A Survey on Securing Image-Centric Edge Intelligence

LI TANG, Yunnan University, China
HAIBO HU*, The Hong Kong Polytechnic University, China
MONCEF GABBOUJ, Tampere University, Finland
QINGQING YE, The Hong Kong Polytechnic University, China
YANG XIANG, Swinburne University of Technology, Australia
JIN LI, Guangzhou University, China
LANG LI, Hainan University, China

Facing enormous data generated at the network edge, Edge Intelligence (EI) emerges as the fusion of Edge Computing and Artificial Intelligence, revolutionizing edge data processing and intelligent decision-making. Nonetheless, this emergent mode presents a complex array of security challenges, particularly prominent in image-centric applications due to the sheer volume of visual data and its direct connection to user privacy. These challenges include safeguarding model/image privacy and ensuring model integrity against various security threats, such as model poisoning. Essentially, those threats originate from data attacks, suggesting data protection as a promising solution. Although data protection measures are well-established in other domains, image-centric EI necessitates focused research. This survey examines the security issues inherent to image-centric EI and outlines the protection efforts, providing a comprehensive overview of the landscape. We begin by introducing EI, detailing its operational mechanics and associated security issues. We then explore the technologies facilitating security enhancement (e.g., differential privacy) and edge intelligence (e.g., compact networks and distributed learning frameworks). Next, we categorize security strategies by their application in data preparation, training, and inference, with a focus on image-based contexts. Despite these efforts on security, our investigation identifies research gaps. We also outline promising research directions to bridge these gaps, bolstering security frameworks in image-centric EI applications.

CCS Concepts: • **Computing methodologies** → **Distributed computing methodologies**; **Artificial intelligence**; • **Security and privacy** → **Privacy protections**; **Privacy-preserving protocols**.

ACM Reference Format:
Li Tang, Haibo Hu, Moncef Gabbouj, Qingqing Ye, Yang Xiang, Jin Li, and Lang Li. 2018. A Survey on Securing Image-Centric Edge Intelligence. *J. ACM* 37, 4, Article 111 (August 2018), 34 pages. https://doi.org/XXXXXXX.XXXXXXX

*Corresponding author

Authors' addresses: Li Tang, tangli@ynu.edu.cn, Yunnan University, China; Haibo Hu, haibo.hu@polyu.edu.hk, The Hong Kong Polytechnic University, China; Moncef Gabbouj, moncef.gabbouj@tuni.fi, Tampere University, Finland; Qingqing Ye, qqing.ye@polyu.edu.hk, The Hong Kong Polytechnic University, China; Yang Xiang, yxiang@swin.edu.au, Swinburne University of Technology, Australia; Jin Li, lijin@gzhu.edu.cn, Guangzhou University, China; Lang Li, 1226835507@qq.com, Hainan University, China.

0004-5411/2018/8-ART111 $15.00
https://doi.org/XXXXXXX.XXXXXXX

1 INTRODUCTION

In this transformative era, artificial intelligence (AI) is flourishing at an unprecedented pace and leading a significant technological revolution across various sectors. This revolution has a profoundly positive impact on our daily lives and public services [51]. Especially noteworthy are the advancements in image-centric AI services, which are indispensable in enhancing safety and convenience for personalized services. For instance, obstacle detection technology is vital in autonomous vehicles, capable of identifying obstacles in real-time to prevent accidents and protect lives and property [31, 181]. Similarly, facial recognition technology is widely applied across various sectors, from unlocking phones and authenticating payments to enhancing public safety and border control, improving service efficiency while also strengthening security and regulatory capabilities [88, 119]. It's foreseeable that image-centric AI demands substantial computational power and memory resources to deliver its outstanding services. Conventional image-centric AI services, operating in a server-centric mode, transfer large volumes of image data from the owner to the server, which incurs high latency, communication costs, and security risks.

Empowered by advanced edge devices and edge-friendly AI technologies, image-centric edge intelligence (EI) emerges, shifting the paradigm from server-centric to edge-centric mode [16, 195]. This approach brings intelligent image processing closer to the source, improving responsiveness, efficiency, and privacy. It reduces network strain, enables real-time processing, and enhances security to some extent by keeping images local, all while optimizing edge computing resources in areas with limited connectivity. Despite these advantages, image-centric EI still has a complex spectrum of security challenges [165]. On one hand, collaborative computing in image-centric AI, driven by its computational demands, involves multiple participants and diverse communication, increasing the risk of security breaches. Those participants, often resource-constrained edge devices, may be honest-but-curious or even malicious, potentially compromising privacy and AI services, both actively and passively. The diverse communication also creates more opportunities for outside attackers to exploit vulnerabilities. On the other hand, tailoring security measures to the functional requirements of EI tasks is challenging. For instance, securing federated learning involves mitigating the impact of malicious participants, accommodating the dropout of legitimate ones, and ensuring efficient model update aggregation. Additionally, heterogeneous distributed data further complicates the balance between functionality and privacy protection. Sophisticated security protocols are required to safeguard against multifaceted security risks while meeting the operational demands.

This survey aims to explore the unique security challenges in image-centric EI, offering a detailed overview of the current security landscape and guiding future research directions. There are three main phases in image-centric AI tasks: data preparation, model training, and inference, each with its own deployments and security issues. We investigate security measures based on their application across these phases, with an in-depth discussion of the specific threats that arise in each phase. Our contributions are summarized as follows:

1) We survey recent security measures for image-centric EI and classify them by application phase: data preparation, model training, and inference. For each phase, we discuss the unique threats, especially those intensified by the EI environment and deployment specifics, and categorize the security schemes by their deployment and security techniques.
2) We provide an overview of attack sources and security threats. Specifically, we categorize potential attackers into internal and external groups, each with distinct privileges and attack vectors, and focus on the threats to both data security and model security.
3) We review the enabling technologies for both EI and security protection, offering a clearer understanding of scheme deployments in EI, the additional threats they introduce, and the fundamental security techniques.

4) We discuss the challenges in securing image-centric EI due to inherent security technologies. Specifically, theoretical and experimental results indicate that differential privacy (DP)-based methods achieve the lowest communication and computation costs while balancing privacy and utility; homomorphic encryption (HE) incurs the highest computation overhead but provides strong security with low communication overhead; and secure multi-party computation (MPC) has lower computation overhead but results in increased communication overhead. Additionally, we outline three future opportunities related to security, efficiency, and model utility.

The survey is structured as follows: We begin by introducing image-centric EI, detailing its operational mechanics and security concerns (Section 2). Next, Section 3 investigates technological advancements in security, such as differential privacy, and edge-friendly AI technologies, including compact networks and distributed learning frameworks. Security strategies are then examined separately in the data preparation (Section 4), model training (Section 5), and model inference phases (Section 6), with a specific focus on image-centric AI. We also outline orthogonal security strategies (Section 7) that could complement those in previous sections. Despite ongoing efforts to strengthen data security in this emerging field, we identify continued research gaps and suggest promising directions for future research to address these gaps and enhance security mechanisms in image-centric EI applications (Section 8). Conclusively, we summarize the paper (Section 9).

1.1 Existing Surveys

To the best of our knowledge, a comprehensive survey on securing image-centric edge intelligence has not yet been conducted, despite the publication of several related surveys in recent years. Villar *et al.* [145] review the available and up-to-date frameworks for implementing secure EI, highlighting the most relevant unaddressed gaps. Meurisch *et al.* [112] review data protection approaches for AI services at three levels: management, system, and AI. Alwarafy *et al.* [5] focus on the security and privacy issues inherent in edge computing-assisted IoT systems. They provide a comprehensive overview of these concerns from multiple perspectives, including the challenges, attacks and threats, as well as the countermeasures and solutions designed to address them. Singh *et al.* [136] discuss security and privacy issues across various layers of the EC architecture, focusing on challenges arising from the networking of heterogeneous devices. The main emphasis is on countermeasures implemented through machine learning and deep learning to address various attacks, such as DDoS, eavesdropping, malware, and more. Wang *et al.* [147] examine additional security and privacy issues introduced by resource-constrained and security-vulnerable devices, security vulnerabilities in RAN communication, and foundational MEC technologies like software-defined networking, all in the context of MEC. They particularly review AI-based approaches to address these challenges. Liao *et al.* [94] focus on the application of blockchain for security and forensics management in MEC-IoT systems. Specifically, they analyze these blockchain-based secure approaches from three aspects: device security, data security, and IoT forensics. AI-Doghman *et al.* [3] provide a comprehensive survey on securing edge computing-based AI microservices, focusing on IoT management and secure decision-making at the edge. They highlight key security requirements and challenges and propose a framework for secure edge AI algorithms utilizing containerization technology.

2 BACKGROUND

In this section, we begin with an overview of EI, followed by its operational mechanics, threat model, and attacks.

2.1 Scope of Edge Intelligence

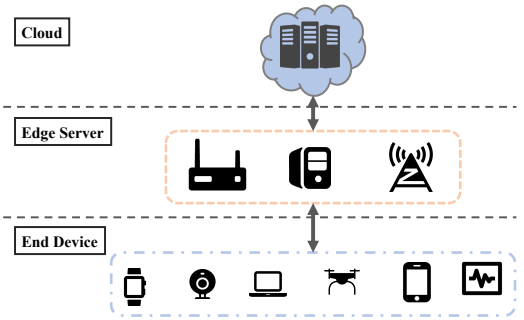

Fig. 1. System Model.

2.1.1 System Model. As shown in Figure 1, EI consists of three main entities: end devices, edge servers, and cloud. End devices, located at the very network edge, typically serve as the primary sources for data generation and the initiators of AI tasks. With the progression of hardware technology, these devices are equipped to handle some simple tasks directly. This means they can quickly process data using the power available at the edge, making data handling more immediate and efficient. Edge servers, positioned at the network edge, are potent computing platforms equipped with superior processing power and abundant storage. They oversee end devices within a certain area, providing services like computation and storage to these devices. This work mode allows them to handle complex tasks timely and efficiently on-site. Although EI emphasizes computation at the network's edge, cloud servers remain integral to the system's framework. Their necessity stems from several key functions: First, equipped with robust resources, they can handle extensive tasks like large-scale model training, offering strong backend support to complement EI. Furthermore, cloud servers monitor the status of edge servers, dynamically allocating tasks and resources to enhance performance and swiftly replacing any non-functional or malicious units to maintain system stability and integrity.

