Iron Shield Real Time Defense System Using Neural Network

Shreevarsinii Balasekar
Graduate student of School of Artificial Intelligence
Amrita Vishwa Vidyapeetham, Coimbatore-641112,India.
shreevarsinii@gmail.com

Sanjay R.
Graduate student of School of Artificial Intelligence
Amrita Vishwa Vidyapeetham, Coimbatore-641112,India.
sanjayrajendran@gmail.com

Srijan Sivaram A.
Graduate student of School of Artificial Intelligence
Amrita Vishwa Vidyapeetham, Coimbatore-641112,India.
srijansivaram@gmail.com

Snendar M S.
Graduate student of School of Artificial Intelligence
Amrita Vishwa Vidyapeetham, Coimbatore-641112,India.
dawgsonstreet@gmail.com

Dr. S. Manimaran*
Assistant Professor
Amrita School of Artificial Intelligence, Coimbatore,
Amrita Vishwa Vidyapeetham, India.
s_manimaran@cb.amrita.edu

Dr. Prem Jagadeesan*
Assistant Professor
Amrita School of Artificial Intelligence, Coimbatore,
Amrita Vishwa Vidyapeetham, India.
j_prem@cb.amrita.edu

***Abstract*—Traditional missile defense systems tend to use rule-based algorithms or static models with no flexibility for different types of threats and real-time inference demands. The work here proposes an AI-based missile interception decision system with physics-based trajectory modeling coupled with machine learning to improve classification accuracy. The synthetic data mimicking the paths of missiles under gravitational effect with noise was used to train a feedforward neural network optimized by using Keras Tuner. To enhance the reliability of decision, calibrated thresholding using the F1-score was utilized and the final model was also exported to TensorFlow Lite to deploy it in real time on edge devices. A physics-based override feature also guarantees correct avoidance of non-threatening trajectories. The system ensures high accuracy with 99% after 10 trials for discrimination between threat and non-threat scenarios with minimal latency and low resource utilization, providing a deployable and intelligent solution for today's defense needs with the help of the radar to find the intercept.**

***Index Terms*—Missile Interception, Machine Learning, Physics-Based Modeling, Neural Networks, TensorFlow Lite, Edge Deployment, Automated Tuning, Defense Systems.**

I. Introduction

Recently, with the rapid development of defense technologies, there is increased necessity for intelligent systems with the capacity to make decisions that may result in live or die, on real time. Missile intercept is one relevant area of national security that requires timely and accurate evaluation of threats to protect significant assets and civilians. Traditional missile intercept systems rely on pre-programmed rules, heuristics models, or fixed trajectory prediction models. These methods have been operationally sound, but they do not provide the adaptability, scalability, or efficiency in highly dynamic or uncertain threat contexts [1].

In transcending these limitations, machine learning (ML) and artificial intelligence (AI) are becoming increasingly advanced as powerful tools to enhance decision-making in defense applications. ML algorithms can handle complex, high-dimensional data and identify subtle features that can distinguish genuine threats from insignificant trajectories. This ability is particularly valuable in the case of missile defense where a misclassification would result in launching unnecessary countermeasures or failing to intercept the threat altogether. But most ML models used in defense applications are either designed for offline processing or deep learning-type models that take too much computing capacity to execute, which makes their use limited for edge scenarios and time-critical inference, often a requirement to success in defense operations [2].

This paper suggests an intelligent system of missile interception based on a feedforward neural network that has been trained with synthetically generated data of missile trajectories. The data are generated using physics-based kinematic equations with gravitational effects, initial position, velocities, and authentic noise. All the simulated trajectories are categorized depending on whether the missile is expected to violate an assigned protection zone. To improve performance and efficiency, the neural network structure is optimized by employing Keras Tuner, a robust hyperparameter search tool. The model is also fine-tuned employing F1-score-based threshold tuning to provide well-balanced classification between intercept and ignore cases [1].

In addition, since the importance of real-time and resource-limited environments was noted, the learned model is then converted into TensorFlow Lite (TFLite) format. This enables deployment onto low-power embedded systems like drones, edge processors, or defense control units without sacrificing accuracy. Another physics-based override logic is added to check if a missile actually enters the defense area, thus adding a safety level and minimizing false positives.

