CARAN: The Car and Retina Automated Navigation System

Muskan Sinha
Department of Computer Science
Netaji Subhas University of Technology,
New Delhi, India
muskan.sinha.ug21@nsut.ac.in

Aparna Srivastava
Department of Computer Science
Netaji Subhas University of Technology,
New Delhi, India
Aparna.Srivastava.ug21@nsut.ac.in

Pulkit Batra
Department of Computer Science
Netaji Subhas University of Technology,
New Delhi, India
pulkit.batra.ug21@nsut.ac.in

***Abstract*—This paper presents a wireless face tracker system that uses ESP32-CAM module as a wireless camera and XBee modules for communication. This system establishes connection with ESP32-CAM to stream frames to a Python script running on local system. These algorithms analyze frames using Python script applying face detection and tracking techniques to compute coordinates and distance of detected faces.**

When detected, the program calculates adjustments that would put the face at the center of the frame. These adjustments are then converted into control signals which are sent via serial communication to an arduino board. The Arduino board is equipped with an XBee module, which sends wireless instructions to motorized vehicle.

Motorized vehicle uses also an XBee module, receives these instructions, and performs driving commands so that it places the identified face at the center of camera's sight. When there are no human being involved directly for this real-time face tracking facility then it shows how efficient it can be in reality.

Further, this capability extends also to defense applications specifically missile target tracking. Through integrating missile guidance systems with this wireless technology for facial recognition, this system may improve accuracy.

I. Introduction

In today's fast-changing security world, there's a growing need for better ways to watch and follow targets in the defense area. Enemies are always coming up with smart tricks to hide or escape, pushing defense teams all over to up their game in finding, tracking, and stopping threats well. For sure, being able to track targets quickly and right on the spot is super key, especially when dealing with fast-moving or hard-to-catch things like missiles.

Old ways of keeping an eye on targets often use fixed or hand-run watch systems, which may not catch up with quick and changing scenes. Also, when people are in the mix, mistakes or delays can happen, making it harder to keep a good watch on targets.

To beat these problems, using new tech like wireless face tracking is a great idea to make defense better. By using the ESP32-CAM module as a wireless camera and XBee modules for talking without wires, this setup is a handy and quick way to track targets in real time. With face finding and tracking tech, it can spot and keep watch on targets well, even when things move fast or in unpredictable ways.

Also, since this system works without wires and does things on its own, there's no need for people to do it by hand. This cuts down on wait times and the chance of human mistakes. This not only makes tracking tasks work better and faster but also lets the team put their time into other important defense work.

Even more, this system's use isn't just for watching over things; it can also make missile aiming much better. By putting wireless face tracking tech into missile aiming systems, defense groups can hit their targets way more exactly and dependably, making their defense work much stronger.

So, making and using wireless face tracking systems is a big step forward in defense tech. It gives a handy and effective way to face the new challenges of keeping an eye on and tracking targets in today's war scenes.

II. Literature Survey

The literature survey covers several relevant research works on the topic of face recognition and its applications in automotive security and defense systems.

a. *Video-based Face Recognition Technology for Automotive Security*: Chen and Zhang [1] explored the use of face recognition technology for car safety, highlighting its potential in addressing car theft risks. The paper focuses on face recognition through video for automotive applications, discussing techniques like AdaBoost for face detection and 2DFLD for feature extraction to improve the system's performance.

b. *An Economical Car Security Authentication System Based On Face Recognition Structure*: Jaikumar and Jaiganesh [2] proposed a low-cost embedded system that combines facial recognition, GPS, and GSM modules for automotive security. The system detects intruders using a hidden camera and sends images to the owner for identification and GPS positioning, providing real-time monitoring and automatic security features.

c. *Raspberry Pi Based Intelligent Car Anti-Theft System Through Face Recognition Using GSM and GPS*: Kaprakkadan et al. [3] suggested a Raspberry Pi-based car anti-theft system that utilizes face detection, GSM, and GPS technologies. The study addresses the limitations of current car security systems and explores the use of PCA-based face detection and GSM/GPS for theft prevention and vehicle tracking.

d. *A Face Detection and Recognition System for Intelligent Vehicles*: Ishak et al. [4] presented a prototype facial recognition system for intelligent vehicles, focusing on the integration of fast and classical neural networks for face recognition and PCA-LDA techniques for face detection. The results demonstrate promising accuracy in daylight conditions, suggesting potential applications in intelligent vehicle systems.

