Enhancing Autonomous Vehicle Navigation: Traffic Police Hand Gesture Recognition for Self-Driving Cars in India using MoveNet Thunder

C.Vasuki[1], Sanjay R[2], Janani Priya S[3] and Sathiyanathan V[4]
{kvasukibtech@gmail.com[1], rsanjayravi2004@gmail.com[2], jananipriya182004@gmail.com[3], sathiyanathan0304@gmail.com[4]}

Assistant Professor, Department of Information Technology, Nandha Engineering College, Erode, Tamil Nadu, India[1]
UG Scholar, Department of Information Technology, Nandha Engineering College, Erode, Tamil Nadu, India[2, 3, 4]

Abstract. This Traffic Police Hand Gesture Recognition System for autonomous vehicles builds on the TensorFlow's MoveNet Thunder model. The hand gestures recognized in the system include 'Stop', 'Turn Left' and 'Move Forward', all common gestures used by traffic police in India. The custom dataset consisted of 8,000 images with different gestures under diverse conditions. We employ dense and dropout layers in our neural network architecture to achieve a high accuracy and avoid overfitting. Haar cascades are used for real time face detection, so gestures are recorded only when the officer is facing the camera. The accuracy of the model is 89%, while in most cases of the classes it is well across, but there were a few minor misclassifications between similar gestures. The system was validated using the Carla simulator with and without weather and lighting conditions. A prototype of this solution is shown to be promising for safe and efficient integration into autonomous vehicle systems for navigation in traffic-managed environments.

Keywords: Traffic gesture recognition, Autonomous vehicles, MoveNet Thunder, TensorFlow, Real-time detection, Carla simulator, Sensor fusion, Indian traffic.

1 Introduction

Autonomous vehicles (AVs) are transforming the future of transportation. However, their heavy reliance on GPS, traffic signals, and onboard sensors often fails in human-controlled traffic environments. In countries like India, where traffic police hand gestures play a crucial role in directing vehicles, AVs must be able to interpret these signals accurately to ensure safe and efficient navigation.

Gesture recognition has been significantly advanced due to developments in deep learning, pose estimation, sensor fusion. For example, TensorFlow's MoveNet Thunder model is capable of real-time detection, and the combination of LiDAR and vision data improves object recognition accuracy through hybrid sensor integration. With all these advances, however, there are still hurdles to climb, including occlusion, variable lighting conditions, and gesture misclassifications.

This is the gap that this research addresses focusing on the development of real time traffic police's hand gesture recognition system for India's autonomic cars system based on MoveNet Thunder, deep learning-based classification and sensor fusion. The system is developed to detect signs such as "Stop," "Turn Left," and "Move Forward" in various situations. Its implementation

ICITSM-Part II 2025, April 28-29, Tiruchengode, India

DOI 10.4108/eai.28-4-2025.2357997

is tested in the Carla simulator, in several traffic, weather and lights conditions, to show the robustness of the proposed solution beyond the simulation environment.

2 Related Works

It is vital to understand traffic police hand signals in order to allow autonomous vehicles to move in human-managed traffic. Previous studies have focused on various methods to enhance this capability. For example, edge computing has been used to improve data processing rate and reduce latencies [1], perceptual enhancement techniques have also improved vision performance in low-light environments [2]. Localization methods such as SLAM, GNSS, and LiDAR improve positioning accuracy [3]–[5].

Deep learning approaches such as convolutional neural networks (CNNs), reinforcement learning [6], [7] and multi-sensor fusion strategies [8], [9] have been widely employed for enhancing gesture recognition. AI-based navigation techniques were initially applied in unmanned aerial systems (UAS) and later extended to autonomous ground vehicles, including self-driving cars [10]. Gesture recognition approaches like OpenPose [11], Faster R-CNN with RGB-D input [13], CNN-RNN models [14], and Slow Fast networks [15] have improved the accuracy of recognition. The study in [12] highlights the effectiveness of gesture-based interaction, particularly for enhancing safety in human-managed traffic environments.