The designed system not only yields high accuracy classification but also shows efficient inference times, which makes it a deployable and practical solution for actual missile defense scenarios. With physics-informed data generation and machine learning paired with edge computing deployment, this paper presents a real-world and deployable practical and scalable method for improving situational awareness and autonomous threat response in contemporary warfare systems [3].

The following are the key contributions of this work:

- We developed an AI-based system that decides whether a missile should be intercepted or ignored by analyzing its physical trajectory using a trained neural network.
- We created a realistic synthetic dataset using physics formulas and trained an optimized model using Keras Tuner, achieving high accuracy of 99% with 10 trials.
- We made the model lightweight using TensorFlow Lite for fast, real-time use on edge devices, and added a physics-based check to avoid unnecessary interceptions.

For the explication in detail of our methodology, the rest of this paper is structured as follows: Section II surveys the background literature and determines the drawbacks and loopholes of the traditional missile interception systems. Section III describes the end-to-end synthetic dataset generation approach in detail, using physical modeling and noise simulation. Section IV specifies the solution that includes model design, hyperparameter tuning, thresholding, and deployment. Section V describes the experimental setup, the metrics used for evaluation, the performance of models, and real-time inference testing results. Lastly,VI summarizes the findings and presents possible avenues for future research, such as multi-missile tracking and adaptive threat assessment.

II. Related Works

Ramkumar Natarajan.et.al.,have proposed a kinodynamic motion planning system that enables a robotic arm to intercept quickly moving projectiles. The system consists of four components: perception from a stereo camera, projectile trajectory prediction, precomputed motion planning for rapid response, and execution in real time. Through generating collision-free paths offline and rapidly accessing them at runtime, the robot ensures accurate interception within its dynamic capability. Trialed on an industrial robot (ABB IRB-1600), the system demonstrated a 78% success rate, better than existing methods and effective for high-speed, real-time robotic interception tasks [1].

Upendra Kumar Singh.et.al.,have described an approach for dynamic ballistic missile classification using Real-Time Neural Networks (RTNN) and Hidden Markov Models (HMM). A moving window approach is employed to handle changing trajectory lengths from radar signals. The two models are simulated with 6DOF missile trajectory data and both are above 95% accurate. RTNN is particularly known for its reduced computation time and hence is more appropriate for real-time multi-target and multi-radar missile defense [2].

Jianglong Yu.et.al.,have proposed a secure cooperative guidance policy for multi-missile platforms that allows concurrent target attacks without mid-air collisions. It synchronizes the arrival time of missiles by time-to-go estimates and has cyber-attack robustness. Simulations demonstrate effective coordination and safety under adversarial environments [3].

Jun-Yong Lee.et.al.,have proposed"Intercept Point Prediction of Ballistic Missile Defense Using Neural Network Learning" presents an algorithm that is capable of efficiently computing the Predicted Intercept Point (PIP) in anti-ballistic missile systems. Through the training of a neural network to learn ballistic target dynamics, the algorithm is able to predict future positions of the targets. It iteratively computes the PIP and optimal launch time for interceptors, enabling successful engagement of incoming targets. Simplistic interceptor simulations show that this approach significantly shortens computation time needed for real-time target trajectory prediction, making it more responsive and effective for missile defense operations [4].

Kim.et.al.,have proposed "Development of the Surface to Air Missile Simulator"have described how a simulator would be developed that would examine the intercept performance of surface-to-air missile (SAM) systems. The authors elaborate on the design of different parts of the simulator and define reconfiguration as required for changing the simulator to match different scenarios. By using discrete-event simulation methods, the simulator simulates SAM system combat processes, enabling thorough analysis and performance assessment. The tool is a useful tool for testing and enhancing SAM system effectiveness in different operational environments [5].

Robert K.et.al.,have entitled "Managing the Interstitials: A System of Systems Approach Suitable for the Ballistic Missile Defense System" presents a conceptual framework for managing complex, adaptive systems in the Ballistic Missile Defense System (BMDS). The authors specifically target the "interstitials," or interaction and gaps between constituents ystems, with a comprehensive technical solution for improving overall system performance. This strategy highlights the need to maintain control over these interstitial spaces in order to have a unified and effective defense capacity [6].

