# **Route Adherence & Predictive Pallet Management**

## **Contents**

1 [Business Objective](https://chatgpt.com/c/67afc129-8a3c-8012-8543-a7b972444bdf#business-objective) 1.1 [Background](https://chatgpt.com/c/67afc129-8a3c-8012-8543-a7b972444bdf#background) 1.2 [Business Case](https://chatgpt.com/c/67afc129-8a3c-8012-8543-a7b972444bdf#business-case)

2 [Project Management](https://chatgpt.com/c/67afc129-8a3c-8012-8543-a7b972444bdf#project-management) 2.1 [Project Roles](https://chatgpt.com/c/67afc129-8a3c-8012-8543-a7b972444bdf#project-roles) 2.2 [Project Plan](https://chatgpt.com/c/67afc129-8a3c-8012-8543-a7b972444bdf#project-plan) 2.3 [Key Risks](https://chatgpt.com/c/67afc129-8a3c-8012-8543-a7b972444bdf#key-risks)

3 [Methodology](https://chatgpt.com/c/67afc129-8a3c-8012-8543-a7b972444bdf#methodology) 3.1 [Data Preparation and Availability](https://chatgpt.com/c/67afc129-8a3c-8012-8543-a7b972444bdf#data-preparation-and-availability) 3.2 [Proposed Approach](https://chatgpt.com/c/67afc129-8a3c-8012-8543-a7b972444bdf#proposed-approach) 3.2.1 [Manual (Non-ML) Threshold-Based Method](https://chatgpt.com/c/67afc129-8a3c-8012-8543-a7b972444bdf#manual-non-ml-threshold-based-method) 3.2.2 [Advanced Self-Improving Methods](https://chatgpt.com/c/67afc129-8a3c-8012-8543-a7b972444bdf#advanced-self-improving-methods)

4 [Analysis](https://chatgpt.com/c/67afc129-8a3c-8012-8543-a7b972444bdf#analysis) 4.1 [Distance Calculation & Thresholding](https://chatgpt.com/c/67afc129-8a3c-8012-8543-a7b972444bdf#distance-calculation--thresholding) 4.2 [Mapping Pallet to Route](https://chatgpt.com/c/67afc129-8a3c-8012-8543-a7b972444bdf#mapping-pallet-to-route) 4.3 [Violation Logging](https://chatgpt.com/c/67afc129-8a3c-8012-8543-a7b972444bdf#violation-logging) 4.4 [Next-Location Estimation](https://chatgpt.com/c/67afc129-8a3c-8012-8543-a7b972444bdf#next-location-estimation) 4.5 [Stationary Pallet Detection (Proposed Extension)](https://chatgpt.com/c/67afc129-8a3c-8012-8543-a7b972444bdf#stationary-pallet-detection-proposed-extension)

5 [Conclusion and Findings](https://chatgpt.com/c/67afc129-8a3c-8012-8543-a7b972444bdf#conclusion-and-findings) 5.1 [Route Adherence & Deviations](https://chatgpt.com/c/67afc129-8a3c-8012-8543-a7b972444bdf#route-adherence--deviations) 5.2 [Next-Location Predictions](https://chatgpt.com/c/67afc129-8a3c-8012-8543-a7b972444bdf#next-location-predictions) 5.3 [Stationary Pallets](https://chatgpt.com/c/67afc129-8a3c-8012-8543-a7b972444bdf#stationary-pallets) 5.4 [Enhanced Analytics (Autoencoder & Random Forest)](https://chatgpt.com/c/67afc129-8a3c-8012-8543-a7b972444bdf#enhanced-analytics-autoencoder--random-forest) 5.5 [Overall Business Impact](https://chatgpt.com/c/67afc129-8a3c-8012-8543-a7b972444bdf#overall-business-impact)

6 [Supporting Documentation](https://chatgpt.com/c/67afc129-8a3c-8012-8543-a7b972444bdf#supporting-documentation)

## **1 Business Objective**

This document details a system for monitoring and managing pallet movements in a logistics network. The primary goal is to ensure each pallet adheres to known transport routes, identify any deviations, and leverage predictive methods to forecast next steps, thereby reducing costs, enhancing visibility, and mitigating theft/loss risks.

