TECHNISCHE HOCHSCHULE DEGGENDORF

SMARTPHONE-BASED

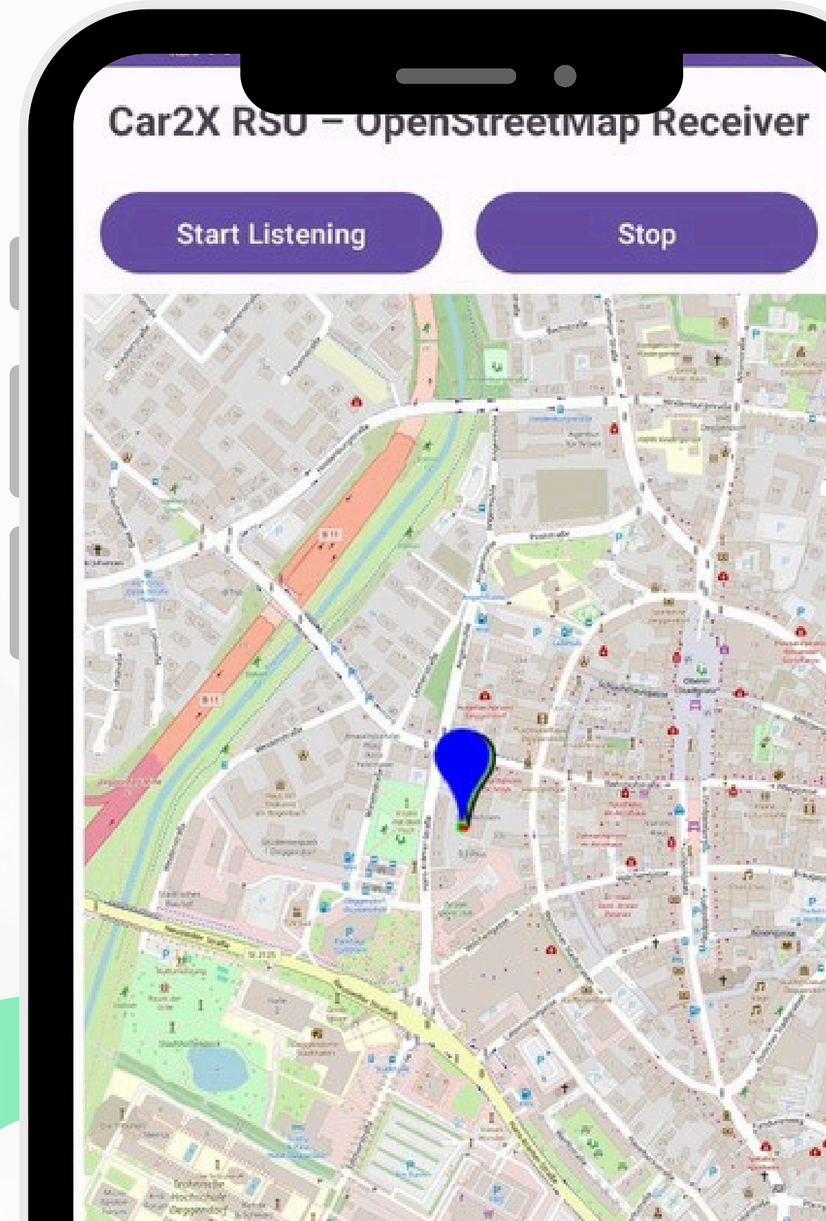
RSU – CAR2X DEMO

Low-Cost Vehicle-to-Infrastructure Communication using Android

Presented by:
Deep Bharatbhai Savaliya - 12501180
Satyajit Sushant Pardeshi - 22408966
Yutsav Hari Bhagat - 12500192
Niraliben Yash Jani - 22402416

Supervisor : Prof. Dr. Andreas Kassler

Study Program : Master Automotive Software Engineering
Semester : 2nd

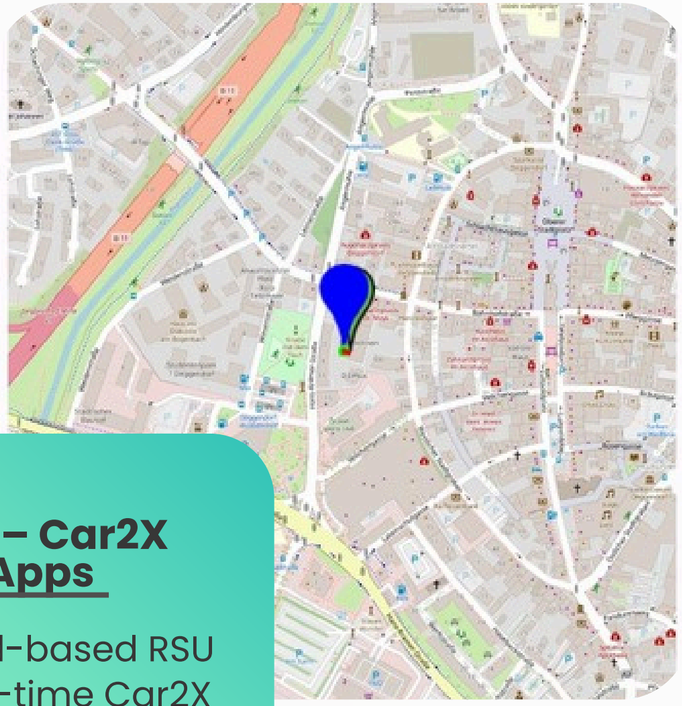


Contribution Details

FirstName	LastName	Matriculation	Email Ids	Ownership
Niraliben Yash	Jani	22402416	niraliben.jani@stud.th-deg.de	System Architecture & Communication Protocols
Deep Bharatbhai	Savaliya	12501180	deep.savaliya@stud.th-deg.de	Vehicle beacon implementation + UI (RSU App)
Satyajit Sushant	Pardeshi	22408966	satyajit.pardeshi@stud.th-deg.de	RSU detection + logging
Yutsav Hari	Bhagat	12500192	yutsav.bhagat@stud.th-deg.de	Analysis + documentation

Agenda

- Introduction to V2X
- Project motivation
- System overview
- CAM & DENM implementation
- Safety logic
- Demo application
- Field testing & results
- Challenges & future work



RSU – Car2X Apps

Android-based RSU for real-time Car2X vehicle detection and warning generation

What is V2X?

