Scifat
Conferences, Events and Awards

SF

COMPUTER
International Research
Awards on Computer Vision

Adaptive sensor attack detection and defense framework for autonomous vehicles based on density

Author: Zujia Miao, Cuiping Shao*, Huiyun Li, Yunduan Cui, Zhimin Tang

Contents

1 Background

2 METHOD

3 Experimental Results

4 Conclusion

/01 Background

Background

A. Sensor on the autonomous vehicle system

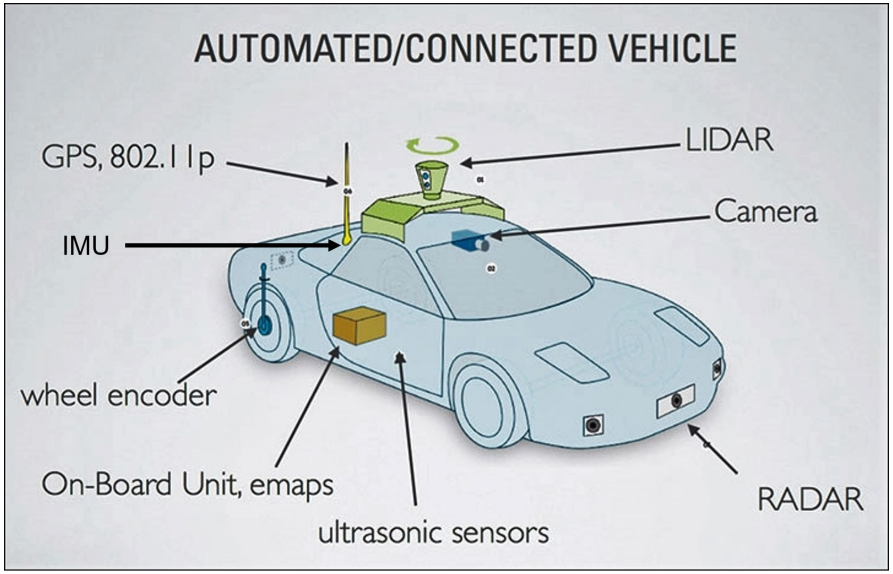


Fig1. Sensors on the autonomous vehicle system

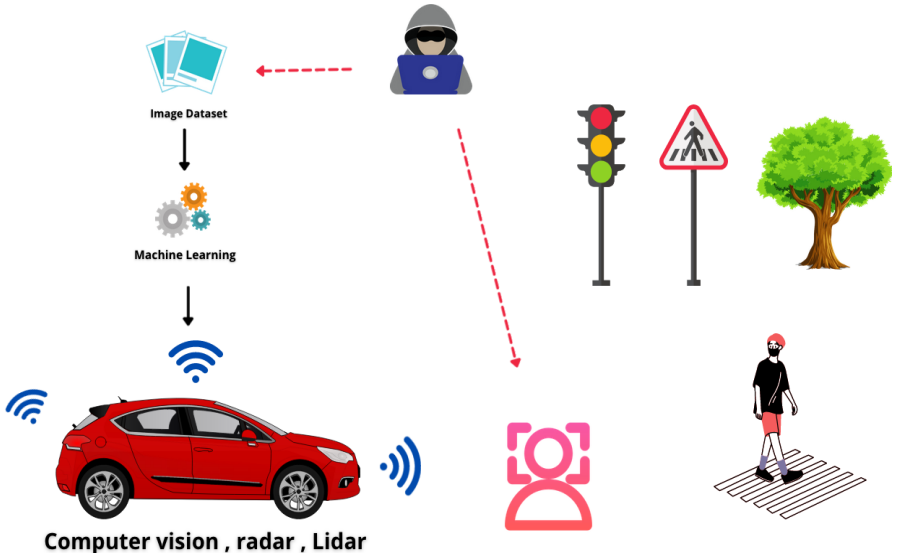
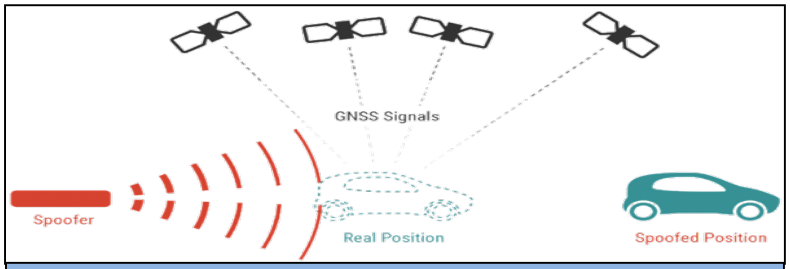


Fig2. Sensors attack

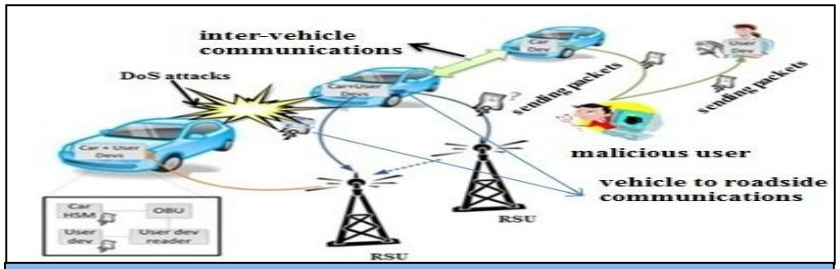
- Environment perception technology is significant in the intelligent transportation system.
- These sensors are vulnerable to external attacks, resulting in the distortion of the information perceived by the unmanned vehicle.
- Attack on autonomous vehicle sensors is a simple, direct, violent, and effective method, which poses an enormous security threat to the autonomous vehicle system.

