FedRD: Privacy-preserving Adaptive *Fed*erated Learning Framework for Intelligent Hazardous *R*oad *D*amage Detection and Warning

Yachao Yuan[a], Yali Yuan*[a], Thar Baker[b], Lutz Maria Kolbe[c] and Dieter Hogrefe[a]

[a]*Department of Computer Science, University of Göttingen, 37077 Göttingen, Germany*
[b]*Department of Computer Science, College of Computing and Informatics, University of Sharjah, 27272 Sharjah, United Arab Emirates*
[c]*Institute of Information Systems, University of Göttingen, 37073 Göttingen, Germany*

ARTICLE INFO

Keywords:
Edge-cloud computing, federated learning, differential privacy, hazardous road damage detection and warning, traffic safety, latency.

ABSTRACT

Road damages have caused numerous fatalities. Therefore, the study of road damage detection, especially hazardous road damage detection and warning, is critical in improving traffic safety. Existing road damage detection systems mainly process data on clouds. However, they are not able to warn users timely due to the long latency. Recent edge-computing techniques mitigate this problem while users can only receive warnings of hazardous road damages within a small area due to the limited communication range of edges. Besides, untrusted edges might misuse users' sensitive information. In this paper, we propose *FedRD*: a novel privacy-preserving edge-cloud and *Fed*erated learning-based framework for intelligent hazardous *R*oad *D*amage detection and warning. In FedRD, a new hazardous road damage detection model is developed leveraging the advantages of hierarchical feature fusion. A novel adaptive federated learning strategy is designed for robust model learning from different edges with limited and unequally-sized datasets. A new individualized differential privacy approach with pixelization is proposed to protect users' privacy before sharing data. Simulation results demonstrate that FedRD achieves a high detection performance and provides fast responses with accurate warning information covering a wider area while preserving users' privacy, even when some edges have limited data.

1. Introduction

Road transportation networks are indispensable social and economic components for all nations [7]. However, road systems are crumbling and sometimes even to a dangerous level due to aging, lacking periodic maintenance, and natural disasters [15]. Road damages, especially hazardous road damages (e.g., big holes, blowups, fractures, and pounding), highly increase the risk of road accidents and might cause serious injuries or fatalities [14]. For example, when a new big hole appears on a road while drivers are not aware of it, one vehicle may collide with another and injure the drivers or passengers when hitting or avoiding the big hole. As reported in [37, 14], road accidents caused by road damages lead to millions of injuries every year and cost around 1.5% to 3% of Gross Domestic Product (GDP) economic losses all over the world. Therefore, detecting hazardous road damages and warning drivers timely is critical for ensuring traffic safety.

However, due to colossal road network volume and cluttered real-world environments, it is challenging for drivers to obtain accurate road damage information (e.g., types, levels, and locations). Besides, current road condition monitoring is predominantly performed by certified inspectors, which is subjective, labor-intensive, costly, and time-consuming [24]. Expensive vehicles equipped with various sensors, high-definition cameras, and illumination devices are also utilized in countries like Germany and the UK. However, it is unaffordable for some developing countries and local road administrations. So far, some road damage detection systems [34, 27] have been proposed, but they can't warn drivers about hazardous road damages in advance. Few researches enable road damage warning [6] by using a central cloud server. In these systems, data is processed on a cloud, and warning messages are sent to drivers from the cloud. However, due to the long latency between the cloud and drivers, drivers may not receive the warnings timely, which might result in serious road accidents. Further, most existing systems are sensor-based [1, 12], so road damages can only be detected when vehicles hit them, which is dangerous and not suitable for accident-preventing.

The authors' previous work [41] provided an edge-cloud computing-based framework named EcRD for low-latency road damage detection and warning by deploying the detection model on edges. Unlike sensor-based systems [1, 12], EcRD detects hazardous road damages before vehicles hitting them, which is much safer. However, in EcRD, drivers can only receive warnings about road damages within a small area due to the limited communication range of edges. For instance, Road Side Units (RSUs) are usually used as edges in intelligent transportation systems. However, their communication range is only around 1000 meters. Vehicles covered by an RSU can only receive road damage information within this range. Moreover, the detection performance of edges is strictly constrained by the amount of data collected. Some edges may fail to detect road damages if they do not have enough data for training. Even if edges or devices can directly collect pre-trained models from the cloud, it requires

*This project is in part supported by the China Scholarship Council (Grant ID: 201706050095).
*Corresponding author
yali.yang001@gmail.com (Yali Yuan*)
ORCID(s):

direct data sharing from edges or devices to the cloud, which has a high privacy leakage risk. Also, the computation power and storage space of IoT Devices are limited. Despite the importance, there is *no* existing work that addresses these problems in this field. Furthermore, data collected from users' Devices at edges contain massive private information, for example, people's faces, locations, and license plate numbers. Only few researches [39, 3] considered the privacy problem for road condition inspection by using cryptographic techniques. However, they have a high computation cost for key generation, authentication, encryption, and decryption. Besides, *no* existing work considers the privacy problem inside image/video data in this field.

To tackle these issues, in this paper, we design *FedRD*: a novel privacy-preserving edge-cloud-based federated learning framework for intelligent hazardous road damage detection and warning. In FedRD, a new map construction approach is introduced. It provides drivers/users a hazardous road damage warning map, which has hundreds or even thousands of times wider coverage than EcRD [41]. Additionally, Federated Learning (FL) strategy [25] is utilized to collaboratively learn hazardous road damage information from decentralized edges without direct data sharing. It improves the model's robustness and protects people's privacy from untrusted cloud servers. Different from [16, 25, 29], the developed Adaptive Federated learning strategy (AFed) ensures high detection performance by only aggregating qualified models selected from top K models received from edges. In this way, high detection performance can be guaranteed within limited computation iterations. Also, it can prevent data poisoning attacks since the local models trained on poisoned data will not be used for global aggregation due to their low performance on the shared testing set on the cloud. Although FL protects privacy from untrusted clouds by keeping data locally on edges, there is still a high privacy leakage risk because FL does not protect privacy from untrusted edges. Additionally, private data can also be recovered only by shared parameter gradients of FL [13]. Hence, Differential Privacy (DP) technique [8] is utilized to fill in this gap. Unlike [9, 30, 38], the proposed Individualized Differential Privacy with Pixelization (IDPP) method preserves privacy on users' devices before uploading data to untrusted edges, which is more private. Moreover, it has $3/4$ less computation cost because noise is added to pixelized images instead of original images. To the best of our knowledge, we are the *first* to propose a privacy-preserving edge-cloud-based federated learning framework for smart hazardous road damage detection and warning addressing the problems of existing systems (e.g., long latency, small coverage, model robustness, and privacy). The main contributions of this paper are summarized as follows:

- A novel edge-cloud computing and Federated learning-based framework (FedRD) for intelligent hazardous Road Damage detection and warning is proposed. FedRD utilizes the advanced edge-cloud computing, federated learning, and differential privacy techniques for fast, accurate, cheap, and private hazardous damage detection and warning.
- An Advanced road Damage Detection model (ADD) for hazardous road damage detection is developed. ADD leverages high feature extraction advantages of deep learning models with hierarchical feature fusion. ADD enables fast, accurate, and robust road damage detection.
- A novel Adaptive Federated learning strategy (AFed) is designed which updates the learning models based on their detection performance and learning speed. It ensures more robust model learning with low communication rounds. Additionally, a new map construction method is introduced to provide road users a global warning map covering a wider area.
- A new privacy protection technique named Individualized Differential Privacy with Pixelization (IDPP) is introduced based on the advanced differential privacy technology. IDPP protects both users' sensitive information (e.g., ID and location) and the privacy inside images/video frames (e.g., people's faces and drivers' plate numbers) collected at users' devices before uploading to edges.
- Extensive evaluations are performed to prove that the proposed FedRD framework can achieve a high detection performance with low latency and provides accurate warning information covering a wider area while preserving privacy, even when some edges have limited data.

The remainder of this paper is organized as follows: Section 2 explains the design of the FedRD framework, including design goals, architectural components, components interactions, and algorithm design. Section 3 introduces adaptive federated learning and warning map construction. Section 4 elaborates individualized differential privacy with pixelization technique. Experimental setup, datasets, and baselines are illustrated in Section 5. The performance of the proposed approaches and the overall FedRD framework is evaluated in Section 6. Section 7 gives a detailed summary and comparison of the related literature. Finally, Section 8 concludes the paper.

2. System Design

This section introduces the design goals, architectural components, components interactions, and details of the proposed AbRS and ADD models of the FedRD framework.

2.1. Design Goals

The following goals drive the design of FedRD:

1. **Latency:** A hazardous road damage detection and warning system must warn users timely about dangerous road damages for accident prevention, which means the latency should be low.