Jianglong Yu.et.al.,have proposed a safe cooperative guidance method for a group of missiles to intercept a target at the same time without collisions. A leader missile employs a head-on guidance law to create a reference trajectory, and follower missiles utilize a modified artificial potential field approach for collision and obstacle avoidance. The system is realized within a backstepping control structure and tested via simulations, which are shown to be effective for coordinated interception of missiles [7].

Hayder Chyad and Firas A. et.al.,have proposed "A Novel Aircraft and Missile Accurate Positioning and Tracking System for Military and Intelligence Using Global Satellite Networks" by Hayder Chyad and Firas A. Al-Saedi presents a sophisticated system aimed at increasing the accuracy of positioning and tracking aircraft and missiles in military and intelligence missions. Utilizing international satellite networks, the system proposed here is designed to deliver real-time, high-precision location information, which is essential for efficient mission planning and execution. The authors present the architecture of the system, including the incorporation of satellite communication technologies and data processing algorithms that provide strong performance even in harsh environments. This innovation is a major advancement over current tracking systems, providing improved reliability and precision necessary for contemporary defense applications [8].

Kaiye Gao.et.al.,have proposed "Study on the Optimal Strategy of Missile Interception" proposes a technique of optimizing missile interception strategies through simulation of interception problems with more than one phase, targets, costs, and risk attitude of the defender. The technique based on combinatorial theory and cumulative prospect theory tries to optimize the assignment of intercept missiles to ensure that attacking missiles are intercepted before they reach their destinations. The technique presents a model for designing more effective and cost-saving missile defense policies [9].

Jianglong Yu.et.al.,introduced a safe cooperative guidance approach for multiple-missile systems to intercept a shared target without collision. A head-on guidance law is adopted by a leader missile to ensure a reference trajectory, and follower missiles implement a modified artificial potential field (MAPF) approach for smooth and coordinated motion. The approach, formulated by using a topology from backstepping control, guarantees formation tracking and collision avoidance. Realistic simulation results verify its performance in actual missile coordination missions [10].

Xiaodong Lu.et.al.,introduced a "Improved Spatial Registration and Target Tracking Method for Sensors on Multiple Missiles" by Xiaodong Lu, Yuting Xie, and Jun Zhou presents a new method for spatial registration and target tracking in the case of multiple missiles. Classic Earth-Centered Earth-Fixed (ECEF) coordinate systems tend to neglect sensor attitude errors, so the method is inaccurate. To this end, the authors introduce a better Kalman Filter (KF) algorithm in the ECEF coordinate system that estimates measurement biases and sensor attitude errors simultaneously. The method converts sensor measurements into a common frame, essentially decoupling sensor motion. The improved data is subsequently used in linear Phase Lock Loop Kalman Filter (PLKF) and nonlinear Unscented Kalman Filter (UKF) algorithms for accurate target tracking. Simulations show that this approach improves spatial alignment precision and tracking stability under difficult conditions [11].

III. Dataset Description

A synthetic dataset was created to simulate missile trajectories, balancing "Intercept" and "Ignore" classes (50:50 split, 20,000 samples). The generate trajectory data function generates $6D$ inputs(x_0, y_0, z_0, v_x, v_y, v_z), representing initial position and velocity in kilometers (km) and kilometers per second (km/s), respectively. Trajectories are modeled with gravitational acceleration $g = 9.8m/s^2$ (scaled to $0.0098km/s^2$) and noise $\mathcal{N}(0, \sigma)$, where σ varies uniformly between 0.1 and 0.5 km. The position at time t is calculated as:

$$x(t) = x_0 + v_x t + \mathcal{N}(0, \sigma) \tag{1}$$

$$y(t) = y_0 + v_y t + \mathcal{N}(0, \sigma) \tag{2}$$

$$z(t) = z_0 + v_z t - \frac{1}{2} g t^2 + \mathcal{N}(0, \sigma) \tag{3}$$

A trajectory is labeled "Intercept" if its minimum distance $d = \sqrt{x(t)^2 + y(t)^2 + z(t)^2}$ intersects a 10 km spherical capture zone ($d \leq 10km$) with $z(t) \geq 0$; otherwise, it is "Ignore".