Background

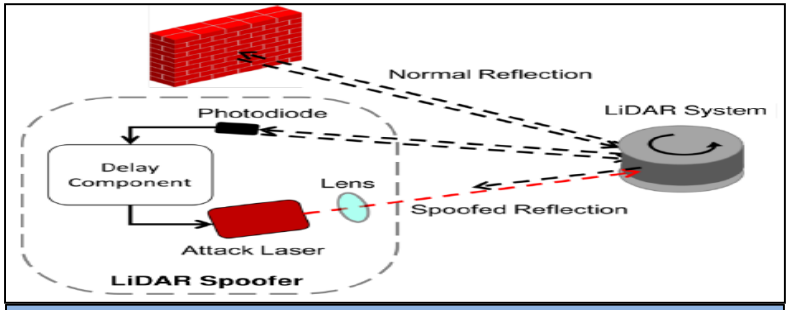
B. Attack on sensors



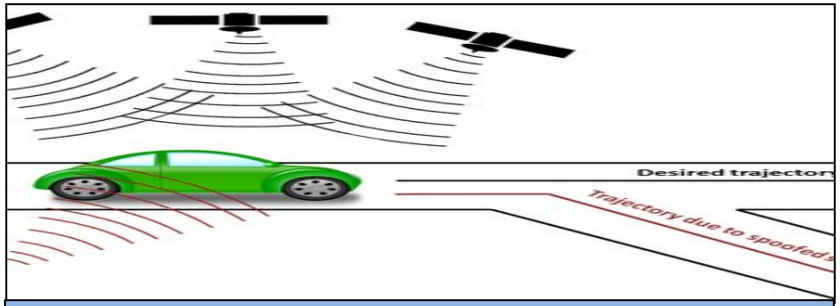
LiDAR spoofing attacks



IMU Dos attack



Lidar relay attacks

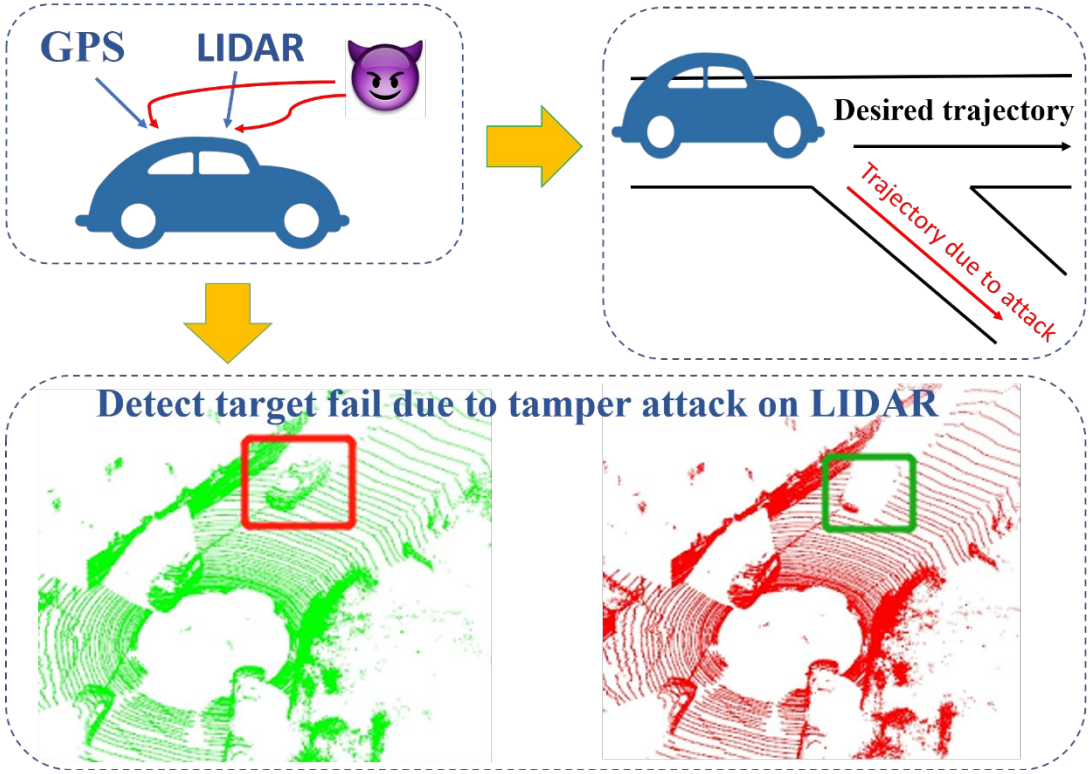


GPS spoofing attack

the development of a method to detect and defend sensor attacks on autonomous vehicle is critical to the safety of autonomous driving

Background

B. Attack on sensors



Common sensor
- GPS
- LiDAR
- IMU

Sensor attack
- Attack outside the system
- have no technology threshold
- Cause serious consequences

Fig 2. Attack on sensors

Background

C. Detection and defense method against sensor attack

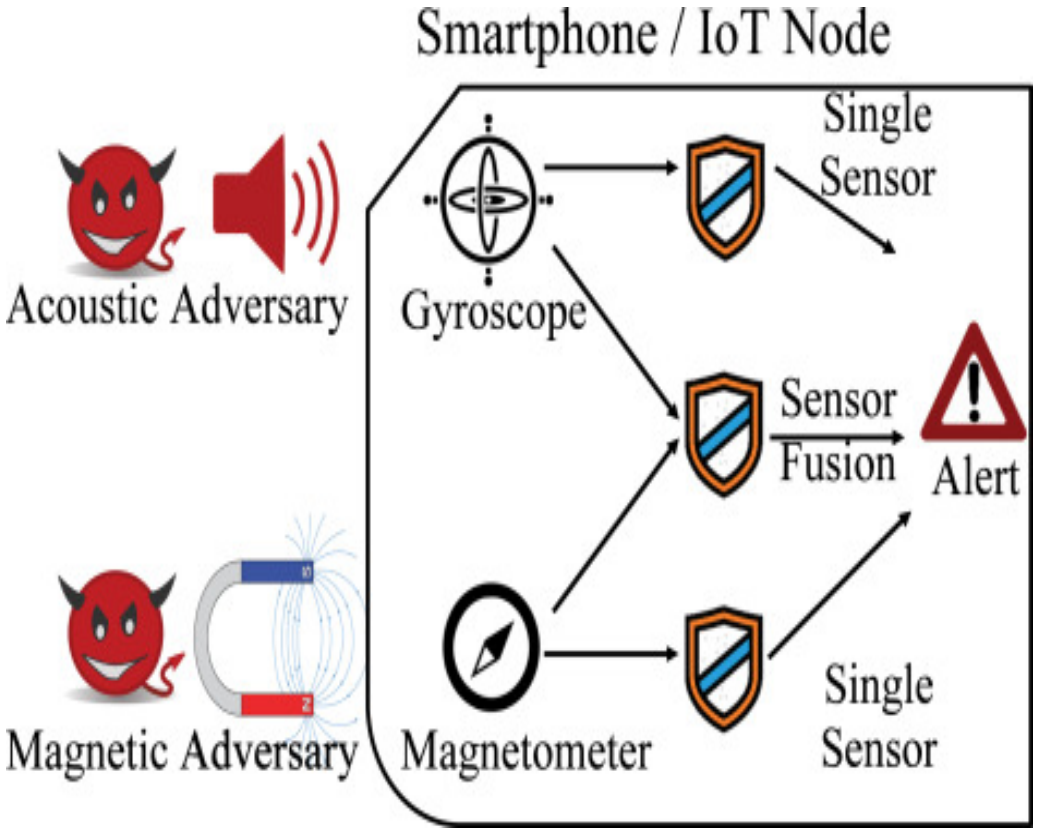


Fig 3. Attack and defense method

Present problems:

- focuses on function development rather than security defense
- few real-time attack defense methods against attacks
- Lack adaptive ability for complex and dynamically changing driving scenarios in autonomous driving

Scifat
Conferences, Events and Awards

SF

COMPUTER
International Research
Awards on Computer Vision

/02 METHOD

METHOD

A. Multi-armed bandit-based DBSCAN algorithm

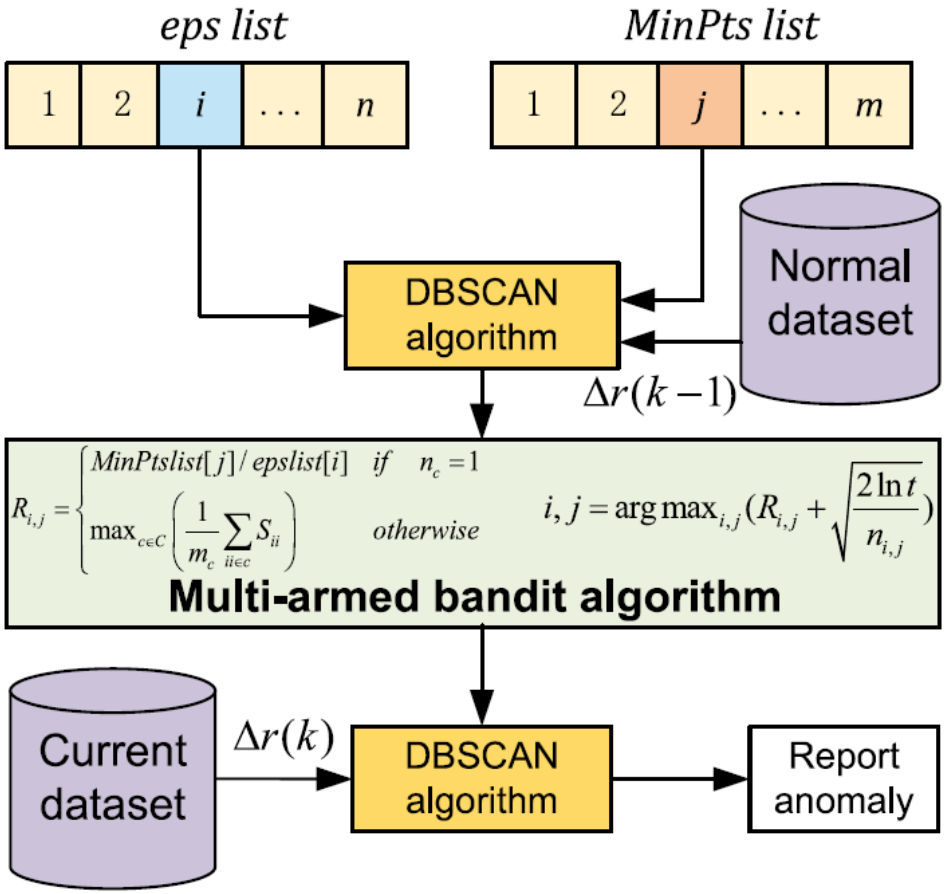


Fig. 5. Multi-armed bandit-based DBSCAN algorithm.

