Implicit Gradient-Modulated Semantic Data Augmentation for Deep Crack Recognition

Zhuangzhuang Chen, Ronghao Lu, Jie Chen, Houbing Herbert Song, *Fellow, IEEE*, and Jianqiang Li, *Senior Member, IEEE*

***Abstract*— Crack detection has attracted extensive attention in an intelligent transportation system (ITS). Despite the substantial progress of deep learning technology on crack recognition tasks, due to the various limitations in traffic, equipment, and time, it is hard to collect copious samples for training deep models. Considering this, *implicitly semantic data augmentation* (ISDA) tries to augment the training set in the feature space. However, when applying it to crack recognition tasks, our empirical studies reveal that those poor-classified augmented samples have little semantic relevance to the crack class, resulting in a non-negligible negative effect on training deep models. Since the augmented features follow the multivariate normal distribution, it is computationally inefficient to explicitly sample those features and filter out the hard-classified augmented features. To this end, we propose the *implicit gradient-modulated semantic data augmentation* (IGMSDA) for addressing the above problems. Concretely, this paper first proposes gradient-modulated (GM) loss to dynamically modulate the gradient of those poor-classified augmented samples by reshaping the standard cross-entropy loss. And then, in the feature space, we derive an upper bound of the expected GM loss on the augmented training set to avoid the costly explicit sampling process. Experiments show that IGMSDA improves the generalization performance of the existing deep models on crack recognition datasets.**

***Index Terms*— Intelligent transportation system, crack detection, semantic data augmentation.**

TABLE I

THE MAINLY USED ACRONYMS IN THIS PAPER

Abbreviation	Description
ITS	Intelligent transportation systems
DL	Deep learning
CNN	Convolutional neural network
GAN	Generative adversarial network
ISDA	*Implicit semantic data augmentation*
CE	Cross-entropy
IGMSDA	Implicit gradient-modulated semantic data augmentation
GM	Gradient-modulated
VDA	Virtual data augmentation
DND	Difficult but not too different augmentation
x_i	Training sample
y_i	Ground truth label of x_i
Σ_{y_i}	Class-conditional covariance matrix of y_i
$\mathcal{M}_e$	Feature extractor of crack recognition model
$\mathcal{M}_c$	Linear classifier of crack recognition model
$\boldsymbol{\Theta}$	Parameters set of $\mathcal{M}_e$
$\boldsymbol{W}$	Parameters of weight parameters matrix of $\mathcal{M}_c$
$\boldsymbol{b}$	Parameters of bias matrix of $\mathcal{M}_c$
C	Number of classes
F	Dimension of the features
$\boldsymbol{a}_i$	Learned features of x_i
$\tilde{\boldsymbol{a}}_i$	Augmented features of x_i
p_{y_i}	Probability of the ground truth class y_i
p_t	Simplification form of p_{y_i}
$\exp(\cdot)$	Exponential function
$\mathcal{N}(\cdot,\cdot)$	Multivariate normal distribution
S	Sampling times
$\mathbb{E}[\cdot]$	Expectation w.r.t. a distribution

I. INTRODUCTION

CONCRETE structure health monitoring plays an important role in intelligent transportation systems (ITS) [1], [2], [3], [4], in which crack damage classification is the first and critical stage [4], [5]. Considering the countries like China, there are highways over 5.0 million Km that will require regular testing and maintenance [6], while repairing cracks before they deteriorate can greatly reduce maintenance costs. Therefore, there is a great need for automated crack damage classification in ITS, so as to maintain road safety, save people's properties, and reduce costs [4], [7].

Recently, with the development of deep learning (DL), the applications of deep learning technology on crack recognition tasks have achieved great success [8], [9]. This can be further explained that those DL methods can automatically extract crack features from training samples without a complex preprocessing prerequisite, showing considerable outperformance over traditional machine learning algorithms [10], [11]. However, a superior Convolutional neural network (CNN) model relies on a large-scale dataset [12], [13], [14]. Especially, for roadway maintenance tasks, crack samples are hard to collect due to the traffic, equipment, and time [10].

To tackle such a problem, many researchers adopt data augmentation techniques to generate additional training samples

Manuscript received 29 June 2023; revised 19 February 2024, 26 June 2024, and 29 July 2024; accepted 29 July 2024. This work was supported in part by the National Natural Science Funds for Distinguished Young Scholar under Grant 62325307; in part by the National Natural Science Foundation of China under Grant 62073225, Grant 62203134, and Grant 62072315; in part by the National Key Research and Development Program of China under Grant 2020YFA0908700; in part by the Natural Science Foundation of Guangdong Province under Grant 2023B1515120038; in part by Shenzhen Science and Technology Innovation Commission under Grant 20220809141216003, Grant JCYJ20210324093808021, and Grant JCYJ20220531102817040; in part by Guangdong "Pearl River Talent Recruitment Program" under Grant 2019ZT08X603; in part by Guangdong "Pearl River Talent Plan" under Grant 2019JC01X235; and in part by the Scientific Instrument Developing Project of Shenzhen University under Grant 2023YQ019. The Associate Editor for this article was J. Hemanth. *(Corresponding author: Jianqiang Li.)*

Zhuangzhuang Chen, Ronghao Lu, Jie Chen, and Jianqiang Li are with the National Engineering Laboratory for Big Data System Computing Technology, Shenzhen University, Shenzhen 518060, China (e-mail: chenzhuangzhuang2016@email.szu.edu.cn; 2110276154@email.szu.edu.cn; chenjie@szu.edu.cn; lijq@szu.edu.cn).

Houbing Herbert Song is with the Department of Information Systems, University of Maryland, Baltimore County (UMBC), Baltimore, MD 21250 USA (e-mail: songh@umbc.edu).

Digital Object Identifier 10.1109/TITS.2024.3441816

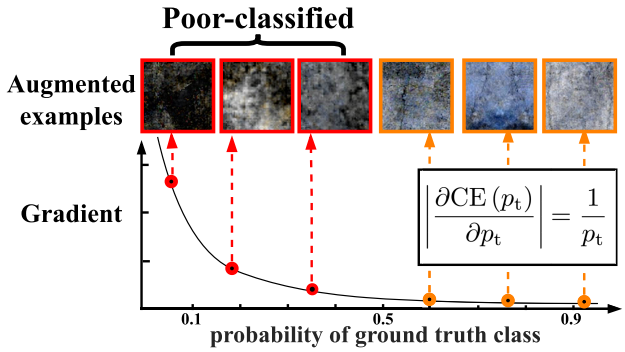


Fig. 1. Considering *implicit semantic data augmentation* for the crack recognition task, the augmented features are mapped back to the pixel space to show semantic changes of the crack images. It can be observed that the augmented features with a lower probability of the crack class have lost their crack semantics. Meanwhile, these samples will result in a large gradient during backward propagation under CE loss, leading to a non-negligible negative effect on the deep models during training. This motivates us to reduce the harmful effect of these poor-classified augmented samples by automatically modulating their gradients.

and expand the crack dataset [10], [15], [16], [17]. Generally speaking, in terms of the crack recognition tasks, data augmentation can be viewed as applying content-preserving transformations on the original images, such as horizontal flipping, cropping, random rotation, and color transformation [18]. However, these augmentation methods are limited in increasing the diversity of training data as they are not capable of performing semantic transformations [18], like changing the texture of the background area. To address this problem, existing works [10] take advantage of Generative adversarial network (GAN) [19], which help in sampling an infinite number of diverse augmented training samples via the generator. However, GAN-based methods incur huge time costs as training generative models and sampling the augmented specimen will prolong the training procedure in real-world applications.

To avoid tremendous time overhead, *implicit semantic data augmentation* (ISDA) [18] is proposed for augmenting the training set efficiently in the feature space. Specifically, certain directions in the deep feature space correspond to meaningful semantic transformations like *varying visual angles*. ISDA acquires these directions for each class by sampling random vectors from a multivariate normal distribution with zero mean and a covariance that is proportional to the intra-class covariance matrix. Note that, the above covariance matrix, which is estimated dynamically during training, is crucial for those semantic transformations to preserve semantics relevance to the augmented class. However, the covariance estimation is not quite informative when the network is not well-trained, especially for a data-limited training set under the crack recognition task. Consequently, those poor-classified samples are more likely to fail in preserving their category semantics, which violates the intentions of ISDA.

As shown in Fig 1, we exploit the reversing convolutional networks [18] to show the semantic changes of the augmented samples in the image space. It can be observed that the augmented samples with lower classification probabilities are likely to be generated by meaningless semantic transformations, resulting in the semantics irrelevance problem (ref. to Section III-B for more examples). In other words, for these poor-classified augmented samples, their intrinsic label is not consistent with the desired augmented class. Consequently, the gradient of these samples is harmful to training the deep models during backward propagation. One simple solution is a naive pipelined process, which explicitly generates augmented samples and filters out those poor-classified features. However, this strategy is highly inefficient when considering explicitly sampling infinite augmented features and filtering those negative samples.

In this paper, we propose *implicit gradient-modulated semantic data augmentation* (IGMSDA) to augment the training set in feature space. Our proposal alleviates the effect of the semantics irrelevance problem without explicit sampling. Specifically speaking, at first, we propose the gradient-modulated (GM) loss that acts as a gradient modulator for the standard cross-entropy (CE) loss. Notably, according to the probability of the ground truth class, our gradient modulator aims to dynamically modulate the gradients of augmented features during backward propagation. Secondly, to improve the efficiency further, we derive a closed-form upper bound of the expected GM loss under all possible augmented features. That way, we can directly minimize the above upper bound instead of explicitly sampling and filtering those poor-classified augmented samples. The advantage of our method is three-fold: (i) There is no need to introduce auxiliary models or an extra computational cost for sampling when alleviating the effect of those semantics irrelevant samples. (ii) Making a complement for powerful traditional augmentation techniques by serving as a plug-and-play loss function. (iii) the proposed method can be conveniently implemented on the top of most crack recognition models.