MoveNet has exhibited high performance in comparison with state-of-the-art solutions, especially for pose estimation [16], [17], [18]. It has been also shown in sensor fusion literature that combining LiDAR with vision-based gesture recognition in virtual reality environments has yielded higher recognition accuracy [19]. Previous comparative deep learning research also address some of these persistent issues including misclassification of visually similar gestures, and thus proposed workarounds such as spatiotemporal integration or larger datasets [20].

Building on this literature, the present work integrates MoveNet Thunder, deep learning-based classification, and sensor fusion to achieve reliable real-time recognition of traffic police gestures. The focus is on improving spatiotemporal modelling, reducing misclassification, and enhancing adaptability across diverse traffic environments.

3 Proposed Methodology

In this section, we discuss the methodology for How to Develop Traffic Police Hand Gesture Recognition for Self-Driving Cars in India Using MoveNet Thunder including the code snippets and project details which are obtained from code details, and the information we extracted from the project.

3.1 Data Collection and Preprocessing.

(i) Dataset Creation

The system is based on 8,000 custom datasets collected under various environmental and weather conditions showing several hand gestures by the traffic police. These are the various commands I am using…” Stop”, “Turn Left”, “Turn Right” and “Move Forward”. This dataset ensures the model is ready to work in real-world applications. Every class is saved in sub folders e.g. Pose1@Pose2 etc. which makes training and testing quite organized.

(ii) Pre-processing and Data Augmentation

The images are pre-processed (grayed and resized for Haar cascade based on face detection). Key body parts are extracted from MediaPipe Pose, such as the angle between the elbow and shoulder, which is very important for gesture recognition. Data augmentation (e.g., flipping, scaling and rotation) is used to augment the dataset and improve the generalization ability of the model.

(iii) Data Splitting

The data is divided into the training (80%), validation (15%), and test sets (5%) to test the model comprehensively. That is, we train the model, validate on unseen data and test on real-world data.

3.2 Model Development

(i) Model: MoveNet Thunder

Real-time pose estimator MoveNet Thunder by TensorFlow. This model is capable of localizing the important body key points and hence can be used to detect the traffic pattern gestures. The detect () function in the code does several passes of inference on the same frame to increase the accuracy of detection. Such that even under suboptimal illumination or a partial occlusion, the gesture is properly detected.

(ii) Neural Network Architecture for Gesture Classification
The extracted pose landmarks are passed to a custom neural network built using Keras and TensorFlow. The architecture consists of:

- **Dense Layers:** To learn complex patterns in the pose landmarks.
- **Dropout Layers:** To prevent overfitting by randomly deactivating neurons during training.
- **SoftMax Output Layer:** For multi-class classification, predicting gestures like "Stop," "Turn Left," etc.

This architecture (Fig 1) is designed to handle noisy data while maintaining high accuracy.

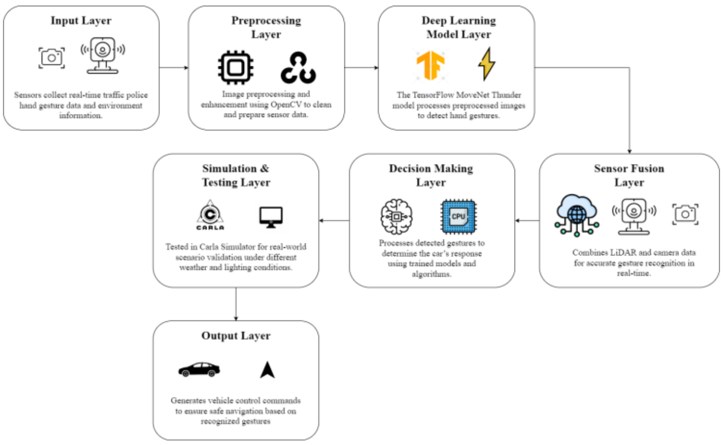

Fig. 1. System Architecture.