- WHY : adaptive choose the optimal parameters of the DBSCAN algorithm according to reinforce learning MBA in dynamic changing driving scenario
- HOW: Multi-armed bandit algorithm defines each combination of eps and $MinPts$ parameters as an arm and utilizes a reward mechanism to evaluate the efficacy of each arm

METHOD

B. Attack detection

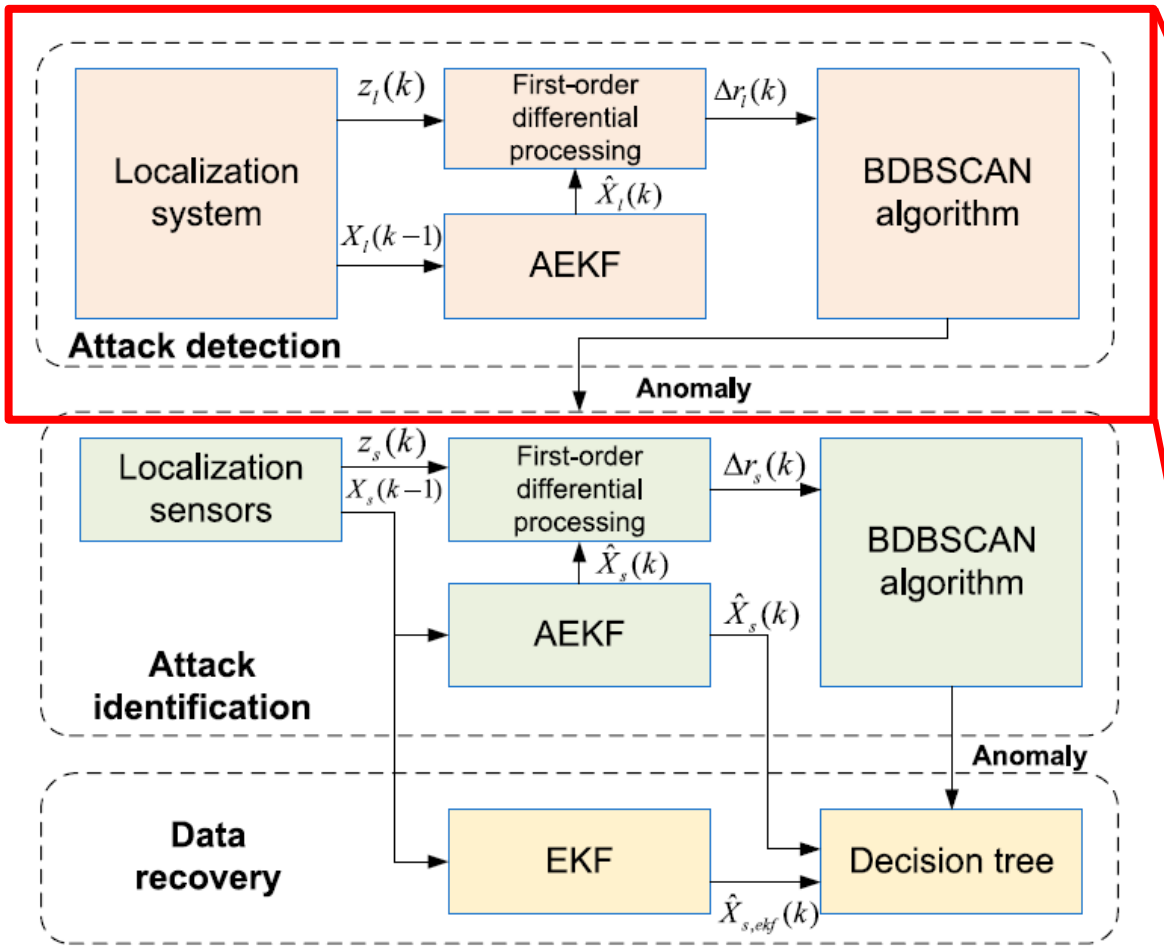


Fig. 4. Illustration of proposed framework.

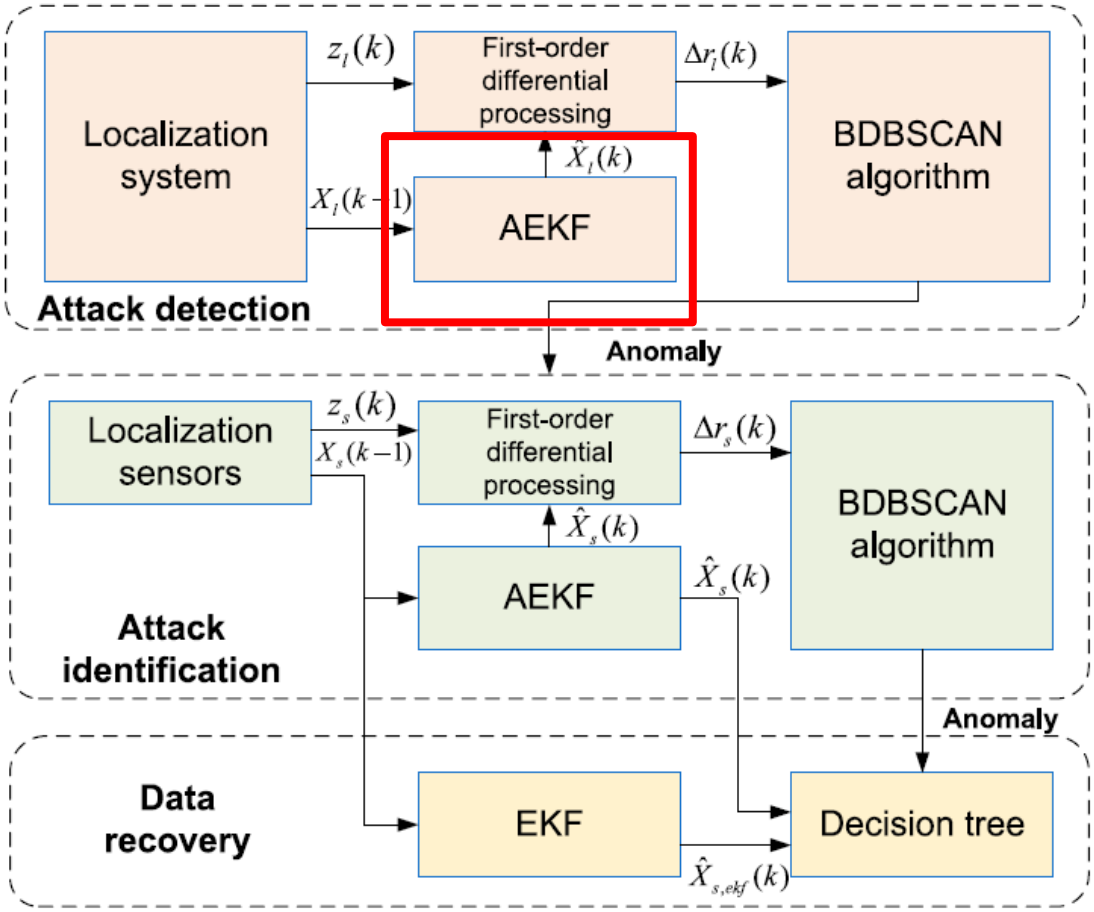
Algorithm 2 Attack detection

Input: Position $(x(k-1), y(k-1))$, orientation $\psi(k-1)$, velocity $v(k-1)$
Output: Alarm signal $\mathcal{A}$

1: $\hat{X}_l(k) \leftarrow AEKF(x(k-1), y(k-1), \psi(k-1), v(k-1))$
2: $r_l(k) = z_l(k) - \hat{X}_l(k)$
3: $\Delta r_l(k) = r_l(k) - r_l(k-1) = \left[\ \Delta r_x(k), \Delta r_y(k), \Delta r_\psi(k) \ \right]^T$
4: $\mathcal{A} \leftarrow BDBSCAN(\Delta r_x(k), \Delta r_y(k))$
5: **if** $\mathcal{A} == True$ **then**
6: Perform Algorithm 3
7: **else**
8: $k = k + 1$
9: **end if**

we deploy a combination of AEKF and BDBSCAN algorithms to the localization system to achieve attack detection.