Our contributions are summarized as follows:

- To the best of our knowledge, we are the first ones that apply the *implicit semantic data augmentation* on the crack recognition task, and our empirical studies reveal that those poor-classified augmented samples are more likely to fail to preserve crack semantics, involving a non-negligible negative effect on training deep models.
- We propose the GM loss that alleviates the negative effect of those poor-classified augmented samples by dynamically modulating their gradient during backward propagation.
- To further improve the efficiency, we propose IGMSDA that addresses the above problem by deriving a closed-form upper bound of the expected GM loss. Our effort here alleviates the semantics irrelevance issue without the requirement for explicit sampling.
- We conduct extensive experiments on several crack recognition benchmark datasets, and demonstrate the superiority of the proposed method, taking a negligible computational burden.

The rest of this paper is organized as follows. Sec. II reviews related works. The details of our proposed method will be described in Sec. III. In Sec. IV, we present comparative

experiments between the proposed method and the existing methods. Sec. V concludes the whole paper.

II. Related Work

In this section, we briefly review existing research on related topics and provide the comparison of the proposed method and existing DL-based crack recognition methods in Table II.

A. Crack Recognition

According to previous research [38], image-based crack recognition via classification approaches for an intelligent transportation system can be clustered into two groups: traditional approaches and deep learning approaches. In the traditional approaches, there are two stages of the method. In the first stage, hand-craft features are extracted by different types of descriptors, e.g., Histogram of oriented gradients [39] and Local binary pattern [40]. And then, a pre-trained classifier is applied to extract potential crack patches [41], including the well-studied Gaussian process [42] and Support vector machine [43]. With the development of deep learning (DL) technology, many existing works integrate deep learning techniques for crack recognition tasks [41], [44]. The typical DL methods first train the Convolution Neural Network (CNN) [8], [20] on sub-images, then use the trained model to scan the high-resolution image with a sliding window [22] to coarsely locate the crack by classifying each sub-image. These works focus on how to design and train a powerful CNN model for identifying crack regions on sub-images of 99 $\times$ 99 pixels [20], [45], 120 $\times$ 120 pixels [21], 200 $\times$ 200 pixels [46], 224 $\times$ 224 pixels [5], and 256 $\times$ 256 pixels [22], [23]. More specifically, Silva et al. [25] adopt the pre-trained Visual Geometry Group 16 (VGG16) [47] on ImageNet, and re-train this model or crack sub-image classification. However, these deep models rely on a large set of training samples, resulting in over-fitting problems. Meanwhile, it is labor-intensive to obtain representative training samples [48], due to the prerequisite of traffic, equipment, and time.

B. Robust Loss Function

Since the proposed IGMSDA serves as a loss function for crack recognition, we give a brief review of related research on this scope. In the work [27], Focal loss is designed to assign a large weight to the hard examples, preventing a large number of easy samples from dominating the training procedure. To learn discriminative features, Center loss [49] simultaneously learns a center for deep features of each class and penalizes the distances between the samples' feature and their corresponding class centers. From a similar perspective, Cosface loss [28] introduces a cosine margin term to further maximize the decision margin in the angular space. To achieve improvements in interpretability, Arcface loss [50] adds the geometric interpretation to a hypersphere to boost the performance of the face recognition tasks. Since a similarity score should be emphasized when it deviates far from the optimum, Circle loss [51] is proposed to re-weight each similarity for highlighting the less-optimized similarity scores. Considering the inconsistent cracks in varying sizes, shapes, and noisy background textures, Chen et al. propose the geometry-aware guided loss (GAGL) that enhances the discrimination ability of learned deep features.

C. Data Augmentation

It is widely accepted that deep models can better overcome over-fitting problems by using larger datasets [52]. Data augmentation is a general technique to enlarge the amount of training set as well as the data diversity. For example, in image classification tasks, classic data augmentation methods like random horizontal or vertical flipping and rotation are applied to increase the diversity of the training set.

Among the existing powerful data augmentation methods, mixup-based methods [29], [30], [53], [54], [55] are simple yet effective, and achieve satisfactory performance. Specifically, Mixup [53] exploits random pairs of training images and their corresponding labels, to obtain more diverse samples. By combing with regional dropout strategies, CutMix [54] cuts and pastes patches among training images where the ground truth labels are also redefined according to the area of the patches. SuperMix [29] generates the augmented samples by taking advantage of the salient regions within input images. By geometrically aligning two images in the feature space, AlignMixup [30] achieves state-of-the-art performance on various tasks. Aiming to find a better augmentation strategy among many candidates, AutoAugment [31] is proposed as one of the automatic data augmentation techniques. Further, recent research proves that semantic data augmentation techniques are effective as well [18]. It applies semantic transformations while preserving class semantics (e.g. only changing backgrounds). This can be achieved by generating extra semantically transformed training samples with customized deep models such as domain adaptation networks [56] or other GAN-based models [32], [33], [34]. Aiming at obtaining the relevant augmented samples with sufficient diversity, Virtual data augmentation (VDA) [36] exploits the masked language model with an adversarial attacks mechanism to augment virtual samples for improving the robustness, and also utilizes well-designed training skills to guarantee semantic relevance and diversity. Difficult but not too different augmentation (DND) [37] is designed to obtain difficult but not too different augmented samples by a customized-design reward function. Although the above methods achieve great success, they unavoidably bring a huge time cost, due to the need to train generative models beforehand and generate the extra augmented samples. To escape from the cumbersome generative model, ISDA performs augmentation in the feature space via semantic direction. However, considering the crack recognition tasks, we reveal that some of the transformations in ISDA will cause the semantics irrelevance problem. To address this issue, we proposed IGMSDA in Section III.

III. Proposed Method

In this section, the notation descriptions are presented in Sec III-A. And then, we show the semantics irrelevance problem in the *implicit semantic data augmentation* scheme when

TABLE II
LITERATURE REVIEW OF THE EXISTING DEEP LEARNING-BASED CRACK RECOGNITION METHODS

Topics	Methods	Low computation burden at training stage	No addition cost at inference stage	Enhancing the diversity of the training set
Customized crack recognition model	[20] [21] [22] [23] [24] [25]	✓		
Loss functions	[26] [27] [28] [5]	✓	✓	
Mixup-based augmentation	[29] [30] [31]		✓	✓
GAN-based augmentation	[32] [33] [34] [35] [36] [37]		✓	✓
Implicit gradient-modulated augmentation	Our	✓	✓	✓

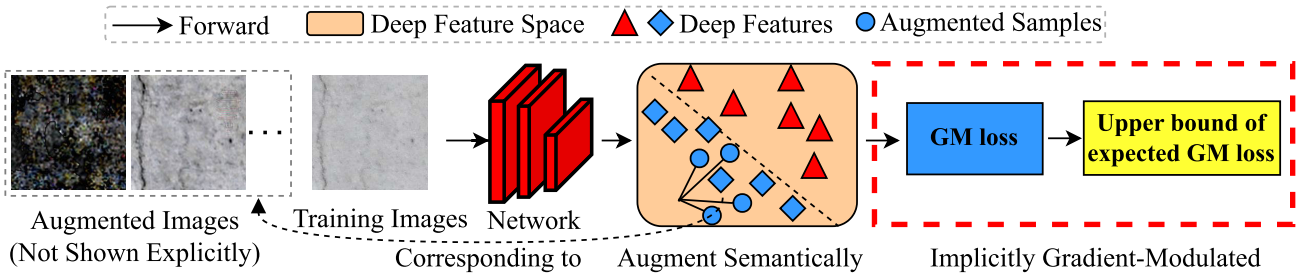


Fig. 2. An overview of IGMSDA. Our effort here alleviates the negative effect of the poor-classified augmented samples via the expected GM loss. Notably, the proposed method does not rely on any cumbersome generative models or the costly explicit sampling process.

applying to the crack recognition tasks. Next, we introduce the proposed GM loss in Sec. III-C to overcome the above problem. To further improve the efficiency, we propose IGMSDA and present the details in Sec III-D. Moreover, we discuss the complexity of the proposed IGMSDA in Sec III-E. To increase readability, the important notations used in this paper are summarized in Table I.

A. Notation Descriptions

Given a training set, let y_i denote the ground truth label of a training sample x_i. Typically, a deep model for crack recognition contains two parts: (1) a feature extractor $\mathcal{M}_e$ with parameters $\boldsymbol{\Theta}$, and (2) a classifier $\mathcal{M}_c$. In our implementation, $\mathcal{M}_c$ is implemented by a fully connected layer with the parameters of weight matrix $\boldsymbol{W} = [\boldsymbol{w}_1, \ldots, \boldsymbol{w}_C]^T \in \mathcal{R}^{C\times F}$ and bias $\boldsymbol{b} = [b_1, \ldots, b_C]^T \in \mathcal{R}^C$, where C is the number of classes and F denotes the dimension of the features. Then, the deep feature $\boldsymbol{a}_i$ of x_i can be obtained as follows:

$$\boldsymbol{a}_i = \mathcal{M}_e(x_i). \tag{1}$$

With the input of $\boldsymbol{a}_i$, the corresponding predicted probability p_{y_i} of the ground truth class y_i can be derived as follows:

$$p_{y_i} = \frac{\exp(\boldsymbol{w}_{y_i}^T \boldsymbol{a}_i^k + b_{y_i})}{\sum_{j=1}^{C} \exp(\boldsymbol{w}_j^T \boldsymbol{a}_i^k + b_j)}, \tag{2}$$

where $\exp(\cdot)$ denotes exponential function.