2.1.2 Work Mode. In EI scenarios, the limited resources necessitate a collaborative computing approach. This is illustrated in Figure 2, which shows that an end device might offload either all or part of its computational tasks to one or several entities. When involving multiple entities, this collaborative approach functions similarly to a pipeline or parallel processing system from the perspective of the end device. These collaborative computing modes are notably applicable across all pivotal phases of an AI application: data preparation, training, and inference.

(1) Data Preparation. This phase encompasses various processing operations such as denoising, feature extraction, and other tasks crucial for preparing data for training. The focus is on enhancing the quality and readiness of the data, a critical prerequisite for effective model training. Typically, there are two common processing modes: (a) remote processing [108, 194], where data is sent to an external server for processing, and (b) distributed processing [86, 134], which is suitable for processing distributed data and can be integrated into distributed learning. Distributed processing allows for local processing and necessitates collaborative efforts to optimize model generation.

(2) Model Training. During this phase, the AI model is trained to identify patterns and relationships within the prepared data. The essence of this phase is the iterative adjustment of the model's parameters to minimize prediction errors. In EI, where computational and storage resources may be limited and data is often distributed across multiple nodes, training the AI model presents significant challenges. To address these resource limitations and optimize resource utilization, the training phase can be managed through: (a) remote training, where the training is partially or entirely outsourced to a dedicated server [116], or (b) distributed training [153, 167], which involves

multiple dedicated entities and enables them to contribute to improving training performance by sharing model updates or gradients.

(3) Inference. In the inference phase, when fed with new, unseen data, the trained model applies the learned patterns to generate actionable insights or support decision-making. Data owners and model owners are often distinct entities, each with a primary concern for protecting their respective assets. Typically, there are three common inference modes: (a) remote inference [11, 177], where the data is sent to the model owner for predictions; (b) model deployment [55], where the model is privately deployed at the data owner's site for local inference; and (c) collaborative inference [65, 102], where both the model owner and the data owner interact during inference without exposing their sensitive information to each other.

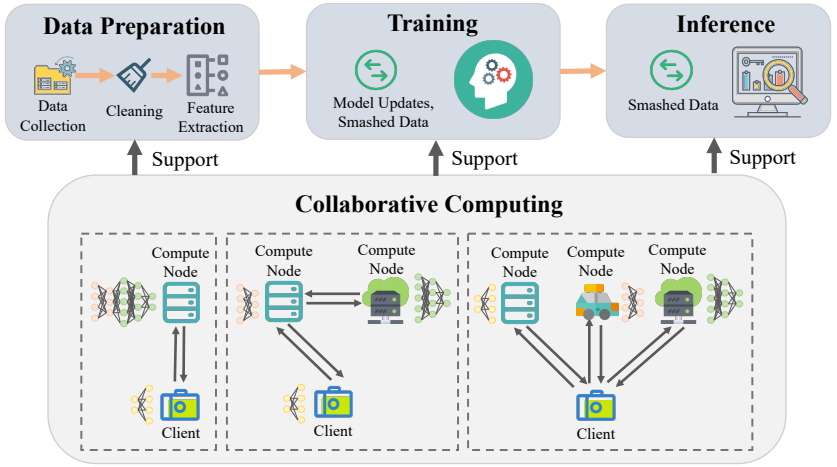

Fig. 2. Work Mode [1].

2.2 Threat Models

1) Internal Attackers. In EI, AI tasks frequently involve multiple participants, some of whom may be attackers. Based on the nature of their attacks, these internal attackers can be categorized into 'honest-but-curious' and malicious types. An 'honest-but-curious' participant adheres to the established protocol but tries to access private information not meant for them, like others' datasets or model parameters, aiming to gain benefit. For instance, in a federated learning scenario, an 'honest-but-curious' participant could analyze shared gradients to deduce sensitive data attributes of others [36, 67], potentially gaining competitive insights or indirectly improving their models. On the other hand, a malicious participant actively compromises privacy and data integrity, possibly for direct financial gain, sabotage, or espionage. They might introduce false data or manipulate the learning process to skew results in their favor. For instance, by poisoning the training data [115] or model updates [7], a malicious participant could degrade the model's performance on specific tasks or ensure it behaves in a predictably incorrect manner under certain conditions, thereby undermining the collaborative effort and gaining an unfair advantage.

2) External Attackers. AI tasks in EI typically necessitate frequent data exchanges; these transmissions become prime targets for external attackers. Specifically, attackers may undermine data privacy with strategies like eavesdropping and covertly monitoring private network communications. Moreover, external attackers can actively manipulate transmitted data by injecting false

[1]The logic depicted in this figure is adapted from [168].

information or altering existing data. These actions directly threaten data integrity, distort AI model training outcomes, and ultimately compromise AI services.

2.3 Attacks

In this subsection, we shift our focus to the primary attacks targeting input data and models, emphasizing the perspective of the objects rather than the methods employed.

2.3.1 Data Stealing. Data stealing refers to attackers attempting to obtain sensitive information about training data or inference data through various attacks. Common attacks include membership inference attack [170] and model inversion attack [179]. The former aims to determine whether a specific data sample was included in the training process by analyzing the model's predictions, ultimately revealing the makeup of the training dataset. The latter is designed to extract sensitive details of the input data and potentially reconstruct the input based on the outputs of the model. These attacks can be problematic, especially in scenarios where inputs reveal sensitive information about an individual. For instance, reconstructed facial images in facial recognition systems may expose personal details and lead to identity theft.

In traditional centralized AI scenarios, these attacks mainly occur during the inference stage, as training on the server is usually assumed to be trusted. However, in EI, which typically involves multiple participants and various transmissions, these attacks can also occur during the training phase. Furthermore, unlike attacks in traditional centralized AI scenarios, which are typically conducted in a black-box manner, attacks in EI settings may take place in gray-box [93] or even white-box [191] modes, presenting more serious threats.

2.3.2 Model Stealing. Model stealing aims to acquire sensitive information about the model, while model extraction is a specific type of attack that often occurs during the inference stage. In model extraction, attackers attempt to replicate a proprietary model's functionality without access to its underlying architecture or training data. This is typically achieved by querying the target model with carefully crafted inputs, collecting the input-output pairs, and then using this data to train a new model that mimics the behavior of the original model. While such attacks primarily target the inference stage, security risks increase during the training phase in EI, where model parameters like gradients may be exchanged, providing attackers with a direct pathway to steal model details [29, 35].

2.3.3 Data Poisoning. Data poisoning involves introducing malicious samples into the training dataset, aiming to mislead the model into learning incorrect patterns or biases, ultimately reducing its accuracy [45, 115]. This attack can be carried out by both external and internal attackers. Adversarial attacks [157] can be viewed as a form of data poisoning that occurs during inference. In these attacks, adversarial examples are crafted to deceive the model by exploiting its vulnerabilities, leading it to make incorrect predictions without directly altering its internal parameters.

2.3.4 Model Poisoning. Model poisoning aims to induce the model to generate inaccurate predictions or behave undesirably under certain conditions. For instance, a backdoor attack [7], a subtype of model poisoning, misleads the model during training to produce a specific incorrect output when it encounters a certain pattern while functioning normally for other inputs. Traditionally, model poisoning is often done through data poisoning by adding malicious samples to the training dataset [45, 115]. However, in more complex training scenarios within EI, attackers may directly manipulate the exchanged model parameters to achieve their objectives [7, 87]. These attacks threaten the model's integrity and can lead to security concerns such as data leaks or information theft.

3 ENABLING TECHNOLOGIES

In this section, we present the enabling technologies for security and edge intelligence.

3.1 Enabling Technologies for Security

EI commonly employs secure schemes based on Differential Privacy (DP), Homomorphic Encryption (HE), and Secure Multi-Party Computation (MPC).

3.1.1 Homomorphic Encryption. Homomorphic Encryption (HE) is a cryptographic technique enabling computations on ciphertexts, yielding an encrypted result that, once decrypted, matches the operation's outcome on the plaintext. This property is characterized by the equation below:

$$Enc(a) \oplus Enc(b) = Enc(a \oplus b) \tag{1}$$

where $Enc : x \rightarrow y$ denotes a HE scheme that transforms plaintext x into ciphertext y, and $\oplus$ denotes the applicable homomorphic operation, such as addition and multiplication.

Regarding the operations they support, HE schemes are categorized into Partially Homomorphic Encryption (PHE) [130], which allows either addition or multiplication; Somewhat Homomorphic Encryption (SHE) [144], which permits a finite sequence of additions and multiplications; and Fully Homomorphic Encryption (FHE) [37], which enables unrestricted additions and multiplications. Typically, computational demands increase progressively from PHE through SHE to FHE.

3.1.2 Secure Multiparty Computation. Secure Multi-Party Computation (MPC) enables multiple participants to collaboratively calculate a function using their inputs without revealing them to each other. Its mathematical definition is as follows.

Let f be the object function and $P_1, P_2, ..., P_n$ denote the involved n participants, each holding inputs $x_1, x_2, ..., x_n$. In MPC, participants collaborate to compute $f(x_1, x_2, ..., x_n)$ while simultaneously ensuring that no paritipant's private input is composed to other participants.