B. Attack detection



$$\boldsymbol{P}(k+1|k) = \boldsymbol{F}(\hat{\boldsymbol{X}}(k))\boldsymbol{P}(k|k)\boldsymbol{F}(\hat{\boldsymbol{X}}(k))^T + \boldsymbol{Q}_F(k)$$

$$\boldsymbol{\Sigma}(k+1) = \boldsymbol{C}(k)\boldsymbol{P}(k+1|k)\boldsymbol{C}(k)^T + \boldsymbol{R}_F(k)$$

$$\boldsymbol{K}(k+1) = \boldsymbol{P}(k+1|k)\boldsymbol{C}(k)^T\boldsymbol{\Sigma}(k+1)^{-1}$$

$$\boldsymbol{P}(k+1|k+1) = [\boldsymbol{I}_n - \boldsymbol{K}(k+1)\boldsymbol{C}(k)]\boldsymbol{P}(k+1|k)$$

$$\boldsymbol{Y}(k+1) = (\boldsymbol{I}_n - \boldsymbol{K}(k+1)\boldsymbol{C}(k))\boldsymbol{F}(\hat{\boldsymbol{X}}(k))\boldsymbol{Y}(k)$$

$$+(\boldsymbol{I}_n - \boldsymbol{K}(k+1)\boldsymbol{C}(k))\boldsymbol{\Phi}(k)$$

$$\boldsymbol{\Omega}(k+1) = \boldsymbol{C}(k)\boldsymbol{F}(\hat{\boldsymbol{X}}(k))\boldsymbol{Y}(k) + \boldsymbol{C}(k)\boldsymbol{\Phi}(k)$$

$$\boldsymbol{\Lambda}(k+1) = [\lambda\boldsymbol{\Sigma}(k+1) + \boldsymbol{\Omega}(k+1)\boldsymbol{S}(k)\boldsymbol{\Omega}(k+1)^T]^{-1}$$

$$\boldsymbol{\Gamma}(k+1) = \boldsymbol{S}(k)\boldsymbol{\Omega}(k+1)^T\boldsymbol{\Lambda}(k+1)$$

$$\boldsymbol{S}(k+1) = \frac{1}{\lambda}\boldsymbol{S}(k) - \frac{1}{\lambda}\boldsymbol{S}(k)\boldsymbol{\Omega}(k+1)^T\boldsymbol{\Lambda}(k+1)\boldsymbol{\Omega}(k+1)\boldsymbol{S}(k)$$

$$\hat{\boldsymbol{X}}(k+1) = \boldsymbol{f}(\hat{\boldsymbol{X}}(k), \boldsymbol{z}(k)) + \boldsymbol{B}(k)\boldsymbol{u}(k) + \boldsymbol{\Phi}(k)\hat{\boldsymbol{\delta}}(k)$$

$$+ \boldsymbol{K}(k+1)\tilde{\boldsymbol{z}}(k+1) + \boldsymbol{Y}(k+1)[\hat{\boldsymbol{\delta}}(k+1) - \hat{\boldsymbol{\delta}}(k)]$$

Fig. 4. Illustration of proposed framework.

B. Attack detection

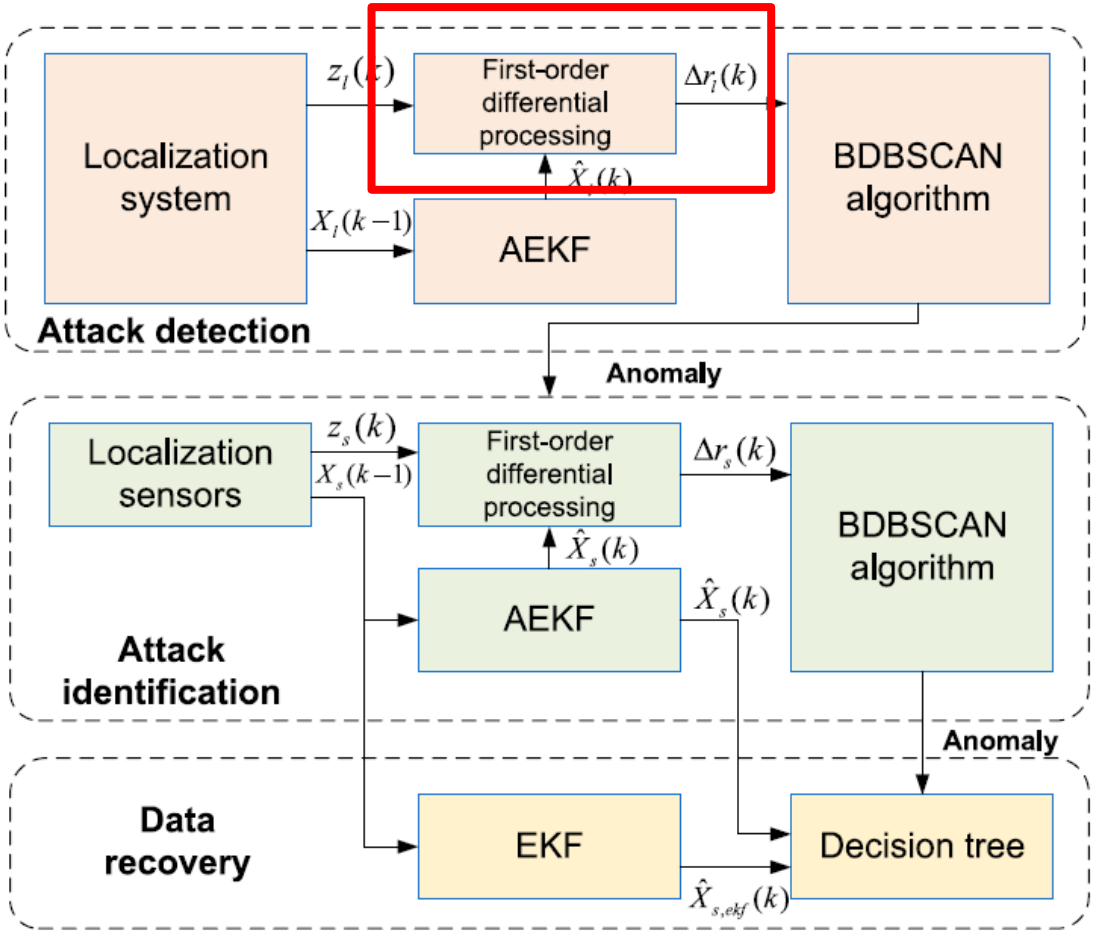


Fig. 4. Illustration of proposed framework.

To further filter out noise and increase data density, we use first-order differencing

$$r_l(k) = z_l(k) - \hat{X}_l(k)$$

$$\Delta r_l(k) = r_l(k) - r_l(k-1) = \left[\Delta r_x(k), \Delta r_y(k), \Delta r_\psi(k)\right]^T$$