B. The Problem in ISDA

Considering *implicit semantic data augmentation* scheme for crack recognition tasks, the training set can be augmented by applying semantic transformations on the deep learned features. Specifically, they first randomly samples vectors from a zero-mean multivariate normal distribution $\mathcal{N}\left(0, \Sigma_{y_i}\right)$, where Σ_{y_i} is the class-conditional covariance matrix estimated from the features of the labeled samples in class y_i. This is done by using the online estimation algorithm [18]. Augmenting x_i in the feature space can be performed by translating $\boldsymbol{a}_i$ along a random direction sampled from $\mathcal{N}\left(0, \lambda\Sigma_{y_i}\right)$ presented in the following:

$$\tilde{\boldsymbol{a}}_i \sim \mathcal{N}\left(\boldsymbol{a}_i, \lambda\Sigma_{y_i}\right), \tag{3}$$

where λ is a positive coefficient to control the strength of semantic data augmentation. Then, an upper bound can be derived via the expectation of the CE loss $\mathcal{L}_{CE}$ under infinite(∞) augmented features. The corresponding equation is shown as follows:

Equation 4, as shown at the bottom of the next page, reveals that it can optimize the upper bound $\overline{\mathcal{L}_{CE}^{\infty}}$, which is equivalent to training the networks under the infinite augmented features with the supervision of CE loss. Due to the small training set of crack recognition tasks, the estimated covariances are not quite informative [18]. Therefore, those augmented samples with a lower probability of the crack class are more likely to fail to preserve their crack semantics. To verify this fact, we randomly sample some augmented features by Equation 3, and then map them back to the pixel space via the most commonly used reversing convolutional networks [18]. As we can see from Fig 3, the reconstructed images of the augmented features with a lower probability have less semantics relevance to the crack class. Thus, it will bring a non-negligible negative effect when equally treating them as augmented crack samples. In addition, it is time-consuming to simply filter out those samples, due to the need for the explicit sampling and filtering process.

C. The Proposed GM Loss

Inspiring by the fact that the smaller gradients have less impact on parameter updates during backward propagation for deeper models [57], we propose the gradient-modulated (GM) loss to alleviate the above issue. Specifically, we add a modulating factor $\frac{\alpha \cdot p_t}{\alpha \cdot p_t + 1}$ to the cross-entropy loss, with

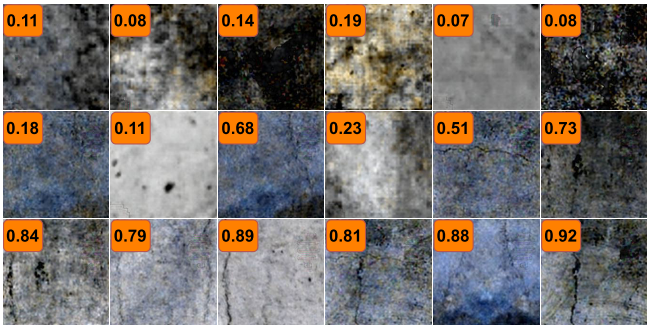


Fig. 3. Visualization of semantic data augmentation via the reconstructed images under different classification probabilities of the crack class.

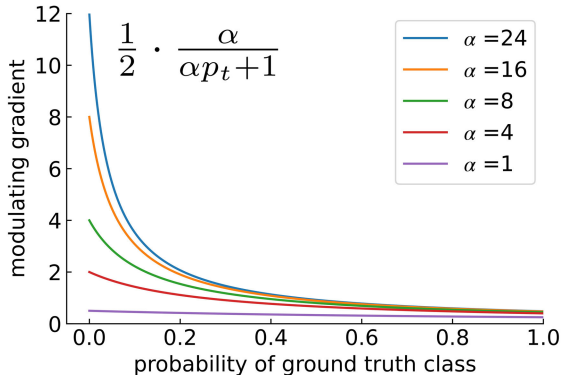


Fig. 4. We propose GM loss that adds a modulating factor $\frac{\alpha\cdot p_t}{\alpha\cdot p_t+1}$ to the CE loss, where p_t denotes the model's predicted probability for the target class t. Compared to CE loss, the gradient of GM loss contains an extra term $\frac{1}{2}\cdot\frac{\alpha}{\alpha p_t+1}$ that modulates the gradient with respect to p_t. We denote this term as "modulating gradient", and visualize this term under different α. Since the augmented features are generated by meaningless semantic transformations, they are likely to have a lower probability of the ground truth class. Setting $\alpha > 0$ reduces the relative gradient for those features, alleviating their negative effects on deep models.

tunable parameter $\alpha > 0$. That way, we are allowed to reduce the gradient of the poor-classified augmented samples according to the model's predicted probability p_t (probability of the ground truth class). Then, GM loss $\mathcal{L}_g$ can be defined in the form of Equation 5:

$$\mathcal{L}_g = -\frac{1}{2}\log(p_t \underbrace{\cdot\frac{\alpha\cdot p_t}{\alpha\cdot p_t+1}}_{\text{Modulating factor}}). \tag{5}$$

As CE loss can be formulated as $\mathcal{L}_{CE} = -\log p_t$, the gradient of $\mathcal{L}_g$ can be derived in two parts: (i) the gradient of $\mathcal{L}_{CE}$ with respect to p_t, is denoted by $\nabla\mathcal{L}_{CE}$, (ii) the modulating gradient, which is presented as the following:

$$\frac{\partial\mathcal{L}_g}{\partial p_t} = \underbrace{-\frac{1}{p_t}}_{\nabla\mathcal{L}_{CE}} + \underbrace{\frac{1}{2}\cdot\frac{\alpha}{\alpha p_t+1}}_{\text{Modulating gradient}}. \tag{6}$$

As shown in Fig. 4, in the GM loss, the modulating gradient increase with the decreases in the probability of the ground truth class. Readers are referred to Equation 6, where the gradient amplitude of augmented features that have lower classification probabilities, will be reduced to a greater extent compared to the augmented samples with higher probabilities. Thus, by dynamically modulating its gradient within GM loss, the effect of meaningless semantic transformations can be alleviated. Since we have $\frac{1}{2}\cdot\frac{\alpha}{\alpha p_t+1} < \frac{1}{2}\cdot\frac{\alpha}{\alpha p_t} < \frac{1}{2}\cdot\frac{1}{p_t}$, the modulating gradient would not change the sign of the gradient corresponding to $\mathcal{L}_g$.

Now, we consider an easy method to modulate the gradient of the augmented features by explicitly sampling from the Equation 3. Specifically speaking, we can sample S times from the distribution $\mathcal{N}(\boldsymbol{a}_i, \Sigma_{y_i})$ to compose an augmented feature set $\left\{\left(\tilde{\boldsymbol{a}}_i^1, y_i\right), \ldots, \left(\tilde{\boldsymbol{a}}_i^S, y_i\right)\right\}$ of size S. Here $\tilde{\boldsymbol{a}}_i^k$ denotes k^{th} sampled augmented features for the sample x_i. Then, the GM loss in Equation 5 can be unfolded by Equation 7 and denoted as $\mathcal{L}_g^S(\boldsymbol{W}, \boldsymbol{b}, \boldsymbol{\Theta})$:

$$\mathcal{L}_g^S(\boldsymbol{W}, \boldsymbol{b}, \boldsymbol{\Theta}) = \frac{1}{N}\sum_{i=1}^{N}\frac{1}{S}\sum_{k=1}^{S}-\frac{1}{2}\cdot\log\left(p_{y_i}^k\cdot\frac{\alpha\cdot p_{y_i}^k}{\alpha\cdot p_{y_i}^k+1}\right), \tag{7}$$

where $p_{y_i}^k$ can be computed via Equation 2. As the network is not well trained in the first few epochs, the features of the original training samples are also likely to have lower classification probabilities. To address this issue, we let $\alpha = (t/T)\times\alpha_0 + 1$ be a function of the current iteration t and the total iteration number T. This setup contributes to avoiding the impact of the GM loss on the non-augmented hard-to-classified examples in the early training stage.