This technique utilizes secret sharing most commonly, involving dividing each participant's input into pieces, distributing them among all parties for joint computation, and then merging the results to produce the final outcome. Secret sharing-based MPC is computationally efficient, yet it inherently exhibits heavy communication costs

3.1.3 Differential Privacy. Differential Privacy (DP) has wide applications [27, 28, 169]. It safeguards individual data by introducing noise while retaining statistical information, ensuring that the analysis results remain unaffected. This can be achieved through various methods, such as Laplace noise, Gaussian noise, the exponential mechanism, and additive noise. The typical mathematical definition is as follows.

A mechanism A is (ϵ, θ)-differentially private if, all pairs of neighboring dataset D and D' that differs by the inclusion or exclusion of one data sample, and for all subsets S of the range of A, the following inequality holds:

$$Pr[A(D) \in S] \leq e^{\epsilon} \cdot Pr[A(D') \in S] + \theta \tag{2}$$

where ϵ is the privacy budget parameter which is negatively correlated with privacy level, and θ is a parameter that loosens the bound of deviation.

3.2 Enabling Technologies For Edge Intelligence

3.2.1 Compact Networks. Numerous initiatives have been undertaken to develop compact network designs that excel in computational efficiency, minimize memory requirements, and even reduce power consumption. These features make them ideal for use in resource-limited environments like mobile devices, embedded systems, and edge computing platforms. Some studies [39, 61] adopt

"bottleneck" approach, where a squeeze stage reduces dimensions or channels to make the model more compact, and an expansion stage restores them to maintain or improve accuracy. Some works [22, 56, 133] use depthwise separable convolutions to create compact network architectures. This technique separates the convolution into a depthwise layer, which independently processes each channel to reduce computation and model size, and a pointwise layer, which combines the results across channels to generate new features. ShuffleNet series [106, 190] adopt pointwise group convolution to ensure compactness and channel shuffle to enhance inter-group feature combination.

3.2.2 Model Compression. Model compression facilitates local computing by streamlining existing models, achieving a balance between reduced complexity and minimal accuracy loss. Here, we provide a brief overview of the advancements in model compression techniques.

a) Pruning. Pruning strategically removes unimportant or redundant parameters (e.g., weights, filters, neurons) from a pre-trained network and fine-tunes the remaining parameters to obtain a compact model without noticeable accuracy loss. This approach maximizes storage and computational efficiency while preserving accuracy. The pruning granularity and the metrics used to assess parameter significance are key to achieving this goal. Regarding granularity, pruning can be applied to various levels, such as individual weights [44, 69, 98], filters [49, 50, 103], and neurons [25, 180]. Weight pruning creates a sparse matrix for model compression but needs specialized hardware or software, while filter pruning reduces model depth or width without such dependencies, and neuron pruning is mainly suited for fully connected layers. Regarding importance evaluation, several key metrics have been proposed, including those based on magnitude [46, 47, 62], sensitivity [26, 82], energy-aware [173, 178], and information theory [25, 34]. Magnitude-based pruning primarily targets individual weights and speeds up convergence, sensitivity-based pruning preserves accuracy by evaluating parameter impact on the cost function, and energy-aware pruning reduces energy consumption by focusing on the most energy-intensive model parts.

b) Quantization. It has been noted that high precision in parameters is often unnecessary for optimal performance, especially when certain data have little effect on the outcome. This has led to the use of data quantization, which reduces bit usage to improve computational efficiency and lower storage costs. Depending on when the quantization is applied, it can be classified into post-training quantization (PTQ) [48, 73, 164] and quantization-aware training (QAT) [23, 101, 117]. PTQ is highly practical, requiring minimal calibration data and no retraining, but it may cause some accuracy loss, making it ideal for scenarios with limited resources and lower accuracy demands. On the other hand, QAT integrates quantization directly into the training process, allowing the model to adapt to quantized weights and activations. This approach enables more aggressive quantization while maintaining accuracy, making it ideal for high-precision applications.

c) Knowledge Distillation. In knowledge distillation, knowledge is transferred from a large, complex model (the teacher) to a smaller, simpler model (the student), aiming to retain important aspects of the teacher model's performance in a more lightweight student model. Knowledge can be categorized as response-based, feature-based, relation-based, etc. Response-based knowledge distillation [17, 185] aims to mimic the teacher's predictions, i.e., output of the last layer. This method leverages the rich informational content inherent in the teacher model's output. For instance, soft targets provide valuable insights into the relationships between different classes and the confidence levels of the predictions. This form of knowledge distillation is notably straightforward and has been successfully applied in various domains, including object detection [17], semantic landmark localization [185], and image classification [54]. However, it overlooks the potential benefits of intermediate-level supervision, which limits its effectiveness in knowledge distillation. Conversely, feature-based knowledge distillation [52, 120] supervises the student model using both intermediate feature maps and final outputs. The process involves matching the features of the teacher and student

models, either directly or indirectly. Features can be matched directly through activations [131] or indirectly through knowledge derived from these features, such as probability distributions [120] and activation boundaries [52]. Relation-based knowledge distillation [83, 121] extends beyond focusing on the output of specific layers; it explores inter-layer or inter-sample relationships, with a crucial focus on the relationship extraction method.

d) Operational Neural Networks. Some works recently introduce Generalized Operational Perceptrons (GOPs) [74, 75, 141–143] to model biological neurons with distinct synaptic connections while ensuring compactness, differing from the approach in [85]. Reflecting the diversity found in biological neural networks, GOPs have achieved excellent performance on challenging problems such as the Two-Spirals and N-bit parity problems [74]. GOPs are now considered state-of-the-art (dense) ANN model [141–143]. Following GOPs footsteps, the authors introduced Operational Neural Networks (ONNs) [76, 77, 109], which not only significantly outperform CNNs but also excel in solving problems where CNNs fail entirely. However, like GOPs, ONNs also have drawbacks, including reliance on a predefined operator set, the need to search for the best operator set for each layer or neuron, and the requirement to fix operator sets for output layer neurons in advance. These limitations reduce network heterogeneity and diversity, affecting learning performance and computational efficiency. More recently, Self-organized ONNs (Self-ONNs) [78, 110] address these drawbacks by using a generative neuron model that avoids prior operator search or training. Each generative neuron customizes operators for kernel connections, achieving greater heterogeneity than ONNs and converting "weight optimization" into "operator generation".

"Generative Adversarial Networks" [43] have been introduced to learn from real-world datasets to create new data instances such as images. In the Computer Vision domain, GANs have been successfully employed for many tasks, including face generation [70], image super-resolution [81], and style transfer [197]. Most importantly, GANs can generate synthetic data in fields like biomedical informatics [80], where large-scale data annotation is costly and impractical. However, the practical use of GANs faces challenges, such as training stability [6], the need for large datasets [71], complex architectures [10], and substantial computational resources [71]. Kiranyaz *et al.* [79] address these limitations with the super-neuron model-based neural networks. This model optimizes non-linear transformations with dynamic, adaptable kernels, which adjust both the optimal location for synaptic operations and the receptive field size.

3.2.3 Model Partitioning. Model partitioning is a common strategy to accelerate inference tasks by distributing them across different computing units in a sequential manner, where the output of one unit becomes the input for the next. To optimize performance, the sequence must be carefully managed to ensure efficient data flow and minimize latency, considering each unit's resources and the interdependencies between model components. The key focuses of model partitioning are (i) identifying the optimal model partition and (ii) minimizing latency. Hu *et al.* [57] propose a Dynamic Adaptive DNN Surgery scheme that minimizes processing delay under light loads and maximizes throughput under heavy loads, addressing network dynamics and partition complexity. Almeida *et al.* [4] propose a data packing approach that adjusts precision levels for different CNN segments and introduces a scheduler to optimize both partition points and precision, improving model adaptability and efficiency. Yang *et al.* [172] optimize CPU usage and reduce latency by fine-grained partitioning of the model both horizontally and vertically. Zhang *et al.* [189] present Fully Decomposable Spatial Partition, a method that seamlessly accommodates resource diversity and network dynamics, complemented by a compression strategy that reduces bandwidth costs.

3.2.4 Distributed Training. Distributed Learning [153, 167] refers to spreading a training task across multiple computing units, facilitating efficient handling of sophisticated models and extensive datasets. It markedly accelerates the training process by utilizing parallel computing resources,

where the parallelization can occur either at the model level or the data level, i.e., model parallelism and data parallelism.

1) Model Parallelism. Model parallelism [13, 90] involves segmenting the model into multiple segments and allocating them to different computing units for collaboratively training. This approach necessitates precise coordination to handle the interdependencies among the segments.

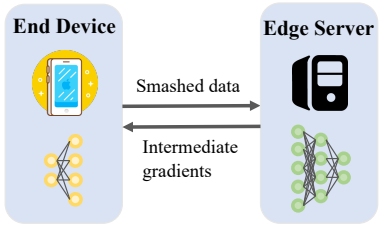

Fig. 3. Split Learning

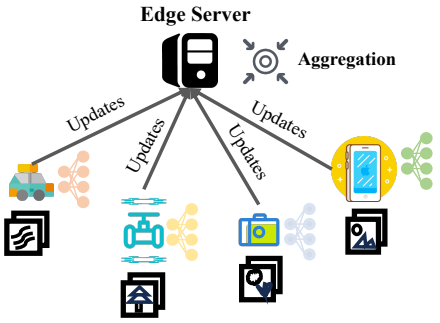

Fig. 4. Federated Learning

a) Split Learning. Split learning (SL) [124, 160, 163] is a model parallelism technique that enhances privacy and efficiency by keeping training data on-site and partitioning a neural network into sub-networks at cut layers. Distinct entities, such as clients, edge servers, and the cloud server, train these sub-networks separately. Figure 3 shows the simplest structure of SL and its training proceeds as follows: The client transmits the output (smashed data) of its sub-network to the server, which performs forward propagation, computes the loss, and starts backward propagation. The server then sends intermediate gradients back to the client, which continues backward propagation to update its sub-network. This iterative process repeats until training is complete. In scenarios with multiple data owners, they collaboratively train the client-side subnetwork like in federated learning, while the server updates the global model.