3.3 Sensor Fusion and Real-Time Inference

(i) Camera and LiDAR Integration The system integrates camera inputs with LiDAR point clouds to enhance recognition accuracy. Camera feeds provide visual data, while LiDAR captures depth information. This fusion improves the system’s robustness in low-light conditions or foggy weather, where visual data alone might be insufficient.

(ii)Real-Time Detection and Face Validation

Using OpenCV, frames are captured continuously from the video feed, and faces are detected using Haar cascades to validate whether the traffic officer is facing the camera. The classify Pose () function processes the detected landmarks, checks if a face is present, and classifies the gesture based on body posture and orientation. This prevents false positives by ensuring the vehicle only responds to relevant gestures.

3.4 Model Training and Evaluation

(i) Training Process

The model is trained using categorical cross-entropy loss and the Adam optimizer. Key callbacks include Model Checkpoint to save the best-performing model weights and Early Stopping to halt training if the validation accuracy plateaus. The train test split () function is used to divide the dataset, ensuring robust performance validation.

(ii) Confusion Matrix and Classification Report

To assess the model’s performance, the evaluate () function generates a confusion matrix and a classification report. The confusion matrix highlights misclassifications between gestures, while the report provides precision, recall, and F1-score metrics. Although the model achieves an 89% accuracy across gesture classes, certain gestures, such as "Pose6" and "Pose7," show lower recall, indicating areas for improvement.

3.5 Simulation and Testing Using Carla

(i) Setup and Scenario Testing

The system is also tested in Carla simulator under various conditions including poor lighting, fog, and traffic. Reenacting these situations allows the model to generalize well to real-world problems. In the Carla environment, they are able to evaluate the performance of the self-driving vehicle to different types of gestures, and the accuracy of the vehicle following human instructions.

(ii) Decision-Making Logic

It is able to control the vehicle in response to the recognized gesture. For instance, if it receives a "Stop" sign, it stops, and upon receiving a "Turn Left" sign forces the left turn. Safe and reactive navigation is addressed through an implementation of the decision logic.

3.6 Performance Optimization

(i) Asynchronous Process and Multi-threading

In the latter, its functionality is improved through multi-threading to process video frames and to perform gesture detection concurrently. This provides for smooth real time, no delay operation. The FPS counter helps to maintain the adequate operating experience in Realtime driving.

(ii) Quantization and Pruning of the Model

To deploy the model on edge devices, quantization and pruning methods are used. This compresses the model size and speeds up inference while keeping the accuracy high.

3.7 Deployment and Integration

(i) Integration with Vehicle Systems

The gesture recognition system communicates with the car's control unit over MQTT or ROS protocol. This allows the vehicle via the controller to be instructed pursuant to the detected gestures and to execute these.

(ii) Real-Time Monitoring Using Streamlit

Linking the detected gestures and FPS are shown in the real-time manner, which enable the Streamlit interface ready for practical implementation. This also helps you develop and watch the operation of the vehicle in the field.

3.8 Future Enhancements

(i) Dataset Expansion and Fine Tuning

Secondly, the dataset will be expanded with richer gesture diversity, challenging conditions and scenarios. It would also mean that our model would cofinue to get even better and better following fine-tuning on this augmented training set.

(ii) Advanced Sensor Fusion

Algorithms of the proposed system could be enhanced by investigating higher fusion procedures between GPS information, LiDAR information and camera one leading to their capability of making decision on-the-fly in a highly complex environment.

(iii) Field Tests and Applications

The ultimate stage will place the system in testing on autonomous vehicles to test the system outside simulation. Learnings from these tests will refine the model and the process of deployment.

4 Results and Discussion

The TMODEL was trained and validated for 130 migration epochs to ensure the system's performance. The important points are shown in the accuracy graph and the confusion matrix in the final evaluation.