D. The Proposed IGMSDA

Notably, the above easy implementation is computationally inefficient when S is large, as the feature set gets enlarged by S times. Herein, we consider the case where S grows to infinity, and find that an easy-to-compute upper bound can be derived for the GM loss function. Our initiative here leads to a highly efficient implementation, while avoiding explicit sampling of the augmented features. Concurrently, the burden of modulating the gradient of these features one by one can

$$\begin{aligned}\mathcal{L}_{CE}^{\infty}(\boldsymbol{W}, \boldsymbol{b}, \boldsymbol{\Theta} \mid \Sigma_1, \cdots, \Sigma_C) &= \frac{1}{N}\sum_{i=1}^{N}\mathbb{E}_{\tilde{\boldsymbol{a}}_i}\left[\log\left(\frac{\sum_{j=1}^{C}\exp(\boldsymbol{w}_j^T\tilde{\boldsymbol{a}}_i + b_j)}{\exp(\boldsymbol{w}_{y_i}^T\tilde{\boldsymbol{a}}_i + b_{y_i})}\right)\right] \\ &\leq \frac{1}{N}\sum_{i=1}^{N}-\log\left(\frac{\exp(\boldsymbol{w}_{y_i}^T\boldsymbol{a}_i + b_{y_i})}{\sum_{j=1}^{C}\exp(\boldsymbol{w}_j^T\boldsymbol{a}_i + b_j + \frac{\lambda}{2}\left(\boldsymbol{w}_j - \boldsymbol{w}_{y_i}\right)^T\Sigma_{y_i}\left(\boldsymbol{w}_j - \boldsymbol{w}_{y_i}\right))}\right) \\ &= \overline{\mathcal{L}_{CE}^{\infty}}.\end{aligned} \tag{4}$$

be dumped. Actually, in the case that S is close to ∞, it is equivalent to considering the expectation of the GM loss under all possible augmented features, thereby further denoting as $\mathcal{L}_g^{\infty}$ in Equation 8:

$$\begin{aligned}\mathcal{L}_g^{\infty}(\boldsymbol{W},\boldsymbol{b},\boldsymbol{\Theta}\mid\Sigma_1,\cdots,\Sigma_C) &= -\frac{1}{2}\log\alpha \\ &+ \underbrace{\frac{1}{N}\sum_{i=1}^{N}\mathbb{E}_{\tilde{\boldsymbol{a}}_i}\left[\log\left(\frac{\sum_{j=1}^{C}\exp(\boldsymbol{w}_j^T\tilde{\boldsymbol{a}}_i+b_j)}{\exp(\boldsymbol{w}_{y_i}^T\tilde{\boldsymbol{a}}_i+b_{y_i})}\right)\right]}_{\text{Part I}} \\ &+ \underbrace{\frac{1}{2N}\sum_{i=1}^{N}\mathbb{E}_{\tilde{\boldsymbol{a}}_i}\left[\log\left(\alpha\frac{\exp(\boldsymbol{w}_{y_i}^T\tilde{\boldsymbol{a}}_i+b_{y_i})}{\sum_{j=1}^{C}\exp(\boldsymbol{w}_j^T\tilde{\boldsymbol{a}}_i+b_j)}+1\right)\right]}_{\text{Part II}}.\end{aligned} \tag{8}$$

If $\mathcal{L}_g^{\infty}$ can be computed efficiently, then we can directly reduce the relative gradient for poor-classified augmented samples among the augmented features without explicit sampling. Since we can get the upper bound of part I by Equation 4, at next, we focus on how to compute part II in Equation 8. Note that, it is difficult to compute part II precisely. Hence, inspired by Equation 4, we also find a possible way to derive an easy-to-compute upper bound for it. We present the idea in the following proposition.

Proposition 1: Suppose that $\tilde{\boldsymbol{a}}_i \sim \mathcal{N}\left(\boldsymbol{a}_i, \lambda\Sigma_{y_i}\right)$, then we have an upper bound of part II, shown as:

$$\begin{aligned}&\frac{1}{2N}\sum_{i=1}^{N}\log(1+\frac{\alpha}{C^2} \\ &\cdot\left(\sum_{j=1}^{C}\exp\left(\left(\boldsymbol{w}_{y_i}^T-\boldsymbol{w}_j^T\right)\boldsymbol{a}_i+\left(b_{y_i}-b_j\right)\right.\right. \\ &\left.\left.+\frac{\lambda}{2}\left(\boldsymbol{w}_{y_i}^T-\boldsymbol{w}_j^T\right)\Sigma_{y_i}\left(\boldsymbol{w}_{y_i}-\boldsymbol{w}_j\right)\right)\right).\end{aligned} \tag{9}$$

Proof: According to the definition of the part II in Equation 8, we have Equation 10, shown at the bottom of the next page. Since the function $log(\cdot)$ is concave, we can follow Jensen's inequality $\mathbb{E}[\log X] \leq \log\mathbb{E}[X]$, and get the inequality in the second line. From Cauchy's inequality, we have Equation 11: ■

$$\begin{aligned}&\left(\sum_{j=1}^{C}\exp(\boldsymbol{w}_j^T\tilde{\boldsymbol{a}}_i+b_j)\right)\left(\sum_{j=1}^{C}\frac{1}{\exp(\boldsymbol{w}_j^T\tilde{\boldsymbol{a}}_i+b_j)}\right)\geq C^2 \\ &\Longleftrightarrow \frac{1}{C^2}\left(\sum_{j=1}^{C}\frac{1}{\exp(\boldsymbol{w}_j^T\tilde{\boldsymbol{a}}_i+b_j)}\right) \\ &\geq \frac{1}{\left(\sum_{j=1}^{C}\exp(\boldsymbol{w}_j^T\tilde{\boldsymbol{a}}_i+b_j)\right)}.\end{aligned} \tag{11}$$

It results in the inequality produced in the third line of Equation 10. Finally, the equation in the last line of Equation 10 can be obtained by leveraging the moment-generating function in Equation 12:

$$\mathbb{E}\left[e^{tX}\right]=e^{t\mu+\frac{1}{2}\sigma^2t^2},\quad X\sim\mathcal{N}\left(\mu,\sigma^2\right). \tag{12}$$

Concurrently, we have the fact that $\left(\boldsymbol{w}_{y_i}^T-\boldsymbol{w}_j^T\right)\tilde{\boldsymbol{a}}_i+\left(b_{y_i}-b_j\right)$ is a Gaussian random variable, shown as: $\left(\boldsymbol{w}_{y_i}^T-\boldsymbol{w}_j^T\right)\tilde{\boldsymbol{a}}_i+\left(b_{y_i}-b_j\right) \sim \mathcal{N}(\left(\boldsymbol{w}_{y_i}^T-\boldsymbol{w}_j^T\right)\boldsymbol{a}_i+\left(b_{y_i}-b_j\right),\lambda\left(\boldsymbol{w}_{y_i}^T-\boldsymbol{w}_j^T\right)\Sigma_{y_i}\left(\boldsymbol{w}_{y_i}-\boldsymbol{w}_j\right))$.

Then, instead of minimizing the exact loss function $\mathcal{L}_g^{\infty}$, we can optimize its joint upper bound, i.e., the part I and part II in a more efficient way. It helps in alleviating the negative effect of those poor-classified augmented samples in ISDA. Herein, the proposed IGMSDA boils down to a novel robust loss function, which can be used as a plug-and-play loss function. As shown in Algorithm 1, the proposed IGMSDA can be efficiently optimized with stochastic gradient descent (SGD) [58]. Besides, it can also be conveniently implemented on top of most existing deep models for crack recognition tasks.

Algorithm 1 The IGMSDA Algorithm

Input: Hyper-parameters λ_0, α_0, training set L, batch size B, and total iteration number T, the feature extractor $\mathcal{M}_e$ with parameters $\boldsymbol{\Theta}$, the classifier $\mathcal{M}_c$ with parameters $\boldsymbol{W}$ and $\boldsymbol{b}$
Randomly initialize $\boldsymbol{\Theta}$, $\boldsymbol{W}$, and $\boldsymbol{b}$
for $t=0$ to T **do**
  Calculate α: $\alpha=(t/T)\times\alpha_0+1$
  Sample a mini-batch $\{x_i,y_i\}_{i=1}^{B}$ from L
  Compute $\boldsymbol{a}_i=M^e(x_i)$, $i=1,\cdots,B$
  Estimate the covariance matrices $\Sigma_1,\Sigma_2,\cdots,\Sigma_C$
  Compute the joint upper bound of the part I in Equation 4 and part II in Proposition 1.
  Update $\boldsymbol{W}$, $\boldsymbol{b}$, and $\boldsymbol{\Theta}$ with SGD
end for
Output: $\boldsymbol{W}$, $\boldsymbol{b}$, and $\boldsymbol{\Theta}$

E. Complexity of IGMSDA

To illustrate the proposed IGMSDA can incur an unremarkable additional computational cost, we present a theoretical analysis in this section. Since our method is based upon the existing *implicit semantic data augmentation*, for a single sample, their computational complexity is $O\left(D^2\right)$ (using the online update formulas) and $O\left(C\times D^2\right)$ (computing the upper bound of the excepted loss in Equation 4). Here the notation D denotes the dimension of feature space and C stands for the number of classes. For IGMSDA, it involves an additional computational complexity $O\left(C\times D^2\right)$ *i.e.,* computing the upper bound of the expected loss in Equation 10. Hence the total complexity of IGMSDA would be $O\left((2C+1)\times D^2\right)$, which is highly dependent on the dimension of feature space and the number of classes. Note that, a typical convolution neural network with L layers requires $O\left(D^2\times K^2\times H\times W\times L\right)$ operations, where K denotes the filter kernel size, H and W denote the height and width

of feature maps, respectively. For example, by considering ResNet-110 [59] on CIFAR-10 [60], if we have $K = 3$, $H = W = 8$ and $L = 109$ (ignoring the last fully connected layer), then the additional computation cost of IGMSDA would be *three orders of magnitude lower* than the total computation cost of ResNet-110. Hence, the proposed method involves a negligible computational burden.

IV. Experimental Results

The essential dataset is described in Sec. IV-A. Then, we provide implementation details in Sec. IV-B and evaluation metrics in Sec. IV-C. To find the most suitable parameters α_0 in IGMSDA, we carry out a series of sensitivity studies in Sec. IV-D. Furthermore, to verify the effectiveness of the proposed method, we not only compared it with existing crack classification approaches in Sec. IV-E, but also compared it with the generator-based augmentation methods in Sec. IV-F and other state-of-the-art loss functions in Sec. IV-G. Importantly, Sec. IV-H shows that our proposed method can complement the traditional augmentation techniques. Moreover, we conduct an ablation study in Sec. IV-I to verify the importance of each component in the proposed method.

