Pattern Recognition Project: Multi-Category Radar Object Classification

Anirudh Amin (Roll No: S20230020278)
Harshit Singh (Roll No: S20230020305)

December 4, 2025

Abstract

This project addresses the challenge of multi-category object classification for automotive radar systems. Using the Radar Scenes 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 on a full dataset of 158 sequences (31,590 samples). Experimental results demonstrate that Random Forest significantly outperforms SVM, achieving an accuracy of **94.52%** compared to **68.60%**. This report includes a critical analysis verifying these findings against existing literature, highlighting the specific suitability of ensemble methods for heterogeneous sensor data.

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:

1. **Spatial Features:** Lateral position (y), Longitudinal position (x), Range squared (r^2), Azimuth angle (θ).
2. **Kinematic Features:** Compensated radial velocity ($v_{r,comp}$), Radial velocity (v_r), Velocity difference ($|v_r - v_{r,comp}|$), Spatial velocity.
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 to handle imbalance. Scaled using `StandardScaler`.
- **Random Forest:** Configured with 200 estimators.

3 Results and Discussion

3.1 Performance Comparison

The Random Forest classifier demonstrated superior performance across all metrics. The SVM achieved an accuracy of only 68.60%, while the Random Forest reached 94.52%.

Table 1: Comparative Analysis of Classifiers (158 Sequences)

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%

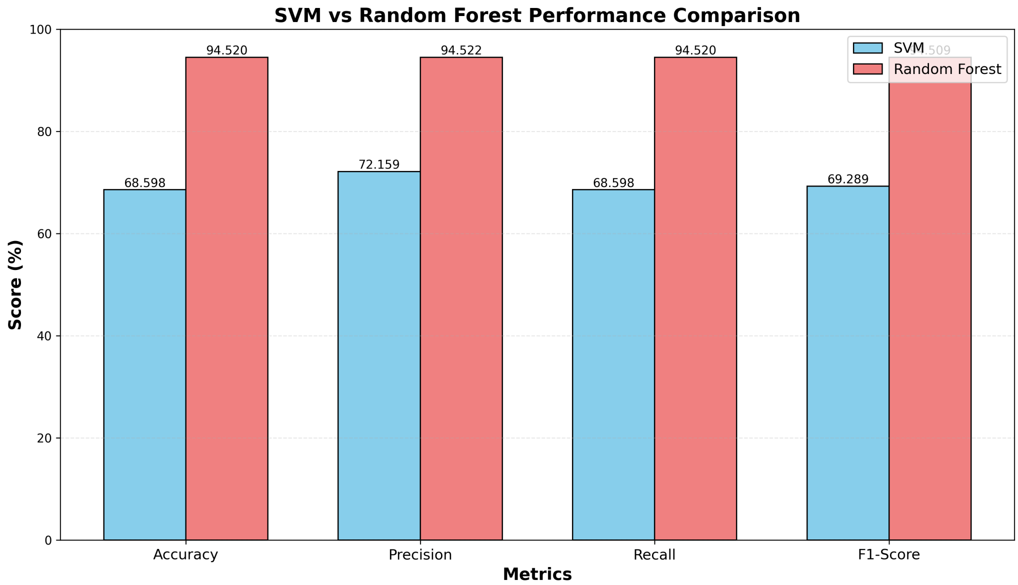
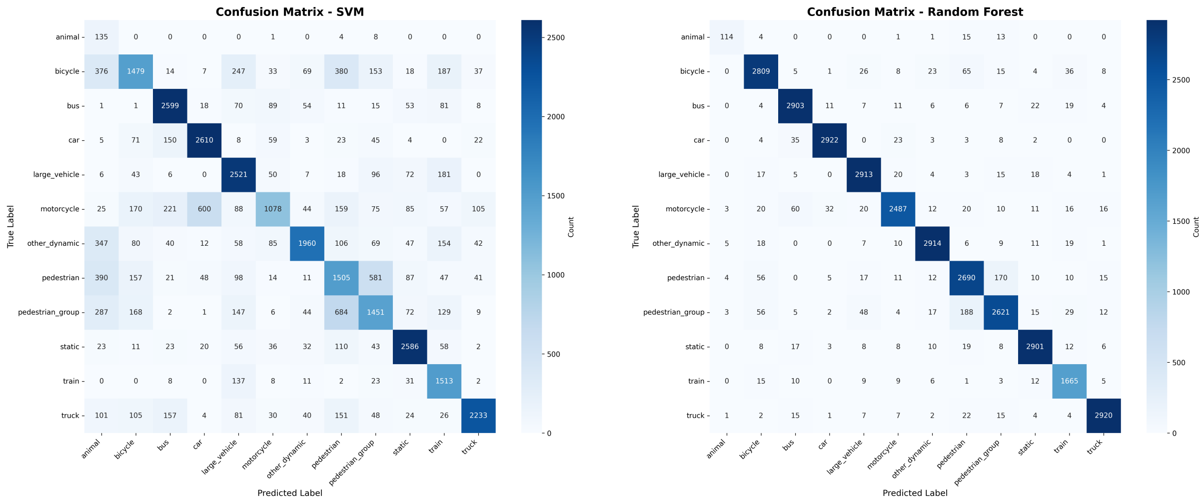