e. *Research Paper on Car Ignition System Based on Face Recognition*: Murthy et al. [5] explored the evolution of vehicle security, from traditional key-based systems to facial recognition-based car ignition systems. The paper addresses the limitations of traditional methods and offers a more secure and affordable solution using facial recognition techniques such as Haar cascade classifier and SVM-based clustering.

f. *Face Recognition with Local Binary Patterns*: Ahonen et al. [6] presented a method for robust face detection under various conditions, combining Independent Component Analysis (ICA) and Local Binary Pattern (LBP) techniques. The evaluation on several facial databases showed promising results for automotive security applications.

g. *Intelligent Face Recognition Based Multi-Location Linked IoT Based Car Parking System*: Rajyalakshmi and Lakshmanna [7] proposed an Intelligent Facial Recognition Multi-Location IoT-Based Parking System (IFRbMLL-IoT-CPS) to improve parking efficiency, reduce delays, and enhance vehicle safety through the integration of IoT and facial recognition technologies.

h. *Intelligent Car Anti-Theft Face Recognition System*: A.V. et al. [8] developed a low-cost vehicle security system using OpenCV's Facial Recognition Subsystem (FDS), GPS, GSM, and Arduino modules. The integration of image processing, biometric identification, and communication technology highlights its potential to improve vehicle safety.

i. *Face Recognition in Vehicles with Near Infrared Frame Differencing*: Kang et al. [9] addressed the challenge of lighting variation for facial recognition systems in outdoor environments, especially in vehicular applications. The proposed system uses infrared (IR) lighting and frame differencing techniques to provide a robust and consumer-friendly solution for driver identification in cars.

j. *Face Recognition Based Vehicle Starter Using Machine Learning*: Archana et al. [10] presented a face recognition-based vehicle starter system using machine learning techniques. The study highlights the importance of biometric technologies in improving vehicle security and convenience.

These research works demonstrate the growing interest and advancements in the integration of facial recognition technologies for automotive security and defense applications, providing a solid foundation for the proposed CARAN system.

III. Proposed Work

FRMN is a lightweight Human Face Recognition Model, which is built around a new mobile architecture called MobileNetV2 and the ArcFace Algorithm, and is specially designed for embedded devices.

FRMN Models are built on MobileNetV2.

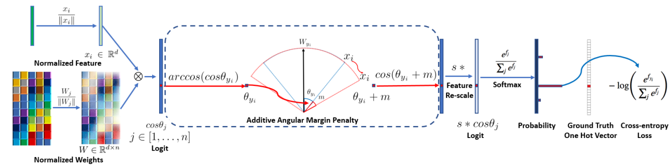

Fig. 1. ArcFace is not only a discriminative model but also a generative model. Given a pre-trained ArcFace model, a random input tensor can be gradually updated into a pre-defined identity by using the gradient of the ArcFace loss as well as the face statistic priors stored in the Batch Normalization layers.

In our training, we use the ArcFace Algorithm, instead of the traditional Softmax Function, and the Cross-Entropy Loss function.

To reduce the computation complexity, we use images of smaller size (56x56) in our training.

A. ArcFace

ArcFace uses similarity learning to enable the solution of classification tasks by learning distance metrics. The loss function can be separated into two different parts, the nominator and denominator. Since we are minimizing the loss, and because our loss function is negative, we would like to increase the nominator and decrease the denominator absolute values:

In the nominator, a cosine similarity between the normalized class embeddings and the class weight is calculated as an inner product between the two vectors. The closer the two vectors are to co-linearity, the closer the cosine similarity would be to 1, the further away, the closer it will be to 0. Thus, the smaller the angles between the two vectors, the larger our nominator, the smaller our loss.

In the denominator, we want to minimize the cosine similarity between our class instance and all the other classes weights.

Algorithm 1 Face Image Generation from the ArcFace Model

Input: model with L BN layers, class label y_i
Output: a batch of generated face images: I^r
Generate random data I^r from Gaussian $(\mu = 0, \sigma = 1)$
Get μ_i, σ_i from BN layers of , $i \in 0, \ldots, L$
for j $= 1, 2, \ldots, T$ **do**
  Forward propagate (I^r) and calculate ArcFace loss
  Get $\tilde{\mu}_i$ and $\tilde{\sigma}_i$ from intermediate activations, $i \in 0, \ldots, L$
  Compute statistic loss $\min \sum_{i=0}^{L} \|\tilde{\mu}_i^r - \mu_i\|_2^2 + \|\tilde{\sigma}_i^r - \sigma_i\|_2^2$,
  Backward propagate and update I^r
end for

[11]

B. MobileNetV2

MobileNetV2 is a convolutional neural network architecture that seeks to perform well on mobile devices. It is based on an inverted residual structure where the residual connections are between the bottleneck layers. The intermediate expansion layer uses lightweight depthwise convolutions to filter features as a source of non-linearity. As a whole, the architecture of

MobileNetV2 contains the initial fully convolutional layer with 32 filters, followed by 19 residual bottleneck layers.