A. Datasets

NPP2021: This dataset is collected from the nuclear power plants [5]. The original image size is 7000×6000 and includes cracks as narrow as 0.05 mm and as wide as 10 mm with the noisy background. For augmentation purposes, the original images are sliced into the size of 224×224 pixels, contributing to a final dataset with 13372 samples. And then, these samples are carefully manually classified into three classes: without (w/o) cracks, with (w/) cracks, and w/ scratches. There are 5317, 4254, and 3801 training samples for each category. By following the existing settings [5], the training, validation, and test set are set at the ratio of $3 : 1 : 1$.

CRACK2019: This dataset is partially from crack500 [20]. More specifically, this dataset is obtained from Temple University and various METU campus buildings, and contains the number of 40000 images with the size of 227×227 pixels [20], [61]. It is divided into two classes: non-crack class and crack class for the crack recognition task. Each class has 20000 images. Note that, this dataset involves a large variance in terms of illumination condition or surface finish, and is automatically acquired from the 458 high-resolution images with the 4032×3024 pixels. Following the previous settings, the training, validation, and test set are set at the ratio of $3 : 1 : 1$. This dataset is publicly available.[1]

SDNET2018: This dataset is a common-used dataset, which contains over 56000 images of crack and non-crack concrete bridge decks, walls, and pavements [23]. The cracks in this dataset are as narrow as 0.06 mm and as wide as 25 mm, which also involves a variety of disturbances, including surface roughness, various edges, and multi-scale holes. Again, the training, validation, and test set are set at the ratio of $3 : 1 : 1$.

B. Implementation Details

Our implementation is based on Pytorch, which is a well-known deep-learning framework in computer vision. The mini-batch SGD [62] optimizer is used to train the model with the mini-batch size 64. By following the previous paper [5], we also select ResNet50 [63] and VGG16 [47] as the feature extractor. We note that, all images are resized into the size of 224×224 with the random horizontal flip. For all three datasets, the deep models are trained with 150 epochs in total. The initial learning rate is set to 0.01 and reduced by a factor of 10 after 65, 95, and 125 epochs. According to our experiments, we set $\alpha_0 = 2.5$, which consistently leads to optimal performance across different settings. λ_0 is the same as the one reported in the paper [18].

All of the experiments in this study are carried out on Ubuntu 18.04.6 equipped with Intel(R) Xeon(R) Gold 6148 CPU clocked at 2.40GHz and one TITAN Xp GPU.

C. Evaluation Metrics

In our paper, we adopt precision-recall analysis [64] as our evaluation metrics. The reason is that it can provide a

[1] https://data.mendeley.com/datasets/5y9wdsg2zt/2

$$
\begin{aligned}
&\frac{1}{2N}\sum_{i=1}^{N}\mathbb{E}_{\tilde{\boldsymbol{a}}_i}\left[\log\left(\alpha\frac{\exp(\boldsymbol{w}_{y_i}^T\tilde{\boldsymbol{a}}_i+b_{y_i})}{\sum_{j=1}^{C}\exp(\boldsymbol{w}_j^T\tilde{\boldsymbol{a}}_i+b_j)}+1\right)\right]\\
&\le\frac{1}{2N}\sum_{i=1}^{N}\left(\log\left(\alpha\mathbb{E}_{\tilde{\boldsymbol{a}}_i}\left[\frac{\exp(\boldsymbol{w}_{y_i}^T\tilde{\boldsymbol{a}}_i+b_{y_i})}{\sum_{j=1}^{C}\exp(\boldsymbol{w}_j^T\tilde{\boldsymbol{a}}_i+b_j)}\right]+1\right)\right)\\
&\le\frac{1}{2N}\sum_{i=1}^{N}\left(\log\left(\frac{\alpha}{C^2}\mathbb{E}_{\tilde{\boldsymbol{a}}_i}\left[\sum_{j=1}^{C}\exp((\boldsymbol{w}_{y_i}^T-\boldsymbol{w}_j^T)\tilde{\boldsymbol{a}}_i+b_{y_i}-b_j)\right]+1\right)\right)\\
&=\frac{1}{2N}\sum_{i=1}^{N}\log(1+\frac{\alpha}{C^2}\\
&\quad\cdot\left(\sum_{j=1}^{C}\exp\left(\left(\boldsymbol{w}_{y_i}^T-\boldsymbol{w}_j^T\right)\boldsymbol{a}_i+\left(b_{y_i}-b_j\right)+\frac{\lambda}{2}\left(\boldsymbol{w}_{y_i}^T-\boldsymbol{w}_j^T\right)\Sigma_{y_i}\left(\boldsymbol{w}_{y_i}-\boldsymbol{w}_j\right)\right)\right).
\end{aligned}
\tag{10}
$$

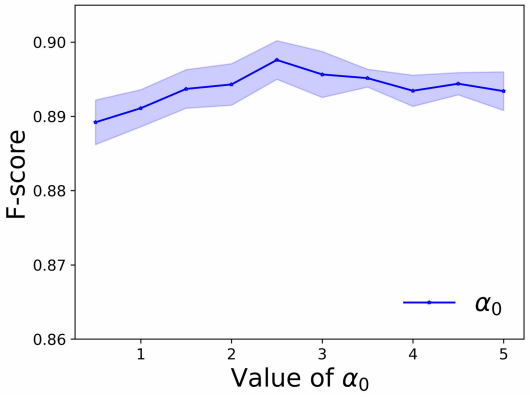


Fig. 5. Hyper-parameter sensitivity study of α_0 with ResNet50 as feature extractor on NPP2021 dataset.

more comprehensive evaluation considering the various data distributions. The precision-recall analysis consists of three metrics, namely the Precision, Recall, and F_{score}. Precision (Equation 13) is defined as the ratio of correctly predicted samples to all the test samples; Recall (Equation 14) is defined as the ratio that the model correctly classified out of all instances for all classes; As a harmonic mean of the Precision and Recall, the F_{score} (Equation 15) provides a comprehensive measure on the crack recognition performance.

$$\text{Precision} = \frac{TP}{TP + FP}, \tag{13}$$

$$\text{Recall} = \frac{TP}{TP + FN}, \tag{14}$$

$$F_{score} = \frac{2 \times \text{Precision} \times \text{Recall}}{\text{Precision} + \text{Recall}}, \tag{15}$$

where TP, TN, FP, and FN indicate true positive, false positive, false negative, and true negative, respectively.

D. Sensitivity Study on Hyper-Parameters

The corresponding hyper-parameter sensitivity study on the NPP2021 dataset is introduced in Fig. 5. α_0 is used to dynamically control the impact of the GM loss on the augmented examples during the training. It can be observed that the hyper-parameter α_0 across a wide range has a small test error increase compared with the lowest, which demonstrates the satisfying stability of our proposed method applied in real-world crack recognition tasks. Based on these observations, we set $\alpha_0 = 2.5$ in our next experiments.

E. Comparison With Existing Crack Recognition Approaches

To verify the effectiveness of our method, we compare the proposed method with various baseline methods on NPP2021 and CRACK2019 dataset.

- ConvNet [20] designs a supervised deep model to classify each image patch of the collected images.
- Crack-CNN [21] proposes a deep learning framework to deal with the noisy background in the image.
- SDNET [23] carries experiments with different network architectures and selects the well-performed AlexNet architecture for detecting the crack.
- AliNet [24] proposes a customized ResNet50 architecture and uses sliding windows approach to detect and localize the crack in the concrete building.
- SilvaNet [25] uses the open-source VGG16 architecture as basis for the design of the crack classification method.

The IGMSDA^R and IGMSDA^V denote ResNet50 [63] and VGG16 [47] under the supervision of the joint loss in Sec. III-D, respectively. Both IGMSDA^R and IGMSDA^V are compared with other approaches in terms of Precision, Recall, and F_{score}.

As shown in Table III, it can be observed that our methods compare favorably to other competitive crack classification approaches. For example, compared with the AliNet [24] that adopts the ResNet50 architecture on the NPP2021 dataset, the Precision, Recall, and F_{score} of IGMSDA with ResNet50 are 89.95 %, 88.07 % and 89.00 %, respectively, while the AliNet only achieves 82.86 % in term of F_{score} metric. For the CRACK2019, IGMSDA refreshes F_{score} about 0.22 % and 0.33 % by combining with ResNet50 and VGG16, when compared with SilvaNet [25] that adopts VGG16 architecture as basis for the crack recognition model. The proposed method shows consistent improvements over the NPP2021 and CRACK2019 datasets. This further illustrates that IGMSDA can efficiently augment the training set for releasing the data scarcity of the training samples, so that the crack model can achieve high performance on the test set.

F. Comparison With Existing Data Augmentation Methods

As the intuition of IGMSDA is to augment the training set, we also compare it with the generator-based augmentation methods on the SDNET2018 dataset and report experiment results with Precision, Recall, and F_{score} metrics. For AC-GAN [32], DA-CGAN [33], and DA-infoGAN [34], the corresponding generator is required to generate images for crack classes. A 100-dimension noise drawn from a standard normal distribution is adopted as input, generating images corresponding to their label. Synthetic images are involved with a fixed ratio in every mini-batch. Based on the experiments on the validation set, the proportions of generated images are set to 1/6, 1/5, and 1/5 for AC-GAN, DA-CGAN, and DA-infoGAN, respectively. For the negative data augmentation (NDA) [35], we adopt the same setting as in [35], and exploit BigGAN [65] as the generative model for conditional image generation. Similar to the previous, the proportions of generalized images are set to 1/5, according to the validation set.