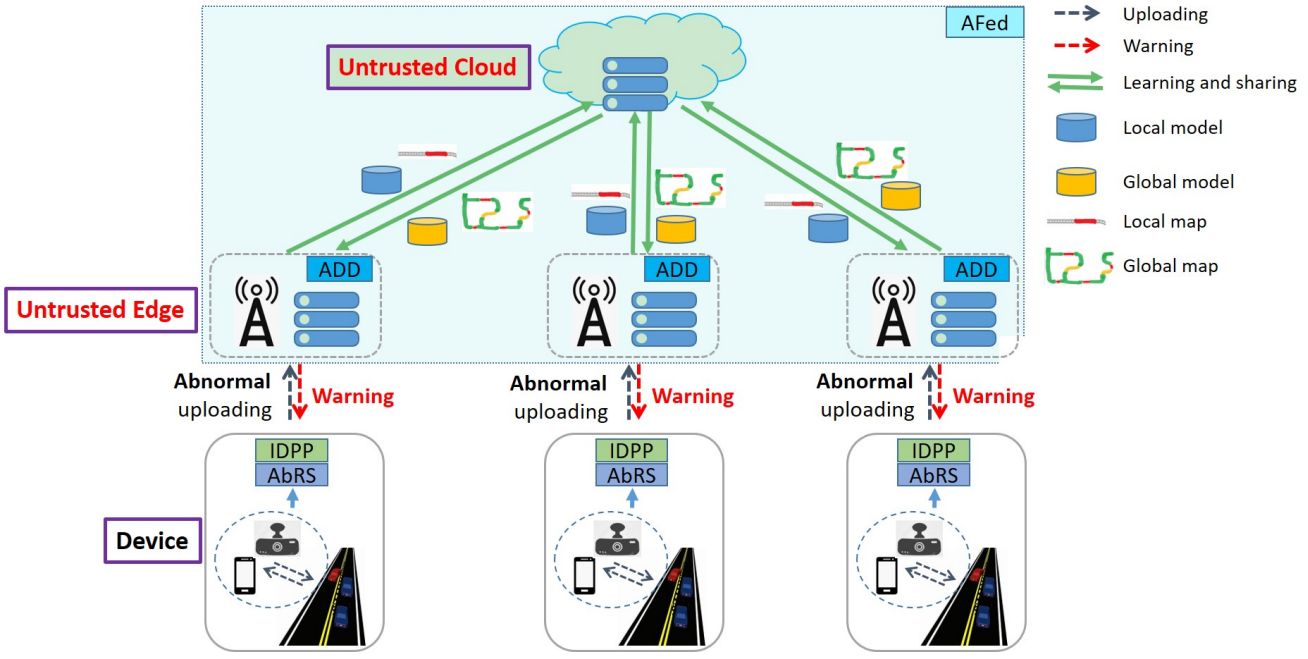


Figure 1: Overall framework of FedRD.

2. **Accuracy:** The proposed system should detect hazardous road damages accurately since miss-detected dangerous road damages are fatal for road users.
3. **Robustness:** The proposed system's performance should be robust to different environments, such as different weather conditions, various illuminations, and obstacles like vehicles and pedestrians. Besides, it should achieve high performance even when some edges only have limited data (which is common in the real world).
4. **Coverage:** The designed framework should provide users with hazardous road damage information with wide coverage for accident prevention and route planning.
5. **Cost:** Image/video analysis is a high resource-demanding task. The size of an image/video is usually large, and the amount of data quickly increases as time goes by. Also, the Internet bandwidth and data storage are expensive. Hence, the developed system requires low data transmission and data storage costs.
6. **Privacy:** There is a high privacy leakage risk from untrusted edges/clouds or during data transmission in open-access environments. The designed framework should protect the privacy of users, such as ID and location, and the privacy inside collected data, e.g., people's faces and license plate numbers.

2.2. Architectural Components

The proposed FedRD framework that satisfies the design goals is illustrated in Fig. 1. The components of this framework are described in detail as follows:

– Devices: This component gathers video data by pervasively used IoT devices (e.g., smartphones) mounted on vehicles. The Abnormal Road Screening module (AbRS) deployed on devices detects suspicious road damages. Then, the data (including users' information) is processed by Individualized Differential Privacy with Pixelization method (IDPP) to protect privacy. Finally, the processed data is sent to the nearest edge (e.g., Road Side Unit (RSU)).

– Edges: The Advanced road Damage Detection model (ADD) is deployed on edges for fast response. ADD detects hazardous road damages. The detection performance of the ADD models is further improved by the Adaptive Federated learning strategy (AFed). Once any dangerous road damages are detected, edges broadcast warning maps to covered users for accident-preventing. The warning maps contain the detected hazardous road damage information, for example, type (e.g., big holes, fractures, blowups, pounding), level (i.e., low, middle, and high), and location (i.e., GPS coordinates). They are then uploaded to the cloud for aggregation.

– Cloud: The cloud serves as an aggregator which aggregates selected ADD models from edges into one model and sends it back to all edges to improve the learning process of the ADD models on edges. Meanwhile, it integrates received warning maps from edges into one warning map and sends it back to the edges for both accident-preventing and route planning.

– AbRS: AbRS is deployed on devices. It is a light-weighted deep learning model and quickly detects suspicious abnormal roads from raw videos recorded by the devices. In this way, normal roads are successfully screened out, which significantly reduces data transmission cost between devices and edges. Then, only

suspicious abnormal road data is transmitted to edges.
- IDPP: The IDPP technique protects privacy on users' devices before sending data to edges, including both users' privacy (e.g., IDs and locations) and privacy inside collected images/videos (e.g., faces and license plate numbers). It is developed based on the advanced differential privacy technique. Besides, a pixelization approach is utilized in IDPP to reduce computation and communication costs.
- ADD: The ADD model functions as a hazardous road damage detector. It is a deep learning-based model with hierarchical feature fusion. It is deployed on edges to reduce latency. ADD enables fast and accurate hazardous road damage detection and warning.
- AFed: The AFed strategy further improves the detection performance of the ADD models on edges by using the cloud as a parameter server without requiring direct data sharing.

2.3. Component Interactions

The detection models deployed on edges and clouds are defined as local models and global models, respectively. Local models learn knowledge from data on edges, while global models assist the learning process of local models by aggregating the local models. The FedRD framework mainly consists of the following four phases that are repeated periodically for efficient hazardous road damage monitoring:

- **Phase 1:** Each vehicle collects road condition data by its carried smart IoT devices. The AbRS model on devices detects suspicious abnormal roads. The detected suspicious abnormal roads are transmitted to the nearest edge after protecting privacy by the IDPP technique.
- **Phase 2:** Each edge detects hazardous road damages by the ADD model based on data received from covered users. Once detected, it broadcasts the hazardous road damage warning information to all users within its communication range. The hazardous road damage warning information from an edge is also called a local map. Then, the edge sends the trained local model and the local map to the cloud.
- **Phase 3:** The cloud selectively aggregates received local models according to their performance to generate a global model. Meanwhile, the cloud integrates all collected local maps into a global map. Afterward, the cloud sends the global model and the global map back to the covered edges. Also, the global map is sent to road administration authorities for timely repair and maintenance.
- **Phase 4:** Edges update their local models with the received global model and broadcast the acquired global map to the covered users. With the global map, users are informed about road conditions in a broader area (e.g., a city) and can select optimal routes for traveling.

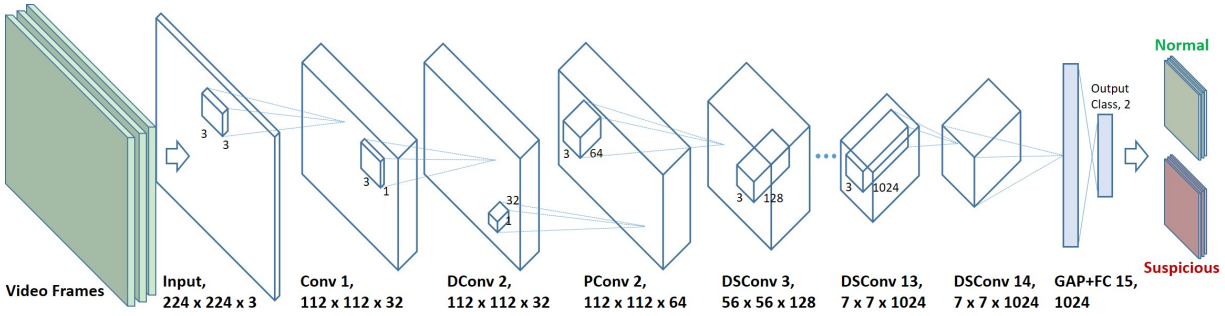
Table 1
Summary of notations.

Notation	Description
V_i	The i-th video
R, C	No. of rows and columns
M_i	The i-th image in $\{M_1, \cdots, M_N\}$
L	Prediction loss of AbRS
$L_{threshold}$	Threshold of prediction loss L
c_i	Different classes of hazardous road damages
l_1, l_2, l_3	Low, middle, and high level hazardous road damages
K	Top K received local models
Q	Top Q best local models
$\mathbf{w}_i^t$	Local model parameters on i-th edge at time t
$\mathbf{w}_t$	Global model parameters at time t on cloud
t	Period of time
D_i	No. of images on the i-th edge
D	Total no. of images on all edges
b	Optimal pixels differ from neighboring images
$1/\epsilon$	Level of privacy
$F(x)$	A random function
$X, \mu, 2b_2$	Random variable, its mean and variance
P	Pixelization function
θ_i	Value from Laplace distribution
δP	L1-sensitivity of the function P
S	No. of subsets
$M_{(r,c)}$	An image with r rows and c columns
l	Length of a square subset
E	No. of epochs.

2.4. Algorithm design

In this section, the proposed algorithms incorporating the Abnormal Road Screening model (AbRS) and the Advanced hazardous road Damage Detection model (ADD) are introduced in detail.

