetV2 , VGG16, and DenseNet121.
- The results of all these architectures were compared with that of my model (DARNET) and provided a comparative analysis.
- According to the observations, for multi-class classification the ensemble model is attaining the best accuracy (99.24%) when compared to any other existing architectures.

# FUTURE SCOPE

- In the future, much more data will be analyzed with AUC Dataset as there is no scope of improvement using the State Farm Dataset (because the results were highly saturated)
- All the currently explored methods and techniques will be implemented with the data from AUC to increase the robustness and recognition accuracy of the system.
- And furthermore to improve efficiency of the training process of models, there is a technique of foreground extraction through segmentation which will remove all unnecessary parameters in the image so that it indirectly leads to better accuracy.

# REFERENCES


- Md. Uzzol Hossain , Md. Ataur Rahman , Md. Manowarul Islam , Automatic driver distraction detection using deep convolutional neural networks, Intelligent Systems with Applications, Volume 14, May 2022, 200075
- Prof. Pramila M. Chawan, Shreyas Satardekar, Distracted Driver Detection and Classification, Journal of Engineering Research and Application, ISSN : 2248-9622, Vol. 8, Issue4 (Part -III) April 2018, pp60-64
- https://www.kaggle.com/competitions/state-farm-distracted-driver-detection/data
- Y.Xing, C.Lv, H.Wang, D.Cao, E.Velenis, F.Y.Wang, Driver activity recognition for intelligent vehicles: A deep learning approach IEEE Trans. Veh. Technol., (2019), 5379—5390.
- Yehya Abouelnaga, Hesham M. Eraqi, Mohamed N. Moustafa, AUC Distracted Driver Dataset, 2017, The American University in Cairo

# ~~THANK YOU~~

- MANOHAR REDDY TIPPIREDDY - (110134650)
- KATHAR PATCHA ABDUL RAHIM - (110134610)
- ASIF KASIM LOHAR - (110137509)