Table IV shows the comparison with the existing GAN-based data augmentation methods. These GAN-based augmentation methods aim to train cumbersome generative models and synthesize many augmented samples. Then, these augmented samples can be used to train the classification model at the training stage. After training, the classification model can be used at the inference stage. We would like to note that our method enjoys the advantage of these augmentation-based methods that are only applied in the training stage without extra computation and memory during inference. For this reason, our method and these GAN-based methods take only 1.8ms for one input image at the inference

TABLE III

COMPARISON WITH THE STATE-OF-THE-ART CRACK CLASSIFICATION METHODS ON VARIOUS DATASETS. WE REPORT THE PRECISION, RECALL, AND F_{SCORE} ON NPP2021 AND CRACK2019 DATASETS

Methods \ Dataset	NPP2021			CRACK2019		
	Precision (%)	Recall (%)	F_{score} (%)	Precision (%)	Recall (%)	F_{score} (%)
ConvNet[20]	79.86	82.54	81.18	97.32	96.18	96.75
Crack-CNN [21]	81.91	82.56	82.23	99.01	98.56	98.78
SDNET [23]	80.51	81.47	81.00	98.83	98.98	98.90
AliNet [24]	83.05	82.68	82.86	99.14	98.65	98.89
SilvaNet [25]	84.90	84.02	84.45	99.39	99.12	99.25
IGMSDA^V	89.03	87.69	88.35	99.63	99.32	99.47
IGMSDA^R	89.95	88.07	89.00	99.54	99.63	99.58

TABLE IV

COMPARISON WITH THE GENERATOR-BASED AUGMENTATION METHODS ON THE SDNET2018 DATASET IN TERMS OF TRAINING TIME AND F_{SCORE}. THE COMMONLY USED RESNET50 IS SELECTED AS THE CLASSIFIER

Method	Training Time (h)	Inference Time (ms)	F_{score} (%)
AC-GAN [32]	6.6	1.8	91.18
DA-CGAN [33]	5.9		90.56
DA-infoGAN [34]	6.3		92.03
NDA [35]	5.7		91.65
IGMSDA	1.2		93.57

TABLE V

COMPARISON WITH THE EXISTING LOSS FUNCTIONS ON NPP2021 DATASET WITH RESNET50

Methods	Precision (%)	Recall (%)	F_{score} (%)
CE Loss [26]	86.74	84.81	85.76
Focal Loss [27]	87.75	85.63	86.68
Center Loss [49]	87.04	84.19	85.59
Cosface Loss [28]	88.19	87.15	87.67
Arcface Loss [50]	87.18	86.53	86.85
Circle Loss [51]	88.85	86.39	87.60
ISDA [18]	87.64	87.12	87.38
GAGL [5]	87.95	87.33	87.64
IGMSDA	89.95	88.07	89.00

stage, as they use the same classification model. Moreover, IGMSDA not only achieves better performance, but is also easier to implement as it does not require explicit sampling at the training stage. For this purpose, our method involves a smaller training burden when compared with the existing methods, as it only takes 1.2 hours for the training. The superiority of IGMSDA can be attributed to two reasons: (i) meaningful semantic transformations can provide more diversity for the training set. (ii) the effect of those negative augmented samples can be alleviated by the upper bound based on the expected GM loss.

G. Comparison With Existing Loss Functions

As mentioned, the proposed IGMSDA can be served as a novel robust loss function. Thus, we also compare it with the existing loss functions, including CE loss [26], Focal loss [27], Center loss [49], Cosface loss [28], Arcface loss [50], Circle loss [51], ISDA [18], and GAGL [5].

Table V provides the quantitative results compared with the existing powerful loss functions. First of all, our IGMSDA outperforms ISDA [18] by 2.31% (Precision), 0.95% (Recall) and 1.62% (F_{score}), respectively. The reason is two-fold: (i) The poor-classified augmented samples within ISDA have a non-negligible negative effect on training deep models. (ii) IGMSDA can efficiently alleviate the negative effect of those poor-classified augmented samples by implicitly modulating their gradient during backward propagation. Then, the crack model can extract crack features better, promoting recognition of crack and non-crack images. Second, the proposed method also performs well on the recall metric. The reason for this effect is that IGMSDA can generate diverse crack samples in the feature space, and then it can provide a better regularization for the crack model, as a result, it can help the model to extract discriminative features for both crack and non-crack samples.

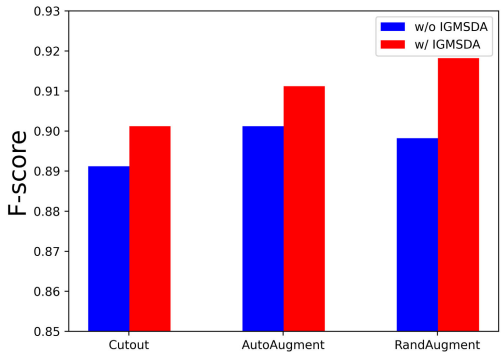


Fig. 6. Comparison of non-semantic augmentation techniques without (w/o) or with (w/) our IGMSDA on the NPP2021 dataset.

H. Complementing Traditional Augmentation Techniques

To further demonstrate our method can be a complement to various traditional augmentation methods, we also conduct a series of experiments that use IGMSDA to train the ResNet50 with these traditional augmentation methods including Cutout [66], AutoAugment [31], and RandAugment [67].

Interestingly, as shown in Fig. 6, the result of our proposed method exceeds those without IGMSDA by a large margin on the NPP2021 dataset, showing that the proposed IGMSDA can further improve the performance of networks that use only these traditional augmentation methods. A reasonable explanation for this phenomenon is that our method can help to increase the diversity of the training set in the deep feature space, which can amplify the effect of these methods.

I. Ablation Study

To further demonstrate the effectiveness of the proposed method, the ablation studies are performed on the NPP2021

TABLE VI
ABLATION STUDY ON THE NPP2021 DATASET WITH RESNET50 AND RESNET110. 'GM' DENOTES THE GRADIENT-MODULATED LOSS COMBINED WITH EXPLICIT SAMPLING

Networks	Method	Additional wall time	F_{score} (%)
ResNet50	ISDA	-	87.38
	GM loss	40.12%	88.73
	IGMSDA	2.5%	89.00
ResNet110	ISDA	-	88.06
	GM loss	35.47%	89.01
	IGMSDA	1.9%	89.43

dataset with ResNet50 and ResNet110. It should be noted that we explicitly sample augmented features 1000 times for each training sample in the implementation GM loss. As we can see from Table VI, our method significantly improves the F_{score} by 1.62% and 1.37% with ResNet50 and ResNet110, when compared with ISDA. Meanwhile, we successfully reduce the huge computation burden of GM loss, while improving performance. Interestingly, in terms of F_{score} metric, it gains more for smaller model configurations. The reasons are two-fold. Firstly, the small network is harder to train by finding the right parameters [68]. Thus, it is hard for these networks to learn better feature representation, which makes the estimated covariances unreliable and generates the semantics irrelevant samples within ISDA. To this end, the proposed method can achieve a better performance, as IGMSDA allows to alleviate the effect of those harmful augmented samples by implicitly modulating their gradients. Considering the efficiency, Table VI reports the additional wall time over ISDA. As expected, the GM loss involves high time costs due to explicit sampling. In contrast, IGMSDA involves negligible additional costs and achieves better results.

V. CONCLUSION AND FURTHER WORK

In this paper, we are the first ones that try to apply the *implicit semantic data augmentation* on the crack recognition task for increasing the diversity of the crack training set. And then, our empirical studies reveal that those poor-classified augmented samples have a non-negligible negative effect on training crack models. Based on these, GM loss is proposed to alleviate those negative effects by explicitly sampling and modulating the gradient of those poor-classified augmented samples. To release the burden of explicit sampling, IGMSDA is presented to address the considered problem simultaneously by deriving a closed-form upper bound of the expected GM loss. Our effort here does not require any auxiliary models or an extra computational cost for explicit sampling. Besides, the proposed method acts as a plug-and-play loss function and can make a complement for the other augmentation techniques. A series of experiments on several competitive crack classification datasets demonstrate the effectiveness of the proposal here.

In future work, we would like to explore the potential of the proposed method on crack segmentation tasks [69]. Moreover, considering a new scenario for the crack recognition task, it is crucial to select the most informative samples for constructing a high-quantity dataset. To this end, the proposed method can be further combined with active learning by looking ahead the effect of semantic data augmentation in the selection of unlabeled samples [70].

REFERENCES

[1] S. M. Khan, S. Atamturktur, M. Chowdhury, and M. Rahman, "Integration of structural health monitoring and intelligent transportation systems for bridge condition assessment: Current status and future direction," *IEEE Trans. Intell. Transp. Syst.*, vol. 17, no. 8, pp. 2107–2122, Aug. 2016.

[2] X. Ma, Z. Dong, and Y. Dong, "Toward asphalt pavement health monitoring with built-in sensors: A novel application to real-time modulus evaluation," *IEEE Trans. Intell. Transp. Syst.*, vol. 23, no. 11, pp. 22040–22052, Nov. 2022.

[3] D. Dan, Y. Ying, and L. Ge, "Digital twin system of bridges group based on machine vision fusion monitoring of bridge traffic load," *IEEE Trans. Intell. Transp. Syst.*, vol. 23, no. 11, pp. 22190–22205, Nov. 2022.

[4] L. Guo, R. Li, and B. Jiang, "A cascade broad neural network for concrete structural crack damage automated classification," *IEEE Trans. Ind. Informat.*, vol. 17, no. 4, pp. 2737–2742, Apr. 2021.

[5] Z. Chen, J. Zhang, Z. Lai, J. Chen, Z. Liu, and J. Li, "Geometry-aware guided loss for deep crack recognition," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2022, pp. 4703–4712.