Abnormal road screening model (AbRS): Video transmission and analysis are high resource-demanding tasks. Directly transferring video data from devices to edges would cause network congestion and seriously affect other services. Fortunately, road condition videos recorded by smart IoT devices are primarily normal roads without dangers (around 80%). Hence, it is essential to detect abnormal roads first and only upload them to edges to minimize network communication burden and data processing and storage costs. Additionally, AbRS is built based on deep learning models since they are more robust for processing real-world data with cluttered backgrounds comparing to traditional machine learning methods combined with hand-craft features [20, 41]. Moreover, considering that the AbRS model is deployed on devices with limited computational power, we choose a lightweighted deep learning model structure (i.e., depthwise separable convolutions) to build our AbRS model similar to MobileNet [19]. As illustrated in Fig. 2, the AbRS model includes one Convolutional layer (Conv), one Depthwise Convolutional layer (DConv), one Pointwise Convolutional layer (PConv), twelve Depthwise Separable Convolutional layers (DSConv), a Global Average Pooling layer (GAP), one Fully-Connected layer (FC), and a softmax classification layer. Dif-



DConv: Depthwise Conv; Pconv: Pointwise Conv; DSConv: Depthwise Separable Conv;
GAP: Global Average Pooling; FC: Fully-Connected

Figure 2: Abnormal road screening model (AbRS).

ferent from standard convolution operations, a depthwise separable convolution is a form of factorized convolutions including a depthwise convolution and a 1×1 pointwise convolution [19], which makes the AbRS model more efficient. The AbRS model classifies each video frame as normal roads and suspicious abnormal roads and only upload suspicious abnormal roads to edges. In this way, the data transmission cost can be significantly reduced (by around 80%), and considerably fewer data need to be processed on edges and the cloud.

The training of the AbRS model requires a large labeled dataset, while abnormal roads are not easy to collect in some areas. Also, data samples are considerably different even within the abnormal road class, making the learning of AbRS even harder. In contrast, normal roads have a high homogeneity. In other words, data samples are similar within the normal road class. Therefore, we only train the AbRS model with normal roads, and the abnormal roads are recognized based on the prediction loss of the model. More specifically, since the model is well-trained with normal roads, it can recognize normal roads with considerably high confidence. If an image's prediction loss is lower than the threshold, then it is considered a normal road. On the contrary, if an image's prediction loss is higher than the threshold, it is classified as an abnormal road.

We define a video as V. The videos collected by a user are denoted as $\{V_1, V_2, \cdots, V_N\}$, where N is the number of videos collected by the user. Additionally, a frame with R row and C column pixels in video V is denoted as M_i. The pixel value at location (r, c) of an image M is defined as $I(r, c)$. As shown in Fig. 2, videos $\{V_1, V_2, \cdots, V_N\}$ are gathered by a road user with a smart IoT device. The AbRS model on the device pre-processes the videos to filter out normal roads. The output of the AbRS model is a prediction loss L, reflecting the probability of being a normal road. If $L > L_{threshold}$, then the input is an abnormal road; otherwise, it is a normal road.

Advanced hazardous road damage detection model (ADD): The Advanced hazardous road damage Detection model (ADD) is introduced to detect hazardous road damages and measure their severity levels. ADD is deployed on edges instead of the cloud for fast response. Besides, IoT devices are not directly employed for hazardous road damage detection due to their low computational power, and each device only has limited training samples. In contrast, edge servers have much higher computational power than IoT devices and much closer to users than the cloud. Once any hazardous road damages are detected by ADD on edges, warning messages are distributed to its covered users instantly, incorporating hazardous road damage types, levels, and locations.

According to [17], deeper models can significantly increase the classification performance but more challenging to train and has a higher computation cost. Fortunately, the deep residual learning structure proposed by [17] is easier to optimize and can achieve high accuracy from considerably increased depth. Hence, ADD is built based on deep residual learning structure, i.e., residual blocks with skip connections. Although the detection accuracy increases with the growth of the number of residual blocks, the data processing time is longer. As a trade-off, ADD only uses five residual blocks, as shown in Fig. 3. Each residual block contains a residual function performed by a shortcut connection and element-wise addition. An example of a residual block is displayed in Fig. 4.

Different residual blocks produce different levels of features. Deep layers produce high-level features, while shallow layers generate low-level features. Generally, only the last layer's feature is utilized for classification. However, low-level features also contain valuable information that can assist the final classification task. In ADD, multiple-level feature maps are extracted from different residual blocks. Extra layers are applied to the output of the four residual blocks (i.e., residual blocks 2, 3, 4, and 5) to fuse the feature maps, as illustrated in Fig. 3. More specifically, when an image feeds into the ADD, we can get one feature map from each residual block (except the first one). The feature maps are then processed by extra feature fusion layers to refine and resize them. The fusion of the four feature maps is used to classify the input image. Based on the classification result of ADD, the input image is further categorized into three dangerous levels according to visual severity. To

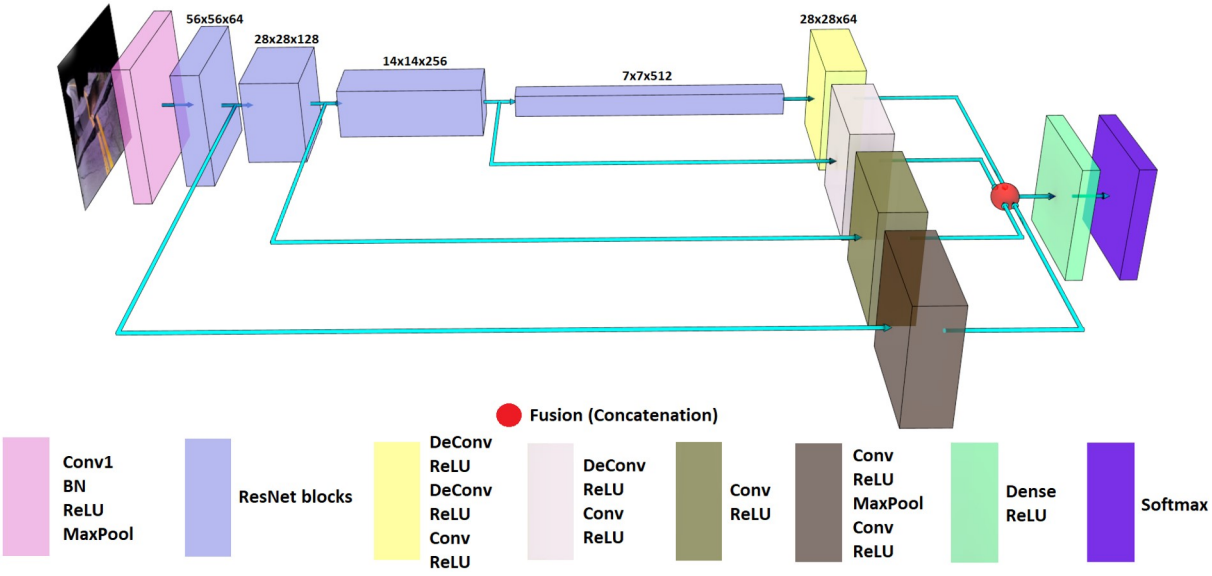


Figure 3: Advanced hazardous road damage detection model (ADD).

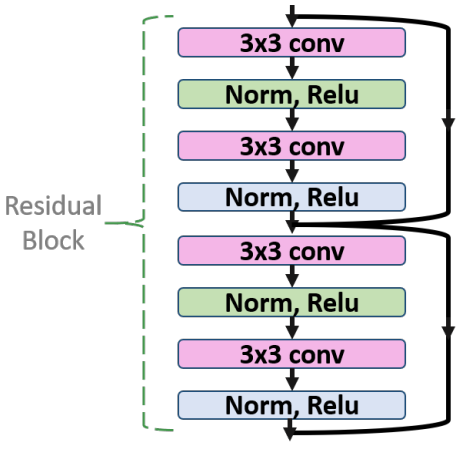


Figure 4: An example of a residual block.

be more specific, images without damages or only with minor damages (i.e., cracks and patches) are treated as low-level. Middle-level and high-level road damages are images with middle severity damages (i.e., potholes and fractures) and high severity damages (i.e., big holes and serious road blowups).

We define the input of ADD as $\{M_1, M_2, \cdots, M_N\}$, where M_i is the i-th image and N is the total number of input images. The input images are classified into six classes by ADD, denoted as $\{c_1, c_2, \cdots, c_N\}$, where, c_i represents different classes of road damages, i.e., cracks, patches, potholes, fractures, big holes, and blowups. Based on the classes of the damages, the output result is further categorized into three levels, i.e., $\{l_1, l_2, l_3\}$, where l_1, l_2, and l_3 represent low, middle, and high dangerous levels.

3. Adaptive Federated Learning (AFed) and Map Construction

Adaptive Federated learning (AFed): Adaptive Federated learning (AFed) is a decentralized adaptive federated learning strategy inspired by [25]. Unlike centralized methods that train a machine learning model with data on a central cloud server, decentralized approaches (e.g., Federated Learning (FL)) train a global model by collaboratively learning from multiple clients without direct data sharing. Different from existing FL strategies [25, 29, 16], AFed selective aggregates Q qualified models from top K local models collected from edges, which complies with the fact that the central server has no control over local clients. Also, by only aggregating high-quality local models, high performance can be ensured in limited communication rounds, and data poisoning attacks can be alleviated to some extent.

There are mainly two phases in AFed, i.e., local update and global aggregation. The local update phase updates the local models at edges periodically, while the global aggregation phase generates a global model by aggregating the selected local models. Local models are utilized for timely hazardous road damage detection and warning. The global model is used for updating local models' parameters to improve their performance.

We consider the AFed strategy with the setting of one cloud and N edges. Let D_i denote the number of data samples in the local database held by the edge i. The local training on the i-th edge aims to obtain a parameter set $\mathbf{w}_i^t$ of the local model at time t by minimizing the loss function. The goal of the cloud is to learn a global model $\mathbf{w}^t$ over data on the selected edges. Specifically, the cloud waits until top K local models are received. Out of them, the cloud aggregates

Q best local models by:

$$\mathbf{w}^t = \sum_{i=1}^{Q} \frac{D_i}{D} \mathbf{w}_i^t, \tag{1}$$

where D_i is the number of samples on the i-th edge, while D is the total number of samples over the selected Q edges. Also, $D = \sum_{i=1}^{Q} D_i$.

