Pattern Recognition Project: Multi-Category Radar Object Classification

November 14, 2025

Abstract

This project addresses the challenge of multi-category object classification for automotive radar systems. Using the RadarScenes dataset, we extracted 15 kinematic and spatial features to classify objects into 12 semantic categories. We performed a comparative analysis between Support Vector Machines (SVM) and Random Forest classifiers. Experimental results show that Random Forest significantly outperforms SVM, achieving an accuracy of **94.52%** compared to **68.60%**. The Random Forest model proved more robust in handling non-linear feature interactions, leading to superior performance.

1 Dataset Selection

We utilized the **RadarScenes** dataset, a real-world automotive database. This selection satisfies all course constraints:

- **Problem Type:** Multi-Category Classification (12 classes).
- **Scale:** The dataset contains over 31,000 labeled instances (Requirement ≥ 1000).
- **Attributes:** We engineered 15 distinct features per instance (Requirement ≥ 10).

2 Methodology

2.1 Feature Extraction

To enhance the discriminative power of the raw radar points, we extracted 15 features. This focus on feature extraction aligns with the project requirement.

1. **Spatial Features:** Lateral position (y), Range squared (r^2), Azimuth angle (θ).
2. **Kinematic Features:** Compensated radial velocity ($v_{r,\text{comp}}$), Velocity difference ($|v_r - v_{r,\text{comp}}|$).
3. **Physical Features:** Normalized RCS, Logarithmic RCS.

2.2 Classification Models

We implemented and compared two classical pattern recognition models:

- **Support Vector Machine (SVM):** Configured with an RBF kernel and class weighting.
- **Random Forest:** Configured with 200 estimators.

3 Results and Discussion

3.1 Performance Comparison

The Random Forest classifier demonstrated superior performance. The SVM struggled with complex dynamic classes, as seen in its low recall for classes like 'bicycle' (0.49) and 'motorcycle' (0.40).

Table 1: Comparative Analysis of Classifiers

Metric	SVM	Random Forest
Accuracy	68.60%	**94.52%**
Precision	72.16%	**94.52%**
Recall	68.60%	**94.52%**
F1-Score	69.29%	**94.51%**

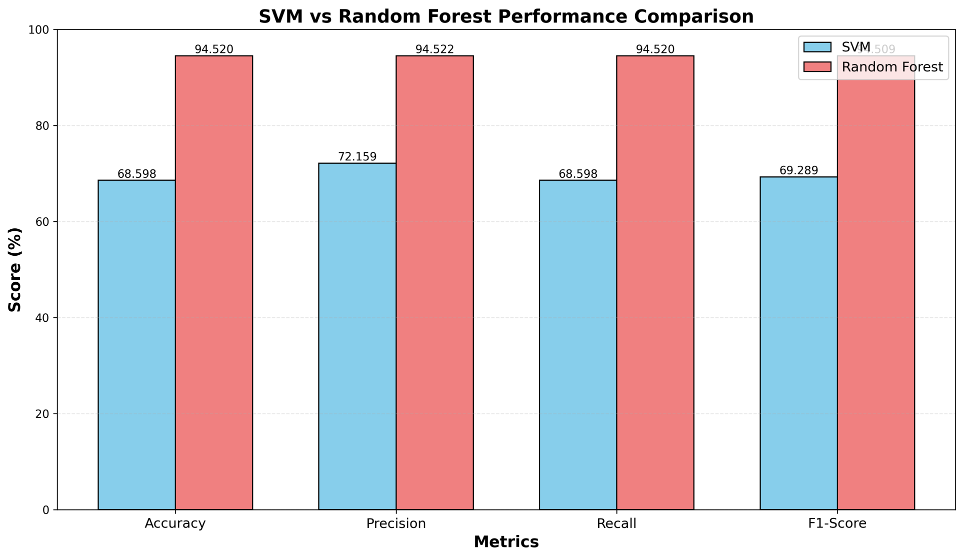
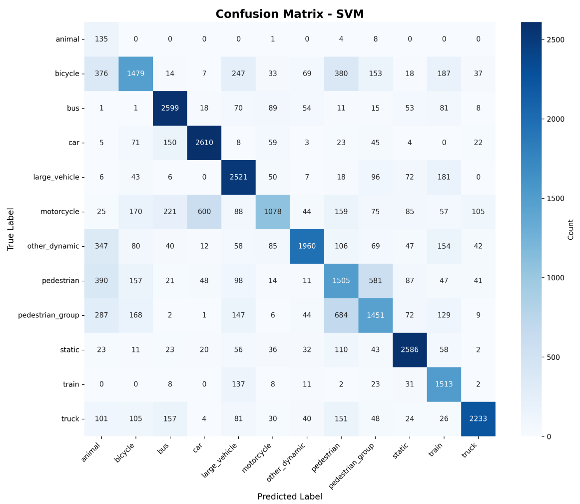