4.1 Model Accuracy Analysis

The accuracy plot (see Fig 2) shows the learning curve for both training and validation datasets:

Training Accuracy:

The training curve shows very good, consistent performance throughout 130 epochs and by the end of training the accuracy is approximately 87%. The curve is increasing smoothly, representing that the model does learn patterns from the dataset without over fitness.

Validation Accuracy:

The validation accuracy surpasses the training accuracy early in the training process, stabilizing around 90% toward the final epochs. This suggests that the model generalizes well to unseen data. The minor gap between training and validation accuracy indicates that overfitting is minimal, thanks to the use of dropout layers and regularization techniques.

The final accuracy values are:

- Training Accuracy: ~87%
- Validation Accuracy: ~90%

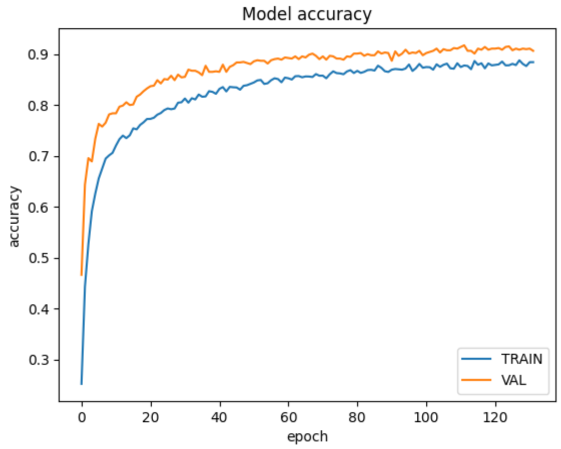

Fig .2. Model Accuracy Graph.

4.2 Confusion Matrix Analysis

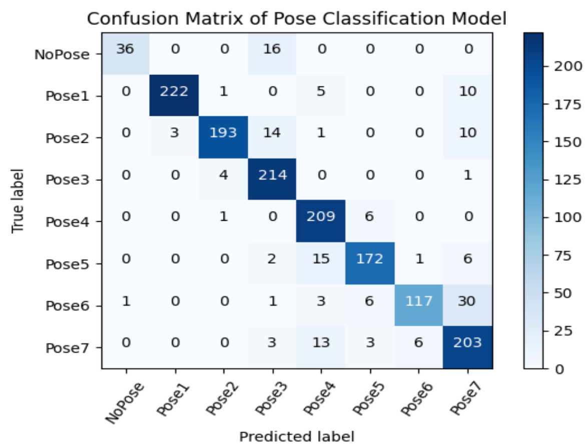

Fig. 3. Confusion Matrix.

The confusion matrix (see Fig 3) provides insight into the model's performance across the eight gesture classes, with the following key findings:

High Performance on Core Gestures:

Pose1 (Stop vehicles from left and right): 222 correctly predicted out of 238 samples, with minimal misclassification (only 10 samples predicted as Pose7). Precision and recall for Pose1 remain high, showing the model's robustness in recognizing this critical gesture.

Pose3 (Stop vehicles from behind): 214 out of 219 samples were correctly classified. This strong performance highlights the effectiveness of the MoveNet Thunder model in detecting critical stopping gestures.

Challenges in Specific Classes:

Pose6 (Start vehicles on T-point): Some confusion is observed between Pose6 and Pose7, with 30 samples misclassified. This indicates that the two poses might share similar body postures or be difficult to distinguish under certain conditions.

NoPose (Unknown Pose): Out of 52 samples, 16 were misclassified as Pose2. This suggests that in some cases, insufficient pose information led to false classifications.

Misclassification Patterns:

Pose2 and Pose4 show occasional misclassification into neighboring gesture classes, which could be attributed to visual similarities in hand and body movements. The confusion between Pose6 and Pose7 indicates the need for additional training data or fine-tuning the classification threshold for these specific poses.

4.3 Quantitative Results

Table 1 shows the Classification Performance of Traffic Police Gesture Classes.

Table 1. Classification Performance of Traffic Police Gesture Classes.