The training process of the AFed strategy involves the following four steps. Firstly, edges locally train models on their datasets to obtain the optimal model parameters $\{\mathbf{w}_1^t, \cdots, \mathbf{w}_i^t, \cdots, \mathbf{w}_N^t\}$ at time t. Secondly, the edges send the locally trained parameters to the cloud. Thirdly, the cloud selects Q qualified local models from top K received models. It stops receiving local models once K local models are collected. The Q qualified local models are selected by comparing the detection performance between the K collected local models. Only the local models with top Q highest detection performance will participate in the aggregation. The detection performance of both local models and the global model is evaluated by a shared testing set on the cloud. Then, the cloud aggregates the parameters of the Q local models from the edges by Eq. (1). The aggregated global model's parameters $\mathbf{w}^t$ are sent back to edges if its performance is better than the edges' local models. Finally, edges update their local models with the global model. More details of the method are presented in Algorithm 1, where $F_{select}(*)$ means the selection of top Q local models; $P(*)$ indicates performance; η is the learning rate, $\bigtriangledown$ is the derivative, $l(*)$ is the loss function, $\mathbf{w}$ and b are the weights and biases; E is the number of epochs.

Algorithm 1: AFed strategy

```
Input  : Local databases from edges and detection
         model ADD.
Output: Optimal local models on edges
for each time period t do
    cloud initialize w^0;
    for every E epochs do
        for each edge i = 1, 2, ..., N do
            w_i^t ← localUpdate(t, w_t^i)
        end
        cloud wait until receive K local models
        w^t ← Σ_{i=1}^{Q} (D_i / D) F_select(w_i^t)
    end
    send w^t to edges if P(w^t) > P(w_i^t)
    for every E epochs do
        update w_i^t by w^t if receive
        w_i^t ← w_i^t − η ▽ l(w_i^t, b_i^t)
    end
    send w_i^t to the cloud
end
```

Map construction: In FedRD, the local and global maps are generated by the local and global models. Based on the collected road condition information at edges, the local model detects dangerous roads and classifies them into three severity levels: low, middle, and high. Then, the local maps, including the types, levels, and locations, are constructed on edges. Each edge provides a fast warning by broadcasting its local map to its neighboring users. After that, each edge sends its local map to the cloud. The global map is created on the cloud by aggregating all local maps. The cloud broadcasts the global map to all users. Then, users can obtain the latest road conditions in a large area (e.g., a whole city), which helps users select optimal routes for traveling and significantly reduces the road accidents caused by hazardous road damages.

4. Individualized Differential Privacy with Pixelization

As mentioned in Section 1, although AFed protects privacy from untrusted clouds, there is still a high privacy leakage risk from untrusted edges. Sending data from users' devices to untrusted edges also poses great threats to users' data (e.g., location and ID) and the collected data (e.g., people's faces and license plate numbers). To protect privacy, users must sanitize all data before sending it to edges. Therefore, we introduce a new privacy-preserving technique named Individualized Differential Privacy Pixelization (IDPP) to fulfill this requirement. IDPP is built based on the powerful differential privacy approach [21]. Unlike [9, 21], IDPP preserves privacy at users' devices before uploading to untrusted edges. Also, images are pixelized before applying DP to reduce the computation cost of devices and the communication cost from devices to edges. Similar to [8], the Laplace mechanism is employed in IDPP.

4.1. Preliminaries

Following [9], we define neighboring images as follows:

Definition 1. *Let two images be M_1 and M_2. We can say that M_1 and M_2 are neighboring images if they have the same dimension and differ by b pixels.*

According to Definition 1, images' sensitive information, such as faces and license plate numbers, can be protected by up to b pixels difference. We can select the optimal b value to customize different levels of privacy according to the requirements of users and the trade-off between detection performance and privacy.

In the following content, the necessary preliminaries related to IDPP are illustrated. Then, we prove that IDPP achieves the ϵ-differential privacy. Following the concept of the differential privacy mechanism from Dwork et al. [8], ϵ-differential privacy is defined as follows:

Definition 2. *(ϵ-differential privacy): A random function F is said to be a ϵ-differential privacy (ϵ-dp) function, if for two different inputs $x, x' \in Dom(F)$ and one output be $z \in Range(F)$, we have:*

$$P(F(x) = z) \leq exp(\epsilon) P(F(x') = z). \tag{2}$$

Remark 1. *The parameter ϵ denotes privacy level. The higher the ϵ value, the more privacy leakage. Hence, ϵ is utilized to measure the trade-off between privacy leakage and detection performance.*

To accomplish the ϵ-differential privacy, the Laplace mechanism is often employed to add noises to the original data, where the noises are generated from the Laplace distribution from Definition 3.

Definition 3. *Laplace distribution: A random variable X follows the Laplace distribution if its probability density function is,*

$$Lap(x|b) = \frac{1}{2b} exp(-\frac{|x-\mu|}{b}), \tag{3}$$

where the localization parameter is μ and scale parameter is b. Furthermore, the mean of the random variable X is μ and the variance of X is $2b^2$.

Remark 2. *If X follows a Laplace distribution with localization parameter μ and scale parameter b, then we write $X \sim Lap(\mu, b)$.*

The concept of the Laplace mechanism is given in Definition 4.

Definition 4. *Laplace mechanism: Given a function $P : R^n \rightarrow R^p$, we define the Laplace mechanism F as,*

$$F(x, P(.), \epsilon) = P(x) + (\theta_1, \cdots, \theta_p), \tag{4}$$

where $\theta_i \sim Lap(0, \frac{\Delta P}{\epsilon})$. The ΔP is the $L1$-sensitivity of the function P, which is illustrated in Definition 5.

Definition 5. *$L1$-sensitivity: The $L1$-sensitivity of a function $P : R^n \rightarrow R$ is defined as:*

$$\Delta P = sup_{x,y \in A} \|P(x) - P(y)\|_1, \tag{5}$$

where $\|.\|_1$ is the L1 norm.

Remark 3. *The sensitivity shows how much the function P can be changed by adding random noise while still preserving privacy.*

4.2. Pixelization

An image is represented as a matrix M, and each pixel value of the image is denoted as $M_{i,j}$, where $i = 1, 2, \cdots, R$, $j = 1, 2, \cdots, C$, and $0 \leq M_{i,j} \leq 255$. The pixelization technique takes blocks/subsets of the image matrix as input. Every element $M_{i,j}$ that belongs to that block is replaced by the average value of that block. In this paper, we take a square subset of length l. Thus, for a matrix with dimension $R \times C$, the total number of square subsets is $S = \lceil \frac{R}{l} \rceil \times \lceil \frac{C}{l} \rceil$.

To include pixelization technique to the differential privacy concept, we define the pixelization global sensitivity in Lemma 1.

Lemma 1. *The $L1$ sensitivity of the pixelization technique is $\Delta P_l = \frac{256b}{l^2}$.*

Proof. By Definition 1, we have two neighboring images M_1 and M_2. These images differ by at most b pixels. Since each pixel ranges from 0-255, we have,

$$sup_{M_1, M_2} |M_1 - M_2| \leq 256b. \tag{6}$$

Now, the pixelization here takes a square of length l. Thus in each square, we have l^2 pixels. Therefore, for the entire image, the global sensitivity is:

$$sup_{M_1, M_2} \|P_l(M_1) - P_l(M_2)\|_1 = \Delta P_l = \frac{256b}{l^2}. \tag{7}$$

□

4.3. IDPP

Algorithm 2 illustrates the procedure of IDPP. In particular, S is the number of pixel subsets. The random variable θ_i of the i-th subset is generated following the Laplace distribution. After that, we add this random noise to the average of each subset. The proof of IDPP is illustrated in Theorem 2.

Algorithm 2: IDPP

Input : image $M_{(r,c)}$ and R, C, l, b, ϵ
Output: image $M_{(r,c)}$ with privacy
Initialize with $S = \lceil \frac{R}{l} \rceil \times \lceil \frac{C}{l} \rceil$, $\Delta P_l = \frac{256b}{l^2}$
for $i = 1, 2, \cdots, S$ **do**
  $\theta_i \sim Lap(0, \frac{\Delta P_l}{\epsilon})$
  $P_l(M_{(r,c)}) = \sum_{(r,c) \in g_i} \frac{M_{(r,c)}}{l^2}$
  $\widetilde{P}_l(M_{(r,c)}) = P_l(M_{(r,c)}) + \theta_i$
  Return $\widetilde{P}_l(M_{(r,c)})$
end

Theorem 2. *The $\tilde{P}_l$ in Algorithm 2 satisfies ϵ-differential privacy.*

Proof. Let two neighboring frames be M_1, M_2, another independent frame be M, and a random variable be $\theta_i \sim Lap(0, \frac{\Delta P_l}{\epsilon})$, where $i = 1, \cdots, K$.

$$\begin{aligned} \frac{P(\widetilde{P}_l(M_1) = M)}{P(\widetilde{P}_l(M_2) = M)} &= \prod_{i=1}^{K} \frac{exp(-\frac{\epsilon |P_l(M_{i,1}) - M_i|}{\Delta P_l})}{exp(-\frac{\epsilon |P_l(M_{i,2}) - M_i|}{\Delta P_l})}, \\ &= \prod_{i=1}^{K} exp(\frac{\epsilon(|P_l(M_{i,2}) - M_i| - |P_l(M_{i,1})_i - M_i|)}{\Delta P_l}). \end{aligned} \tag{8}$$

Applying the triangular inequality, and from Eq. (8) we obtain,

$$\leq \prod_{i=1}^{K} exp(\frac{\epsilon |P_l(M_{i,1}) - P_l(M_{i,2})|}{\Delta P_l}). \tag{9}$$

By $L1$ norm and $L1$ sensitivity in Definition 5, from Eq. (9), we have,

$$= exp(\frac{\epsilon \| P_l(M_1) - P_l(M_2) \|_1}{\Delta P_l}),$$
$$\leq exp(\epsilon).$$

Thus,

$$\frac{P(\widetilde{P}_l(M_1) = M)}{P(\widetilde{P}_l(M_2) = M)} \leq exp(\epsilon). \quad (10)$$

Finally, the proof of Theorem 2 is concluded as below,

$$P(\widetilde{P}_l(M_1) = M) \leq exp(\epsilon) P(\widetilde{P}_l(M_2) = M). \quad (11)$$

□

5. Experimental setup and dataset

In this section, the experimental setup and the dataset are introduced first. Then, we describe the comparison baselines. Following this, experimental results are presented.