In MobileNetV2, there are two types of blocks. One is a residual block with stride of 1. Another one is a block with stride of 2 for downsizing. There are 3 layers for both types of blocks.

This time, the first layer is a 1x1 convolution with ReLU6.

The second layer is the depthwise convolution.

The third layer is another 1x1 convolution but without any non-linearity. It is claimed that if ReLU is used again, the deep networks only have the power of a linear classifier on the non-zero volume part of the output domain.

Input	Operator	Output
$h \times w \times k$	1x1 conv2d, ReLU6	$h \times w \times (tk)$
$h \times w \times tk$	3x3 dwises=s, ReLU6	$h \times (tk)$
$h \times tk$	linear 1x1 conv2d	S

And there is an expansion factor t. And t=6 for all main experiments. If the input got 64 channels, the internal output would get 64 × t = 64 × 6 = 384 channels.

[12]

IV. Methodology

The research methodology involves integrating IoT components and machine learning algorithms for autonomous navigation, outlined as follows:

1) **Hardware Setup:** Assemble ESP32 cam, Arduino Uno, XBee modules, and L293D motor driver. Configure ESP32 cam for frame capture and TCP/UDP connection. Install XBee modules for wireless communication.
2) **Software Configuration:** Develop Python script for frame retrieval from ESP32 cam. Integrate OpenCV for face detection and algorithms for positional estimation.
3) **Communication Protocol:** Establish TCP/UDP communication between Python script and ESP32 cam. Configure UART serial communication between Arduino Uno and XBee modules.
4) **Face Detection and Position Estimation:** Utilize machine learning for real-time face detection. Develop algorithms for positional changes and distance estimation.
5) **Control Algorithm:** Design logic for interpreting positional data and issuing directional commands to Arduino Uno. Translate commands for motor driver control.
6) **System Integration and Testing:** Integrate hardware and software components. Test functionality, interoperability, and performance.
7) **Iterative Optimization:** Refine algorithms and control logic based on feedback. Optimize system parameters and fine-tune machine learning models and control algorithms.

This methodology facilitated the development of a fully functional IoT system with machine learning for autonomous navigation, ensuring successful implementation of each component and achieving research objectives.

