Iron Shield Real Time Defense System Using Neural Network

SUBJECT : 24MAT112 & 24AIM113

Team Members

Sanjay R - CB.AI.U4AIM24143
Shreevarsinii B - CB.AI.U4AIM24144
Srijan Sivaram A - CB.AI.U4AIM24145
Snendar M S - CB.AI.U4AIM24127

Introduction

- Missile threats demand split-second decisions in defense systems.
- Traditional methods use rule-based or static models with limited flexibility.
- The proposed system leverages Artificial Intelligence (AI) to classify missile threats in real-time.
- A neural network is trained on synthetic trajectory data to decide whether to intercept or ignore an incoming missile.
- The system is designed for fast, accurate, and deployable usage on edge devices.

Problem Statement

- *Can we intelligently and efficiently identify whether a missile will enter a protected zone using only its initial trajectory parameters?*
- *How do we ensure low-latency, high-accuracy decisions suitable for real-time defense deployment?*

LITERATURE REVIEW

S. No	Author(s)	Title	Key Contributions
1	Ramkumar Natarajan et al.	Kinodynamic Motion Planning for Robotic Arms	Uses stereo cameras and precomputed paths for fast robotic interception; achieved 78% success rate.
2	Nigerian Defence Academy	Mathematical Model of Surface-to-Air Missile	Simulates altitude-based atmospheric effects; determines optimal interception via numerical modeling.
3	Upendra Kumar Singh et al.	Real-Time Missile Classification using RTNN and HMM	Uses radar data with neural networks and HMM; 95% accuracy
4	Jianglong Yu et al.	Secure Cooperative Guidance for Multi-Missile Platforms	Enables multiple missiles to attack simultaneously without collision; resilient to cyber-attacks.

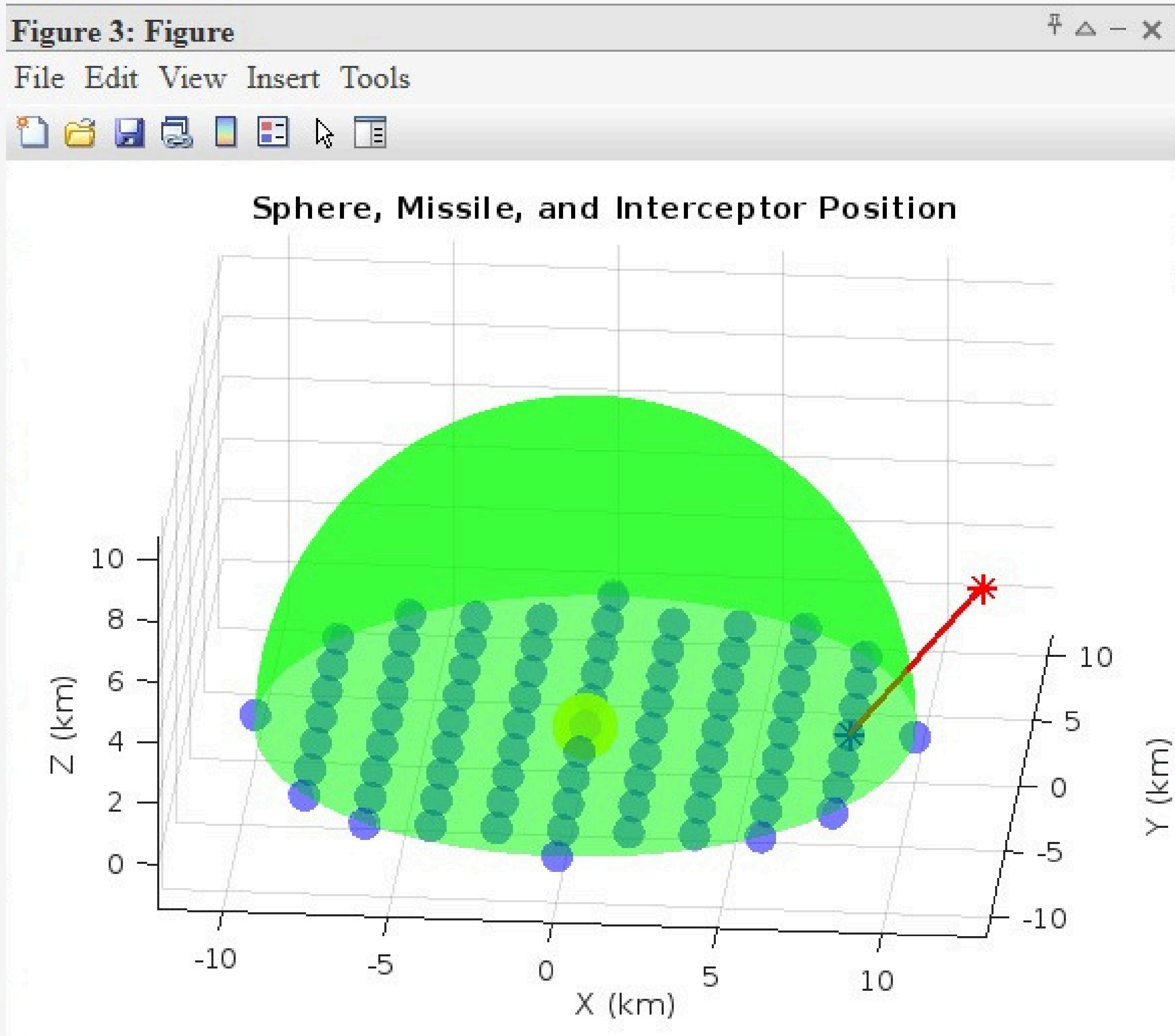
MATHEMATICAL EXPLANATION:

- These equations model 3D trajectories using:
  - Initial position x_0, y_0, z_0
  - Velocity components v_x, v_y, v_z
- The term $N(0,\sigma)$ represents Gaussian noise with a mean of 0 and standard deviation σ

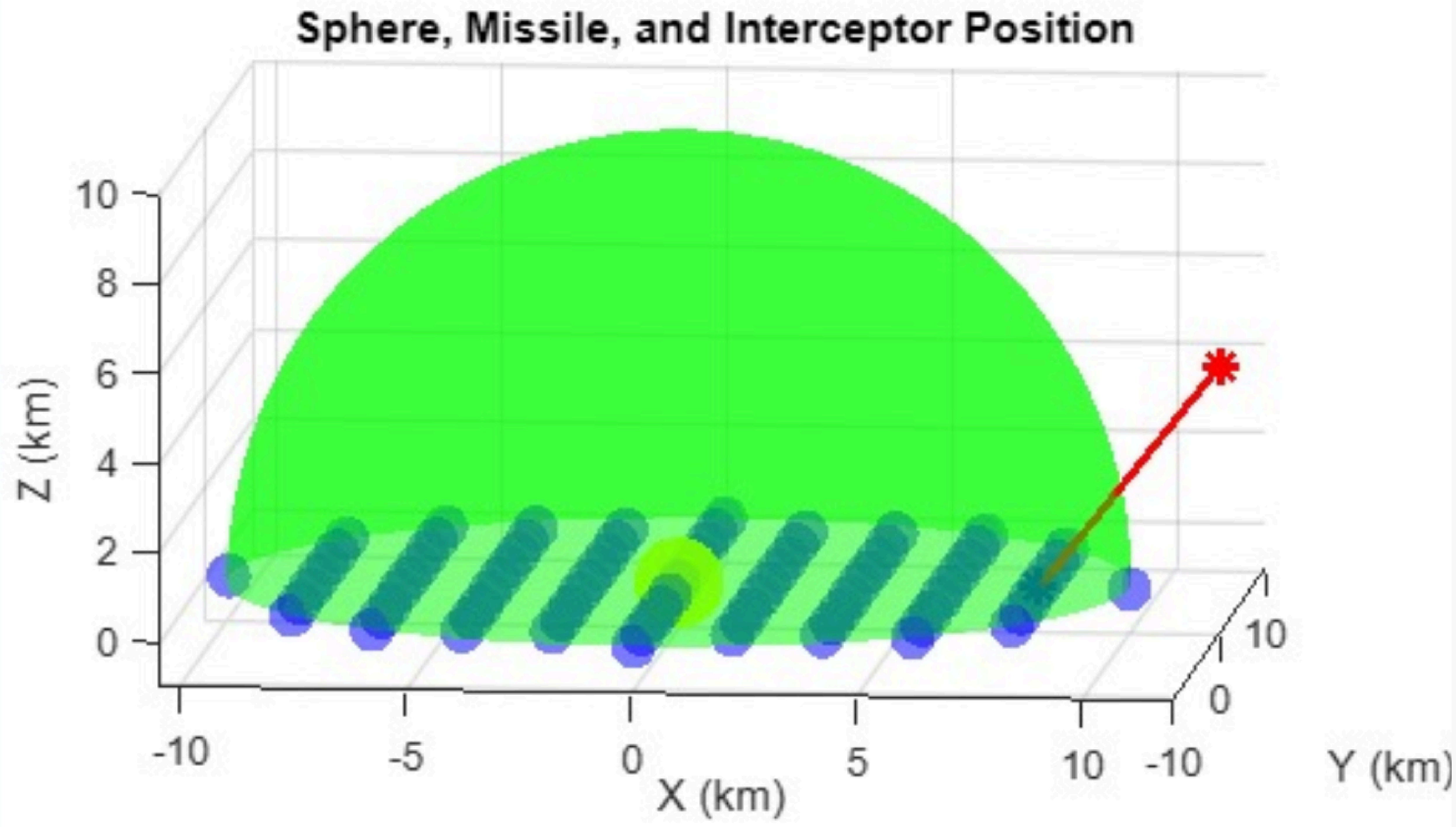
$$x(t) = x_0 + v_{xt}\, t + N(0,\sigma)$$