### **1.1 Background**

Organizations often face significant challenges when tracking high-value or time-sensitive shipments. Palletized goods may deviate from known routes due to routing errors, driver mistakes, or malicious actions. Detecting these deviations swiftly is crucial to prevent losses, maintain operational efficiency, and meet customer delivery expectations.

### **1.2 Business Case**

* **Route Adherence**: Real-time route adherence tracking can reduce costly rerouting, fuel waste, and potential theft or misplacement of goods.
* **Predictive Insight**: Being able to forecast a pallet’s next location enables proactive decision-making—rerouting or customer communication can occur before a problem escalates.
* **Stationary Pallet Detection**: Pallets that remain idle too long can create bottlenecks and delay time-critical deliveries.
* **Scalability & Cost**: The solution must be efficient and capable of handling large volumes of GPS data, minimizing both software complexity and hardware footprint.

## **2 Project Management**

### **2.1 Project Roles**

* **Logistics/Operations Manager**: Oversees end-to-end supply chain processes and sets thresholds for route deviations.
* **Data Engineer**: Cleans, merges, and formats GPS data (historical and real-time) and prepares it for analysis.
* **Data Scientist/Analyst**: Designs and implements the deviation detection logic, next-location prediction models, and advanced ML methods (autoencoders, random forests).
* **IT/Infrastructure Team**: Maintains servers, dashboards, and notification systems (e.g., SMTP alerts).
* **Business Stakeholders**: Define success criteria (e.g., 95% on-route adherence) and ensure alignment with operational goals.

### **2.2 Project Plan**

1. **Assessment & Definition**
   * Collect business requirements (key route adherence metrics, real-time alerts).
   * Identify data sources and define thresholds.
2. **Data Preparation**
   * Clean and validate historical route data.
   * Set up real-time ingestion process for ongoing trip data.
3. **Implementation**
   * Develop route adherence engine using shapely for distance calculations.
   * Implement logging of deviations and real-time alerts.
   * Integrate next-location prediction module.
4. **Testing & Validation**
   * Run pilot on select routes.
   * Validate system performance against known route benchmarks.
5. **Deployment & Monitoring**
   * Roll out solution across multiple routes/pallets.
   * Set up dashboards and alert systems.
   * Conduct ongoing refinements.

### **2.3 Key Risks**

* **GPS Data Quality**: Poor or inconsistent GPS signals can lead to false deviations.
* **Threshold Sensitivity**: Setting a distance threshold too large may allow significant deviations; too small may generate excessive false alerts.
* **Scalability**: High data volumes could tax system resources, potentially delaying real-time alerts.
* **Model Drift**: Over time, new routes or logistical conditions may reduce the accuracy of predictive models if not periodically retrained.

## **3 Methodology**

### **3.1 Data Preparation and Availability**

* **Historical Route Data (known\_route\_repo.csv)**
  + **Columns**: [Route\_ID, timestamp, latitude, longitude]
  + **Usage**: Transformed into shapely LineString objects representing known/ideal paths.
* **Ongoing Trip Data (On-going trip data.csv)**
  + **Columns**: [Pallet\_ID, timestamp, latitude, longitude]
  + **Usage**: Compared in near real-time to identify off-route behavior.
* **Processing & Assumptions**
  + **Timestamp Conversion**: Ensure all date-time fields use consistent formats.
  + **Coordinate System**: Validate that all lat/long data shares the same reference system.
  + **Data Quality**: Confirm minimal missing entries/duplicates.

### **3.2 Proposed Approach**

#### **3.2.1 Manual (Non-ML) Threshold-Based Method**

1. **Route Building with Shapely**
   * Convert each Route\_ID in the historical dataset into a LineString.
   * For each incoming pallet location, compute the minimum distance to each known route.
2. **Deviation Detection**
   * Choose a threshold (e.g., 500m or 1000m).
   * If the minimum distance to any route exceeds this threshold, flag the pallet as off-route.
3. **Next-Location Prediction**
   * Use simple timestamp-based interpolation along the matched historical route to guess the next location.
   * No complex ML involved; relies on time differences between successive route points.
4. **Stationary Pallet Detection (Extension)**
   * Monitor dwell time at known depots or customer locations.
   * If a pallet remains idle beyond typical dwell times, trigger an alert.