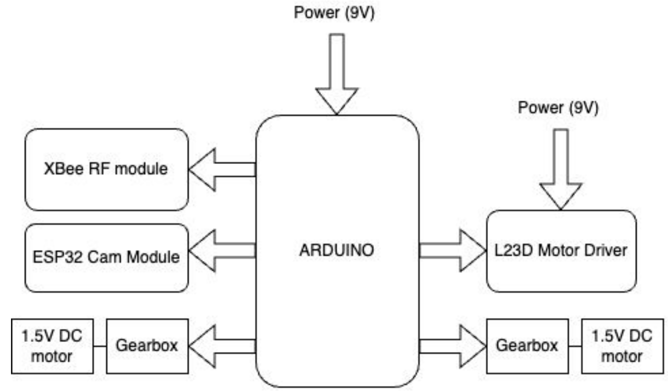

Fig. 2. Block Diagram of the System

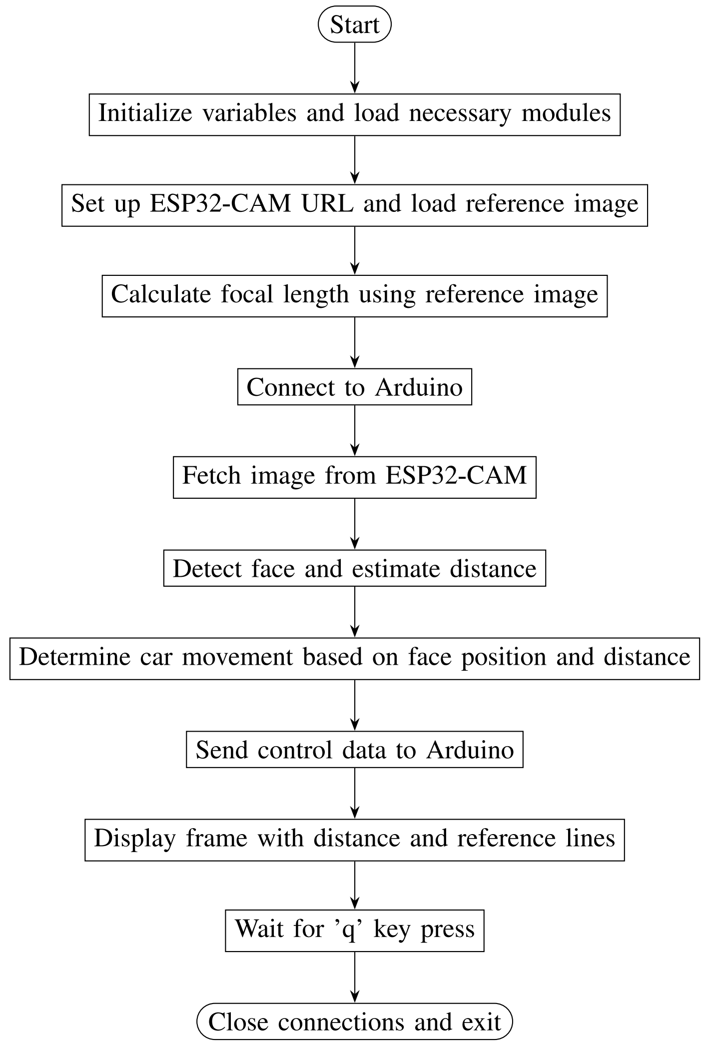

Fig. 3. Flowchart of the Face Detection and Distance Estimation System

V. Result

The conducted research has successfully demonstrated the development and implementation of an integrated IoT system combined with machine learning techniques for autonomous