$$y(t) = y_0 + v_{yt}\, t + N(0,\sigma)$$

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



- *If the missile's minimum distance from the center is ≤ 10 km and it's still above ground (z ≥ 0) it is a threat then Intercept.*
- *If the missile never enters the 10 km zone it is safe then Ignore.*


Sphere, Missile, and Interceptor Position
Z (km)
10
8
6
4
2
0
-10
-5
0
5
10
X (km)
-10
0
10
Y (km)

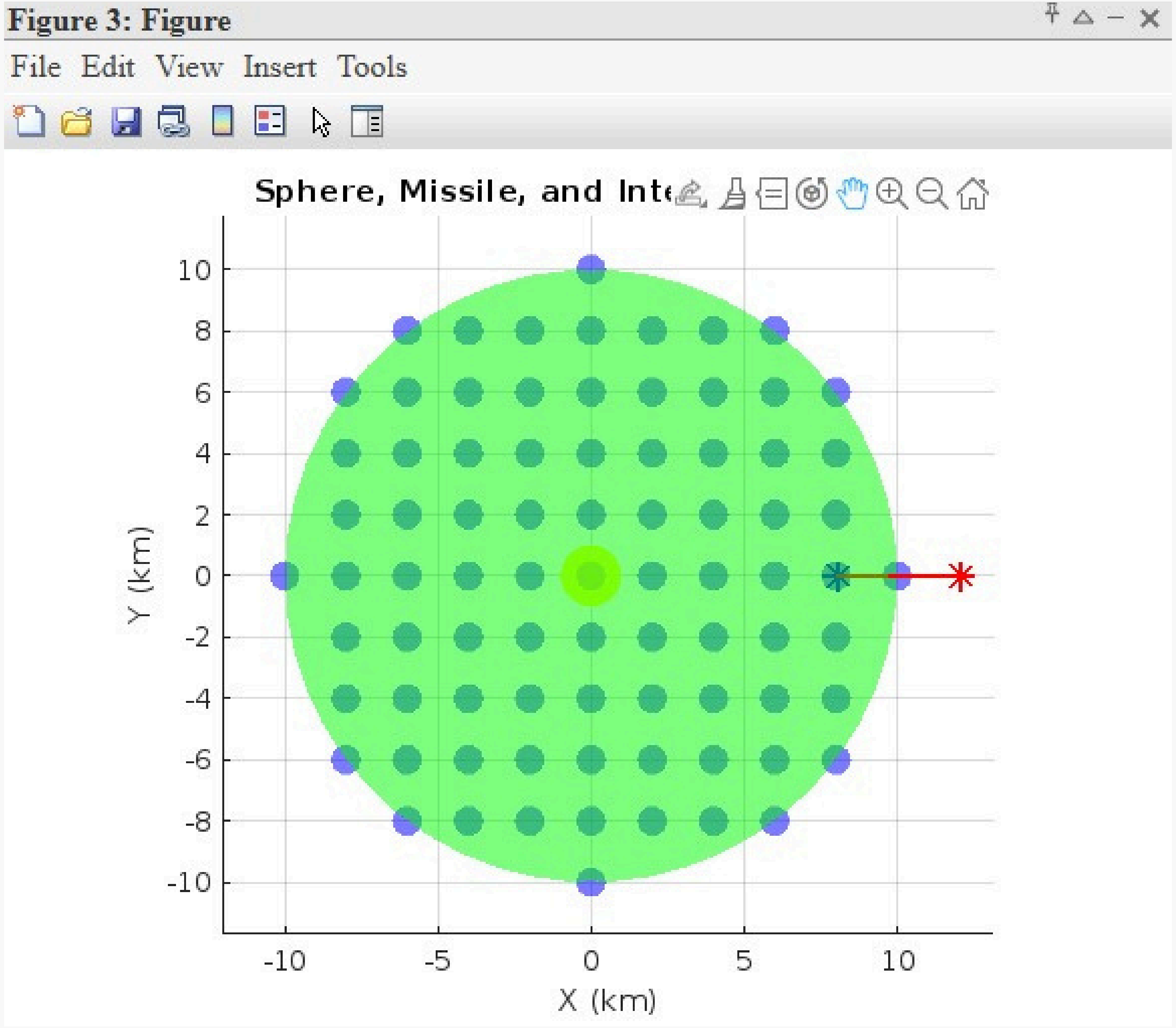


Figure 3: Figure
File Edit View Insert Tools
Sphere, Missile, and Inte
Y (km)
X (km)
10
8
6
4
2
0
-2
-4
-6
-8
-10
-10
-5
0
5
10

DATA STRUCTURE AND ALGORITHM

- ***Used arrays to store missile data.***
- ***Applied distance calculation to label data (Intercept or Ignore).***
- ***Added random noise using Gaussian distribution.***
- ***Trained a model using gradient descent algorithm.***
- ***Tuned the model using search algorithms (Keras Tuner).***
- ***Used thresholding logic to improve prediction accuracy.***

Workflow:

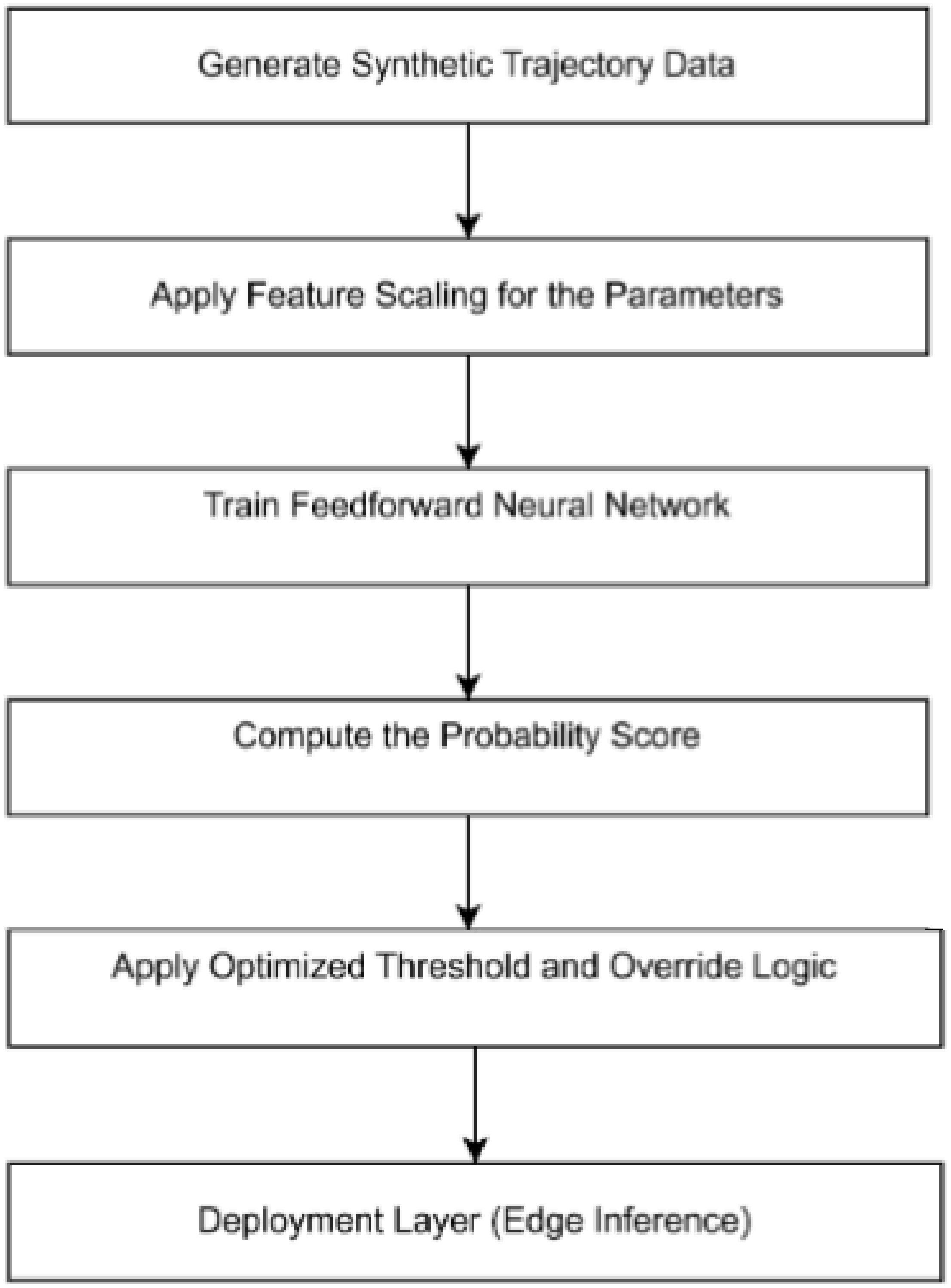


Fig. 1. Work Flow of