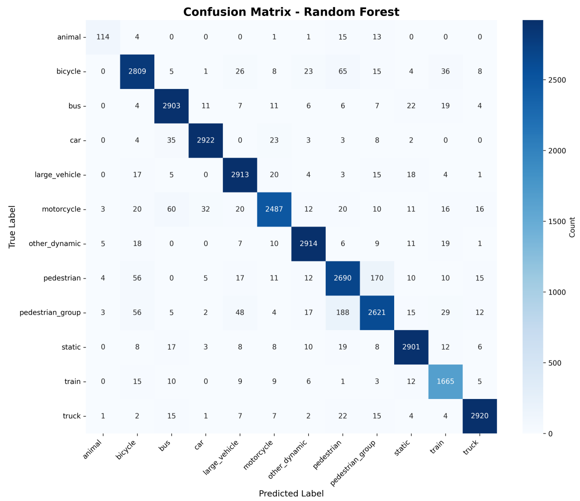
Figure 1: Bar chart comparison of SVM vs Random Forest performance.

3.2 Confusion Matrix Analysis

The confusion matrices (Figure 2) show where each model failed. The Random Forest matrix has a much stronger diagonal. The SVM matrix shows significant confusion.



(a) Confusion Matrix for SVM Classifier



(b) Confusion Matrix for Random Forest Classifier

Figure 2: Comparison of Confusion Matrices

3.3 Feature Importance

Analysis of the Random Forest model revealed that **Lateral Position** (y) (13.93%) and **Compensated Velocity** (13.61%) were the most critical features.

Top 10 Most Important Features:

Feature	Importance (%)
Lateral Position (y)	13.93
Compensated Velocity ($v_{r,\text{comp}}$)	13.61
Velocity Difference ($\lvert v_r - v_{r,\text{comp}} \rvert$)	11.53
Azimuth Angle (θ)	9.14
Radial Velocity (v_r)	9.10
Azimuthal Velocity	7.30
Longitudinal Position (x)	6.16
Range Squared (r^2)	5.97
Range (r)	5.68
Spatial Velocity	5.03

Table 2: Top 10 Feature Importances from Random Forest Classifier

4 Code Repository

The complete Python implementation (`Main.py`) and dataset loaders are available at: `https://github.com/dogtooth-maker/Radar_Object_Classification`

A Appendix

A.1 Individual Contributions

Anirudh Amin (Roll No: S20230020278)

Role: Random Forest Implementation & Feature Engineering

- Implemented the Random Forest classifier and optimization.
- Designed the feature extraction logic for kinematic and spatial attributes.
- Performed the Feature Importance analysis.

Harshit Singh (Roll No: S20230020305)

Role: SVM Implementation & Data Preprocessing

- Implemented the Support Vector Machine (SVM) classifier pipeline.
- Handled data normalization (StandardScaler) and class balancing.
- Conducted failure analysis on SVM results (Confusion Matrix interpretation).