5.1. Experimental setup

A cheap smartphone (VG30+) is used as an example of IoT devices for data acquisition. A laptop (Dell Latitude 5880, 64-bit Windows 10 Operating system, 16 G RAM, Intel Core i7 i7-7820HQ CPU with 2.9GHz) is used as our edge server to simulate an RSU. A high-performance server (Ubuntu 16.04 LTS system, 125.8 GB of RAM, 5.93 TB of hard disk, and 8 GTX 1080 Ti GPUs) is used as our cloud server. The client application is implemented using Python 3.6. For fair comparison and good performance, we set the learning rate as 0.001, the total number of epochs as 1000, the number of epochs for local update E as 30, batch size as 64, momentum as 0.9, and weight-decay as 0.0001. To simulate the model training with limited and unequally-sized datasets on different edges, we split the training set into three subsets, i.e., 80%, 60%, and 40%, for edge1, edge2, and edge3.

5.2. Datasets

To effectively evaluate the performance of FedRD, more than 75 road videos (more than 49 minutes of each) with the resolution of 1280×720 pixels from drivers' front viewpoint are collected by the IoT device. We build our training set by automatically extracting frames from the collected videos with an interval of 0.5 seconds. The training set of the AbRS model contains 1560 normal road images. Fig. 5 presents some examples of normal roads and abnormal roads. The training set of the ADD model includes 300 abnormal road images predicted by AbRS and 600 hazardous road damage images collected from the Internet. We augment the dataset to 2500 by rotation, skew, crop, adding noise, and padding. Some examples of different levels (i.e., low, middle, and high) of hazardous road damages are shown in Fig. 6. Each edge has its local dataset on which it trains its local model. The cloud has a global dataset and global model. Local models and the global model share the same architecture. Further, the cloud has a small test dataset (with 60 images, 20 images per dangerous level) to decide the local models involved for global aggregation. For testing, we build a video including 5000 frames with 1080 abnormal roads and 3920 normal roads.

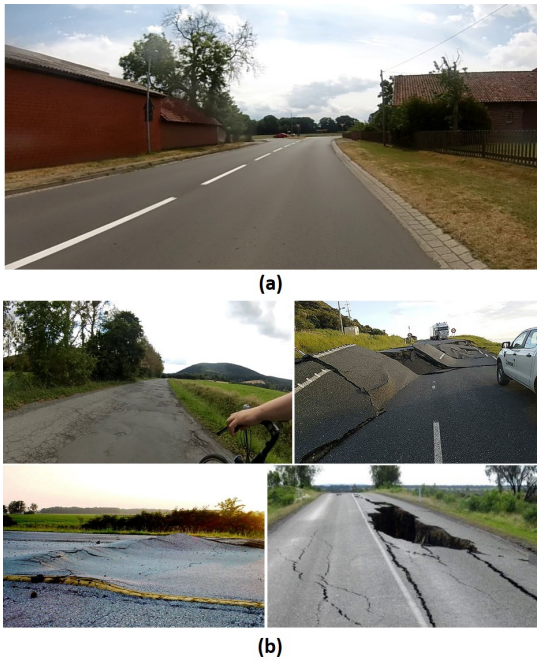


Figure 5: Image examples of (a) Normal road, (b) Abnormal road.

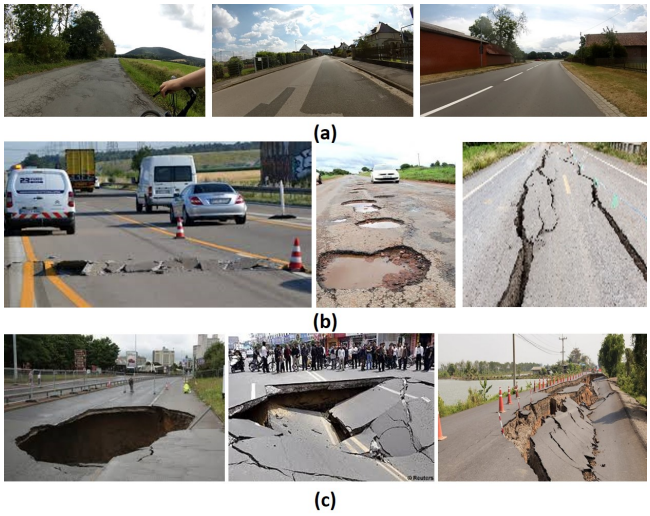


Figure 6: Image examples of abnormal roads in (a) Low level, (b) Middle level, (c) High level.

5.3. Baselines

To evaluate the proposed ADD, AFed, and FedRD, we compare their performance with the following baselines.

Baselines for the evaluation of ADD:

1) cloudRD [10]: It is a method deploying the detec-

tion model on the cloud. The detection model mainly contains seven convolutional + batch normalization + ReLU layers, six max-pooling layers, one fully-connected layer, and a softmax layer.

2) edgeRD [22]: It deploys the detection model on edges instead of the cloud for fast response. The detection model is a CNN-based model with four convolutional + max-pooling layers, one global average pooling layer, two fully-connected layers, and a softmax layer.
3) ResNet [17]: ResNet is a CNN-based model using five residual blocks, one average pooling layer, one fully connected layer, and a softmax layer. Each residual block has an extra shortcut connection and element-wise addition.

Baselines for the evaluation of AFed:

1) No-AFed: it represents that the detection model (ADD) learns without the AFed strategy.
2) AFed: it denotes that ADD learns with the AFed strategy.

Baselines for the evaluation of FedRD:

1) FedRD-no-IDPP: It is a variant of FedRD, without our IDPP technique. By comparing with FedRD (with IDPP), we will know the influence of IDPP on the performance of FedRD and the best trade-off.
2) cloudRD [10] and edgeRD [22]: They are variants of FedRD. edgeRD deploys the hazardous road damage detection task on edges while cloudRD deploys it on the cloud. Additionally, both of them are not using federated learning strategies. The performances of both cloudRD and edgeRD are compared to FedRD to evaluate the effectiveness of our edge-cloud-based Federated learning framework.
3) EcRD [41]: Similar to FedRD, EcRD is also an edge-cloud-based hazardous road damage detection framework that detects hazardous road damages on edges for fast response. However, it uses a different detection model (i.e., HDD) and not using any federated learning strategy. By comparing with EcRD, the performance of the proposed ADD and the necessity of using the federated learning strategy in FedRD can be proved.

6. Experiments and Evaluation

In this section, the performance of the FedRD framework and the proposed methods (i.e., AbRS, ADD, AFed, and IDPP) are evaluated by the evaluation metrics defined in [41], including Precision, Recall, Accuracy, F1-score, runtime, and latency.

6.1. AbRS and ADD Results and Evaluation

The experimental results of the AbRS model are illustrated in Table 2. As shown in the table, the F1-score of AbRS is 98.79%, which is 18.11%, 2.87%, and 1% times higher than AlexNet, GoogleNet, and VGG16, respectively. The result shows that although AbRS is more lightweight than AlexNet, GoogleNet, and VGG16, it achieves better abnormal road detection results. The reason behind this is that abnormal roads, especially for those with hazardous road damages, are vastly different from normal roads, as shown in Fig. 5, making the separation of abnormal roads and normal roads simpler. Moreover, the runtime of AbRS is 89.21%, 92.99%, and 90.93% times lower than AlexNet, GoogleNet, and VGG16, respectively. Likewise, AbRS has the lowest runtime because it has a small model size for using deep separable convolutions instead of normal convolutions. Given the high performance of the AbRS model, the amount of data that needs to be transmitted from devices to edges is significantly reduced by using the AbRS model. Despite the high F1-score, there are still some miss-detected abnormal roads. Some results of AbRS are illustrated in Fig. 7. The results show that AbRS may fail to detect abnormal road frames when the frames are very bright, very dark, with shadows, or very light damages. The experimental results of the AbRS model are illustrated in Table 3. As shown in the table, ADD achieves a 92.52% F1-score, which is 26.34%, 15.51%, and 9.53% times more accurate than cloudRD, edgeRD, and ResNet. ADD achieves worse runtime than the baselines because it contains extra feature fusion layers to capture valuable multi-scale features from different layers. However, ADD enables accurate and relatively fast hazardous road damage detection on edges. Although it achieves a relatively high F1-score, some hazardous road damages are wrongly classified due to illuminations and shadows. Also, if the hazardous road damages are not small for some classes, for example, minor blowups and small holes in Fig. 8(b), Fig. 8(d), and Fig. 8(f), they tend to be miss-classified as lower-level hazardous road damages, which is reasonable in practice. Fig. 8 illustrates some good and bad results of different hazardous road damage levels predicted by ADD.

Table 2
AbRS evaluation results.

Model	Accuracy (%)	F1-score (%)	runtime (s)
AlexNet [26]	82.20	83.64	0.063
GoogleNet [33]	94.80	96.03	0.097
VGG16 [31]	96.50	97.81	0.075
AbRS (Ours)	**97.38**	**98.79**	**0.0068**

Table 3
ADD evaluation results.