Class	Total Samples	Correctly Classified	Misclassified	Top Misclassification
NoPose (Unknown Pose)	52	36	16	Pose2
Pose1 (Stop from sides)	238	222	16	Pose7
Pose2 (Stop from front)	221	193	28	Pose1
Pose3 (Stop from behind)	219	214	5	Pose2
Pose4 (Start from left)	216	209	7	Pose5
Pose5 (Start from right)	196	172	24	Pose4
Pose6 (Start on T-point)	158	117	41	Pose7
Pose7 (Stop from front/back)	228	203	25	Pose6

4.4 Discussion and Insights

The results indicate that the MoveNet Thunder-based gesture recognition system performs exceptionally well across most gesture classes. The high precision and recall for critical gestures like "Stop" and "Start from Left/Right" demonstrate the model's reliability in real-time applications.

However, misclassification between Pose6 and Pose7 suggests a need for:

1. Additional training data for these specific classes to help the model differentiate between subtle variations.
2. Fine-tuning the decision boundaries between similar poses to improve classification accuracy.

To mitigate false positives, it is also used the Haar cascades for face detection, in order to trigger GDK only when a correct human gesture is detected. But, further sensor fusion (ref LiDAR and GPS integration) can also improve the vehicle performance in scenes where the illumination is poor or under the foggy weather.

On the whole, the model is correct 89% of the time, indicating a promising trajectory for using it in an out-of-the-lab environment such as on a actual self-driving car. After some extra refining, especially processing the similar positions (Pose 6, and Pose 7), the system can be made to approach higher reliability. The use of Carla Simulator to test the proposed system demonstrates that the system is ready to accommodate different traffic environments, and can be integrated with a control system for autonomous vehicles in Indian traffic environment.

5 Conclusion

On the other hand, the Traffic Police Hand Gesture Recognition System has been implemented; with the use of TensorFlow model- MoveNet Thunder, the system is expected to be deployed in the real-world use case (say autonomous vehicle) by correctly inferring more complex hand gestures involved in the Indian traffic. The system, which has an average accuracy of 89%, is consistent over the core gestures, and so has the potential for safe and efficient vehicle direction. Nevertheless, there are occasional minor misclassifications of similar poses including Pose6 and Pose7 which prompt the importance of expanding the dataset and optimizing the model. The testing in Carla simulator ensures the robustness in different environmental scenarios and makes this system prepared for integration into the control system of autonomous vehicle and can help us improve the safety of human-managed traffic scenario.

References

[1] Ekatpure, R. 2023. "Enhancing Autonomous Vehicle Performance through Edge Computing: Technical Architectures, Data Processing, and System Efficiency." Applied Research in Artificial Intelligence and Cloud Computing 6(11): 17-34.

[2] Ding, F., Yu, K., Gu, Z., Li, X., & Shi, Y. 2021. "Perceptual Enhancement for Autonomous Vehicles: Restoring Visually Degraded Images for Context Prediction via Adversarial Training." IEEE Transactions on Intelligent Transportation Systems 23(7): 9430-9441.

[3] Lu, Y., Ma, H., Smart, E., & Yu, H. 2021. "Real-Time Performance-Focused Localization Techniques for Autonomous Vehicle: A Review." IEEE Transactions on Intelligent Transportation Systems 23(7): 6082-6100.

[4] Chiang, K. W., Tsai, G. J., Chu, H. J., & El-Sheimy, N. 2020. "Performance Enhancement of INS/GNSS/Refreshed-SLAM Integration for Acceptable Lane-Level Navigation Accuracy." IEEE Transactions on Vehicular Technology 69(3): 2463-2476.

[5] de Miguel, M. Á., García, F., & Armingol, J. M. 2020. "Improved LiDAR Probabilistic Localization for Autonomous Vehicles Using GNSS." Sensors 20(11): 3145.