METHOD

B. Attack detection

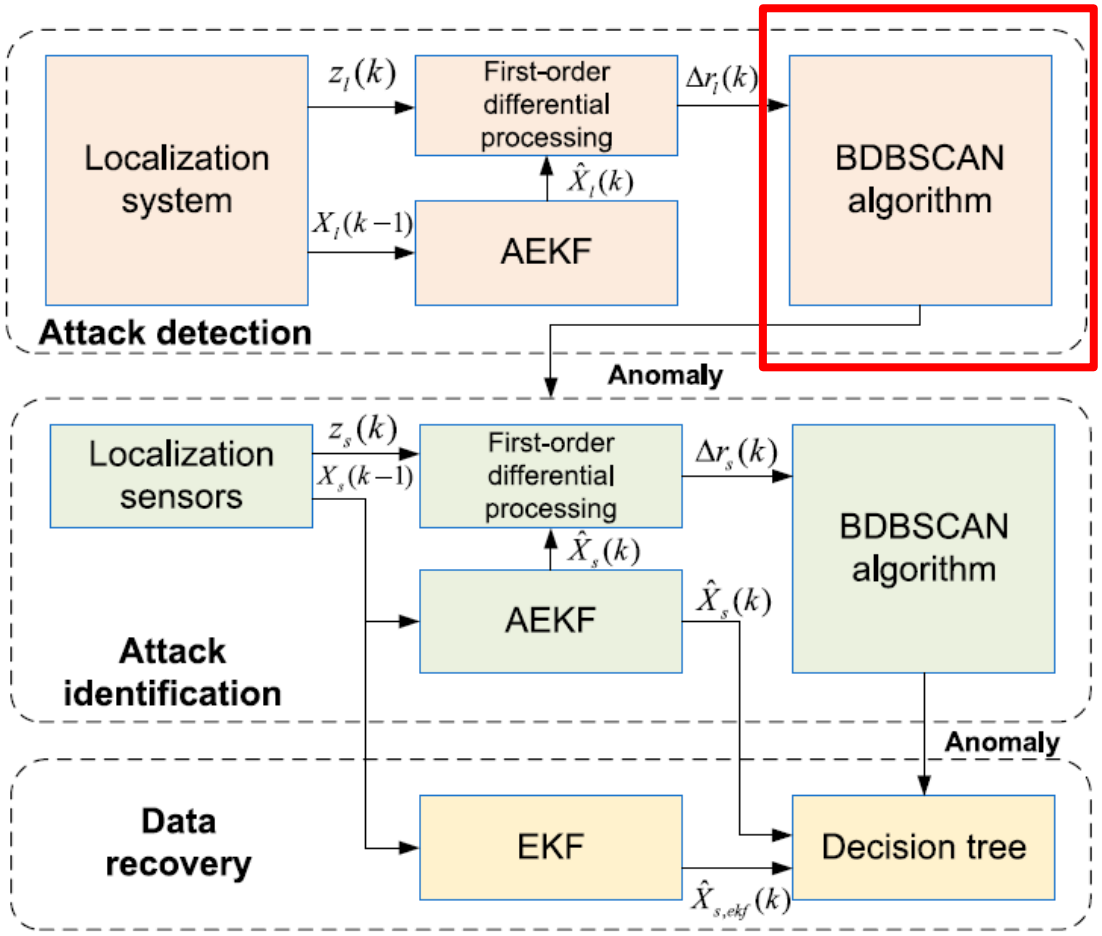


Fig. 4. Illustration of proposed framework.

Algorithm 1 BDBSCAN algorithm

Input: Normal dataset **D**, data feature $\Delta r(k)$, parameter list(*eps list* and *MinPts list*)

Output: Report anomaly

```
i, j ← Equation (20), (21)
Anomaly signal ← DBSCAN(D, Δr(k), eps list[i], MinPts list[j])
if Anomaly signal == True then
    Report anomaly
else
    D ← D ∪ Δr(k)
    k=k+1
end if
```

If the BDBSCAN algorithm reports anomaly, it indicates that the localization system suffer sensor attacks and attack defense will be performed.

C. Attack Defense

1) Attack identification

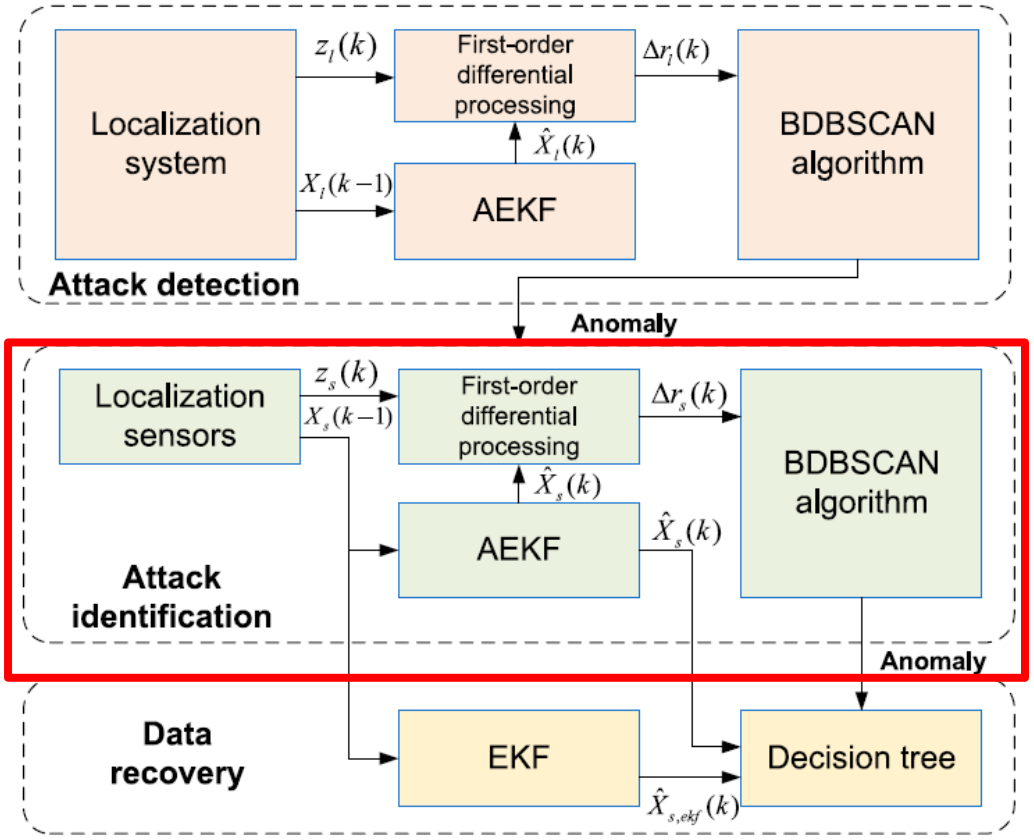


Fig. 4. Illustration of proposed framework.

Algorithm 3 Attack defense, including attack identification and data recovery

Input: GPS position (x_{GPS}, y_{GPS}), LiDAR position (x_{LiDAR}, y_{LiDAR}), orientation ψ, velocity v, alarm signal $\mathcal{A}$, acceleration a.
Output: Attacked sensor S

for s in $[GPS, LiDAR, IMU]$ **do**
  # Perform attack identification
  if $s == GPS$ **then**
    $\hat{X}_s(k) \leftarrow AEKF(x_{GPS}(k-1), y_{GPS}(k-1), \psi(k-1), v(k-1))$
  end if
  if $s == LiDAR$ **then**
    $\hat{X}_s(k) \leftarrow AEKF(x_{LiDAR}(k-1), y_{LiDAR}(k-1), \psi(k-1), v(k-1))$
  end if
  if $s == IMU$ **then**
    $\hat{X}_s(k) \leftarrow AEKF(v(k-1), \psi(k-1), a(k-1))$
  end if
  $r_s(k) = z_s(k) - \hat{X}_s(k)$
  Calculate data feature of sensor: $\Delta r_s(k) = r_s(k) - r_s(k-1)$
  $\mathcal{A} \leftarrow BDBSCAN(\Delta r_s(k))$
  if $\mathcal{A} == True$ **then**
    $S = s$
    # Perform data recovery
    $\hat{X}_{s,ekf}(k) \leftarrow EKF(x_{GPS}(k-1), y_{GPS}(k-1), x_{LiDAR}(k-1), y_{LiDAR}(k-1), \psi(k-1), v(k-1))$
    $z_a(k) \leftarrow Decision\ Tree(\hat{X}_s(k), \hat{X}_{s,ekf}(k))$
  end if
end for