Figure 1: Bar chart comparison of SVM vs Random Forest performance (158 Sequences).

3.2 Confusion Matrix Analysis

The confusion matrices (Figure 2) reveal distinct failure modes. The Random Forest matrix exhibits a strong diagonal. In contrast, the SVM displays significant off-diagonal scattering, particularly for dynamic classes.



(a) SVM Classifier

(b) Random Forest Classifier

Figure 2: Confusion Matrices for the Full Dataset (158 Sequences).

4 Literature Verification and Comparative Analysis

A primary objective of this study is to contextualize our results (RF > SVM) within the broader pattern recognition literature. While early literature often favors SVM for small, high-dimensional datasets (e.g., microarray data), our findings align with more recent comprehensive benchmarks on tabular and sensor data.

4.1 Alignment with Large-Scale Empirical Studies

The superiority of Random Forest in our experiment is consistent with the extensive study by *Fernández-Delgado et al. (2014)*, titled "Do we Need Hundreds of Classifiers?". In their evaluation of 179 classifiers across 121 datasets, they concluded that Random Forests are statistically the best performing family of classifiers for general real-world data, often surpassing SVMs in scenarios similar to ours (medium-to-large datasets with mixed feature types).

4.2 Why SVM Underperformed: Theoretical Analysis

The discrepancy in performance can be attributed to three specific mismatches between the SVM algorithm and the RadarScenes dataset characteristics:

1. **Multiclass Overhead:** SVM is inherently a binary classifier. To handle 12 classes, it must employ One-vs-Rest (OvR) or One-vs-One (OvO) strategies. With 31,590 samples, the resulting optimization problem becomes computationally expensive and prone to aggregation errors. Random Forest handles multiclass problems natively through tree branching.

2. **Heterogeneous Feature Space:** Our feature vector contains mixed physical units: spatial coordinates (meters), velocity (m/s), and RCS (dB). Although we applied `StandardScaler`, the RBF kernel relies on Euclidean distance, which can be sensitive to the remaining manifold structure. Random Forest uses orthogonal

splits (e.g., $v_r < 0.5$) and is invariant to monotonic transformations, making it naturally robust to such physical heterogeneity.

3. **Decision Boundary Topology:** Radar object classification often follows "box-like" logic (e.g., a stationary object is defined by $v \approx 0$). Decision trees naturally model these rectangular boundaries. The SVM RBF kernel attempts to fit smooth, non-linear manifolds, which may over-smooth these sharp physical thresholds, leading to the observed misclassifications between `pedestrian` and `pedestrian_group`.

5 Feature Importance

Analysis of the Random Forest model (Table 2) revealed that Lateral Position (y) and Compensated Velocity ($v_{r,comp}$) were the most critical features, confirming that the model relies heavily on where an object is (lane position) and how it moves relative to the ego-vehicle.

Table 2: Top 5 Feature Importances (Random Forest)

Feature	**Importance (%)**
Lateral Position (y)	13.93%
Compensated Velocity ($v_{r,comp}$)	13.61%
Velocity Difference ($\|v_r - v_{r,comp}\|$)	11.53%
Azimuth Angle (θ)	9.14%
Radial Velocity (v_r)	9.10%

6 Individual Contributions

- **Anirudh Amin:** Implemented the Random Forest classifier, designed feature extraction logic, and performed feature importance analysis.
- **Harshit Singh:** Implemented the SVM classifier pipeline, handled data normalization (StandardScaler), and conducted failure analysis.