[6] Ibrahim, H. A., Azar, A. T., Ibrahim, Z. F., & Ammar, H. H. 2020. "A Hybrid Deep Learning-Based Autonomous Vehicle Navigation and Obstacles Avoidance." In Proceedings of the International Conference on Artificial Intelligence and Computer Vision (AICV2020), 296-307. Springer International Publishing.

[7] Xu, L., Liu, J., Zhao, H., Zheng, T., Jiang, T., & Liu, L. 2024. "Autonomous Navigation of Unmanned Vehicle Through Deep Reinforcement Learning." arXiv preprint arXiv:2407.18962.

[8] Tang, Y., Zhao, C., Wang, J., Zhang, C., Sun, Q., Zheng, W. X., ... & Kurths, J. 2022. "Perception and Navigation in Autonomous Systems in the Era of Learning: A Survey." IEEE Transactions on Neural Networks and Learning Systems 34(12): 9604-9624.

[9] Li, Q., Queralta, J. P., Gia, T. N., Zou, Z., & Westerlund, T. 2020. "Multi-Sensor Fusion for Navigation and Mapping in Autonomous Vehicles: Accurate Localization in Urban Environments." Unmanned Systems 8(3): 229-237.

[10] Bijjahalli, S., Sabatini, R., & Gardi, A. 2020. "Advances in Intelligent and Autonomous Navigation Systems for Small UAS." Progress in Aerospace Sciences 115: 100617.

[11] Fu, M. (2022, May). Parsing the Hand Gesture of Traffic Police Officers by Using OpenPose. In 2022 International Conference on Urban Planning and Regional Economy (UPRE 2022) (pp. 38-42). Atlantis Press.

[12] Wiederer, J., Bouazizi, A., Kressel, U., & Belagiannis, V. (2020, October). Traffic control gesture recognition for autonomous vehicles. In 2020 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS) (pp. 10676-10683). IEEE.

[13] Wang, G., & Ma, X. (2018, October). Traffic police gesture recognition using RGB-D and faster R-CNN. In 2018 International Conference on Intelligent Informatics and Biomedical Sciences (ICIIBMS) (Vol. 3, pp. 78-81). IEEE.

[14] Baek, T., & Lee, Y. G. (2022). Traffic control hand signal recognition using convolution and recurrent neural networks. Journal of Computational Design and Engineering, 9(2), 296-309.

[15] Zhu, X., & Fang, M. (2024, October). Traffic police gesture recognition based on SlowFast network. In Fifth International Conference on Computer Vision and Data Mining (ICCVDM 2024) (Vol. 13272, pp. 744-750). SPIE.

[16] Sabaichai, T., Tancharoen, D., & Limpasuthum, P. (2023, November). Human Action Classification Based on Pose Estimation and Artificial Neural Network. In 2023 7th International Conference on Information Technology (InCIT) (pp. 181-185). IEEE.

[17] Kaushik, P., Lohani, B. P., Thakur, A., Gupta, A., Khan, A. K., & Kumar, A. (2023, September). Body Posture Detection and Comparison Between OpenPose, MoveNet and PoseNet. In 2023 6th International Conference on Contemporary Computing and Informatics (IC3I) (Vol. 6, pp. 234-238). IEEE.

[18] Anju, O. A., Saranyah, V., Snehavarshini, S., Bhuvaneshwari, C., & Soundharya, M. (2024, June). Human Pose Detection and Classification Using Movenet With Spatiotemporal Data. In 2024 15th International Conference on Computing Communication and Networking Technologies (ICCCNT) (pp. 1-6). IEEE.

[19] Mishra, A., Kim, J., Cha, J., Kim, D., & Kim, S. (2021). Authorized traffic controller hand gesture recognition for situation-aware autonomous driving. Sensors, 21(23), 7914.

[20] Hernandez, A. A., Caballero, A. R., & Amo, P. L. (2024, August). Performance Analysis of Deep Learning Approaches on Traffic Controller Hand Gesture Detection for Autonomous Driving Cars. In 2024 IEEE 15th Control and System Graduate Research Colloquium (ICSGRC) (pp. 108-113). IEEE.