C. Attack Defense

1) Attack identification

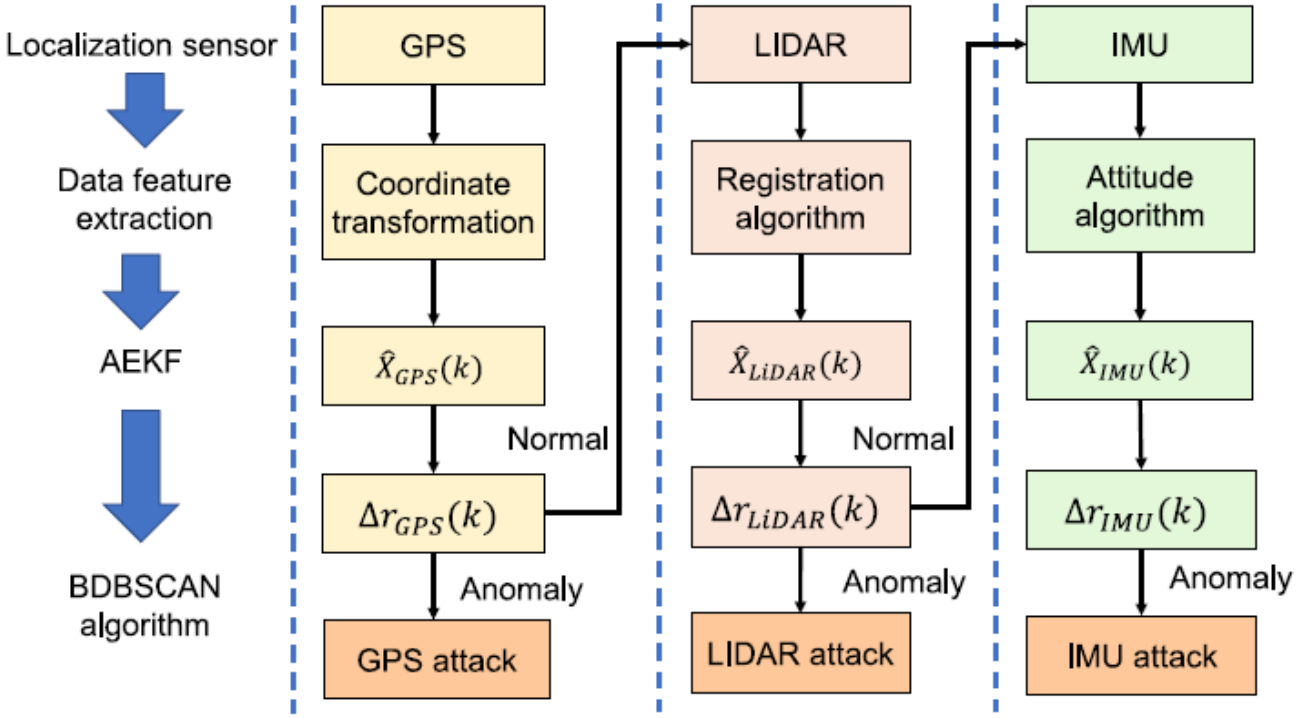


Fig. 6. The flowchart of attack identification.

- The importance of GPS, LiDAR, and the IMU diminishessequentially.
- If this sensor fails to detect anomalies, the next localizationsensor is used for attack identification.

C. Attack defense

2) Data recovery

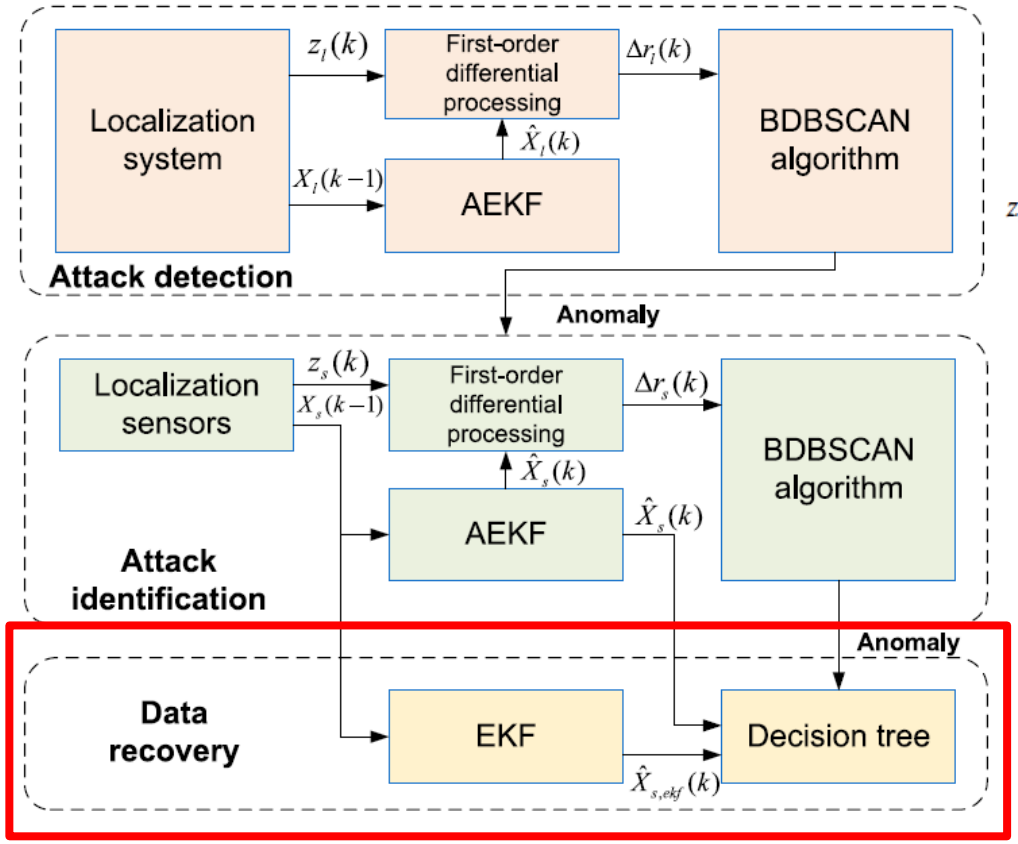


Fig. 4. Illustration of proposed framework.

$$label = \begin{cases} 1 & if \quad RMSE(\hat{X}_s(k), z_s(k)) < RMSE(\hat{X}_{s,ekf}(k), z_s(k)) \\ 0 & otherwise. \end{cases}$$

$$z_a(k) = \begin{cases} \hat{X}_s(k) & if\ DecisionTree(\hat{X}_s(k), \hat{X}_{s,ekf}(k), z_s(k-1)) = 1 \\ \hat{X}_{s,ekf}(k) & if\ DecisionTree(\hat{X}_s(k), \hat{X}_{s,ekf}(k), z_s(k-1)) = 0 \end{cases}$$

- replace attacked sensor data in real-time, which achievesreal-time data recovery.
- employs redundancy technology to developthe EKF used in localization sensors to obtain redundancy state estima-tion.

/03 Experimental Results

Experimental Results

A. Experiment setup

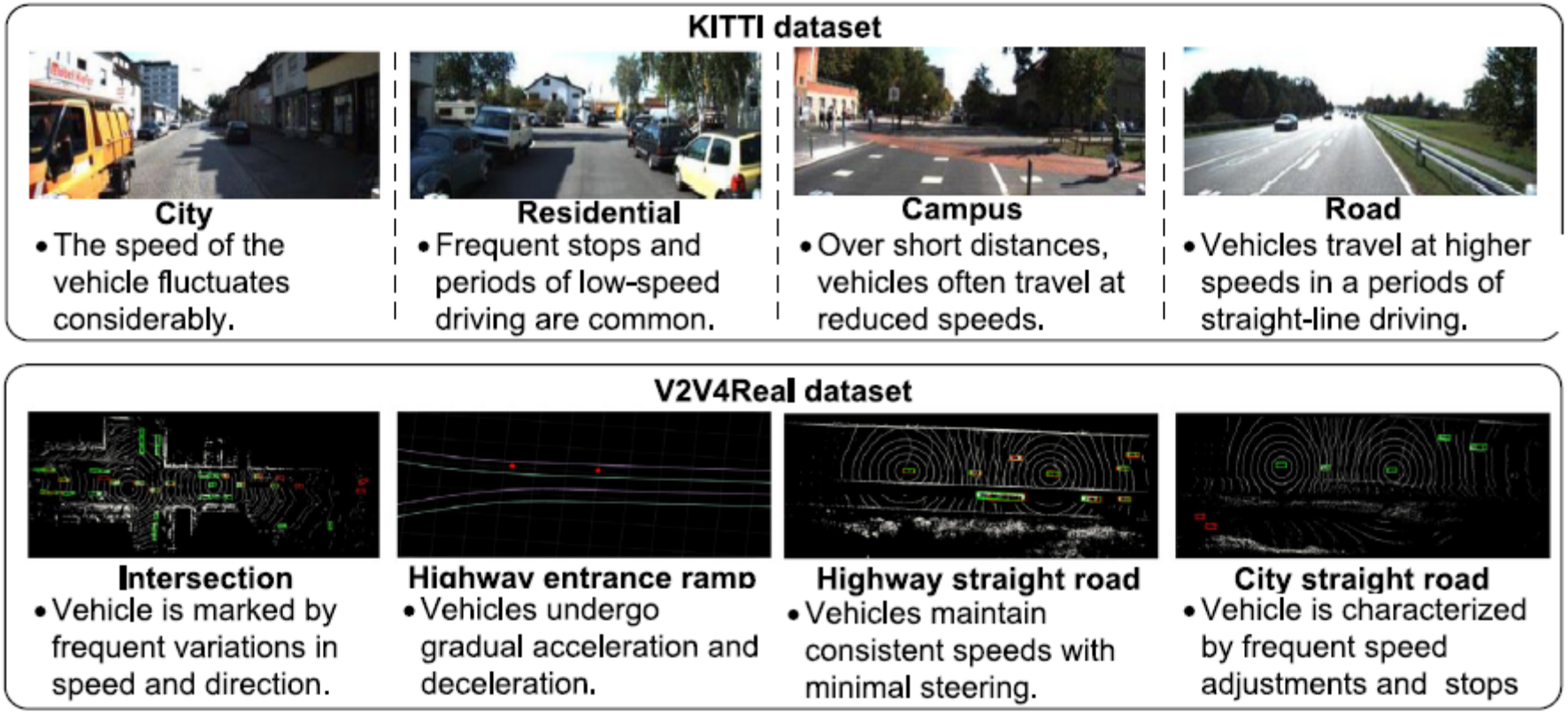


Fig. 7. Driving scenarios.