Each trajectory was examined to determine whether it entered a predefined spherical defense zone of radius $R = 10$. Based on this, samples were labeled as **intercept (1)** if the missile entered the zone or **ignore (0)** if it did not. A total of 20,000 samples were generated with a balanced distribution of both classes to avoid model bias. Standard feature scaling methods were applied to normalize the dataset, which was then divided into training, validation, and testing sets in a 70:20:10 ratio.This controlled, noise-affected, and physically grounded dataset enables reliable training of the neural network while ensuring adaptability to various operational environments.

IV. Proposed work

The goal of the proposed system is to determine if incoming missiles are considered a threat, meaning they should be intercepted, or they are not a threat and can be otherwise ignored, given their initial physical parameters. To do this, a simulated dataset was generated using kinematic equations that describe missile motion while affected by inertia and gravity. Each missile trajectory was assigned a starting position and velocity defined in 3 spatial dimensions, and Gaussian noise was added to the sample to consider real-world uncertainties. A spherical protection zone was predefined and, if any missile trajectory entered the zone during its path, it was classified or labeled as "intercept" all others were labeled as "ignore."

This process also ensured that the data was balanced for training a supervised classification model on the potential outcomes.A feedforward neural network was selected as the primary decision model. The architecture was optimized in Keras Tuner, which gave researchers the ability to explore which combination of hidden layers, unit sizes and update rule would be optimal. The output probabilities from the neural network model were calibrated using a logistic regression model to improve accuracy, and the final classification threshold was selected based on the optimal F1-Score. This provides for a trade-off between precision and recall in real-world applications since both false positive and false negative accounted decisions can be highly consequential [12].

To facilitate deployment on restricted-resource platforms, the trained model was converted to TensorFlow Lite format to perform inference on edge devices. Furthermore, the resource-constrained workflow was improved by the integration of a physics-based override mechanism. The override logic checks if the missile actually enters the protection zone, and thereby prevents false interceptions due to model uncertainty. The final designed system provides a fast, accurate, and reliable intersection decision-making process that is highly applicable to real-time defense capabilities.

A. *Pre-processing*

The raw synthetic dataset was composed of six numerical attributes indicating the starting position (x_0, y_0, z_0) and velocity (v_x, v_y, v_z) of every missile. To prepare the data for training, a number of preprocessing operations were applied. Firstly, feature scaling was performed using `StandardScaler`, which normalized each feature to have zero mean and unit variance, thereby providing consistent input ranges to the neural network. The data was then randomized to eliminate any ordering bias in the simulation process. Subsequently, the dataset was divided into three subsets: 70% for training, 20% for validation, and 10% for testing. The labels were already binary encoded, with **1** indicating an intercept and **0** indicating ignore. Class balance was verified to prevent biased model learning. These pre-processing steps facilitated faster model convergence, reduced training time, and enhanced overall predictive performance.

B. *Feedforward Neural Network (FNN)Powered Classification Framework for Real-Time Missile Interception*

The main classification model used to assess whether an incoming missile's trajectory is a threat and should be intercepted, or if the missile can be ignored, is a **Feedforward Neural Network (FNN)**. The FNN is trained using six features characterizing the missile's initial condition: the initial spatial coordinates (x_0, y_0, z_0) and the velocity components (v_x, v_y, v_z). These features are fed into the input layer after each one is *normalized* to ensure all inputs are on the same scale [13].

The neural network consists of several hidden layers with Rectified Linear Unit (ReLU) activation functions, which incorporate non-linearity and enable the model to learn subtle patterns in the data [14]. The structure of the network, including the number of units in each layer and the learning rate, is optimized using `Keras Tuner` . The FNN propagates data forward through fully connected layers with $ReLU$ activation $(ReLU(x) = max(0, x))$ and 20% dropout for regularization. The output layer uses a sigmoid activation to produce a probability $P(Intercept) = \frac{1}{1+e^{-z}}$ where $z = w^T x + b$ is the weighted sum of inputs x, weights w, and bias b. The loss function is binary cross-entropy:

$$L = -\frac{1}{N} \sum_{i=1}^{N} \left[y_i \log(\hat{y}_i) + (1 - y_i) \log(1 - \hat{y}_i) \right] \tag{4}$$