2) Data Parallelism. Data parallelism [8, 66, 91] typically splits the dataset into smaller chunks and distributes these across various computing units. Each unit trains the model on its chunk and uploads the intermediate training results to a central unit. The central unit then aggregates the uploaded results and updates the central model. In traditional data parallelism, the dataset is usually centralized or shared among the units, meaning all chunks follow the same distribution.

a) Federated Learning. Federated learning (FL) [60, 161, 184, 187] is a specialized distributed learning mechanism where each unit holds local data samples without needing to exchange or centralize these data. In FL, the training data at each unit are typically unbalanced and non-Independent and Identically Distributed (non-IID). The primary benefits of FL encompass enhanced data security and privacy, along with the efficient utilization of distributed data resources. By utilizing data locally and only sharing model updates (as shown in Figure 4) across the network, DL addresses concerns related to data breaches and unauthorized access. Through collaborative training, FL maximizes the utilization of distributed computing resources and bandwidth. A typical FL process works as follows: Initially, a central master (server or edge device) initializes a global model and distributes it to the slaves, i.e., the edge devices. Each slave then trains the model locally with its data and sends its model updates—often in the form of gradients or updated weights, rather than raw data—back to the master. The master aggregates these updates using a simple average or a more complex algorithm to update the global model, which is then sent back to the slaves for further training. This process continues until the model achieves the desired performance.

4 SECURE EDGE DATA PREPARATION

In this section, we outline the related security issues in the data preparation phase. Firstly, we introduce the primary attacks, followed by a discussion on secure schemes.

4.1 Attacks

As mentioned in Section 2.1.2, data preparation involves organizing and refining data to improve its quality and readiness for model training or inference. However, since data is often generated and stored by resource-constrained entities, it usually needs to be processed externally, which increases the risk of data theft and poisoning.

4.1.1 Data Poisoning. During the data preparation phase, data poisoning primarily occurs by injecting harmful samples into the training data. These malicious samples can introduce incorrect patterns, biases, or even hidden backdoors into the model, ultimately reducing accuracy or controlling behavior during inference. This threat is especially severe in EI, where data from multiple sources makes detecting and preventing poisoning attacks more challenging, increasing risks to system reliability and security.

4.1.2 Data Stealing. In the data preparation phase, data stealing is particularly serious because transmitted data can directly reveal sensitive information to adversaries without requiring further analysis. This risk is heightened in EI environments, where frequent data transmission between entities increases opportunities for interception and unauthorized access.

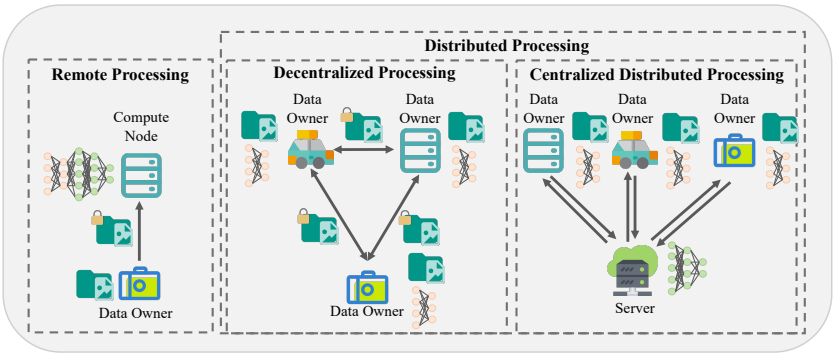

Fig. 5. Taxonomy of Edge Data Preparation.

4.2 Taxonomy of Secure Schemes

As illustrated in Figure 5, the data preparation process can be categorized into remote processing and distributed processing. Table 1 provides a summary of the corresponding secure schemes.

4.2.1 Secure Remote Processing. Data preparation for image-centric AI tasks encompasses various image processing operations, some of which, like denoising and feature extraction, can be computationally intensive and may require offloading. In such scenarios, protecting data privacy when delegating image processing tasks to an external server is crucial. This necessitates the use of security measures such as encryption, secure multi-party computation, and differential privacy to prevent unauthorized access and maintain data confidentiality.

4.2.2 Secure Distributed Processing. AI tasks in EI typically involve non-identically distributed (non-IID) data, which can lead to local optima issues in distributed learning. Meanwhile, while participants contribute their data to enhance model performance, they still prioritize data privacy. In

this context, the challenge of collaboratively processing data, with or without a central coordinator, to optimize model generation while ensuring local data privacy has garnered significant attention.

Table 1. Literature for Securing Edge Data Preparation

Scenario	Method	Ref.	Processing	Highlights	Effectiveness
Remote Processing	Perturbation	[194]	denoising	one-way trapdoor permutations, shape context-based denoising.	verifiable denoising, $O(1)$ complexity.
	HE	[192]	denoising	garbled circuits for computations, similar patches for denoising.	denoising quality nearly matches the optimal performance in plaintext.
		[108]	feature extraction	homomorphic matrix multiplication for parallel computing, accurate vector normalization	1.9× speedup on 200 × 256 datasets, 25× speedup on 60000 × 256 datasets, 0.285 R2 accuracy improvement.
		[64]	feature extraction	aligns descriptors by adjusting extremum points' locations and eliminating unstable points.	high computing efficiency
		[171]	feature extraction	private linear-transforming matrices, encrypted kernel matrix construction.	high accuracy with lower overhead on computation and communication.
	MPC	[193]	denoising	DNN-based denoising, garbled circuits for computation, protect both image and DNN model.	near plaintext-level denoising quality, cost-efficient local processing and affordable cloud-side costs.
		[92]	feature selection	3PC model with an honest majority, Gini impurity for feature ranking.	tolerate at most one malicious server, accuracy improvement of classifiers, several seconds to an hour runtimes.
		[100]	feature extraction	comparison protocol, simplify orientation assignment by median approximation.	64 rounds of inter-server interaction.
		[59]	feature extraction	trusted party for random generation, building blocks for computations, secure interaction protocols for accurate CNN layer execution.	no approximation, notably lower latency and overhead.
		[12]	feature extraction	secure interaction protocols	obviously computation reduction
Distributed Processing	Perturbation	[86]	feature extraction	Gaussian stochastic dual-gate for selection probability estimation, local perturbation for privacy.	optimal global model with higher accuracy and fast convergence.
	HE	[188]	data selection	hierarchical data sample selection, pre-selection and dynamic selection.	protect local data privacy and the server's label set privacy
	MPC	[134]	feature extraction	horizontally partitioned datasets, local private feature extraction, distributed data for learning.	highly-unbalanced training data, accelerate training time, increase network accuracy.
	Blockchain	[100]	feature fusion	smart contracts for collaboration, feature-fusion based model-fusion,	run on untrusted distributed nodes, train on decentralized data.

4.3 Secure Remote Processing

Certain tasks like denoising, feature selection, and feature extraction often require substantial computational resources and are typically outsourced in EI, raising data privacy concerns. Next, we discuss efforts to develop privacy-preserving processing systems.

1) *Permutation.* Some approaches use permutations to protect image privacy during processing by a delegated server. The main challenge is balancing data utility with privacy. Zhou *et al.* [194] employ one-way trapdoor permutations to ensure privacy and use a shape context-based denoising technique within their outsourced, verifiable, privacy-preserving denoising framework.
2) *HE.* Some works use HE to encrypt images, allowing them to be processed by a delegated server without decryption. The primary challenges researchers face in this area are reducing

the computational overhead and enabling complex computations on encrypted data to support tasks such as denoising, feature selection, and feature extraction. Zheng *et al.* [192] use garbled circuits to compute on HE encrypted data, facilitating privacy-preserving denoising. Specifically, this approach allows the server to search its encrypted database for similar patches and utilize the inherent redundancy for more effective denoising. Ma *et al.* [108] address the challenges of efficient covariance matrix computation and accurate homomorphic vector normalization in outsourced privacy-preserving PCA feature extraction. They introduce tailored optimized homomorphic matrix multiplication techniques for parallel computing, along with a novel PowerMethod circuit that incorporates a universal vector normalization strategy. These advancements enhance the efficiency, accuracy, and practicality of PCA on encrypted data. Jiang *et al.* [64] propose a privacy-preserving SIFT feature extraction scheme using an innovative encoding method and Leveled Homomorphic Encryption (LHE). They develop four protocols, including homomorphic comparison and division algorithms, to handle SIFT's complex operations. This scheme aligns SIFT descriptors with their originals by adjusting extremum points and removing unstable points. Yang *et al.* [171] employ vector homomorphic encryption (VHE) to ensure secure feature extraction from encrypted images, maintaining user privacy. Their approach involves designing private linear-transforming matrices to efficiently extract HOG feature vectors and construct encrypted kernel matrices for SVM-based pedestrian detection, all directly on encrypted images.

3) *MPC.* Some works utilize MPC to securely outsource data processing to multiple servers. Key challenges in this area include ensuring security against specific malicious threats and enabling complex computations on encrypted data for tasks like denoising, feature selection, and feature extraction. Zheng *et al.* [193] adopt a two-server architecture with semi-honest, non-colluding servers. They use additive secret sharing to encrypt both images and DNN models, ensuring privacy on the servers, and employ garbled circuits for privacy-preserving DNN-based denoising. Li *et al.* [92] use a 3PC model with an honest majority, tolerating up to one malicious server. They apply Gini impurity, a filter method that ranks features based on their predictive ability, to achieve outsourced privacy-preserving feature selection. Liu *et al.* [100] use two non-colluding but untrusted servers for secure SIFT feature extraction. They design a comparison protocol that allows the servers to securely compare two absolute values and simplify orientation assignment using a median value for approximation. Their scheme supports all SIFT operations with low communication overhead, requiring only 64 rounds of server-to-server interaction for feature extraction, demonstrating its efficiency. Huang *et al.* [59] propose a privacy-preserving CNN feature extraction framework involving two semi-honest, non-colluding servers, and a trusted party for random value generation. They create building blocks for efficient secret-sharing computations and design protocols to securely execute essential CNN layers—Convolutional, ReLU, Pooling, and Fully-Connected—without approximations, ensuring accuracy. Similarly, Cai *et al.* [12] outsource CNN feature extraction to two semi-honest, non-colluding servers. Their design includes secure interaction protocols for mixed multiplication, comparison, and addition to reduce communication and computational overhead.