Experimental Results

A. Experiment setup

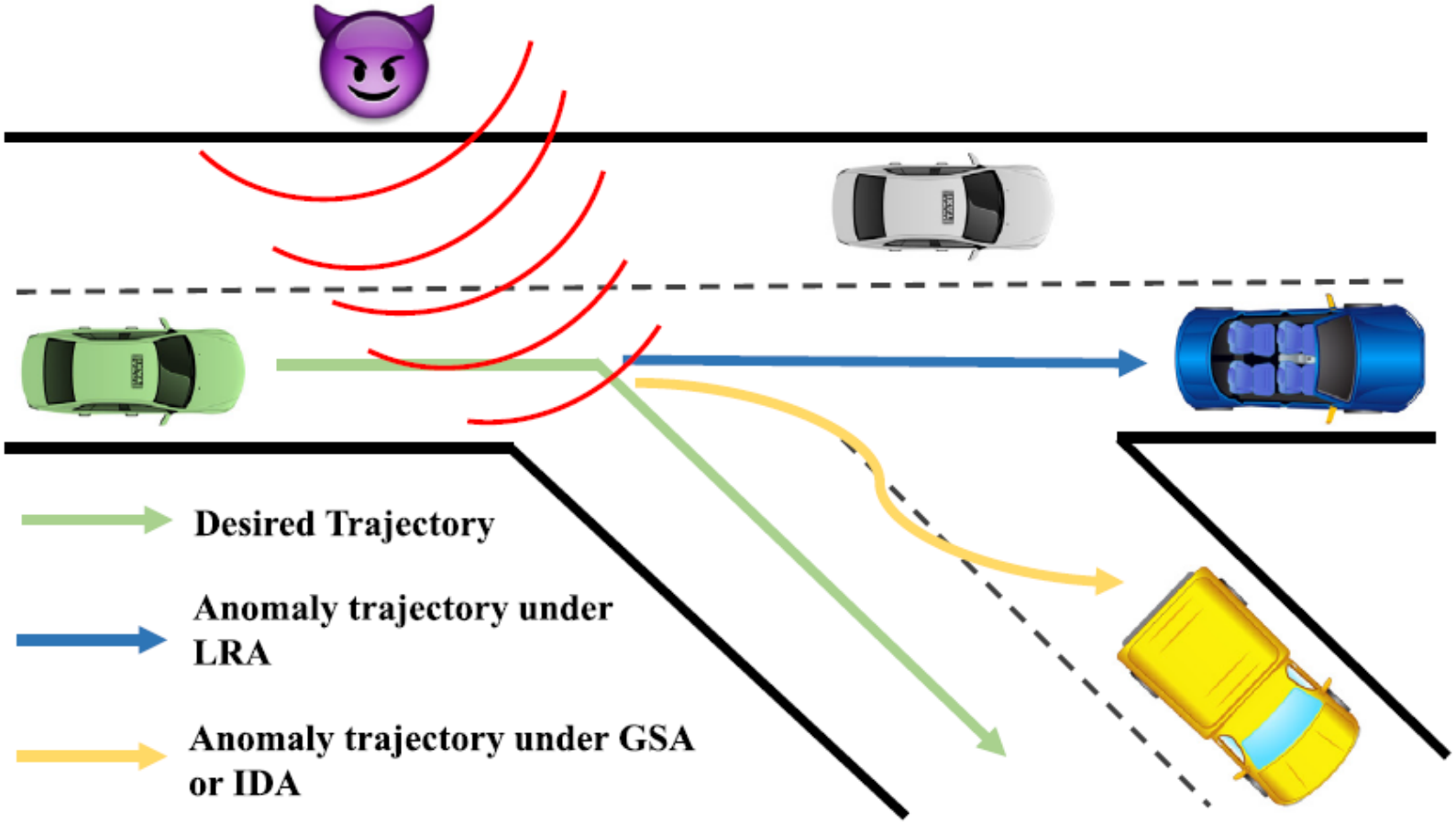


Fig. 3. Possible trajectory changes caused by sensor attacks.

Experimental Results

B. Experiment Result

1) Precision

Table 4

Performance of attack detection.

Metric	Proposed framework	ADCUSUM	SVMDA
ACC[a]	100.00% **(+4.44%)**	98.12%	95.56%
FAR[a]	0.00% **(−4.51%)**	1.88%	4.51%
ADT (ms)[a]	156.44 **(−21.91%)**	85.21	200.35
ACC[b]	100.00% **(+6.38%)**	93.62%	94.33%
FAR[b]	0.00% **(−6.24%)**	6.24%	5.13%
ADT (ms)[b]	155.79 **(−23.38%)**	86.25	203.33
Memory usage (MiB)	128.33 **(−8.42%)**	58.32	140.14
CPU consumption (μJ)	50 354.00	1665.00	312 011.00
DRAM consumption (μJ)	1587.00	826.00	17 517.00

[a] KITTI dataset include four driving scenarios.

[b] V2V4Real dataset include a hybrid driving scenarios.

- relies on data density detection rather than data distribution
- abnormal boundaries are adaptively adjusted rather than fixed.
- integrated with vehicle kinematic data

Experimental Results

B. Experiment Result

1) Precision

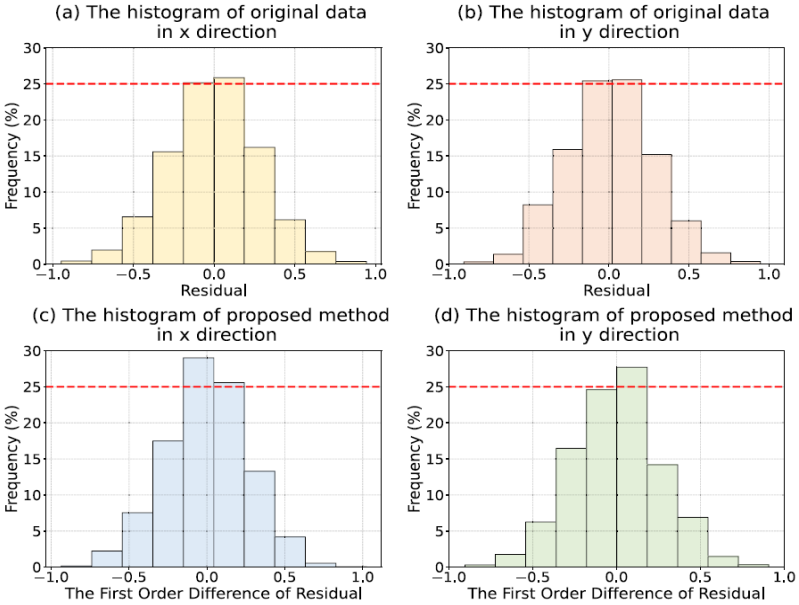


Fig. 13. Compare the original data and process data from the histogram in x and y direction.