Vehicle-to-Everything (V2X) enables communication between:

- Vehicle ↔ Vehicle (V2V)
- Vehicle ↔ Infrastructure (V2I)
- Vehicle ↔ Pedestrians (V2P)

Image Source

Benefits

- Safety
- Traffic efficiency
- Collision avoidance
- Real-time warnings

Role of RSU in V2X

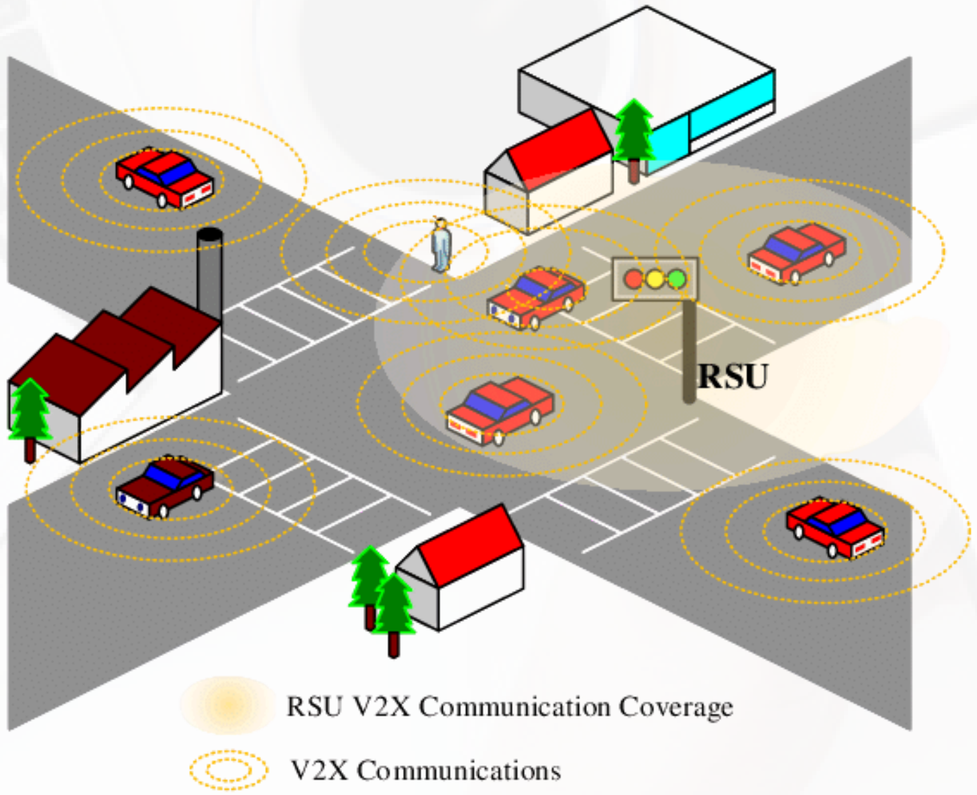


Image Source

Road Side Unit (RSU):

- Fixed infrastructure node
- Receives vehicle messages
- Performs processing & decision making
- Broadcasts warnings

In real systems:
ITS-G5 / C-V2X hardware

In our project: Smartphones

Motivation of This Project

Problem

- Real-world Car2X systems are complex and difficult to observe directly
- Safety message behavior (CAM/DENM) is often studied only theoretically

Motivation

- To practically understand how Car2X safety messages work
- To visualize vehicle-to-RSU interaction in real time
- To experiment with distance-based safety logic in real environments

Why this Project Matters

- Bridges theory (ETSI V2X messages) with hands-on implementation
- Demonstrates real-time safety decision making
- Helps validate concepts like CAM, DENM, RSU behavior