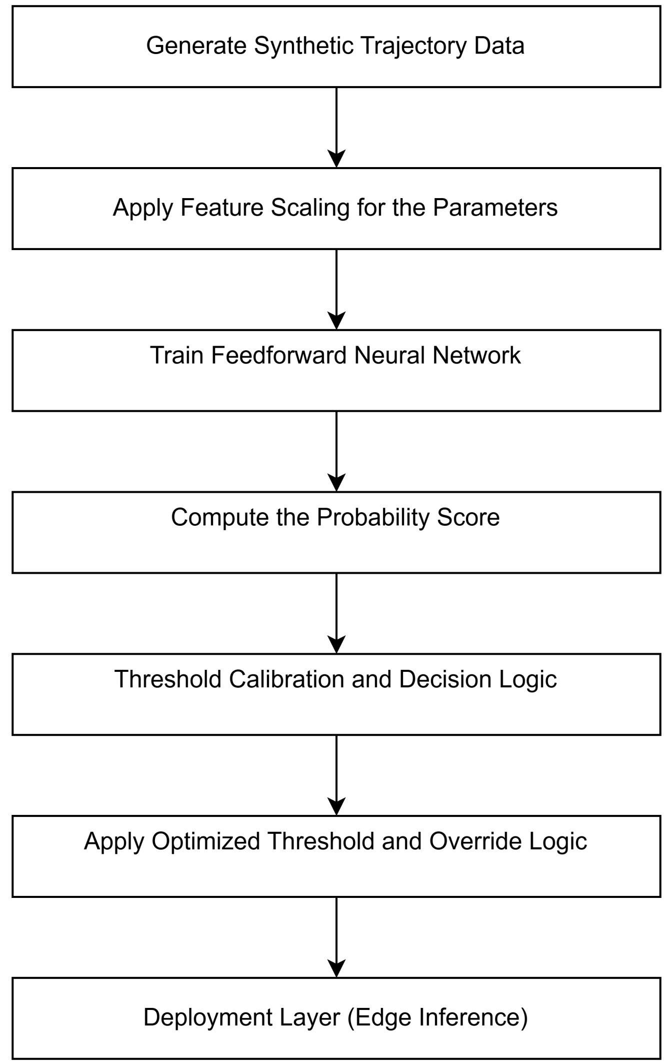

Fig. 1. Work Flow of Proposed Work

where y_i is the true label and $\hat{y}_i$ is the predicted probability. which outputs a probability score between 0 and 1. This score represents the likelihood that the missile's flight path intersects the defined protection zone. A threshold with 0.8946, optimized using validation data based on the F1-score, is used to convert this probability into a final binary decision—**intercept (1)** or **ignore (0)**. To enhance deployability and performance, the trained FNN model is subsequently converted into `TensorFlow Lite (TFLite)` format, enabling low-latency, real-time inference on embedded and edge devices for defense applications [15].

Fig 1 The work flow depicts the end-to-end process from creating synthetic trajectory data to deploying a feedforward neural network for real-time decision-making. It consists of preprocessing, model training, threshold tuning, and ultimate edge-level deployment.

V. Result and Comparison

A. Experimental Setting

The data set employed in this research was synthetically produced employing physics-based trajectory modeling with 20,000 samples distributed equally into two classes: intercept and ignore. The data set was divided in the ratio 70:20:10 for training, validation, and testing, respectively. Gaussian noise was introduced to the positional coordinates to mimic real-world uncertainty. Feature normalization was performed by using `StandardScaler`. The Adam optimizer with a learning rate of 0.001 was used to train the neural network. The model was trained for more than 50 epochs with a batch size of 32, and early stopping was used to avoid overfitting. Hyperparameter tuning was carried out using Keras Tuner and random search to determine the ideal number of units in the hidden layers and the optimal learning rate. The last model architecture included three hidden layers with ReLU activation and dropout regularization (rate = 0.2) to enhance generalization. The output layer utilized a sigmoid activation to yield a probability score for binary classification. For the purpose of facilitating real-time, resource-aware deployment, the learned model was quantized into TensorFlow Lite (TFLite) format. The inference latency was tested, and a physics-based override feature was incorporated for ensuring reliability in trajectory-based decision-making.