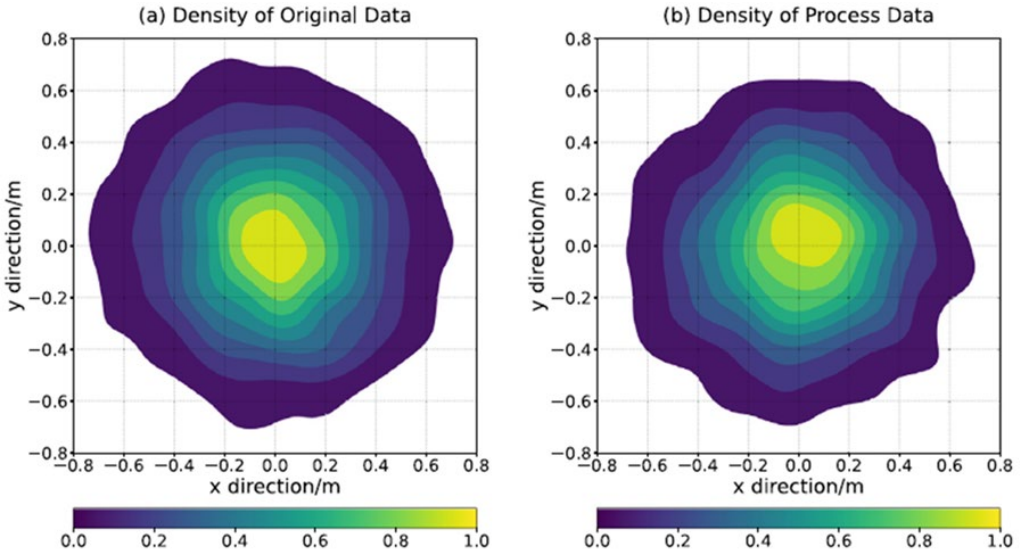


Fig. 12. Compare the original data and proposed from the density in x and y direction.

Table 5

The comparison results of original data and proposed framework in time series.

Metrics	Proposed framework	Original data
Range	7.1841(**−19.64%**)	8.9404
Standard deviation	0.2875(**−13.71%**)	0.3332

First-order difference can filter noise, improve the density of data, and highlight data changes.

Experimental Results

B. Experiment Result

2) Adaptive

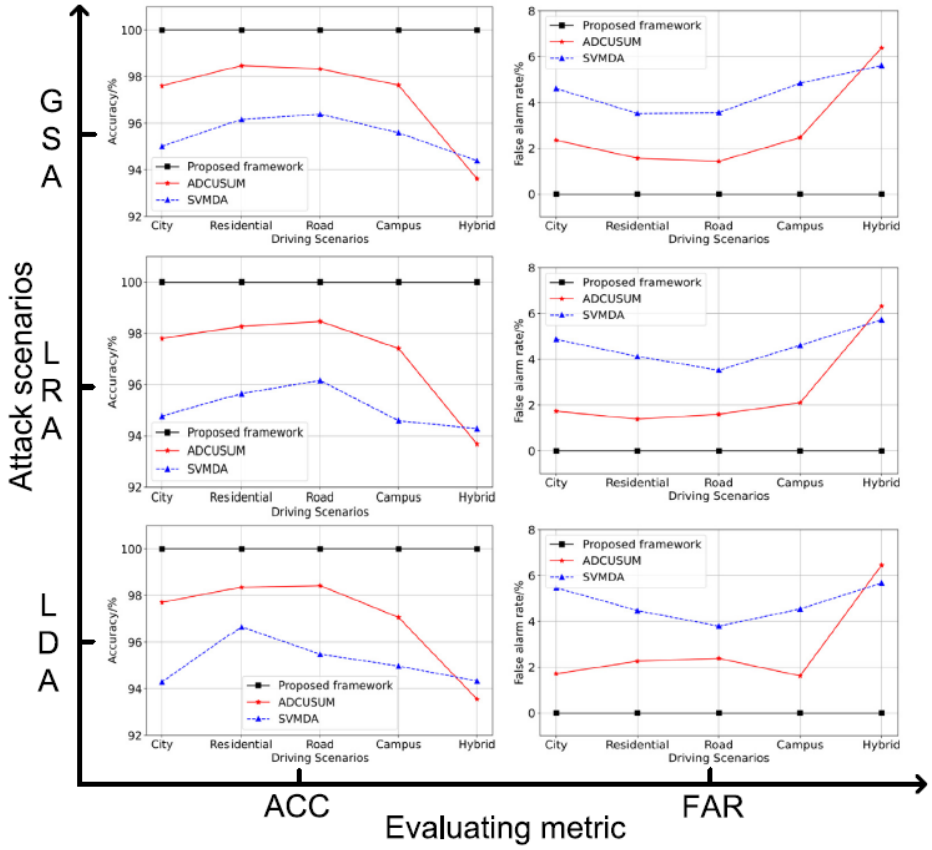


Fig. 11. Adaptive analysis in various driving and attack scenarios.

- achieves 100% accuracy in attack detection against three common sensor attacks in the KITTI and V2V4Real dataset, while ADCUSUM and SVMDA achieve 95.87% and 94.94% accuracy, respectively.
- applies data density for detection and employs the AEKF which can adeptly adjust to variations in noise.
- apply the multi-armed bandit algorithm based on reinforcement learning operates without the need for prior knowledge

Experimental Results

B. Experiment Result

3) Complexity and energy consumption analysis

Table 4

Performance of attack detection.

Metric	Proposed framework	ADCUSUM	SVMDA
ACC[a]	100.00% (+4.44%)	98.12%	95.56%
FAR[a]	0.00% (−4.51%)	1.88%	4.51%
ADT (ms)[a]	156.44 (−21.91%)	85.21	200.35
ACC[b]	100.00% (+6.38%)	93.62%	94.33%
FAR[b]	0.00% (−6.24%)	6.24%	5.13%
ADT (ms)[b]	155.79 (−23.38%)	86.25	203.33
Memory usage (MiB)	128.33 (−8.42%)	58.32	140.14
CPU consumption (μJ)	50 354.00	1665.00	312 011.00
DRAM consumption (μJ)	1587.00	826.00	17 517.00

[a] KITTI dataset include four driving scenarios.

[b] V2V4Real dataset include a hybrid driving scenarios.

Experimental Results

B. Experiment Result

3) Complexity and energy consumption analysis

Table 4
Performance of attack detection.

Metric	Proposed framework	ADCUSUM	SVMDA
ACC[a]	100.00% **(+4.44%)**	98.12%	95.56%
FAR[a]	0.00% **(−4.51%)**	1.88%	4.51%
ADT (ms)[a]	156.44 **(−21.91%)**	85.21	200.35
ACC[b]	100.00% **(+6.38%)**	93.62%	94.33%
FAR[b]	0.00% **(−6.24%)**	6.24%	5.13%
ADT (ms)[b]	155.79 **(−23.38%)**	86.25	203.33
Memory usage (MiB)	128.33 **(−8.42%)**	58.32	140.14
CPU consumption (μJ)	50 354.00	1665.00	312 011.00
DRAM consumption (μJ)	1587.00	826.00	17 517.00

[a] KITTI dataset include four driving scenarios.

[b] V2V4Real dataset include a hybrid driving scenarios.

- the performance of the proposed framework using anIntel Core i9 CPU (65 W capacity) and two Micron DDR4 3200MHz DRAM (9.72 W).
- consumes 0.3 W on CPU, lessthan 4.2 W for general autonomous driving applications and 0.1 W on DRAM.

/04 Conclusion

Conclusion

- This paper develops a novel real-time adaptive framework basedon density for sensor attack detection and defense in dynamic drivingscenarios on localization systems of autonomous vehicles. The defensemechanism includes attack identification and data recovery processes.
- Moreover, our framework demands only a minimal dataset of normaloperations and exhibits the novel BDBSCAN algorithm to adapt prioriparameters in dynamic driving scenarios.
- The precision, complexityand energy consumption and adaptability of the framework are eval-uated compared with two conventional methods on two real-worldautonomous vehicle datasets.
- In the future, the application of the proposed framework can befurther explored across various unmanned systems, including maritimeand aerial vehicles. In addition, we plan to deploy the framework onother recent datasets, such as nuScenes, PandaSet, and the WaymoOpen dataset, to evaluate and analyze the performance durabilityfrom multiple perspectives regarding real-world factors, including sen-sor performance degradation and changes in climate and weather.

Scifat
Conferences, Events and Awards

SF

COMPUTER

International Research Awards on Computer Vision

Thank you !