4.4 Secure Distributed Processing

In EI, data is typically distributed across multiple nodes, leading to the development of distributed learning techniques. Consequently, some works focus on processing data in a distributed manner.

1) *Permutation.* Some methods use permutations to protect local data privacy while facing the challenge of balancing data utility with privacy. Li *et al.* [86] integrate privacy-preserving feature selection into the vertical federated learning process, rather than treating feature selection and

learning as separate steps. Specifically, they design FedSDG-FS, which uses Gaussian stochastic dual-gate to efficiently estimate the selection probability of each feature and employs local perturbation techniques aligned with differential privacy to protect data privacy.

2) *HE.* To enable private and collaborative training across distributed entities, some works utilize HE to protect privacy. Key challenges in this area include reducing HE's computational overhead and enabling complex computations on encrypted data. Zhang *et al.* [188] propose an efficient hierarchical data sample selection mechanism for federated learning that protects both local data privacy and the server's label set privacy. This mechanism employs private set intersection and the determinantal point process to select clients before training and uses an erroneous-aware importance-based method to dynamically choose important clients and samples during training.
3) *MPC.* Given multiple entities, some approaches integrate MPC into distributed processing to safeguard local privacy. The primary challenge in this area is collaboratively optimizing processing to improve model performance. Sarmadi *et al.* [134] develop a privacy-preserving feature extraction method for horizontally partitioned datasets, aimed at collaborative model training. This technique involves each data owner, i.e., participant, extracting local features via her or his local private feature extractor, encrypting these features with secret-sharing technology, and distributing the encrypted data among participants. The parties then combine their own and received encrypted features for distributed learning, promoting both data privacy and collaborative model development.
4) *Blockchain.* Some approaches use blockchain technology to share data among entities, addressing the challenges posed by distributed data. Liu *et al.* [96] employ a blockchain-based feature fusion strategy in TDLearning to address the distributed data challenge in collaborative learning, simultaneously mitigating privacy concerns inherent in direct data sharing. This method reduces privacy risks associated with user data sharing, though it still permits the exposure of certain sensitive information.

5 SECURE EDGE TRAINING

In this section, we outline the key security concerns during the training phase. We first present the primary attacks and provide a taxonomy of secure schemes. We then offer a detailed discussion of each secure scheme.

5.1 Attacks

In edge training, training tasks are often delegated, either partially or entirely, to one or more parties. That is, the training process involves extensive transmission of training data or model updates among multiple participants with varying levels of resistance to attacks. In such a scenario, attackers have increased opportunities to attack communication channels and compromise participants with weaker resistance, enabling them to steal or poison training data or models. In the subsequent sections, we outline the key attacks as follows.

5.1.1 Model Poisoning. Malicious actors, whether insiders or outsiders, can poison the model by either introducing strategically crafted training samples [45, 115], altering model updates [7, 87] during the training process, or modifying the model's parameters post-training. Backdoor Attack is a typical model poisoning approach, designed to mislead the model to produce a predetermined incorrect output when encountering input that contains a specific pattern while functioning normally for other inputs.

5.1.2 Model Stealing. In certain distributed learning scenarios, such as federated learning, model updates are transmitted among multiple parties, creating increased opportunities for malicious outsiders to access model parameters. These outsiders may attempt to intercept these updates

during transmission or exploit resource-constrained participants, which are more susceptible to compromise. In some other distributed learning scenarios, such as split learning, intermediate activations are transmitted to the honest-but-curious server. The server can exploit the intermediate activations to steal the functionality of the client-side model via various attacks [29, 35].

5.1.3 Data Stealing. Throughout the training process, both gradients and various model outputs, such as intermediate activations and final outputs, are susceptible to exploitation for the training data reconstruction. Based on the leveraged information, attacks are classified into two main categories: model inversion attacks and gradient inversion attacks.

- *Model Inversion Attack.* It is designed to uncover sensitive input data and potentially reconstruct the input based on the models' outputs, including intermediate activations, final results, etc. The model inversion attack introduced by Zhang *et al.* [191] operates in a white-box context, where adversaries possess access to the target network along with some supplementary knowledge. Erdougan *et al.* [29] introduced a gray-box model inversion attack where the attacker server, despite lacking access to the client-side network, understands its structure, allowing them to infer model parameters and reconstruct input data during split learning. Gao *et al.* [35] propose PCAT, a potent attack capable of reconstructing client input data without requiring any knowledge of the model.
- *Gradient Inversion Attack.* This attack exploits gradients exchanged during training to infer private training data. Most gradient inversion attacks operate in a white-box manner [36, 67, 118], where the adversary is assumed to be either the client or the server. These inside attackers not only have access to gradients (from other clients or the aggregated gradients) but also possess internal knowledge of the model. Liang *et al.* [93] propose EGIA, a gradient inversion attack that allows external attackers to exploit dummy gradients obtained from querying a server with dummy inputs, even without access to the model's internals.

5.2 Taxonomy of Secure Schemes

As shown in Figure 6, edge training can be categorized into remote training and distributed training. We summarize the related secure schemes in Table 2.

5.2.1 Secure Remote Training. Within numerous learning frameworks, training data must be transmitted to another entity for the execution of learning tasks, which serves to reduce the computational load on the data owner. In particular, the term "training data" encompasses both unprocessed raw data and derived feature representations. Given the close association between training data and user privacy, significant measures are undertaken to ensure their security.

5.2.2 Secure Distributed Training. Many distributed learning schemes, especially federated learning, involve transmitting and aggregating model updates. However, if these updates are stolen, it can compromise the privacy of both the training data and the model. Moreover, injecting malicious or manipulated updates can undermine the model's integrity. In such a scenario, secure distributed learning schemes are indispensable.

5.3 Secure Remote Training

Due to resource constraints, end devices often outsource data for model training, prioritizing the protection of training data privacy. We outline the main approaches as follows.

1) *DP.* Nasr *et al.* [116] analyze Differentially Private Stochastic Gradient Descent by modeling an adversary's ability to distinguish between similar datasets. They establish that the lower bounds on privacy risk closely match the theoretical upper bounds, suggesting that differential privacy offers conservative yet effective protections, with actual data leakage risks being lower under

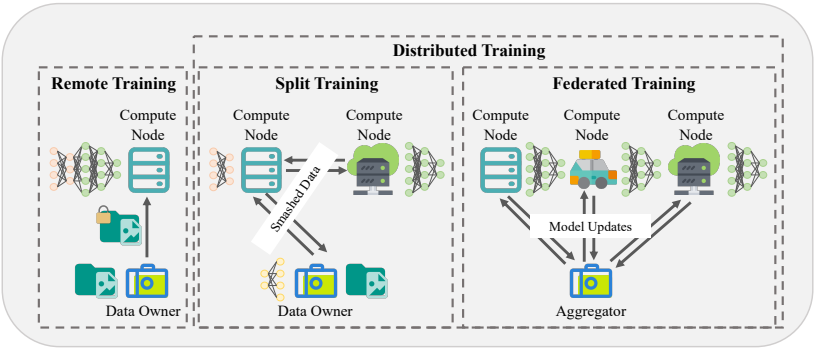

Fig. 6. Taxonomy of Edge Training

realistic adversary limitations. Lyu *et al.* [105] introduce FORESEEN, a mechanism for noisy training that ensures fast, private, and accurate inference at the edge. It perturbs features using random projection, noise addition, and data nullification, and uses mixed-precision deep models to meet IoT devices' accuracy and resource constraints.

2) *HE.* Gonzalez *et al.* [42] use a partial homomorphic cryptosystem to encrypt training data and outsource it for training. They tackle challenges in SVM training and testing on encrypted data by converting integer-valued data with a specific solver and preventing privacy leakage using a semiparametric SVM model with data prototypes not in the training set. Hesamifad *et al.* [53] introduce CryptoDL for secure neural network training on untrusted servers using HE. They provide a theoretical basis for approximating activation functions like ReLU, Sigmoid, and Tanh with low-degree polynomials, minimizing accuracy loss.
3) *MPC.* MPC enhances privacy by allowing multiple parties to compute jointly without accessing raw data. The primary challenge of this approach lies in ensuring the various computations on diverse data types. SecureML [114] uses this approach where encrypted datasets are uploaded to two servers, which collaboratively train the model while learning only the model parameters. It supports both linear and nonlinear functions and improves efficiency by representing shared numbers as integers and using pre-generated multiplication triplets.

5.4 Secure Distributed Learning

Distributed learning, particularly federated learning, typically entails the transmission and aggregation of model updates. The exposure of such updates can compromise sensitive information regarding not only the training data but also the model parameters. Malicious updates can directly undermine the model. In this part, we introduce various strategies to secure model updates, i.e., secure distributed learning.