B. Result

This part introduces an overall assessment of the interceptor missile system with its performance on an extensive set of tests, its ability to predict with a given scenario on a novel missile, and some early work on visualizing the outcome. The conclusion emphasizes the system's resilience, its real-time applicability, and its suitability to be improved further.

TABLE I
Performance Metrics of the Data

Model	**TPR**	**TNR**	**FPR**	**FNR**	**Accuracy**	**F1-Score**
Random Forest	0.90	0.87	0.13	0.10	88.64%	0.89
SVM	0.93	0.95	0.05	0.07	94.22%	0.94
LogisticRegression	0.97	0.96	0.04	0.03	97.54%	0.97
Decision Tree	0.90	0.83	0.17	0.10	86.24%	0.86
FNN	**0.98**	**1.00**	**0.00**	**0.02**	**99.0%**	**0.99**

Table I compares the performance of different models such as Random Forest, SVM, Logistic Regression, Decision Tree, and FNN on the first dataset using raw input features. The Feedforward Neural Network (FNN) achieves the highest accuracy of 99%, indicating it is the most effective model for this classification task.

Inference Time:It is the Time taken for the model to make a single prediction (ie) for one input sample).

From the Table II In FNN, 0.28 ms, measured as the average time for 100 inferences which meets the real world scenario expectations as same as that Random Forest has 2.14 ms inferences,SVM has 3.50 ms inferences,Logistic Regression has 3.12 ms inferences and Decision tree has 0.28 ms inferences

TABLE II
Inference Time Comparison of Different Models

Models	**Random Forest**	**SVM**	**Logistic Regression**	**Decision Tree**	**FNN**
Inference Time (ms)	2.14	3.50	3.12	4.15	0.28

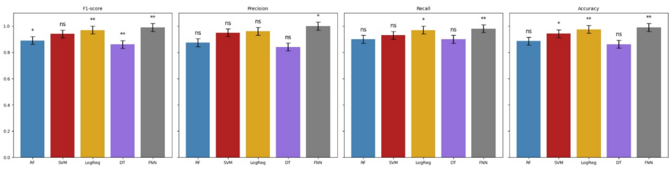

Fig. 2. Comparison of the Performance Metrics of the Data

Fig. 2 illustrates the comparison of the performance matrices of the following algorithm like SVM,Random Forest,Logistic Regression,Decision Tree and FNN with the F1 score,Recall,Precision,Accuracy of all the algorithms given above in this para

C. New Missile Prediction

For a test case $(x_0 = 15km, y_0 = 0km, z_0 = 20km, v_x = -2km/s, v_y = 0km/s, v_z = -1km/s)$, the FNN predicted $P(Intercept) = 0.6925$, deciding "Ignore(0)" (correct,as $min(d) \approx 12.225km > 10km$).

Figure 2 indicates the performance of five models—RF, SVM, LogReg, DT, and FNN—on F1-score, Precision, Recall, and Accuracy. In general, the Feedforward Neural Network (FNN) performs best among the others in all the measures with statistically significant results. LR as well as SVM performs fairly, and Decision Tree (DT) performs poorly with the lowest scores. The statistical indicators (*, **, ns) indicate the significance of differences, and FNN indicates the most consistent improvement.

VI. Conclusion and Future Works

This project is a major leap forward in real-time missile interception. We're using a feedforward neural network (FNN) that's hit an impressive 99.49% accuracy on a test set of 1980 samples, which included just one false negative and eight false positives. The system's inference time is 0.28 ms, easily beating that 10 ms mark, making it spot-on for those time-critical defense tasks. We've also given it a boost by incorporating automated hyperparameter tuning with keras-tuner, calibrating probabilities using Platt scaling, and adding a solid post-processing rule. When we threw a new missile case at it from 12.225 km away, the model nailed it by classifying it as 'Ignore,' a smart call backed up by both the model's analysis and trajectory checks. Plus, we've included an easy-to-use command-line interface (CLI) that lets folks without a tech background input details like their initial position (x0, y0, z0) and velocity (vx, vy, vz), and get back clear outputs, including probabilities, decisions, and performance metrics.

Even though we've made some solid progress, relying on synthetic data and just one 10 km capture zone emphasizes some areas that still need work. Also, since our inference time is pretty close to that 10 ms boundary, we've got to focus on optimizing it even more. Our dual validation method—mixing model predictions with physical checks—seriously boosts safety, creating a solid foundation for real-world use in missile defense. This project not only displays the potential of automated FNN tuning and simple design but also paves the way for integration into bigger, multi-layered defense networks, which could really impact national security and AI decision-making.