[6] R. Zhou et al., "The development and practice of China highway capacity research," *Transp. Res. Proc.*, vol. 15, pp. 14–25, Jan. 2016.

[7] D. Yu, S. Ji, X. Li, Z. Yuan, and C. Shen, "Earthquake crack detection from aerial images using a deformable convolutional neural network," *IEEE Trans. Geosci. Remote Sens.*, vol. 60, 2022, Art. no. 4412012.

[8] K. Gopalakrishnan, S. K. Khaitan, A. Choudhary, and A. Agrawal, "Deep convolutional neural networks with transfer learning for computer vision-based data-driven pavement distress detection," *Construction Building Mater.*, vol. 157, pp. 322–330, Dec. 2017.

[9] Z. Chen et al., "The devil is in the crack orientation: A new perspective for crack detection," in *Proc. IEEE/CVF Int. Conf. Comput. Vis. (ICCV)*, Oct. 2023, pp. 6653–6663.

[10] Y. Que et al., "Automatic classification of asphalt pavement cracks using a novel integrated generative adversarial networks and improved VGG model," *Eng. Struct.*, vol. 277, Feb. 2023, Art. no. 115406.

[11] W. Li, J. Li, M. Ma, X. Hong, and X. Fan, "Multi-scale spiking pyramid wireless communication framework for food recognition," *IEEE Trans. Multimedia*, early access, Mar. 14, 2024, doi: 10.1109/TMM.2024.3368964.

[12] B. Pu, K. Li, S. Li, and N. Zhu, "Automatic fetal ultrasound standard plane recognition based on deep learning and IIoT," *IEEE Trans. Ind. Informat.*, vol. 17, no. 11, pp. 7771–7780, Nov. 2021.

[13] L. Zhao et al., "FARN: Fetal anatomy reasoning network for detection with global context semantic and local topology relationship," *IEEE J. Biomed. Health Informat.*, vol. 28, no. 8, pp. 4866–4877, Aug. 2024.

[14] W. Li, X.-L. Zhao, Z. Ma, X. Wang, X. Fan, and Y. Tian, "Motion-decoupled spiking transformer for audio-visual zero-shot learning," in *Proc. 31st ACM Int. Conf. Multimedia*, Oct. 2023, pp. 3994–4002.

[15] L. Pei, Z. Sun, L. Xiao, W. Li, J. Sun, and H. Zhang, "Virtual generation of pavement crack images based on improved deep convolutional generative adversarial network," *Eng. Appl. Artif. Intell.*, vol. 104, Sep. 2021, Art. no. 104376.

[16] D. Xia, H. Liu, L. Xu, and L. Wang, "Visible-infrared person re-identification with data augmentation via cycle-consistent adversarial network," *Neurocomputing*, vol. 443, pp. 35–46, Jul. 2021.

[17] J. Li, T. Liu, X. Wang, and J. Yu, "Automated asphalt pavement damage rate detection based on optimized GA-CNN," *Autom. Construction*, vol. 136, Apr. 2022, Art. no. 104180.

[18] Y. Wang, G. Huang, S. Song, X. Pan, Y. Xia, and C. Wu, "Regularizing deep networks with semantic data augmentation," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 44, no. 7, pp. 3733–3748, Jul. 2022.

[19] I. Goodfellow et al., "Generative adversarial nets," in *Proc. Adv. Neural Inform. Process. Syst.*, 2014, pp. 1–9.

[20] L. Zhang, F. Yang, Y. Daniel Zhang, and Y. J. Zhu, "Road crack detection using deep convolutional neural network," in *Proc. IEEE Int. Conf. Image Process. (ICIP)*, Sep. 2016, pp. 3708–3712.

[21] F.-C. Chen and M. R. Jahanshahi, "NB-CNN: Deep learning-based crack detection using convolutional neural network and Naïve Bayes data fusion," *IEEE Trans. Ind. Electron.*, vol. 65, no. 5, pp. 4392–4400, May 2018.

[22] Y.-J. Cha, W. Choi, and O. Büyüköztürk, "Deep learning-based crack damage detection using convolutional neural networks," *Comput.-Aided Civ. Infrastruct. Eng.*, vol. 32, no. 5, pp. 361–378, 2017.

[23] S. Dorafshan, R. J. Thomas, and M. Maguire, "SDNET2018: An annotated image dataset for non-contact concrete crack detection using deep convolutional neural networks," *Data Brief*, vol. 21, pp. 1664–1668, Dec. 2018.

[24] L. Ali, F. Alnajjar, H. A. Jassmi, M. Gocho, W. Khan, and M. A. Serhani, "Performance evaluation of deep CNN-based crack detection and localization techniques for concrete structures," *Sensors*, vol. 21, no. 5, p. 1688, Mar. 2021.

[25] W. R. L. D. Silva and D. S. D. Lucena, "Concrete cracks detection based on deep learning image classification," in *Proc. 18th Int. Conf. Experim. Mech.*, Jun. 2018, vol. 2, no. 8, p. 489.

[26] S. Mannor, D. Peleg, and R. Rubinstein, "The cross entropy method for classification," in *Proc. 22nd Int. Conf. Mach. Learn. (ICML)*, 2005, pp. 561–568.

[27] T.-Y. Lin, P. Goyal, R. Girshick, K. He, and P. Dollár, "Focal loss for dense object detection," in *Proc. IEEE Int. Conf. Comput. Vis. (ICCV)*, Oct. 2017, pp. 2980–2988.

[28] H. Wang et al., "CosFace: Large margin cosine loss for deep face recognition," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, Jun. 2018, pp. 5265–5274.

[29] A. Dabouei, S. Soleymani, F. Taherkhani, and N. M. Nasrabadi, "SuperMix: Supervising the mixing data augmentation," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2021, pp. 13789–13798.

[30] S. Venkataramanan, E. Kijak, L. Amsaleg, and Y. Avrithis, "AlignMixup: Improving representations by interpolating aligned features," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2022, pp. 19174–19183.

[31] E. D. Cubuk, B. Zoph, D. Mane, V. Vasudevan, and Q. V. Le, "AutoAugment: Learning augmentation policies from data," 2018, *arXiv:1805.09501*.

[32] A. Odena, C. Olah, and J. Shlens, "Conditional image synthesis with auxiliary classifier GANs," in *Proc. Int. Conf. Mach. Learn.*, 2017, pp. 2642–2651.

[33] O. Bailo, D. Ham, and Y. M. Shin, "Red blood cell image generation for data augmentation using conditional generative adversarial networks," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. Workshops (CVPRW)*, Jun. 2019, pp. 1039–1048.

[34] J. Li, Z. Chen, J. Chen, and Q. Lin, "Diversity-sensitive generative adversarial network for terrain mapping under limited human intervention," *IEEE Trans. Cybern.*, vol. 51, no. 12, pp. 6029–6040, Dec. 2021.

[35] A. Sinha, K. Ayush, J. Song, B. Uzkent, H. Jin, and S. Ermon, "Negative data augmentation," in *Proc. Int. Conf. Learn. Represent.*, 2020, pp. 1–18.

[36] K. Zhou, W. X. Zhao, S. Wang, F. Zhang, W. Wu, and J.-R. Wen, "Virtual data augmentation: A robust and general framework for fine-tuning pre-trained models," in *Proc. Conf. Empirical Methods Natural Lang. Process.*, 2021, pp. 3875–3887.

[37] J. Kim, D. Kang, S. Ahn, and J. Shin, "What makes better augmentation strategies? Augment difficult but not too different," in *Proc. Int. Conf. Learn. Represent.*, 2022, pp. 1–25.

[38] S. D. Nguyen, T. S. Tran, V. P. Tran, H. J. Lee, M. J. Piran, and V. P. Le, "Deep learning-based crack detection: A survey," *Int. J. Pavement Res. Technol.*, vol. 16, pp. 943–967, Apr. 2022.

[39] C. Tomasi, "Histograms of oriented gradients," *Comput. Vis. Sampler*, pp. 1–6, Sep. 2012.

[40] M. Pietikäinen, "Local binary patterns," *Scholarpedia*, vol. 5, no. 3, p. 9775, 2010.

[41] F. Fang, L. Li, Y. Gu, H. Zhu, and J.-H. Lim, "A novel hybrid approach for crack detection," *Pattern Recognit.*, vol. 107, Nov. 2020, Art. no. 107474.

[42] H. Oliveira and P. L. Correia, "Automatic road crack detection and characterization," *IEEE Trans. Intell. Transp. Syst.*, vol. 14, no. 1, pp. 155–168, Mar. 2013.

[43] M. Quintana, J. Torres, and J. M. Menéndez, "A simplified computer vision system for road surface inspection and maintenance," *IEEE Trans. Intell. Transp. Syst.*, vol. 17, no. 3, pp. 608–619, Mar. 2016.

[44] A. Sekar and V. Perumal, "CFC-GAN: Forecasting road surface crack using forecasted crack generative adversarial network," *IEEE Trans. Intell. Transp. Syst.*, vol. 23, no. 11, pp. 21378–21391, Nov. 2022.

[45] M. Eisenbach et al., "How to get pavement distress detection ready for deep learning? A systematic approach," in *Proc. Joint Int. Conf. Neural Netw. (IJCNN)*, 2017, pp. 2039–2047.

[46] K. Zhang, H. D. Cheng, and B. Zhang, "Unified approach to pavement crack and sealed crack detection using preclassification based on transfer learning," *J. Comput. Civil Eng.*, vol. 32, no. 2, pp. 4–18, Mar. 2018.

[47] K. Simonyan and A. Zisserman, "Very deep convolutional networks for large-scale image recognition," 2014, *arXiv:1409.1556*.