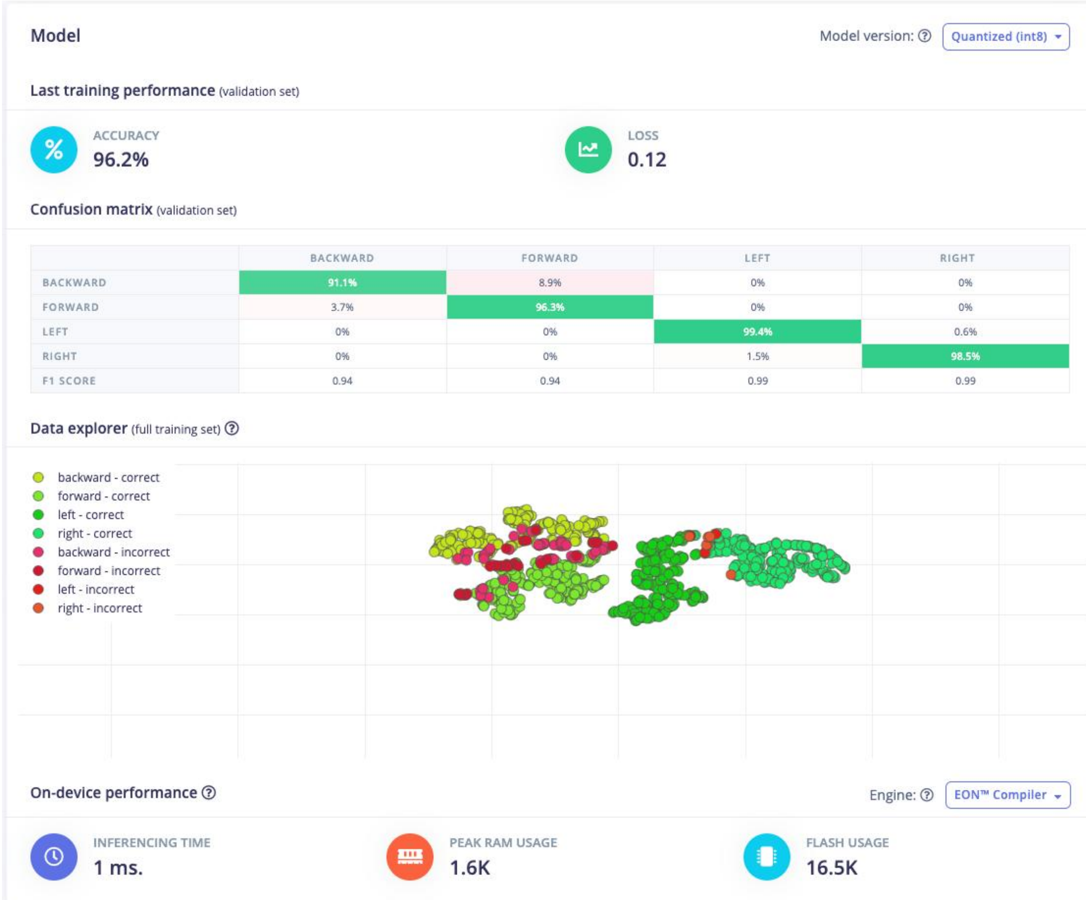

Fig. 4. Result

navigation. Through the utilization of ESP32 cam, Arduino Uno, and XBee modules, coupled with machine learning algorithms for face detection and positional estimation, the system has shown promising results in achieving autonomous navigation capabilities.

The evaluation of the system's performance revealed satisfactory levels of accuracy, responsiveness, and robustness under varying environmental conditions. The machine learning algorithms effectively detected faces in real-time, while the control algorithms interpreted positional data and issued directional commands accordingly. The system demonstrated the ability to navigate autonomously based on the detected faces, adjusting its movements to maintain a centered position and approximate distance from the camera lens.

Furthermore, iterative optimization efforts have led to enhancements in system efficiency and reliability. Fine-tuning of machine learning models and control algorithms has improved the overall accuracy and responsiveness of the system, making it suitable for practical applications in surveillance, robotics, and human-machine interaction.

Overall, the developed system represents a significant contribution to the field of IoT and machine learning, showcasing the potential for integrating these technologies to enable autonomous navigation. Future research directions may include further optimization of algorithms, exploration of additional applications, and scalability to accommodate more complex environments and scenarios.

In conclusion, the presented research underscores the feasibility and effectiveness of leveraging IoT and machine learning for autonomous navigation, laying the groundwork for advancements in intelligent systems and robotics.

References

[1] R. B. Chen and S. Zhang, "Video-based face recognition technology for automotive security," in *2010 International Conference on Mechanic Automation and Control Engineering*, 2010, pp. 2947–2950.

[2] K. Jaikumar and B. Jaiganesh, "An economical car security authentication system based on face recognition structure," *International Journal of Technology Enhanced and Emerging Engineering Research*, vol. 2, no. 8, pp. 28–31, 2014.

[3] M. S. Kaprakkadan, A. T, and R. Ram, "Raspberry pi based intelligent car anti-theft system through face recognition using gsm and gps," 2016.

[4] K. A. Ishak, S. A. Samad, and A. Hussain, "A face detection and recognition system for intelligent vehicles," *Information Technology Journal*, vol. 5, pp. 507–515, 2006.

[5] Y. P. Murthy, C. Y. Goud, N. H. Vardhan, and T. A. Kumar, "Face recognition system based car security system," *International Journal of Future Mobile Robotics*, vol. 5, no. 6, pp. 120–128, Nov-Dec 2023.

[6] T. Ahonen, A. Hadid, and M. Pietikäinen, "Face recognition with local binary patterns," in *European Conference on Computer Vision*, 2004, pp. 469–481.

[7] V. Rajyalakshmi and K. Lakshmanna, "Intelligent face recognition based multi-location linked iot based car parking system," *IEEE Access*, vol. 11, pp. 84 258–84 269, 2023.

[8] A.V., S.K., S.P., S.D., R. Gopi, and S. T, "Intelligent car anti-theft face recognition system," *International Journal of Online Engineering*, vol. 17, no. 01, pp. 120–128, Jan 2021.

[9] J. Kang, D. Anderson, and M. Hayes, "Face recognition in vehicles with near infrared frame differencing," in *2015 IEEE International Workshop on Signal Processing Education*, 2015, pp. 358–363.

[10] P. Archana, P. Divyabharathi, S. Balaji, N. Kumareshan, P. Veeramanikandan, S. Naitik, S. Rafi, P. Nandankar, and G. Manikandan, "Face recognition based vehicle starter using machine learning," *Measurement: Sensors*, vol. 24, p. 100575, 2022.

[11] J. Deng, J. Guo, N. Xue, and S. Zafeiriou, "Arcface: Additive angular margin loss for deep face recognition," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 43, no. 10, pp. 3814–3828, 2021.

[12] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, Kaiser, and I. Polosukhin, "Attention is all you need," *arXiv preprint arXiv:1801.04381*, 2017.