Model	Accuracy (%)	F1-score (%)	runtime (s)
cloudRD [10]	75.54	73.23	0.038
edgeRD [22]	81.21	80.10	0.054
ResNet [17]	86.35	84.47	**0.032**
ADD (Ours)	**93.48**	**92.52**	0.085

Figure 7: Experimental results of the AbRS model. (a) Good results of normal roads, (b) Bad results of normal roads, (c) Good results of abnormal roads, (d) Bad results of abnormal roads.

Figure 8: Experimental results of the ADD model. (a) Good results of low-level road damages, (b) Bad results of low-level road damages, (c) Good results of middle-level road damages, (d) Bad results of middle-level road damages, (e) Good results of high-level road damages, (f) Bad results of high-level road damages.

6.2. Adaptive Federated Learning Results and Evaluation

Experiments with three edges and one cloud are conducted to evaluate the performance of the proposed AFed strategy. In the experiments, local models on edges update their weights every 30 iterations. The goal is to improve the detection performance of local models on edges without directly sharing data.The evaluation results of the proposed AFed strategy are presented in Table 4. This table shows that the F1-score of edges after applying AFed improves by maximally 6.95% than without AFed. Similarly, the accuracy of edges after using AFed increases by maximally 3.66% than that without AFed. The cloud waits until the top two local models are received, complying with the fact that the cloud has no control over edges.

Table 4
AFed evaluation results.

Setting	Edge	Accuracy (%)	F1-score (%)
No-AFed	edge1	89.82	84.57
	edge2	88.35	87.26
	edge3	89.39	88.41
AFed	edge1	93.48	91.52
	edge2	90.68	89.76
	edge3	92.94	91.37

6.3. IDPP Results and Evaluation

The performance of the FedRD before and after applying IDPP is given in Table 5. The table shows that only 1.08% F1-score and 0.7% accuracy are reduced after using IDPP, while data privacy can be well-protected. Also, the computation time of FedRD is only increased by 0.0008 s after

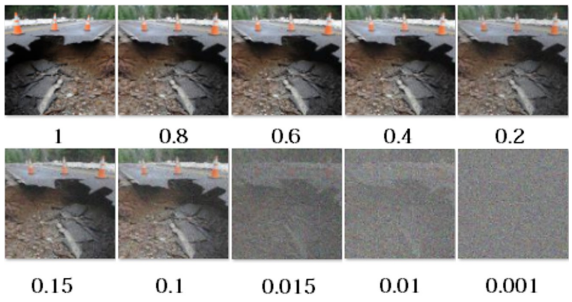


Figure 9: Effect of ϵ on images.

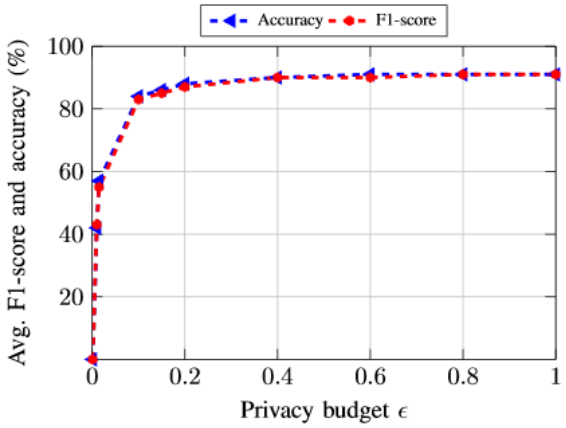


Figure 10: F1-score and accuracy of FedRD with different privacy budget ϵ.

applying IDPP. Fig. 10 illustrates the detection performance

Table 5
FedRD and FedRD-no-IDPP comparison results with $\epsilon = 0.4$.

Model	Accuracy (%)	F1-score (%)	Latency (s)
FedRD-no-IDPP	91.53	91.40	0.0318
FedRD	90.83	90.32	0.0326

of the ADD model after applying AFed with different privacy budgets ϵ, which measures the amount of noise added to the original data. Typically, the higher is the ϵ, the more noise is added, and the more private the data becomes. As shown in the figure, the detection accuracy raises with the increase of the ϵ. The detection accuracy of FedRD with $\epsilon = 0.1$ is 47.37% higher compared with $\epsilon = 0.015$. Also, the detection accuracy of FedRD with $\epsilon = 0.4$ improves by 6.00% compared to that of $\epsilon = 0.1$. However, the performance of FedRD with $\epsilon = 0.4$ is close with $\epsilon = 0.8$ and $\epsilon = 1.0$. Moreover, the F1-score of FedRD has a similar trend with the accuracy of FedRD. Fig. 9 shows the effect of ϵ in IDPP on images. It shows that the more noise added to the image, the more private is the image content. However, if too much noise is added, e.g., when $\epsilon = 0.001$, no valuable information can be observed, including hazardous road damages without any sensitive information. Therefore, we select $\epsilon = 0.4$ as a good trade-off for high detection performance and good privacy-preserving.

6.4. FedRD Overall Performance Evaluation

The performance of FedRD framework is compared with cloudRD [10], edgeRD [22], and EcRD [41] to evaluate its effectiveness. The comparison results are illustrated in Table 6. Overall, the results clearly show that FedRD outperforms its variant and other baselines regarding the accuracy, F1-score, latency, coverage range (i.e., local or global), and privacy risk. Specifically, the detection accuracy achieved by FedRD is as high as 90.83% which is good considering the small road damage dataset as well as high inter-class divergence of the road damages as illustrated in Fig. 6. In addition, compared to the baselines, FedRD has the lowest privacy leakage risk for the usage of AFed and IDPP. The cloudRD has a very high privacy leakage risk because clouds are usually highly untrusted. The edgeRD and cloudRD also have a high privacy leakage risk due to untrusted edges and data transmission between devices and edges in open-access networks. Concerning latency, cloudRD has around 45 times higher latency than FedRD and edgeRD for hazardous road damage detection tasks. With such a high latency, users may not be able to receive life-threatening warning information in time. Therefore, the hazardous road damage detection and warning task should be deployed on edges to ensure high quality of service (QoS). Moreover, existing edge-based frameworks, such as edgeRD [22] and EcRD [41], can only broadcast hazardous road damage warning information covered by an edge to nearby users. FedRD widens the coverage area hundreds or even thousands of times.

To our best knowledge, it is the first research on edge-cloud federated learning-based frameworks for intelligent hazardous road damage detection and warning system. The proposed FedRD efficiently utilizes the available resources from devices, edges, and the cloud for hazardous road damage detection and warning, which satisfies the design goals listed in Section 2.1 in the following ways: Firstly, the FedRD framework deploys the hazardous road damage detection model (ADD) on edges. Hazardous road damage warning messages are sent to users immediately from the edge once any hazardous road damage is detected. The latency is extremely low because edges are very close to users (satisfying design goal 1). Secondly, the AFed strategy improves local models' detection performance by updating their parameters with the latest global model when necessary (design goal 2). In this way, high performance can be guaranteed even when some edges only have limited data, which improves the robustness of hazardous road damage detection (design goal 3). Also, the robustness is further improved by training the model with diverse data, such as from different scenarios, various illuminations, and obstacles (design goal 3). Thirdly, by leveraging the edge-cloud-based global map construction strategy, drivers/users can receive road damage information on a much broader area (satisfying design goal 4). For example, if FedRD (with 500 edges and one cloud) monitors a city's roads, and each edge's communication range is 1000 meters. Then, drivers can receive hazardous road damage information of the whole city, which may reach 500,000 meters. Fourthly, the communication and storage costs of FedRD are

Table 6
FedRD framework Evaluation Results.

framework	Accuracy (%)	F1-score (%)	Latency (s)	Global warning	Privacy risk
cloudRD [10]	86.39	86.22	2.49	**Yes**	very high
edgeRD [22]	81.76	81.81	0.054	No	high
EcRD [41]	**91.96**	**92.43**	**0.003**	No	high
FedRD (ours)	90.83	90.32	0.0326	**Yes**	**very low**

considerably reduced (design goal 5). The communication cost is significantly reduced by around 80% by filtering out normal roads using the AbRS model before transmitting to edges. The storage cost is minimized by only saving abnormal road data. Also, the 75% computation cost is reduced for applying DP by using pixelization. Finally, the FedRD framework protects data privacy (design goal 6). On the one hand, no data is directly shared from edges to the cloud, protecting privacy from untrusted clouds. On the other hand, IDPP preserves data privacy at users' devices before sending to untrusted edges and prevents privacy leakage risk during data transmission in open-access networks.

Despite the outstanding advantages, FedRD has the following limitations. Firstly, although the ADD model achieves high accuracy (93.48%), the runtime (0.085s) is not low enough for real-time hazardous road damage detection and warning. A more lightweight model should be designed to improve the accuracy and reduce the runtime. Secondly, the classification results in Fig. 8 show that classifying images only based on the damage types is not always correct, especially when some damages are small. The results would be better if the road damages can be localized on the images and use both the size and type for road damage rating. Depth information can also be utilized if 3D data is collected. Finally, this paper only tests hazardous road damage as an example for general road danger detection and warning applications. More road danger types, such as traffic accidents, fallen trees, and icy roads, can also be explored in the following researches.

7. Related Work

This section reviews the related literature about road damage classification techniques, cloud/edge computing systems for hazardous road damage inspection, and privacy.

Road damage classification techniques: Most state-of-the-art hazardous road damage classification methods can be categorized into two classes: traditional approaches and deep-learning-based approaches.