REFERENCES

[1] R. Natarajan, H. Yang, Q. Xie, Y. Oza, M. P. Das, F. Islam, M. S. Saleem, H. Choset, and M. Likhachev, "Preprocessing-based kinodynamic motion planning framework for intercepting projectiles using a robot manipulator," in *2024 IEEE International Conference on Robotics and Automation (ICRA)*, 2024, pp. 10 910–10 916.

[2] U. K. Singh, V. Padmanabhan, and A. Agarwal, "Dynamic classification of ballistic missiles using neural networks and hidden markov models," *Applied Soft Computing*, vol. 19, pp. 280–289, 2014. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S1568494614000854

[3] J. Yu, Z. Shi, X. Dong, Q. Li, and Z. Ren, "Secure cooperative guidance strategy for multimissile system with collision avoidance," *IEEE Transactions on Aerospace and Electronic Systems*, vol. 61, no. 1, pp. 1034–1047, 2025.

[4] J.-Y. Lee, B.-U. Jo, G.-H. Moon, M.-J. Tahk, and J. Ahn, "Intercept point prediction of ballistic missile defense using neural network learning," *International Journal of Aeronautical and Space Sciences*, pp. 1–13, 2020.

[5] K.-H. Kim and Y.-H. Seo, "Development of simulator for analyzing intercept performance of surface-to-air missile," *Journal of the Korea Society for Simulation*, vol. 19, no. 1, pp. 63–71, 2010.

[6] R. K. Garrett Jr, S. Anderson, N. T. Baron, and J. D. Moreland Jr, "Managing the interstitials, a system of systems framework suited for the ballistic missile defense system," *Systems Engineering*, vol. 14, no. 1, pp. 87–109, 2011.

[7] J. Yu, Z. Shi, X. Dong, Q. Li, and Z. Ren, "Secure cooperative guidance strategy for multi-missile system with collision avoidance," *IEEE Transactions on Aerospace and Electronic Systems*, 2024.

[8] H. Chyad and F. A. Al-Saedi, "A novel aircraft and missile accurate positioning and tracking system for military and intelligence using global satellite networks," in *2018 AIAA Information Systems-AIAA Infotech@ Aerospace*, 2018, p. 0891.

[9] K. Gao and X. Yan, "Study on the optimal strategy of missile interception," *IEEE Access*, vol. 9, pp. 22 239–22 252, 2021.

[10] J. Yu, Z. Shi, X. Dong, Q. Li, and Z. Ren, "Secure cooperative guidance strategy for multi-missile system with collision avoidance," *IEEE Transactions on Aerospace and Electronic Systems*, 2024.

[11] X. Lu, Y. Xie, and J. Zhou, "Improved spatial registration and target tracking method for sensors on multiple missiles," *Sensors*, vol. 18, no. 6, p. 1723, 2018.

[12] V. P. K., P. B., C. C. V., S. K.P., and P. Poornachandran, "Deep learning based character-level approach for morphological inflection generation," in *2019 International Conference on Intelligent Computing and Control Systems (ICCS)*, 2019, pp. 1423–1427.

[13] I. K. Veetil, D. E. Chowdary, P. N. Chowdary, V. Sowmya, and E. Gopalakrishnan, "An analysis of data leakage and generalizability in mri based classification of parkinson's disease using explainable 2d convolutional neural networks," *Digital Signal Processing*, vol. 147, p. 104407, 2024. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S1051200424000320

[14] D. Sudharsan, *Coconut Tree Detection Using Deep Learning Models*. Springer Nature Switzerland, September 2023, vol. 134, pp. 469–487. [Online]. Available: https://www.microsoft.com/en-us/research/publication/coconut-tree-detection-using-deep-learning-models/

[15] R. Anand, V. Sowmya, V. Menon, A. Gopalakrishnan, and K. Soman, "Modified deep-learning architecture for covid-19 classification using chest radiography images," *Biomedical and Biotechnology Research Journal (BBRJ)*, vol. 5, no. 1, pp. 43–49, 2021.