1) *DP.* Mao *et al.*[111] enhance privacy in collaborative training by combining differential privacy with model partitioning, placing the first convolutional layer on the user's device and the rest on the edge server. The pivotal strategy involves introducing noise to the device's output with differential privacy technologies before it reaches the server. Notably, they provide insights into the best partitioning strategy based on tailored metrics, such as privacy loss. Geyer *et al.* [38] employ a randomized mechanism involving random sub-sampling of participants and distorting the sum of updates, concealing clients' contributions during training. Abadi *et al.* [1] enhance the security of training data by applying differential privacy, introducing perturbations to parameters prior to their update. Their research stands out by specifically targeting the effective management and control of cumulative privacy loss but still suffers from slow convergence. Fu

Table 2. Literature for Securing Edge Training

Scenario	Method	Ref.	Highlights	Effectiveness
Remote Training	DP	[116]	perturb SGD with differential privacy.	theoretical analysis for minimum privacy safeguards.
		[105]	perturb the extracted features with random projection, random noise addition, and data nullification.	mixed-precision models for IoT.
	HE	[42]	convert real-valued training data to integers efficiently for encryption.	prevent privacy leakage from intact instances.
		[53]	approximating activation functions with low-degree polynomials.	minimal utility reduction.
	MPC	[114]	decimals to integers, pre-generated multiplication triplets, truncate shares for same precision.	several orders of magnitude faster.
Distributed Learning	DP	[111]	privacy-preserving split learning, add noise to intermediate results.	best partitioning strategy based on tailored metrics.
		[38]	random sub-sampling of participants, distort the sum of updates.	minor utility reduction.
		[135]	privacy-preserving federated learning, selectively share perturbed parameters.	enhanced model utility.
		[1]	perturb gradients with cumulative privacy loss control.	manageable cost in complexity, efficiency, and quality.
		[32]	differentially private gradient, select effective gradients.	faster convergence.
	Anonymity	[182]	unlinkable pseudonyms, identify and eliminate the malicious anomaly model parameters.	obvious latency reduction, communication reduction.
	MPC	[9]	secure aggregation on model updates.	constant communication rounds.
	Blockchain	[125, 146]	log local updates, distort training data with DP.	traceable federated learning.
		[72]	smart contract.	mitigate single-point-of-failure risks.
	Outlier detection	[128]	hierarchical aggregation, majority vote.	algorithm redundancy, massive gains in accuracy and speed.
		[137]	random masks for privacy, distance-based outlier removal, Reed-Solomon error correction code, verifiable secret sharing.	privacy protection, convergence.
		[99]	pearson correlation coefficients, perturb encrypted gradients.	privacy protection.
		[113]	log global model on blockchain, encrypt model updates with HE, cosine similarity.	privacy protection, convergence.

et al. [32] address this by secretly selecting effective updates that lead to convergence, employing a clipping strategy and a threshold mechanism.

2) *Anonymity.* Yuan *et al.* [182] introduce FedComm, a protocol that preserves user privacy in federated learning by using unlinkable pseudonyms to obscure the connection between model parameters and the users. Additionally, FedComm leverages similarity to identify and eliminate malicious anomaly model parameters, ensuring accurate training.
3) *MPC.* Bonawitz *et al.* [9] propose a protocol for secure aggregation of model updates, in which the parameter center (i.e., the server) only accesses the aggregated updates and remains unaware of the individual updates. The entire scheme is built upon MPC and secret sharing, requiring a constant number of additional communication rounds.

4) *Blockchain.* Some works [125, 146] employ blockchain technology to log local updates from legitimate clients and perform aggregation. They also leverage differential privacy to distort the training data, assuming that legal users will not be malicious. Kim *et al.* [72] proposed BlockFL which deploys a smart contract on the blockchain to secure exchange and validation of local model updates. Additionally, BlockFL enhances reliability by mitigating single point of failure risks through the verification of local training results.
5) *Outlier Detection.* To address the presence of Byzantine users, robust aggregation schemes are needed to handle participant dropout and mitigate poisoning attacks by identifying and removing outliers. Rajput *et al.* [128] enhance robustness against Byzantine users by employing algorithm redundancy, where participants are grouped to process the same data, and the server performs majority voting on gradients. This method reduces Byzantine gradient impact but doesn't prioritize user privacy since the server has full access to the training data. In contrast, most distributed learning frameworks require participants to keep their local datasets private. So *et al.* [137] introduce BREA, a robust federated learning scheme that protects data privacy and counters poisoning attacks. BREA combines the multi-Krum algorithm to filter out malicious gradients and uses Reed-Solomon decoding for error correction, while verifiable secret sharing ensures user privacy. However, this approach results in high communication overhead due to the extensive interaction required among participants. Liu *et al.* [99] propose a robust aggregation method in federated learning involving a collaborative approach between a service provider and an honest-but-curious cloud platform. Participants encrypt their gradients, and the cloud platform, using HE, helps identify and eliminate outliers through Pearson correlation and aggregation, all while preserving data privacy. Besides identifying and isolating malicious gradients using cosine similarity, Miao *et al.* [113] employ blockchain to log the global model, enhancing the transparency and integrity of the learning process.

6 SECURE EDGE INFERENCE

In this section, we address security concerns during inference, covering primary attacks and secure scheme taxonomy, followed by detailed discussions of each scheme.

6.1 Attacks

In edge inference, inference tasks are often outsourced to external parties, raising major concerns about the theft of training data or models. The subsequent sections discuss the main attack vectors.

6.1.1 Data Stealing. Data stealing involves an adversary attempting to uncover sensitive information from input data through various attacks. We outline the key attacks below.

- *Membership Inference Attack.* It aims to determine if a specific data sample was included in the training process by analyzing model predictions, ultimately revealing the makeup of the training dataset [170]. Chen *et al.* [19] explore membership inference attacks in collaborative inference, demonstrating how adversaries can use intermediate activations transmitted from clients to servers to unveil sensitive information about the training data.
- *Model Inversion Attack.* This attack focuses on reconstructing the input from the outputs of the model, such as intermediate activations and final outcomes. Typically occurring during the inference phase, a model inversion attack involves querying the model to gather pairs of inputs and outputs. Adversaries then learn the relationships between these inputs and outputs, utilizing identified correlations to reconstruct the object input. Yin *et al.* [179] introduced Ginver, a robust model inversion attack designed for collaborative inference, capable of operating in both black-box and white-box scenarios.

6.1.2 Model Stealing. Model stealing seeks to replicate the functionality of a proprietary model without direct access to its underlying architecture or training data [166]. It usually works as follows: the attackers query the target model with carefully crafted inputs and collect the input-output pairs, with which they can train a new model that mimics the original model's behavior.

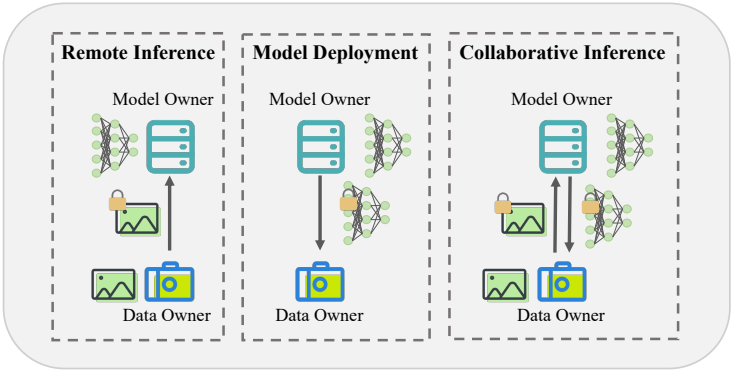

Fig. 7. Taxonomy of Edge Inference

6.2 Taxonomy of Secure Schemes

Based on the work mode of inference shown in Figure 7, we can categorize secure schemes into three types: secure remote inference, secure model deployment, and secure collaborative inference. Table 3 summarizes the works.

6.2.1 Secure Remote Inference. Clients typically send their data to an edge server for inference. In some schemes, users send data directly to the server, necessitating private inference methods that work on encrypted or perturbed inputs. In other schemes involving model partitioning, users process data with client-side subnetworks before sending it to the server, which raises privacy concerns due to the exchange of "smashed" data. This processed data makes both the input and the client-side network more vulnerable to attacks, especially from insiders with partial model knowledge. Secure schemes are needed to protect data privacy while ensuring private server-side inference.

6.2.2 Secure Model Deployment. In some cases, the model owner deploys the model to end devices for local inference, prioritizing model privacy. Secure schemes supporting encrypted model inference are essential.

6.2.3 Secure Collaborative Inference. In certain cases, when both data and the model are outsourced to a third party for collaborative inference, schemes must safeguard both data and model privacy.

6.3 Secure Remote Inference

Due to resource constraints on edge devices, user data is often sent to one or more entities for inference, posing a risk to the user's privacy. Various measures are being undertaken to protect user privacy in these circumstances.

1) *Perturbation.* When inference tasks are delegated to a third party, a common method to safeguard the privacy of inference samples involves adding noise to them. Wang *et al.* [149] split the DNN into client-side feature extraction and cloud-side inference components. They ensure privacy by adding noise to features before transmission, necessitating pre-training on noisy samples to

Table 3. Literature for Securing Edge Inference

Scenario	Method	Ref.	Highlights	Effectiveness
Remote Inference	Perturbation	[149]	add noises to features	low complexity
		[177]	tune the privacy budge, perturb and normalize the confidence score vectors.	high classification accuracy.
	HE	[40]	computation approximation with low-degree polynomials.	protect neural networks.
		[183]	convolutional blocks, LSTM-based sequence analysis layers, a weighted unit and a sequence voting layer.	analysis of encrypted time-series medical images, over 0.94 AUC.
		[11]	computation-friendly data representation, transfer learning	10× speedup, wider networks.
	MPC	[68]	seamlessly data conversation between HE and garbled circuit.	integration of HE and 2PC.
		[97]	nonlinear function approximation by polynomial splines, batch processing with SIMD.	re-training-free, lower latency and bandwidth cost.
		[132]	Yao's Garbled Circuit and oblivious transfer, GC-optimized components.	58× throughput.
	TEE	[84]	accelerate inference inside the enclaves.	3.6× speedup.
Model Deployment	TEE	[55]	obfuscates the model, enclave handles the denoising and noising.	model privacy, 7.8× increase in efficiency, 27× reduction in memory usage.
Collaborative Inference	HE	[65]	HE-based matrice operation, SIMD for batch processing.	0.45s per image.
		[18]	Multi-Key HE, bootstrapping techniques.	simpler and faster relinearization, support multiple data providers.
	MPC	[89]	secret sharing and triplet generation, asynchronous computation and SIMD, garbled circuit and average-pooling.	obvious latency reduction, communication reduction.
		[174]	homomorphic matrix-vector product and convolution, minimizing interactions between the parties.	2-4× speedup.

maintain accuracy. Ye *et al.* [177] propose a differential privacy-based defense mechanism to protect against membership inference and model inversion attacks. They perturb and normalize confidence score vectors while preserving score order to ensure classification accuracy.