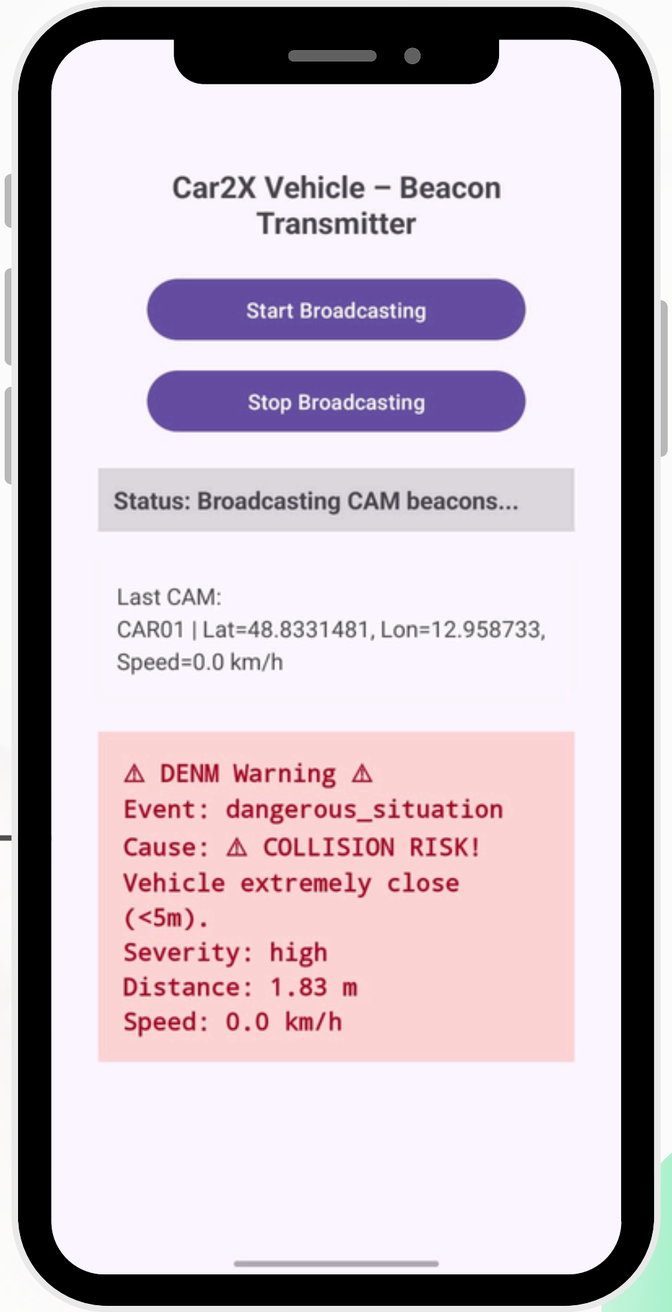
Project Goal & Scope

Goal:

Build a functional Car2X demo using smartphones

Scope:

- Vehicle broadcasts position & speed
- RSU detects approaching vehicles
- RSU generates safety warnings
- Real-world outdoor testing



System Overview

Two Android Applications:

1. Vehicle App (OBU)
2. RSU App (Receiver + Processor)

Network:

- Local WiFi hotspot
- UDP communication
- Low latency (<100 ms)

End-to-End Flow:

1. Vehicle generates CAM messages
2. RSU receives and filters data
3. Distance & risk evaluated
4. DENM generated if required
5. Warning sent back to vehicle

Key Property:

- Real-time, event-driven processing

Communication Protocol

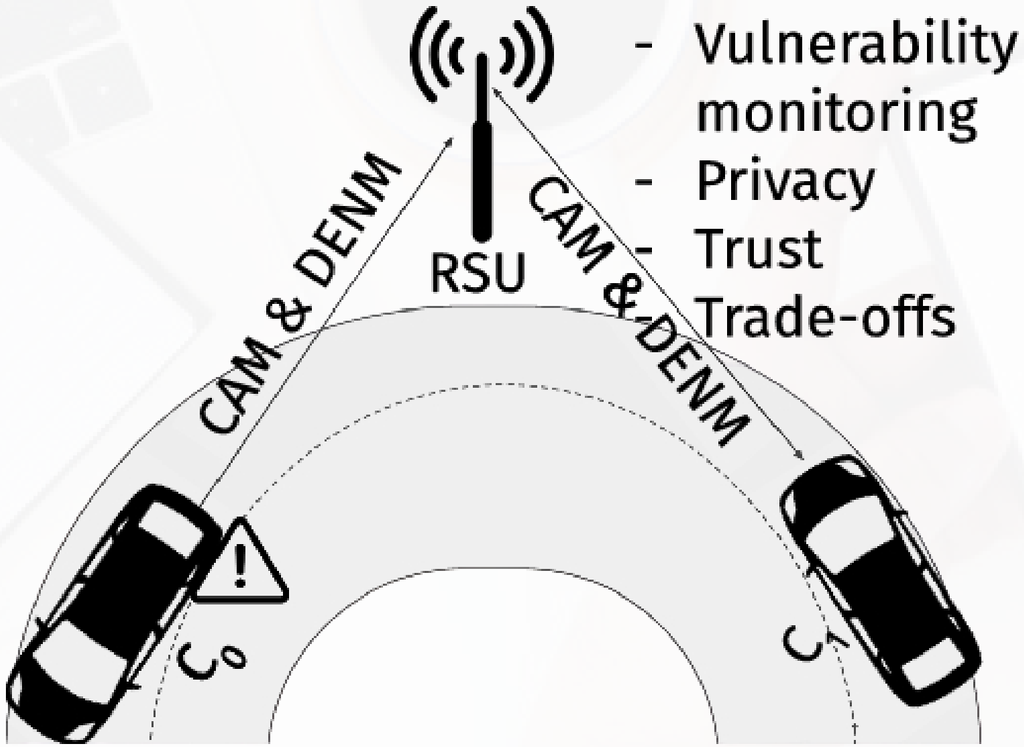


Image Source

Why UDP?

- Low latency
- No connection overhead
- Suitable for safety messages

Ports Used:

- CAM → Port 30001
- DENM → Port 30002

CAM: Cooperative Awareness Message

CAM Characteristics:

- Periodic message (2 Hz in our system)
- Broadcasted by vehicles
- Contains:
  - Vehicle ID
  - Latitude & Longitude
  - Speed
  - IP address

Purpose:

Maintain awareness of nearby vehicles

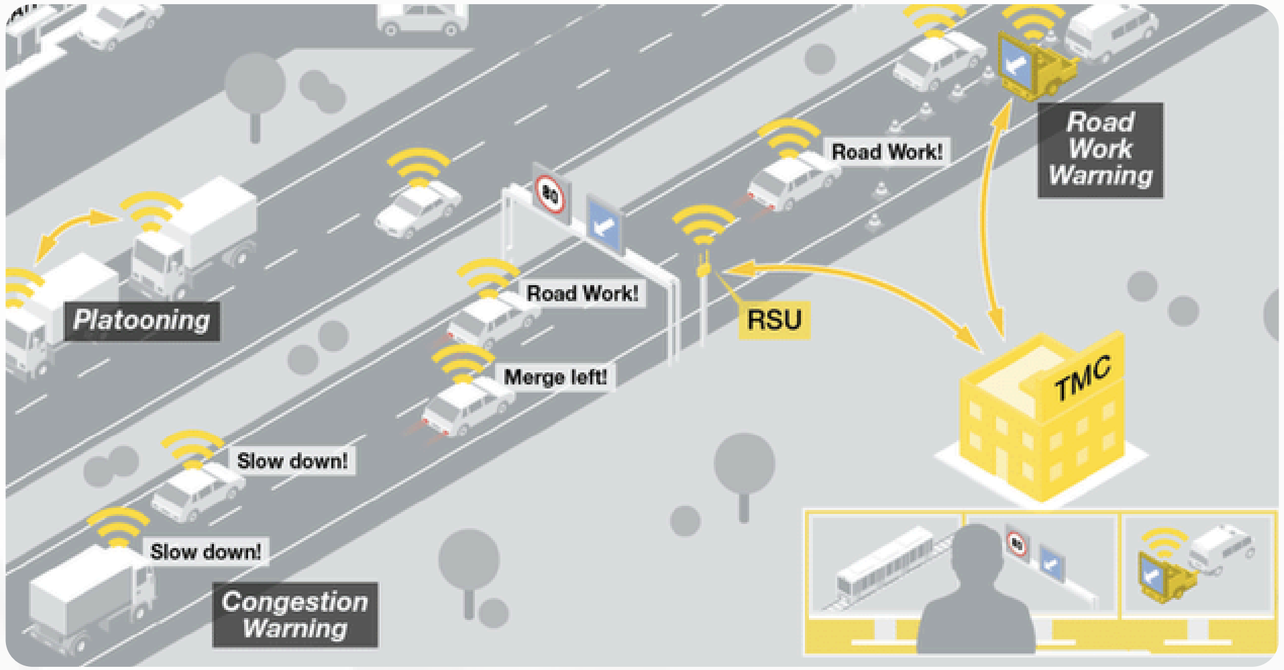


Image Source

DENM: Event-Based Warning

DENM Characteristics:

- Event-triggered (not periodic)
- Generated by RSU
- Contains:
  - Event type
  - Cause
  - Severity

Purpose:

Warn about dangerous situations

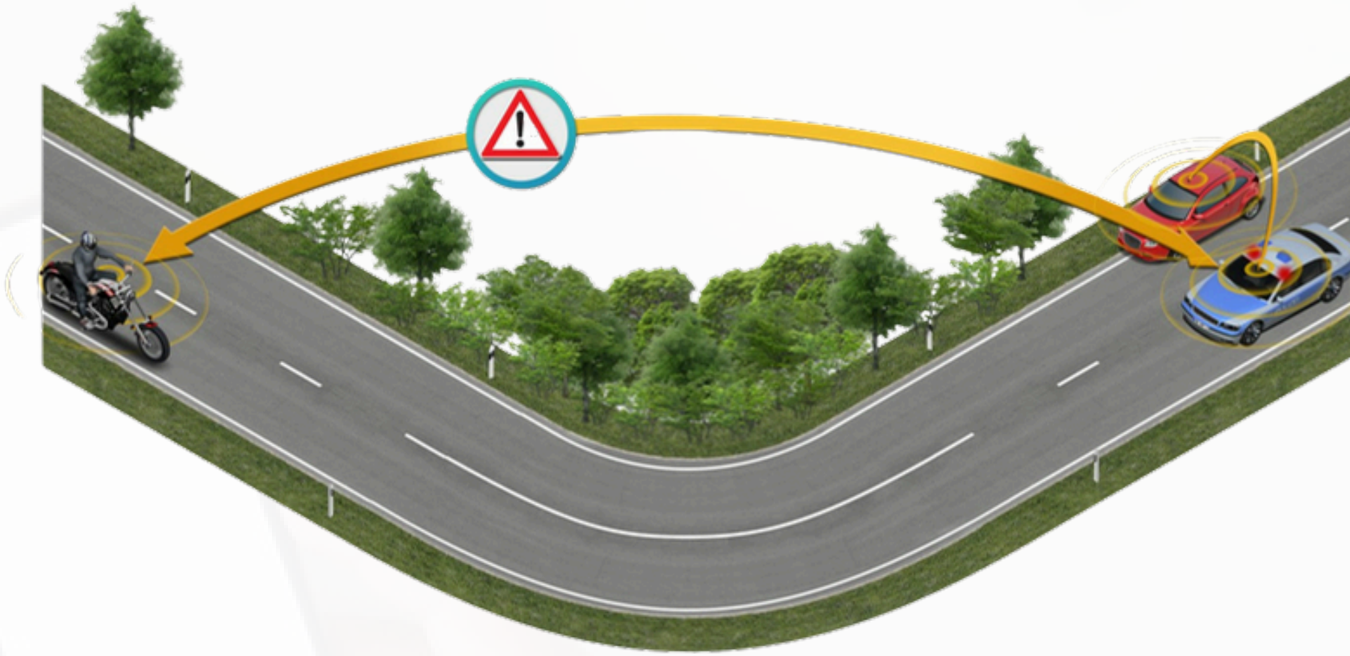


Image Source

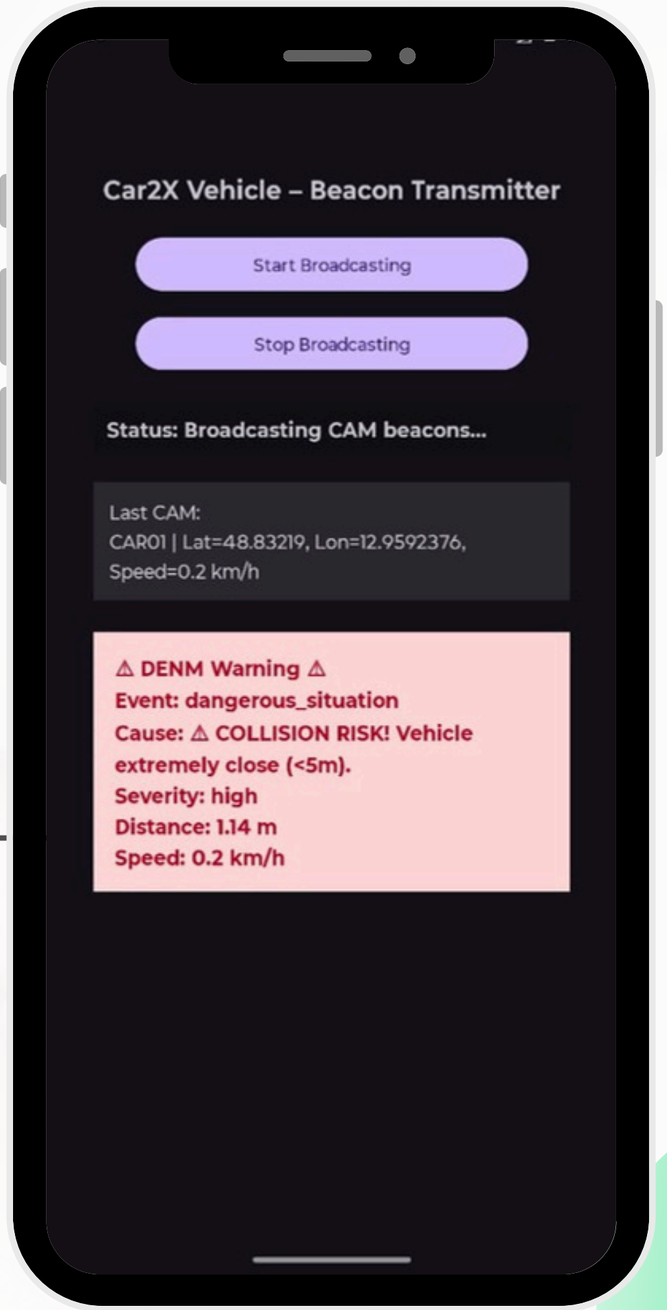
Vehicle Application (OBU)

Functions:

- GPS acquisition
- Speed calculation
- CAM generation
- UDP broadcasting
- DENM reception & alert display

UI Highlights:

- Start/Stop broadcasting
- Last CAM display
- Warning panel



GPS Filtering Logic (Vehicle Side)

Problem: GPS jitter causes false speed spikes

Solution:

- Reject GPS accuracy > 25 m
- Reject unrealistic speed jumps (>43 km/h)
- Smooth position updates

Result:

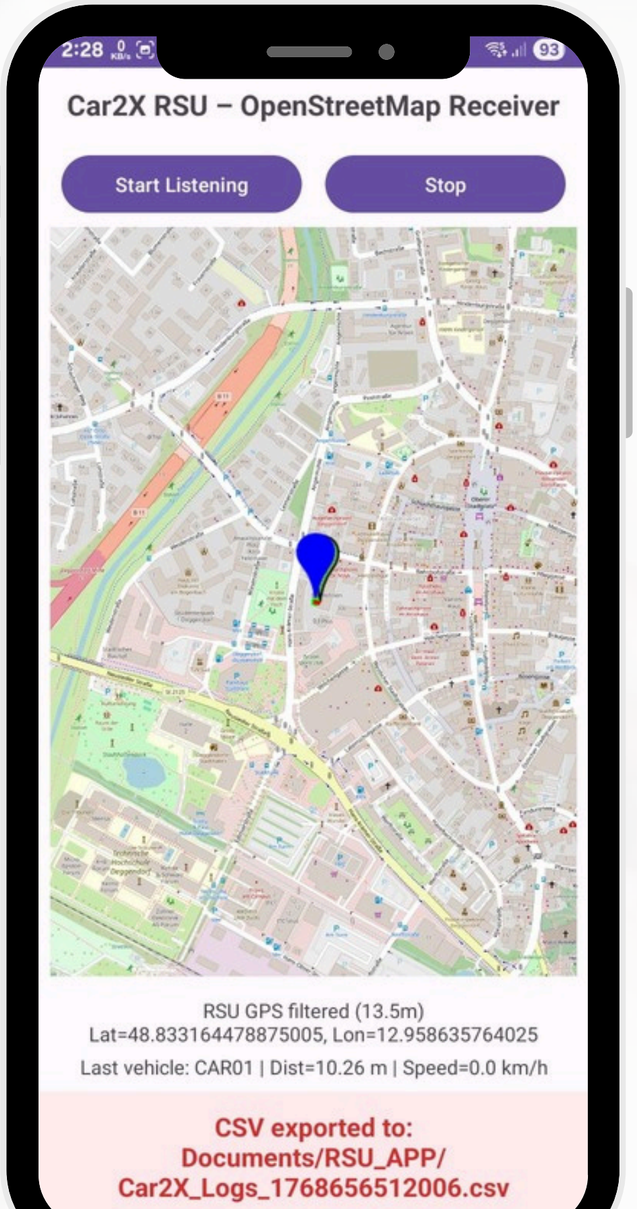
Stable speed & position data

Image Source

RSU Application Design

Role of RSU in Car2X

- Acts as a stationary infrastructure node
- Receives periodic CAM messages from vehicles
- Performs local safety assessment
- Generates event-based warnings (DENM)