Proposed Work

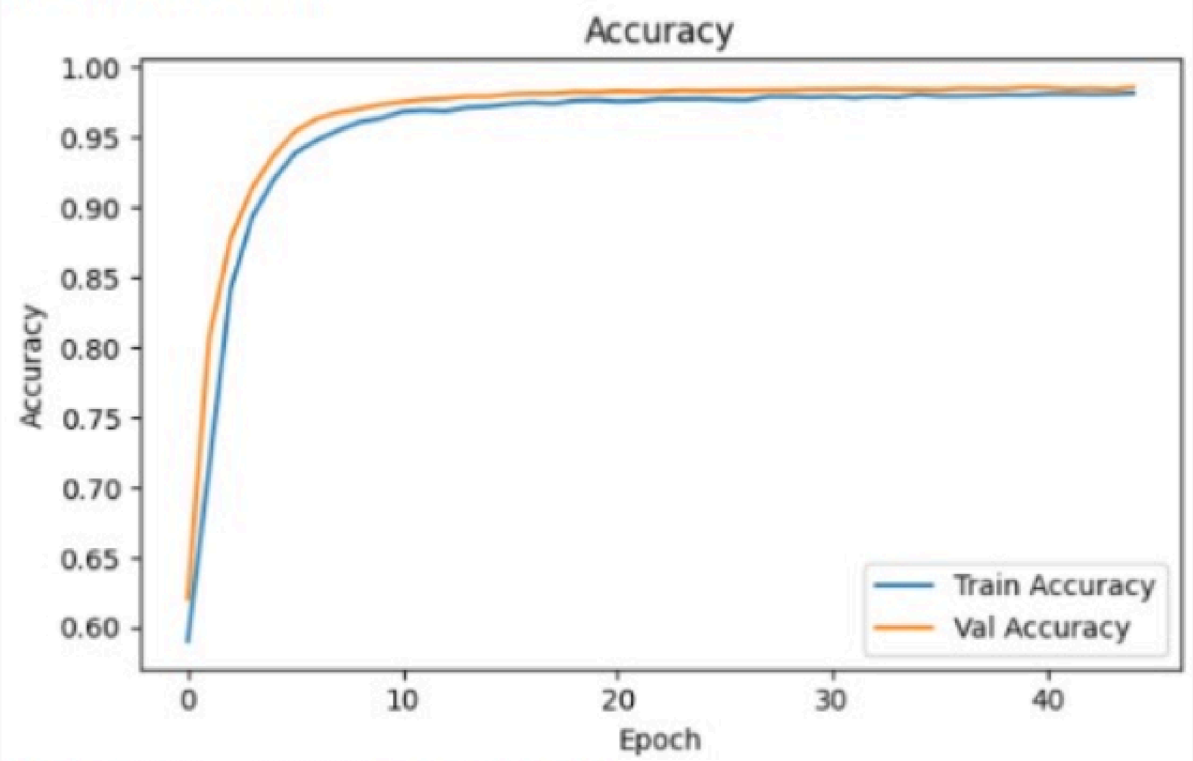
Result:

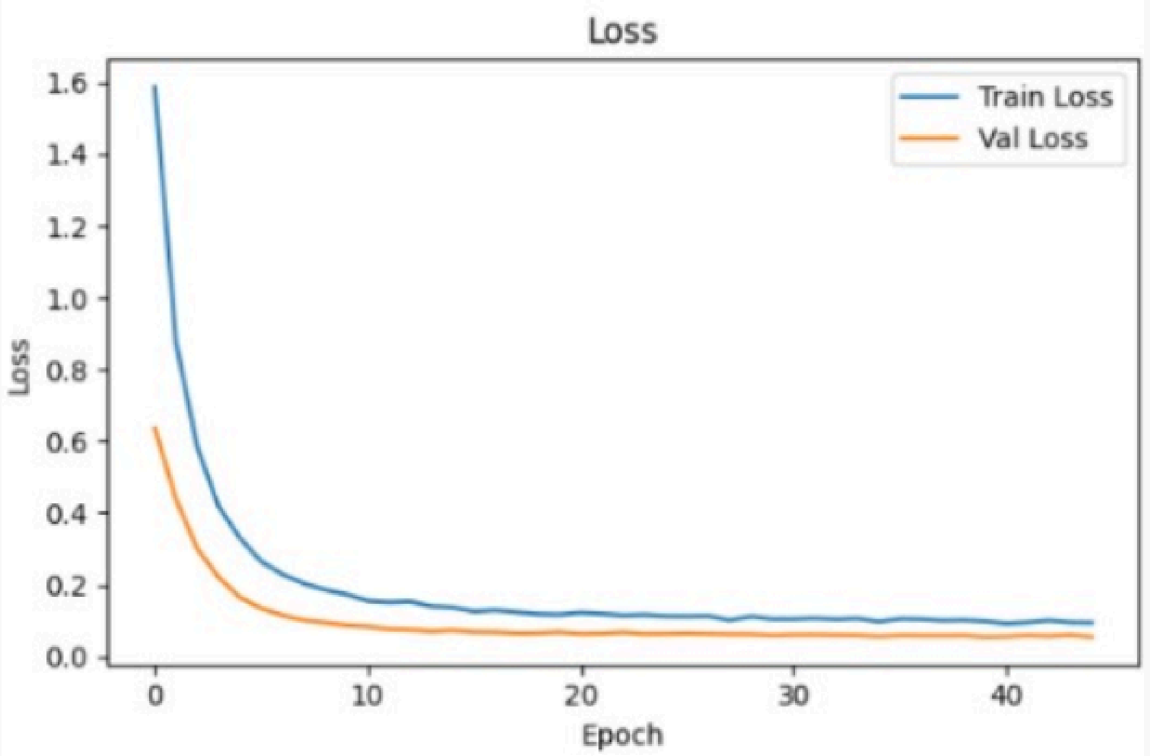
```
Class distribution: Intercept=10000, Ignore=10000
Reloading Tuner from tuner_dir/missile_interception/tuner0.json
/usr/local/lib/python3.11/dist-packages/keras/src/layers/core/dense.py:87: UserWarning: Do no
  super().__init__(activity_regularizer=activity_regularizer, **kwargs)
/usr/local/lib/python3.11/dist-packages/keras/src/saving/saving_lib.py:757: UserWarning: Skip
  saveable.load_own_variables(weights_store.get(inner_path))

Test Accuracy: 0.9949
62/62 ━━━━━━━━━━━━━━━━━━━━ 0s 2ms/step
Optimal Threshold (F1 Score): 0.8946
Confusion Matrix (Threshold 0.8946):
[[995   8]
 [  0 977]]
Saved artifact at '/tmp/tmpscz9i7sx'. The following endpoints are available:

* Endpoint 'serve'
  args_0 (POSITIONAL_ONLY): TensorSpec(shape=(None, 6), dtype=tf.float32, name='keras_tensor'
Output Type:
  TensorSpec(shape=(None, 1), dtype=tf.float32, name=None)
Captures:
  133115529050448: TensorSpec(shape=(), dtype=tf.resource, name=None)
  133115363932112: TensorSpec(shape=(), dtype=tf.resource, name=None)
  133115363933456: TensorSpec(shape=(), dtype=tf.resource, name=None)
  133115363930000: TensorSpec(shape=(), dtype=tf.resource, name=None)
  133115363926736: TensorSpec(shape=(), dtype=tf.resource, name=None)
  133115363930768: TensorSpec(shape=(), dtype=tf.resource, name=None)
  133115363926160: TensorSpec(shape=(), dtype=tf.resource, name=None)
  133115363930576: TensorSpec(shape=(), dtype=tf.resource, name=None)

New Missile Prediction: 0.9925 -> Decision: Ignore (Threshold: 0.8946)
Inference Time: 0.28 ms
```





Thank you