Traditional approaches are mainly based on statistics, filters, and models. Statistical-based methods leverage statistical information of the image, e.g., the distribution of image pixel value. For example, [11] combined both gray level co-occurrence matrix (GLCM) and local binary pattern (LBP) feature where GLCM was used for feature extraction, and LBP was utilized for feature's robustness improvement. The KNN classifier then classifies the features. Filter-based methods describe texture information of images by several filters. This kind of approach works well when classifying road damages with strong texture features. For example, to detect surface defects, the authors of [2] used phase-only Fourier transform to detect saliency regions of images. Then, the detected saliency areas were matched with the corresponding template regions. Model-based methods construct a mixture model with some base models according to certain distributions or other attributes. For example, the authors of [40] utilized two mixture models to calculate pattern likelihoods. Defects were detected automatically by simple parametric thresholding. Despite good performance on texture-oriented defects, they may fail for data with heterogeneous textures or when there are considerable variations of defects or backgrounds. Also, most of them have high complexity, time-inefficient, and prone to errors. Therefore, it is not suitable for real-world applications.

Deep-learning-based approaches achieved state-of-the-art results for many applications [36]. For example, Faster R-CNN and its variants are widely used in road damage danger detection in civil infrastructure like [32, 5]. For faster computing, SSD and MobileNet are utilized. MobileNet is lightweight and specially designed for mobile applications with limited resources. To produce more robust and abundant feature representations, different deep future fusion strategies are designed. For example, in [28], a multi-scale pyramidal pooling network was proposed for defect classification, which accepts different input image sizes. The authors of [18] introduced a multilevel-feature fusion network (MFN), which fused multilevel hierarchical features from different layers of the CNN backbone into the same dimension for defect recognition. Similarly, [42, 4, 35] also fused multilevel features to build discriminative hyper features for defect monitoring.

Following the success of deep-learning-based approaches, we used the deep separable convolutions as utilized in the MobileNet to build our AbRS model for real-time suspicious abnormal road screening at users' IoT devices. Also, we developed the ADD model based on ResNet structure and multilevel feature fusion for hazardous road damage detection at edges.

Cloud/edge computing systems for road damage inspection: Vision-based road damage inspection is a high resource-demanding task. Suitable computing platforms must be selected to ensure high Quality of Service (QoS), such as fast response and high accuracy. Some researchers deploy the road inspection task on clouds for their high computing power and storage capacity for data processing. For exam-

ple, in [39], a cloud server was utilized to process data received from vehicles, and the results were then sent to the traffic monitoring center. Similarly, in [10], data collected by cameras was processed by machine learning or deep learning algorithms on a cloud server to automatic the monitoring process. In this way, the calculation and storage burden is transferred to the cloud, thus, less burden for users or vehicles.

Nevertheless, cloud-based approaches still have many issues, for example, high latency and high bandwidth costs caused by continuously transmitting large amounts of data to the cloud. With the emergence of edge computing, some researchers explored edge-based inspection systems due to the advantages, e.g., location awareness, large scalability, and low latency. The collected data (by users/vehicles) are uploaded to the nearest edge in edge-based systems instead of the cloud in centralized cloud-based systems. For example, a new system for road condition inspection with edge computing was proposed by [3, 23]. Edge servers directly process data from users by specific algorithms and then transfer the results to a cloud. The cloud stores the results for later use. Kawano et al. [22] used edge computing for road damaged lane markings detection. Following its success, the authors' previous work [41] proposed an edge-cloud computing framework (EcRD) for intelligent road damage detection and warning. EcRD exploits the fast-responding benefit of edge and the high computational power and enormous storage space advantages of the cloud. However, there are still some limitations of EcRD: firstly, drivers are only informed about hazardous road damages within a small area covered by one edge. Secondly, since the detection model at each edge is just trained with the data collected from that edge, it cannot ensure the detection performance of the edges that have limited data.

Therefore, in this paper, edges are utilized for fast hazardous road damage detection and warning. Edges warn users/drivers immediately once any hazardous road damages are detected. It is much faster than cloud-based approaches. Also, the detected hazardous road damage information is transmitted to a central cloud. The cloud aggregates it and sends it back to users via edges. In this way, users can receive warnings from a wider range compared with [41]. Further, to improve the performance of edges with limited data, we used the Federated Learning strategy to learn from multiple edges collaboratively without direct data sharing.

Privacy: Despite the success of edge/cloud computing for road damage inspection, the privacy issue is still not addressed in many existing schemes [39, 10, 23]. Only a few systems considered the privacy issues, e.g., [39, 3, 41]. [39] preserved privacy against clouds by receiving data in ciphertext format. After validating the data source by the cloud and the authority, only data from legitimate vehicles are chosen. In [3], the privacy-preserving certificateless aggregate signcryption scheme (CLASC) was proposed for road condition monitoring by vehicular crowd-sensing using edge computing. The scheme is computing-efficient. However, it did not consider location privacy. Moreover, the authors' previous work [41] filtered out sensitive information by a road segmentation model at edges to protect users' privacy. However, there is still a high privacy leakage risk when sending data from users' devices to edges.

In summary, although cryptographic techniques or image/video pre-processing approaches were utilized to protect users' privacy, none of the existing works considered the privacy information within images/videos. Therefore, in this paper, we protect privacy via the Differential Privacy technique [8]. Different from [38, 30, 9], our method preserves privacy at users' devices before uploading to untrusted edges, which is more private. Moreover, it has $3/4$ less computation cost because it adds noise to pixelized images instead of original images.

8. Conclusion and Future Work

This work investigates the importance of hazardous road damage detection and warning for traffic safety. We study the critical limitations of the existing cloud/edge-based systems, such as limited warning scope, low performance of edges with no or limited data, and high data privacy leakage risks. To tackle these problems, we propose the FedRD framework. In FedRD, a new map construction approach is introduced to aggregate hazardous road damage information of a wide area into a global map, which is hundreds or thousands of times wider than existing edge-based systems. The global map contains rich hazardous road damage information and helps to improve users' travel experience and reduce road accidents. Additionally, Adaptive federated learning (AFed) is designed to improve the performance of local models on edges, especially for the edges with no or limited data. AFed ensures high detection performance in limited communication rounds by selective aggregating qualified local models for global model aggregation on the cloud. Moreover, the IDPP technique is proposed to protect data privacy. It protects privacy at users' devices before uploading to untrusted edges, and it reduces $3/4$ computation and communication costs by using pixelization. Simulation results show that the FedRD framework detects hazardous road damages accurately and warns drivers with low latency. It is robust on challenging datasets and still performs well when some edges are lacking data for training. It covers a broader area and has small computation, communication, and storage cost. Data privacy is also preserved. As part of future work, we will collect and integrate more hazardous road damages from both the real world and the Internet to further evaluate the robustness of our proposed model. In addition, we will test the proposed system in the real world to verify the robustness and explore the challenges.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Acknowledgement

We would like to thank Trevor Khwam Tabougua and Shahroz Shahjahan for their valuable help in doing the experiments related to Resnet, federated learning, and differential privacy. We also thank M. Ali Hamza and Cristhian Balta for their assistance in the design of IDPP. This work in part supported by the China Scholarship Council(Grant ID: 201706050095).

References

[1] Anaissi, A., Khoa, N.L.D., Rakotoarivelo, T., Alamdari, M.M., Wang, Y., 2019. Smart pothole detection system using vehicle-mounted sensors and machine learning. Journal of Civil Structural Health Monitoring 9, 91–102.

[2] Bai, X., Fang, Y., Lin, W., Wang, L., Ju, B.F., 2014. Saliency-based defect detection in industrial images by using phase spectrum. IEEE Transactions on Industrial Informatics 10, 2135–2145.

[3] Basudan, S., Lin, X., Sankaranarayanan, K., 2017. A privacy-preserving vehicular crowdsensing-based road surface condition monitoring system using fog computing. IEEE Internet of Things Journal 4, 772–782.

[4] Cao, J., Yang, G., Yang, X., 2020. A pixel-level segmentation convolutional neural network based on deep feature fusion for surface defect detection. IEEE Transactions on Instrumentation and Measurement 70, 1–12.

[5] Cha, Y.J., Choi, W., Suh, G., Mahmoudkhani, S., Büyüköztürk, O., 2018. Autonomous structural visual inspection using region-based deep learning for detecting multiple damage types. Computer-Aided Civil and Infrastructure Engineering 33, 731–747.

[6] Chellaswamy, C., Famitha, H., Anusuya, T., Amirthavarshini, S., 2018. Iot based humps and pothole detection on roads and information sharing, in: 2018 International Conference on Computation of Power, Energy, Information and Communication (ICCPEIC), IEEE. pp. 084–090.

[7] Colin, M., Palhol, F., Leuxe, A., 2016. Adaptation of transport infrastructures and networks to climate change. Transportation Research Procedia 14, 86–95.

[8] Dwork, C., Roth, A., et al., 2014. The algorithmic foundations of differential privacy. Foundations and Trends® in Theoretical Computer Science 9, 211–407.

[9] Fan, L., 2018. Image pixelization with differential privacy, in: IFIP Annual Conference on Data and Applications Security and Privacy, Springer. pp. 148–162.

[10] Fan, R., Bocus, M.J., Zhu, Y., Jiao, J., Wang, L., Ma, F., Cheng, S., Liu, M., 2019. Road crack detection using deep convolutional neural network and adaptive thresholding, in: 2019 IEEE Intelligent Vehicles Symposium (IV), IEEE. pp. 474–479.

[11] Fauzi, A., Utaminingrum, F., Ramdani, F., 2020. Road surface classification based on lbp and glcm features using knn classifier. Bulletin of Electrical Engineering and Informatics 9, 1446–1453.

[12] Fox, A., Kumar, B.V., Chen, J., Bai, F., 2017. Multi-lane pothole detection from crowdsourced undersampled vehicle sensor data. IEEE Transactions on Mobile Computing 16, 3417–3430.

[13] Geiping, J., Bauermeister, H., Dröge, H., Moeller, M., 2020. Inverting gradients–how easy is it to break privacy in federated learning? arXiv preprint arXiv:2003.14053 .

[14] Gleave, S.D., 2014. Eu road surfaces: Economic and safety impact of the lack of regular road maintenance, study. Policy Department Structural and Cohesion Policies .