In this project:

- RSU functionality is emulated using an Android smartphone
- Processing is done locally without backend servers

RSU Map Visualization

Purpose of Visualization:

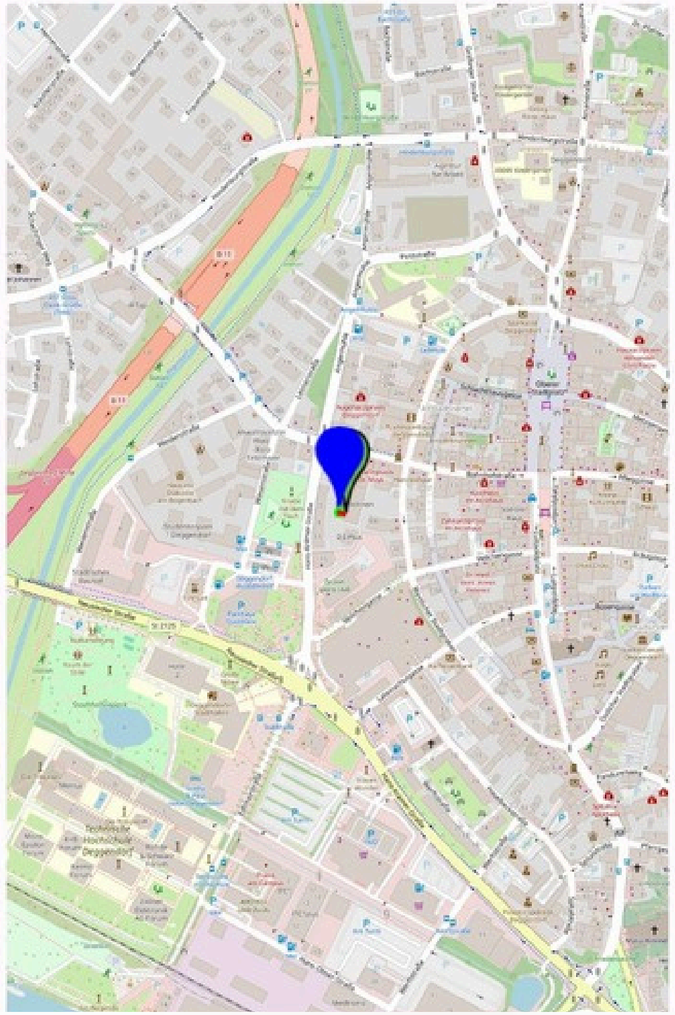
- Improve situational awareness
- Observe vehicle movement in real time

Implementation:

- OpenStreetMap (OSM)
- Displays:
  - RSU position (fixed)
  - Vehicle position (dynamic)
  - Safety zones and trajectories

Why OSM?

- Open-source
- Offline capable
- Commonly used in ITS prototypes



RSU Location Stabilization

Problem:

- GPS drift even when RSU is stationary

Solution Applied:

- Location smoothing using filtering techniques
- Prevents false distance variations

Benefit:

- More accurate distance calculation
- Reduced false collision warnings

Geofencing Zones

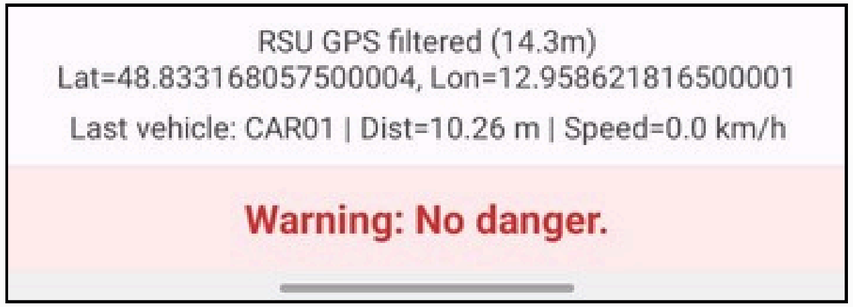
Defined Safety Zones:

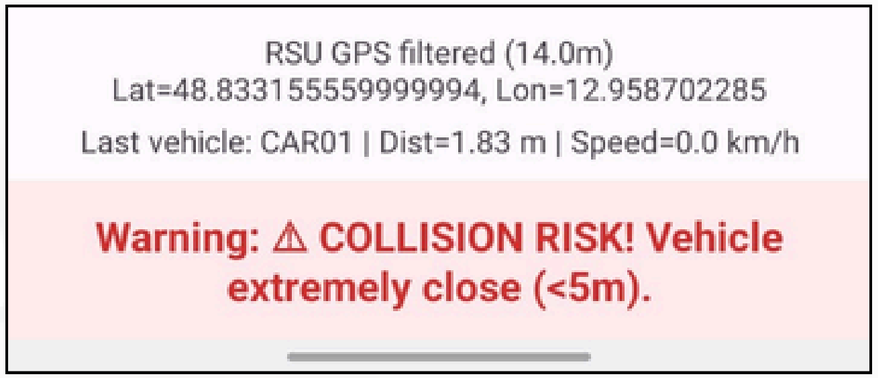
🟢 Warning Zone → Distance < 7 m

🔴 Critical Zone → Distance < 5 m

Function:

- Classifies risk levels visually
- Supports fast decision making at RSU





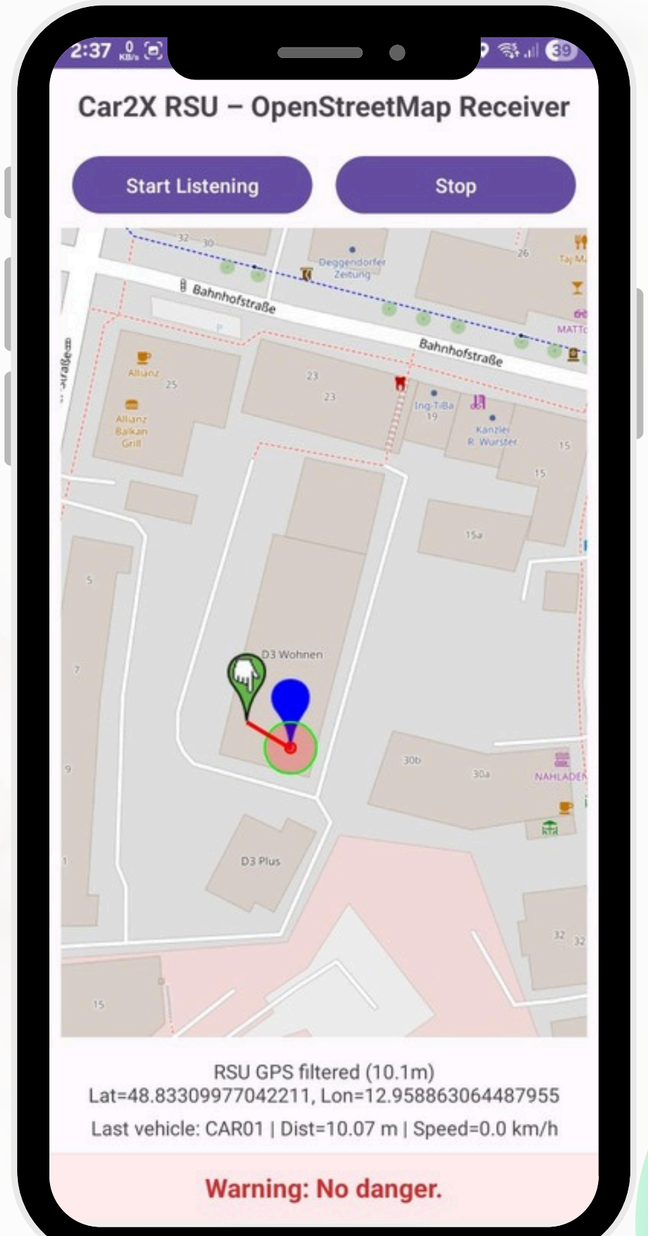
Trajectory Visualization

What is shown:

- Line between RSU and vehicle
- Direction of approach

Why important:

- Helps interpret vehicle behavior
- Supports collision risk understanding
- Useful for debugging and demonstrations



Safety Logic & Algorithms

Safety logic is RSU-centric:

- Vehicle only broadcasts data
- RSU evaluates risk

Inputs:

- Distance
- Speed

Output:

- Risk level
- DENM generation decision

Distance Calculation

Method:

- GPS coordinates
- Android Location API (Haversine-based)

Why distance matters:

- Core parameter for collision detection
- Used in all risk thresholds

The distance between the vehicle and the RSU is calculated using GPS coordinates.

Formula (Haversine / Android Location API concept):

$$\mathbf{d} = R \times \arccos(\sin\phi_1 \sin\phi_2 + \cos\phi_1 \cos\phi_2 \cos(\lambda_2 - \lambda_1))$$

Meaning of Each Term

d – Distance between two locations (in meters)

R – Radius of the Earth (≈ 6,371,000 meters)

ϕ_1, ϕ_2 – Latitudes of point 1 and point 2 (in radians)

λ_1, λ_2 – Longitudes of point 1 and point 2 (in radians)

arccos – Inverse cosine function

Risk Evaluation Logic

Condition	Criteria	Risk
A	d < 5 m & speed > 30 km/h	Critical
B	d < 5 m	Collision Risk
C	d < 7 m	Approaching

Design Principle:

- Simple, deterministic logic
- Suitable for real-time processing

Warning Generation

DENM Characteristics:

- Event-triggered
- Asynchronous
- Sent only when risk detected

DENM Content:

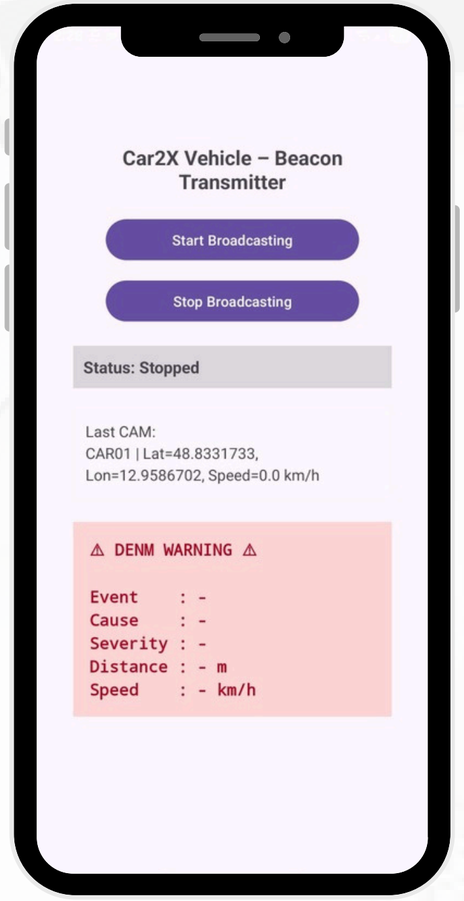
- Event type
- Severity
- Cause

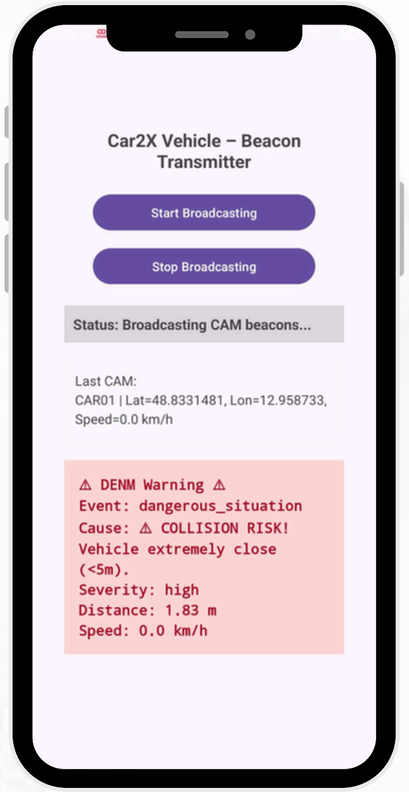
Purpose:

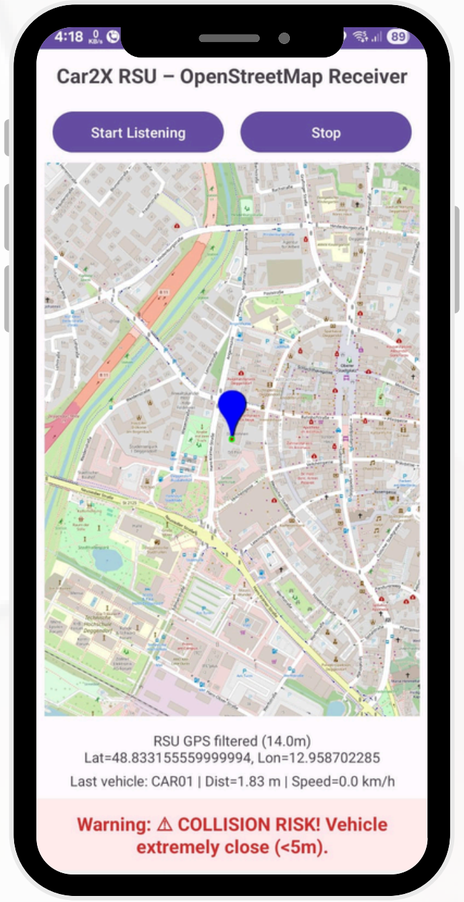
- Immediate driver awareness

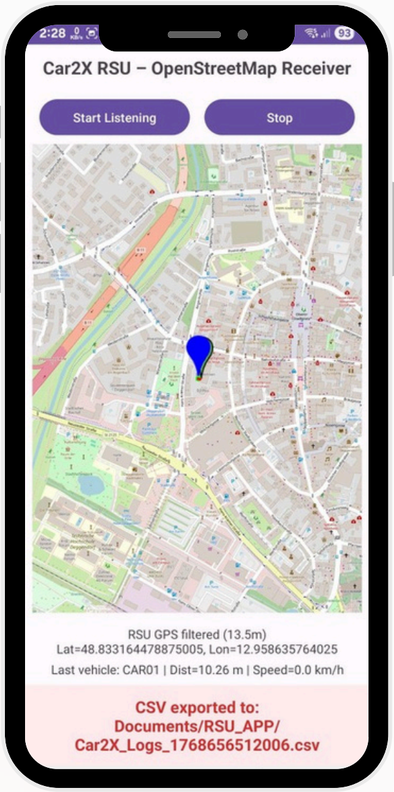
App Screenshots

The Vehicle (OBU) application broadcasts periodic CAM messages, while the RSU application receives, processes, and visualizes vehicle data to generate safety warnings.









Field Testing & Evaluation

Test Environment:

- Outdoor
- Real GNSS conditions

Scenarios:

- Stationary
- Walking speed
- Running / cycling
- Overspeed approach

Goal:

- Validate accuracy
- Measure reliability

Results & Observations

Results:

- Reliable vehicle detection
- Low latency warning delivery
- Stable visualization

Logging:

- CSV file
- Timestamp, distance, speed, warning status
- Upon clicking the 'Stop' button in the RSU application, the system exports a CSV file containing communication and safety event data, saved locally on the mobile device.

Conclusion & Future Improvements

Key Achievements:

- Functional smartphone-based RSU
- Real-time Car2X demo
- Practical validation of CAM & DENM concepts

Educational Value:

- Bridges lecture theory with real implementation

Future Improvements:

- Kalman filter for prediction
- Multi-vehicle handling
- BLE / C-V2X instead of WiFi
- Intersection-based scenarios

References

- **Prof. Dr. Andreas J. Kassler**, Wireless and Car-X Communication – V2X Intro and Messages, Technische Hochschule Deggendorf, WS 2025/26.
- **Android Developers**, Location Services & Fused Location Provider, https://developer.android.com/training/location
- **OpenStreetMap Contributors**, OpenStreetMap Data, https://www.openstreetmap.org
- **Project Report**, Smartphone-Based RSU – Car2X Demo, Master's in Automotive Software Engineering, TH Deggendorf

TECHNISCHE HOCHSCHULE DEGGENDORF

Thank You

Presented by:
Deep Bharatbhai Savaliya - 12501180
Satyajit Sushant Pardeshi - 22408966
Yutsav Hari Bhagat - 12500192
Niraliben Yash Jani - 22402416

Supervisor : Prof. Dr. Andreas Kassler