2) *HE.* The main challenge of HE-based approaches is conducting the complex computations required during inference, which often involves approximation techniques and significant computational overhead. Typically, these approaches require rewriting linear functions using homomorphic operations and approximating non-linear functions, which may necessitate re-training the model to maintain accuracy. CryptoNets [40] employs HE for secure inference by enabling collaboration between the device and server without data exposure. It approximates common neural network computations, such as weighted sums and RELU activations, using low-degree polynomials. Yue *et al.* [183] create a privacy-focused system for encrypted time-series medical images using FHE. The system employs convolutional blocks and LSTM layers for feature extraction, and integrates these features with a weighted unit and sequence voting layer to enhance diagnostic accuracy and reduce false negatives. The main bottleneck of these approaches lies in their computational complexity. Brutzkus *et al.* [11] accelerate HE-based privacy-preserving inference by crafting representations and employing transfer learning, achieving over a 10× speedup and facilitating inference on wider networks.

3) *MPC.* Several research initiatives [68, 97, 132] utilize MPC, often employing secret sharing, garbled circuit, oblivious transfer, or integrating with HE. Gazelle [68] presents protocols that seamlessly convert data between the representations needed for HE and garbled circuits. This facilitates the integration of the diverse computational paradigms of HE and 2PC within a unified workflow. MiniONN [97] offers a method to convert an existing neural network into an oblivious one without requiring re-training. It handles linear functions and approximates nonlinear functions with polynomial splines, and supports batch processing using SIMD operations. DeepSecure [132] builds on Yao's Garbled Circuit (GC) protocol and oblivious transfer to optimize communication and computation. It reduces costs by minimizing the number of non-XOR gates in GC and requires dimensionality reduction by the client and neuron pruning by the server. Its main drawback is the limited number of processing instances per round.
4) *TEE.* Lee *et al.* [84] use a secure enclave on an untrusted server to perform secure inference. Data is sent directly from the client to the enclave through a secure channel, which ensures the integrity of the model while safeguarding user privacy.

6.4 Secure Model Deployment.

Certain end devices, equipped with advanced computational capabilities and memory, can host externally trained models for inference. Nevertheless, the model owners may seek to safeguard the privacy of their models during this process, necessitating efforts towards model privacy protection. Hou *et al.* [55] focus on deploying server models to edge devices for real-time inference without disclosing model specifics, utilizing secure enclaves. The server obscures the model with noise and transfers it to the end device. During inference, the end device performs computationally intensive tasks like convolution outside the enclave, which handles denoising and noise addition.

6.5 Secure Collaborative Inference.

In scenarios where both data and model are delegated to a third party for inference, that third party must not be able to reveal either.. The efforts aimed at ensuring this are summarized below.

1) *HE.* Jiang *et al.* [65] is the first effort to achieve this by adopting HE. Specifically, they utilize HE to secure a matrix and execute arithmetic operations on encrypted matrices, covering both fundamental matrix arithmetic and more complex tasks like matrix transposition and rectangular matrix multiplication. Additionally, by employing SIMD techniques to encrypt multiple matrices within a single ciphertext, they significantly boost efficiency. Chen *et al.* [18] also employ HE for safeguarding data and model privacy. Their method boosts security through the use of Multi-Key Homomorphic Encryption (MKHE) [102], moving beyond the limitations of a common shared encryption key to provide more sophisticated and versatile protection.
2) *MPC.* MPC typically employs secret sharing or HE techniques to safeguard both data and model privacy. Some works [89, 174] adopt secret sharing, which involves two non-colluding servers. The scheme proposed by Li *et al.* [89] serves as a notable example in this context. It dedicates significant efforts towards improving efficiency, particularly through the development of triplet generation techniques that decrease the dependence on heavy cryptographic operations. HE-based Pio [174] executes matrix-vector product and convolution homomorphically, minimizing interactions between the parties and bypassing expensive permutation tasks.

7 ORTHOGONAL SECURITY MEASURES

In this section, we outline orthogonal security strategies that complement the previous schemes.

7.1 Image Encryption

Some studies [14, 148] design visually meaningful image encryption systems that protect image privacy by hiding the plain image in a cover image while preserving its visual information. Chai *et al.*[15] achieve double-color encryption by combining 2D compressive sensing with an embedding technique, compressing and encrypting two color images simultaneously, and embedding them into a color carrier image. Wang *et al.*[155] efficiently encrypt double-image by scrambling and compressing them, then embedding the processed images into a carrier image through FFT. These methods produce visually meaningful cipher images that are the same size as the plain image. Yang *et al.* [175] further reduce the cipher image size and ensure reversibility, even under attack, using lossless compression and multi-round encryption.

7.2 Perceptual Integrity

Images are often manipulated, posing risks like misinformation. Protection strategies focus on detecting and defending against these manipulations.

a) Detection. Some studies adopt watermarks to identify manipulations. Wang *et al.*[151] achieve robust watermarking with an encoder-decoder architecture and channel coding, encoding an identity tag to detect DeepFake images. SepMark [162] uses an encoder-decoder design with separate decoders for robust and semi-robust watermark extraction, tracing the source of the marked face and detecting any alterations. Liu *et al.*[95] combine fragile and robust watermarking for face replacement detection and tracing. Tang *et al.*[139] embed links to facial features in videos using robust watermarking, using these features for detection.

b) Defense. Several studies propose adversarial watermarks to counter manipulations by inducing visible distortions. Huang *et al.* [58] realize the Cross-Model Universal Adversarial Watermark, using a cross-model attack pipeline and a two-level perturbation fusion strategy to target multiple deepfake models. Wang *et al.*[154] improve adversarial watermarks to ensure distorted images are detectable by passive detectors. Wang *et al.*[150] enhance adversarial attacks in image reconstruction with perceptually-aware perturbations and sparse noise. Qu *et al.*[126] maintain adversarial watermark performance under lossy compression in online social networks by using a Compression Approximation GAN integrated into the target Deepfake model.

7.3 Image Copyright

Image copyright is a critical issue, and robust watermarking—whether visible or invisible—is commonly used for protection. It must be resilient to various pre-processing, post-processing, and attacks.. Zhu *et al.*[196] use optional noise layers in an encoder-decoder architecture to improve robustness against distortions like cropping. However, this can lead to redundant features due to poor coupling between the encoder and decoder. Fang *et al.*[30] address this by using an invertible neural block for simultaneous watermark embedding and extraction, ensuring precise feature matching. Jia *et al.* [63] improve robustness against cropping with an additive diffusion block and improve data recovery using Squeeze-and-Extraction blocks and a “message processor”. Ma *et al.* [107] combine invertible and non-invertible mechanisms for high imperceptibility and robustness. The invertible part achieves high imperceptibility using a diffusion & extraction and a fusion & split module. The non-invertible component enhances robustness by handling asymmetric extraction under strong noise attacks with an attention-based module and a noise-specific selection module.

7.4 Privacy-preserving Models

Membership inference attacks aim to determine if an input is from a model's training dataset. Defense strategies aim to make model predictions indistinguishable between members and non-members Wang *et al.*[156] use a weight pruning algorithm to identify a subnetwork with indistinguishable predictions from an over-parameterized network. Tang *et al.*[140] propose an ensemble training method, dividing data into random subsets and using outputs from sub-models not trained on the input sample to achieve indistinguishability. Chen *et al.* [21] train multiple discriminators on disjoint partitions of the training data and approximate a composite distribution, enhancing generalization while achieving indistinguishability. Yang *et al.*[176] enhance indistinguishability by modifying confidence scores, adding Gaussian noise, and altering prediction labels, though this impacts accuracy.

Defending against model inversion attacks is complex. Sun *et al.*[138] degrade reconstructed data quality by perturbing data representations while preserving utility. Wang *et al.*[152] improve the utility-privacy tradeoff by adding a regularizer to reduce the correlation between model input and output. Peng *et al.*[122] enhance this tradeoff by minimizing input reliance and maximizing output correlation. Gong *et al.*[41] use a GAN-based approach, incorporating fake samples and modifying loss functions to add misleading features to protected labels.

7.5 Model Copyright

Watermarking models is an effective method for copyright protection, allowing identification of cloned models through specific watermarks in their output images. Zhang *et al.*[186] use spatial invisible watermarking for image processing models, but it only works if the attacker has the input-output pairs of the target model. Wu *et al.*[159] train the target model and watermark extraction network together, so the surrogate model inherently outputs watermarked images. Cong *et al.*[24] protect pre-trained encoders with watermarks and shadow training to resist model stealing. Luo *et al.*[104] watermark NeRF models' color representations, ensuring robust extraction with a distortion-resistant scheme. Adi *et al.* [2] use backdoor behaviors for watermarking to detect model stealing. Gan *et al.* [33] improve watermark robustness against removal attacks with a minimax formulation. Alternatively, non-invasive fingerprinting methods [123] determine ownership by comparing unique fingerprints extracted from models. Chen *et al.*[20] protect model copyright by testing similarities between victim and suspect models using various metrics and algorithms. Quan *et al.*[127] protect image restoration models with fingerprints based on critical images and verify ownership through color histogram and local gradient pattern comparisons.

8 CHALLENGES AND FUTURE OPPORTUNITIES

Current secure schemes often struggle with complex AI models and resource-constrained edge environments. These challenges arise because secure schemes are often model-specific, involve heavy computation and communication, and may require approximations that reduce utility and limit model size. To address these issues, it's crucial to improve efficiency and model utility while maintaining security. This section first explores the challenges of current security technologies and then discusses potential research directions for developing practical secure schemes.