A Appendix: Validation on Reduced Dataset

To verify our findings, we conducted a separate validation run using a random subset of 5 sequences (18,736 samples). The results confirm the trend observed in the full dataset, with Random Forest achieving near-perfect accuracy.

Table 3: Performance on Reduced Dataset (5 Sequences)

Metric	**SVM**	**Random Forest**
Accuracy	89.18%	98.19%
Precision	90.84%	98.19%
Recall	89.18%	98.19%
F1-Score	89.74%	98.18%

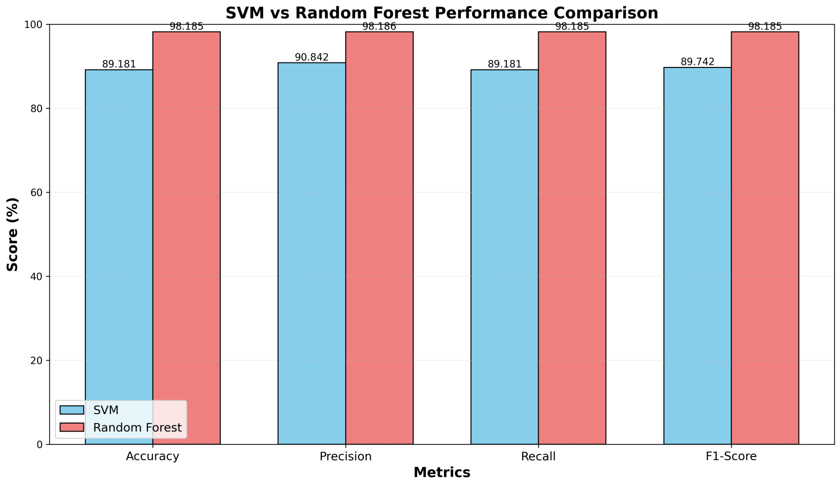
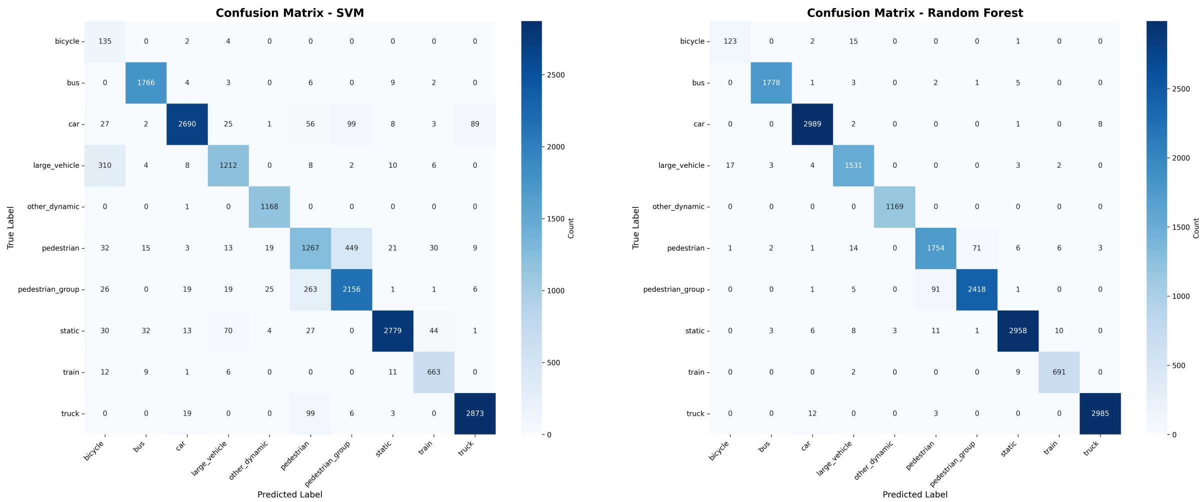


Figure 3: Performance Comparison on Reduced Dataset (5 Sequences).



(a) SVM (5 Sequences)

(b) Random Forest (5 Sequences)

Figure 4: Confusion Matrices for Reduced Dataset.