[48] H. Li, D. Song, Y. Liu, and B. Li, "Automatic pavement crack detection by multi-scale image fusion," *IEEE Trans. Intell. Transp. Syst.*, vol. 20, no. 6, pp. 2025–2036, Jun. 2019.

[49] Y. Wen, K. Zhang, Z. Li, and Y. Qiao, "A discriminative feature learning approach for deep face recognition," in *Proc. Eur. Conf. Comput. Vis.*, 2016, pp. 499–515.

[50] J. Deng, J. Guo, N. Xue, and S. Zafeiriou, "ArcFace: Additive angular margin loss for deep face recognition," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2019, pp. 4690–4699.

[51] Y. Sun et al., "Circle loss: A unified perspective of pair similarity optimization," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2020, pp. 6397–6406.

[52] J. Deng, W. Dong, R. Socher, L.-J. Li, K. Li, and L. Fei-Fei, "ImageNet: A large-scale hierarchical image database," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2009, pp. 248–255.

[53] H. Zhang, M. Cisse, Y. N. Dauphin, and D. Lopez-Paz, "mixup: Beyond empirical risk minimization," in *Proc. Int. Conf. Learn. Represent.*, 2018, pp. 1–13.

[54] S. Yun, D. Han, S. Chun, S. J. Oh, Y. Yoo, and J. Choe, "CutMix: Regularization strategy to train strong classifiers with localizable features," in *Proc. IEEE/CVF Int. Conf. Comput. Vis. (ICCV)*, Oct. 2019, pp. 6023–6032.

[55] V. Verma et al., "Manifold mixup: Better representations by interpolating hidden states," in *Proc. Int. Conf. Mach. Learn.*, 2019, pp. 6438–6447.

[56] K. Bousmalis, N. Silberman, D. Dohan, D. Erhan, and D. Krishnan, "Unsupervised pixel-level domain adaptation with generative adversarial networks," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jul. 2017, pp. 3722–3731.

[57] W. Cai, M. Zhang, and Y. Zhang, "Batch mode active learning for regression with expected model change," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 28, no. 7, pp. 1668–1681, Jul. 2017.

[58] N. Ketkar, "Stochastic gradient descent," in *Deep Learning With Python*. Berkeley, CA, USA: Apress, 2017, pp. 113–132.

[59] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2016, pp. 770–778.

[60] A. Krizhevsky and G. Hinton, "Learning multiple layers of features from tiny images," Univ. Toronto, Toronto, ON, Canada, 2009.

[61] Ç. F. Özgenel and A. G. Sorguç, "Performance comparison of pretrained convolutional neural networks on crack detection in buildings," in *Proc. Int. Symp. Autom. Robot. Construction (IAARC)*, Jul. 2018, pp. 1–8.

[62] P. Goyal et al., "Accurate, large minibatch SGD: Training ImageNet in 1 hour," 2017, *arXiv:1706.02677*.

[63] K. He, X. Zhang, S. Ren, and J. Sun, "Identity mappings in deep residual networks," in *Proc. Eur. Conf. Comput. Vis.*, 2016, pp. 630–645.

[64] P. Flach and M. Kull, "Precision-recall-gain curves: PR analysis done right," in *Proc. Adv. Neural Inform. Process. Syst.*, vol. 28, 2015, pp. 1–9.

[65] A. Brock, J. Donahue, and K. Simonyan, "Large scale GAN training for high fidelity natural image synthesis," in *Proc. Int. Conf. Learn. Represent.*, 2018, pp. 1–35.

[66] T. DeVries and G. W. Taylor, "Improved regularization of convolutional neural networks with cutout," 2017, *arXiv:1708.04552*.

[67] E. D. Cubuk, B. Zoph, J. Shlens, and Q. V. Le, "Randaugment: Practical automated data augmentation with a reduced search space," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. Workshops (CVPRW)*, Jun. 2020, pp. 702–703.

[68] Y. Zhang, T. Xiang, T. M. Hospedales, and H. Lu, "Deep mutual learning," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, Jun. 2018, pp. 4320–4328.

[69] Z. Chen, Z. Lai, J. Chen, and J. Li, "Mind marginal non-crack regions: Clustering-inspired representation learning for crack segmentation," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2024, pp. 12698–12708.

[70] Z. Chen, J. Zhang, P. Wang, J. Chen, and J. Li, "When active learning meets implicit semantic data augmentation," in *Proc. IEEE Eur. Conf. Comput. Vis. (ECCV)*, Oct. 2022, pp. 56–72.

Zhuangzhuang Chen received the B.S. degree from Jiangnan University in 2016 and the M.E. degree from Shenzhen University in 2019, where he is currently pursuing the Ph.D. degree with the Computer and Software College.

His main research interests mainly include robotics, the Internet of Things, crack detection, active learning, and intelligent transportation systems.

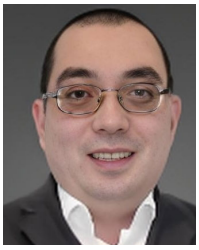
Ronghao Lu received the B.Sc. degree from Shenzhen University, Shenzhen, China, in 2020, where he is currently pursuing the M.S. degree.

His current research interests include crack segmentation, deep learning, and intelligent transportation systems.

Jie Chen received the B.Sc. degree in EE from Taizhou University, Taizhou, China, in 2006, the M.Sc. degree in CS from Zhejiang University, Hangzhou, China, in 2008, and the Ph.D. degree in CS from the National University of Singapore (NUS), Singapore, in 2013.

He was a Post-Doctoral Associate with Singapore MIT Alliance for Research and Technology (SMART), Singapore. In 2018, he joined Shenzhen University, Shenzhen, China, as a Research-Track Associate Professor. His research interests include machine learning, multiagent systems, robotics, intelligent transportation systems, and intelligent healthcare system.

Dr. Chen received the Dean's Graduate Research Excellence Award from NUS in 2013. He was a Program Committee Member of IJCAI, AAAI, and ICRA.



Houbing Herbert Song (Fellow, IEEE) received the Ph.D. degree in electrical engineering from the University of Virginia, Charlottesville, VA, USA, in August 2012.

He is currently a Professor; the Director of the NSF Center for Aviation Big Data Analytics (Planning); the Associate Director for Leadership of the DOT Transportation Cybersecurity Center for Advanced Research and Education (Tier 1 Center); and the Director of the Security and Optimization for Networked Globe Laboratory (SONG Lab, www.SONGLab.us), University of Maryland, Baltimore County (UMBC), Baltimore, MD, USA. Prior to joining UMBC, he was a Tenured Associate Professor of electrical engineering and computer science with Embry-Riddle Aeronautical University, Daytona Beach, FL, USA. He is the editor of eight books, the author of more than 100 articles, and the inventor of two patents. His research has been sponsored by federal agencies (including the National Science Foundation, the National Aeronautics and Space Administration, U.S. Department of Transportation, and Federal Aviation Administration, and among others) and industry. His research has been featured by popular news media outlets, including IEEE GlobalSpec's Engineering360, Association for Uncrewed Vehicle Systems International (AUVSI), *Security Magazine*, *CXOTech Magazine*, Fox News, U.S. News & World Report, The Washington Times, and New Atlas. His research interests include cyber-physical systems/the Internet of Things, cybersecurity and privacy, and AI/machine learning/big data analytics.

Dr. Song is an IEEE Fellow for contributions to big data analytics and integration of AI with the Internet of Things, and an ACM Distinguished Member for outstanding scientific contributions to computing. He received the Research.com Rising Star of Science Award in 2022, the 2021 Harry Rowe Mimno Award bestowed by IEEE Aerospace and Electronic Systems Society, and more than ten Best Paper Awards from major international conferences, including IEEE CPSCom-2019, IEEE ICII 2019, IEEE/AIAA ICNS 2019, IEEE CBDCom 2020, WASA 2020, AIAA/ IEEE DASC 2021, IEEE GLOBECOM 2021, and IEEE INFOCOM 2022. He serves as an Associate Editor for IEEE TRANSACTIONS ON ARTIFICIAL INTELLIGENCE (since 2023), IEEE INTERNET OF THINGS JOURNAL (since 2020), IEEE TRANSACTIONS ON INTELLIGENT TRANSPORTATION SYSTEMS (since 2021), and IEEE JOURNAL ON MINIATURIZATION FOR AIR AND SPACE SYSTEMS (since 2020). He was an Associate Technical Editor of *IEEE Communications Magazine* (2017–2020). He is an ACM Distinguished Speaker (since 2020), an IEEE Vehicular Technology Society (VTS) Distinguished Lecturer (since 2023, and an IEEE Systems Council Distinguished Lecturer (since 2023). He has been a Highly Cited Researcher identified by Clarivate in 2021 and 2022.

Jianqiang Li (Senior Member, IEEE) received the B.S. and Ph.D. degrees from South China University of Technology, Guangzhou, China, in 2003 and 2008, respectively.

He is currently a Full Professor and the Executive Director of the National Engineering Laboratory for Big Data System Computing Technology, China; and also the Vice Dean of the College of Computer Science and Software Engineering, Shenzhen University. He has led three projects for the National Natural Science Foundation; and five projects for the Natural Science Foundation of Guangdong, China. His major research interests include robotics, hybrid systems, the Internet of Things, and embedded systems. He is a fellow of IET. He received the National Science Fund for Distinguished Young Scholars of China in 2023. He was a recipient of numerous awards and honors, including the Wu Wen-Jun Artificial Intelligence Award. He served on the editorial board for seven journals and has been selected for the list of the world's top scientists' lifelong influences by Stanford University.