[15] Gopalakrishnan, K., 2018. Deep learning in data-driven pavement image analysis and automated distress detection: A review. Data 3, 28.

[16] Guha, N., Talwalkar, A., Smith, V., 2019. One-shot federated learning. arXiv preprint arXiv:1902.11175 .

[17] He, K., Zhang, X., Ren, S., Sun, J., 2016. Deep residual learning for image recognition, in: Proceedings of the IEEE conference on computer vision and pattern recognition, pp. 770–778.

[18] He, Y., Song, K., Meng, Q., Yan, Y., 2019. An end-to-end steel surface defect detection approach via fusing multiple hierarchical features. IEEE Transactions on Instrumentation and Measurement 69, 1493–1504.

[19] Howard, A.G., Zhu, M., Chen, B., Kalenichenko, D., Wang, W., Weyand, T., Andreetto, M., Adam, H., 2017. Mobilenets: Efficient convolutional neural networks for mobile vision applications. arXiv preprint arXiv:1704.04861 .

[20] Jiang, H., Wang, J., Yuan, Z., Wu, Y., Zheng, N., Li, S., 2013. Salient object detection: A discriminative regional feature integration approach, in: Proceedings of the IEEE conference on computer vision and pattern recognition, pp. 2083–2090.

[21] Kairouz, P., Oh, S., Viswanath, P., 2014. Extremal mechanisms for local differential privacy, in: Advances in neural information processing systems, pp. 2879–2887.

[22] Kawano, M., Yonezawa, T., Nakazawa, J., 2017. Deep on edge: Opportunistic road damage detection with city official vehicles, in: Proceedings of The Third International Conference on Smart Portable, Wearable, Implantable and Disability-oriented Devices and Systems (SPWID 2017).

[23] Khaliq, K.A., Chughtai, O., Shahwani, A., Qayyum, A., Pannek, J., 2019. Road accidents detection, data collection and data analysis using v2x communication and edge/cloud computing. Electronics 8, 896.

[24] Koch, C., Georgieva, K., Kasireddy, V., Akinci, B., Fieguth, P., 2015. A review on computer vision based defect detection and condition assessment of concrete and asphalt civil infrastructure. Advanced Engineering Informatics 29, 196–210.

[25] Konečnỳ, J., McMahan, H.B., Yu, F.X., Richtárik, P., Suresh, A.T., Bacon, D., 2016. Federated learning: Strategies for improving communication efficiency. arXiv preprint arXiv:1610.05492 .

[26] Krizhevsky, A., Sutskever, I., Hinton, G.E., 2012. Imagenet classification with deep convolutional neural networks, in: Advances in neural information processing systems, pp. 1097–1105.

[27] Maeda, H., Sekimoto, Y., Seto, T., Kashiyama, T., Omata, H., 2018. Road damage detection and classification using deep neural networks with smartphone images. Computer-Aided Civil and Infrastructure Engineering 33, 1127–1141.

[28] Masci, J., Meier, U., Fricout, G., Schmidhuber, J., 2013. Multi-scale pyramidal pooling network for generic steel defect classification, in: The 2013 International Joint Conference on Neural Networks (IJCNN), IEEE. pp. 1–8.

[29] Reisizadeh, A., Mokhtari, A., Hassani, H., Jadbabaie, A., Pedarsani, R., 2020. Fedpaq: A communication-efficient federated learning method with periodic averaging and quantization, in: International Conference on Artificial Intelligence and Statistics, PMLR. pp. 2021–2031.

[30] Seif, M., Tandon, R., Li, M., 2020. Wireless federated learning with local differential privacy, in: 2020 IEEE International Symposium on Information Theory (ISIT), IEEE. pp. 2604–2609.

[31] Simonyan, K., Zisserman, A., 2014. Very deep convolutional networks for large-scale image recognition. arXiv preprint arXiv:1409.1556 .

[32] Suh, G., Cha, Y.J., 2018. Deep faster r-cnn-based automated detection and localization of multiple types of damage, in: Sensors and Smart Structures Technologies for Civil, Mechanical, and Aerospace Systems 2018, International Society for Optics and Photonics. p. 105980T.

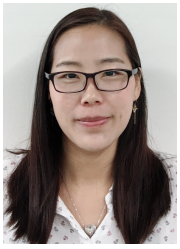
[33] Szegedy, C., Liu, W., Jia, Y., Sermanet, P., Reed, S., Anguelov, D., Erhan, D., Vanhoucke, V., Rabinovich, A., 2015. Going deeper with convolutions, in: Proceedings of the IEEE conference on computer vision and pattern recognition, pp. 1–9.

[34] Tedeschi, A., Benedetto, F., 2017. A real-time automatic pavement crack and pothole recognition system for mobile android-based devices. Advanced Engineering Informatics 32, 11–25.

[35] Wang, J., Luo, L., Ye, W., Zhu, S., 2020a. A defect-detection method

of split pins in the catenary fastening devices of high-speed railway based on deep learning. IEEE Transactions on Instrumentation and Measurement 69, 9517–9525.
[36] Wang, J., Wang, Z., Gao, C., Sang, N., Huang, R., 2016. Deeplist: Learning deep features with adaptive listwise constraint for person reidentification. IEEE Transactions on Circuits and Systems for Video Technology 27, 513–524.
[37] WANG, R.b., LI, L.h., JIN, L.s., GUO, L., ZHAO, Y.b., 2007. Study on binocular vision based obstacle detection technology for intelligent vehicle. Journal of Image and Graphics 12.
[38] Wang, T., Mei, Y., Jia, W., Zheng, X., Wang, G., Xie, M., 2020b. Edge-based differential privacy computing for sensor–cloud systems. Journal of Parallel and Distributed computing 136, 75–85.
[39] Wang, Y., Ding, Y., Wu, Q., Wei, Y., Qin, B., Wang, H., 2018. Privacy-preserving cloud-based road condition monitoring with source authentication in vanets. IEEE Transactions on Information Forensics and Security 14, 1779–1790.
[40] Xie, X., Mirmehdi, M., 2007. Texems: Texture exemplars for defect detection on random textured surfaces. IEEE Transactions on Pattern Analysis and Machine Intelligence 29, 1454–1464.
[41] Yuan, Y., Islam, M.S., Yuan, Y., Wang, S., Baker, T., Kolbe, L.M., 2020. Ecrd: Edge-cloud computing framework for smart road damage detection and warning. IEEE Internet of Things Journal , 1–1.
[42] Zhong, J., Liu, Z., Han, Z., Han, Y., Zhang, W., 2018. A cnn-based defect inspection method for catenary split pins in high-speed railway. IEEE Transactions on Instrumentation and Measurement 68, 2849–2860.

Yachao Yuan received the M.Sc. degree from University of Chongqing, Chongqing, China, in 2017. She is currently working towards her Ph.D. degree in Telematics Group, Computer Science Institute at University of Göttingen, Göttingen, Germany. Her research interests include machine learning, deep learning image/video analysis, security and privacy.



Yali Yuan received the M.Sc. degree from University of Lanzhou, Lanzhou, China, in 2015 and the Ph.D. degree in University of Göttingen, Göttingen, Germany in 2018 where she is currently working as a Postdoctoral Fellow. Her research interests include various topics related to wireless networks, in particular for the intelligent network and security.

Thar Baker is Associate Professor in the Department of Computer Science at the College of Computing and Informatics, University of Sharjah in UAE. He has received his PhD in Autonomic cloud Applications from Liverpool John Moores University (LJMU, in 2010), where he also worked as a Senior Lecturer and Reader in Cloud Engineering from 2013-2020. Before then, he was a Lecturer in Computer Science in Manchester Metropolitan University (MMU). Dr Baker has published numerous refereed research papers in multidisciplinary research areas including: cloud Computing, Distributed Software Systems, Big Data, Algorithm Design, and Autonomic Web Science. He has been actively involved as member of editorial board and review committee for a number peer reviewed international journals, and is on programme committee for a number of international conferences.

Lutz Maria Kolbe is the former Dean of the Faculty of Management and Economic Sciences and a full professor of Information Management at the University of Göttingen, Germany. He served as an IT executive in the financial industry prior to his academic career. His publications appear in outlets such as Information Systems Journal, Journal of the AIS and further journals such as Communications of the Association of Information Systems, MIS Quarterly Executive and Energy Policy as well as in many conferences such as ICIS.

Dieter Hogrefe graduated from Philips Exeter Academy, Exeter, NH, USA, in 1976, and received the Ph.D. degree in computer science and mathematics in 1985 from the University of Hannover, Hanover, Germany.,He has been a Full Professor (C4) for telematics with the University of Göttingen, Göttingen, Germany, since 2002. From 1983 to 1986, he was with SIEMENS Research Center, Munich, Germany, where he researched in the area of analysis of telecommunication systems. He was responsible for the protocol simulation and analysis of the CCS No. 7. Since 2003, he has been the Director of the Institute of Computer Science, University of Goettingen. From 1996 to 2010, he was a Chairman of the Technical Committee Methods for Testing and Specification with the European Telecommunication Standards Institute, Sophia Antipolis, France. From 2011 to 2013, he was a Dean of the Faculty of Mathematics and Computer Science. He held Full Professor positions with the University of Bern, Bern, Switzerland; the University of Lübeck, Lübeck, Germany; and the University of Goettingen and visiting positions with the University of Dortmund, Dortmund, Germany; Technical University Budapest, Budapest, Hungary; the University of California at Berkeley, Berkeley, CA, USA; and Hamilton University, Clinton, NY, USA. He has published numerous papers and two books on Internet technology, security of wireless sensor networks, analysis, simulation, and testing of formally specified communication systems. His current research interests include computer networks and communication software engineering. (Based on document published on 28 August 2019).