#### **3.2.2 Advanced Self-Improving Methods**

1. **Autoencoder for Anomaly Detection**
   * Train on “normal” lat/lon sequences.
   * High reconstruction error signals subtle route deviations that may be missed by fixed thresholds.
2. **Random Forest Regressor for Next-Location**
   * Features: speed, time of day, route ID, etc.
   * Achieved ~86% F1 on test data, offering more robust predictions than simple interpolation.
3. **Additional Tools and Integrations**
   * **KD-Tree** for nearest neighbor queries (e.g., detect when multiple pallets cluster).
   * **SMTP Alerts**: Automated emails upon threshold violations or excessive dwell times.
   * **Dashboards**: Using Streamlit/Tableau for real-time views and geospatial analysis.
   * **Battery Consumption Monitoring**: Predict battery drain patterns, schedule mid-journey swaps.

## **4 Analysis**

### **4.1 Distance Calculation & Thresholding**

* Employed shapely’s LineString.distance() to obtain planar distances in degrees, then converted degrees to meters for practical thresholds (e.g., 500m, 1000m).
* The threshold depends on route complexity, cargo value, and acceptable risk.

### **4.2 Mapping Pallet to Route**

* For each real-time pallet coordinate, compute the distance to every known LineString route.
* Assign (pallet → route) based on the smallest distance found.
* Store (matched\_route, distance\_to\_route, is\_deviating) for auditing.

### **4.3 Violation Logging**

* If distance\_to\_route > threshold, log an event in violations.csv.
* Captures time of day, route ID, and pallet location for further investigation.

### **4.4 Next-Location Estimation**

* Identify the nearest historical index i on the matched route.
* Project time difference Δt from i → i+1 to estimate next location at pred\_timestamp.
* Store (pred\_timestamp, pred\_lat, pred\_lon) in a predictions table or feed it into the alert system.

### **4.5 Stationary Pallet Detection (Proposed Extension)**

* Monitor dwell time using known coordinates for depots or demand centers.
* If a pallet’s last reported positions remain within a small radius over a predefined threshold, create a “stationary” alert.
* Helps manage supply chain bottlenecks by highlighting potential hold-ups.

## **5 Conclusion and Findings**

### **5.1 Route Adherence & Deviations**

* The threshold-based system reliably flags off-route pallets with minimal complexity.
* Deployment is straightforward, requiring basic geospatial libraries (e.g., shapely).

### **5.2 Next-Location Predictions**

* Simple interpolation offers a quick, real-time forecast of pallet position.
* For improved accuracy, integrating a random forest or other ML approach can enhance predictive power.

### **5.3 Stationary Pallets**

* Identifying idle pallets at depots or demand centers reduces supply chain blockages and ensures smooth throughput.
* Alerts for prolonged inactivity enable immediate corrective actions.

### **5.4 Enhanced Analytics (Autoencoder & Random Forest)**

* **Autoencoder**: Captures subtle, high-dimensional deviations that threshold methods might miss.
* **Random Forest**: Increases next-location accuracy by incorporating additional features (time of day, speed, route ID, etc.).

### **5.5 Overall Business Impact**

* **Improved Visibility**: Real-time knowledge of pallet location and adherence significantly reduces losses and potential delays.
* **Reduced Costs**: Early detection of deviations minimizes fuel overages, re-routing fees, and operational disruptions.
* **Scalability**: A modular, Python-based design (via the RouteMapper class, CSV exports, and dashboards) can scale to new routes, expanded fleets, or broader sets of data.
* **Future Expansion**: Integrations for battery life monitoring, advanced geofencing, and ML pipelines can evolve the system from simple detection to comprehensive route intelligence.

## **6 Supporting Documentation**

* **Route Data and Ongoing Trip Data**: CSVs (known\_route\_repo.csv, On-going trip data.csv) containing lat/long details, timestamps, and route IDs.
* **Violations Log**: violations.csv, capturing off-route events with timestamps.
* **ML Model Artifacts (Optional)**: Stored model files (autoencoder, random forest) for advanced predictions.
* **Dashboards and Visualization**: Links or references to Streamlit/Tableau dashboards showcasing real-time route adherence and historical route mapping.