8.1 Challenges Inherent From Security Technologies

Current secure schemes primarily leverage DP, HE, and MPC. Each of these approaches has its own advantages and disadvantages.

1) DP-based Secure Schemes. DP-based solutions consistently demonstrate high efficiency and excel in privacy protection. However, ensuring input data privacy with DP often necessitates

complementary training and inference adjustments. While DP effectively mitigates privacy risks, its implementation may result in a slight decline in model performance. Moreover, adding noise to individual data within DP schemes can increase the difficulty in detecting malicious data, ultimately making defense against model poisoning more challenging.

2) HE-based Secure Schemes. The main advantage of HE-based schemes is their ability to perform computations on encrypted data without revealing plaintext. However, in EI, they face challenges such as controlling computational overhead and realizing complex non-polynomial functions. The costly computational overhead and accumulated approximation errors also impose restrictions on the height and width of the model that HE-based schemes can effectively protect.

3) MPC-based Secure Schemes. MPC enables multiple parties to conduct computations without exposing their data to others, making it suitable for computation tasks involving distributed data. However, the bottleneck of MPC-based schemes lies in their communication complexity, as multiple machines collaborate, necessitating extensive communication and customized protocols.

8.2 Enhancing Security in AI Tasks

Here, we present two avenues for bolstering security: one entails devising schemes that offer comprehensive protection against various forms of attacks, while the other seeks to fortify security in light of advancements in quantum computing.

8.2.1 Privacy-preserving Model Poisoning Defense Schemes. In the realm of edge intelligence, there is a discrepancy in security practices: one emphasizes privacy preservation without sufficient defense against model poisoning, while the other concentrates on mitigating model poisoning, frequently sidelining privacy issues. This indicates a lack of a bridge that effectively combines model poisoning defense with privacy protection. Solutions that address both aspects simultaneously are still in their formative stages. The prevailing approach [99, 137] leverages similarity comparisons on homomorphically encrypted data during the aggregation phase within federated learning. Note that this approach is chosen over DP, which complicates similarity-based outlier detection due to noise addition. There is a pressing demand for optimizing similarity measures and selection methods, such as determining the optimal k in k-nearest neighbors, to enhance similarity-based outlier detection effectiveness. Moreover, in federated learning, the benchmark for outlier detection is derived from the homomorphically encrypted gradients contributed by a multitude of participants. However, in scenarios like split learning, where each participant performs distinct computations, accessing real-time ground truth becomes challenging. Addressing this challenge necessitates the innovation of novel privacy-preserving methods that can define an optimal benchmark for outlier detection, which is crucial for fighting model poisoning in this context. This scenario underscores the expansive opportunity for future research to bridge these gaps, ensuring robust security without compromising privacy in edge intelligence systems.

8.2.2 Schemes with Quantum-Safe. Quantum computing offers great potential for handling complex computations but poses dual-edged security implications. On one hand, it renders many traditional cryptography systems, such as RSA and ECC [129, 158], ineffective. Consequently, security schemes built upon these cryptography systems face obsolescence. On the other hand, its formidable computing power can be harnessed to meet the heavy computational demands of cryptography systems, thereby extending protection to a broader spectrum of assets, including larger-scale models. Therefore, there is a pressing need for security solutions that leverage quantum-safe technologies capable of resisting quantum servers while also harnessing the computational capabilities of quantum computing to provide swift security protection. In image-centric EI, combining quantum computing on powerful servers with resource-constrained edge devices may require integrating quantum and traditional computing, possibly using new transfer protocols.

8.3 Enhancing Efficiency in Privacy-Preserving AI Tasks

Efficiency remains a significant obstacle to deploying secure mechanisms. Crafting diverse strategies to address this challenge is crucial. Efforts to enhance efficiency include optimizing cryptographic approaches, leveraging advanced hardware, applying parallel computation techniques, and compressing models. This discussion opens up a potential research avenue aimed at optimizing the efficiency of processing numerous tasks.

8.3.1 Optimizing Image Feature Selection for Privacy and Efficiency. In image-centric EI, leveraging images for diverse intelligent analyses stands as a core function. This process inherently involves the extraction of features from images, which serve as the foundation for these analyses. Notably, the features extracted for different types of analysis vary, both in terms of the attributes they capture and the extent to which they overlap with features used in other analyses. A significant challenge arises when features are stored in an untrusted database and the model trainer needs to access private features from it. A straightforward method, which might involve creating separately encrypted versions of features for each analysis task, is not only inefficient but also resource-intensive. The database must facilitate private feature selection to enhance both storage and computational efficiency. The untrusted database must remain unaware not only of the feature values and selection but also of the specific training task, as any knowledge could compromise privacy. We then delve into the challenges posed by three key security technologies. While implementing DP introduces complexities in providing task-specific protection levels, HE struggles with efficiency. Utilizing SIMD operations offers a promising solution for efficient HE but demands careful management of feature interactions. Furthermore, MPC involving non-colluding servers increases communication costs, with the additional challenge of accommodating server dropouts. Furthermore, the incorporation of a feature selection method is crucial to the mechanism and should be carefully considered.

8.4 Enhancing Model Utility in Privacy-Preserving AI Tasks

Many privacy-preserving schemes, notably those involving DP and HE, inherently introduce errors, making error minimization crucial. For DP-based approaches, an effective method is to tailor DP mechanisms that account for the dependencies between features and the variable precision needs of feature values, harmonizing privacy with utility. In the context of HE-based schemes, where errors primarily arise from approximating non-polynomial functions, leveraging TEE offers a promising solution to mitigate these errors. This strategy is further elaborated below.

8.4.1 TEE-based Privacy-Preserving Schemes. HE-based private training/inference often necessitates approximation, leading to error introduction. A viable strategy for optimization involves refining approximation techniques, potentially by integrating with other technologies such as MPC to minimize approximation operations. An alternative promising solution is the deployment of TEEs, which provide a secure and isolated execution space for crucial computations and error mitigation, thereby enhancing both the precision and security of AI tasks. Nonetheless, deploying TEEs effectively demands a nuanced consideration of the trade-offs among security, performance, and the computational burden they introduce. With ongoing advancements in technology, incorporating TEEs into secure, privacy-conscious computational frameworks remains a dynamic field for exploration and innovation, presenting novel avenues to bolster the dependability and security of distributed computing infrastructures.

9 CONCLUSION

Edge Intelligence emerges as a powerful new paradigm for processing and making intelligent decisions on the deluge of edge data. Nonetheless, this emerging paradigm also brings complex

security challenges, such as model/image privacy and model integrity, particularly for privacy-sensitive and resource-intensive image-centric applications. In this survey, we have explored the essential origins of those threats, namely data attacks, during the phases of data preparation, training, and inference, and discussed relevant security strategies to mitigate them. Additionally, we present some orthogonal security measures that could complement the aforementioned security strategies focused on data attacks. Furthermore, we outline challenges and discuss promising research opportunities.

ACKNOWLEDGMENT

This work was supported by National Natural Science Foundation of China (Grant No: 92270123, 62072390), and the Research Grants Council, Hong Kong SAR, China (Grant No: 15203120, 15226221), and the Scientific and Technological Research Program of Chongqing Municipal Education Commission (KJZD-K202300601).

REFERENCES

[1] Martin Abadi, Andy Chu, Ian Goodfellow, H Brendan McMahan, Ilya Mironov, Kunal Talwar, and Li Zhang. 2016. Deep learning with differential privacy. In *Proceedings of the 2016 ACM SIGSAC conference on computer and communications security*. 308–318.

[2] Yossi Adi, Carsten Baum, Moustapha Cisse, Benny Pinkas, and Joseph Keshet. 2018. Turning your weakness into a strength: Watermarking deep neural networks by backdooring. In *27th USENIX Security Symposium (USENIX Security 18)*. 1615–1631.

[3] Firas Al-Doghman, Nour Moustafa, Ibrahim Khalil, Nasrin Sohrabi, Zahir Tari, and Albert Y Zomaya. 2022. AI-enabled secure microservices in edge computing: Opportunities and challenges. *IEEE Transactions on Services Computing* 16, 2 (2022), 1485–1504.

[4] Mario Almeida, Stefanos Laskaridis, Stylianos I Venieris, Ilias Leontiadis, and Nicholas D Lane. 2022. Dyno: Dynamic onloading of deep neural networks from cloud to device. *ACM Transactions on Embedded Computing Systems* 21, 6 (2022), 1–24.

[5] Abdulmalik Alwarafy, Khaled A Al-Thelaya, Mohamed Abdallah, Jens Schneider, and Mounir Hamdi. 2020. A survey on security and privacy issues in edge-computing-assisted internet of things. *IEEE Internet of Things Journal* 8, 6 (2020), 4004–4022.

[6] Martin Arjovsky, Soumith Chintala, and Léon Bottou. 2017. Wasserstein generative adversarial networks. In *International conference on machine learning*. PMLR, 214–223.

[7] Eugene Bagdasaryan, Andreas Veit, Yiqing Hua, Deborah Estrin, and Vitaly Shmatikov. 2020. How to backdoor federated learning. In *International conference on artificial intelligence and statistics*. PMLR, 2938–2948.

[8] Yixin Bao, Yanghua Peng, Yangrui Chen, and Chuan Wu. 2020. Preemptive all-reduce scheduling for expediting distributed DNN training. In *IEEE INFOCOM 2020-IEEE Conference on Computer Communications*. IEEE, 626–635.

[9] Keith Bonawitz, Vladimir Ivanov, Ben Kreuter, Antonio Marcedone, H Brendan McMahan, Sarvar Patel, Daniel Ramage, Aaron Segal, and Karn Seth. 2017. Practical secure aggregation for privacy-preserving machine learning. In *proceedings of the 2017 ACM SIGSAC Conference on Computer and Communications Security*. 1175–1191.

[10] Andrew Brock, Jeff Donahue, and Karen Simonyan. 2018. Large scale GAN training for high fidelity natural image synthesis. *arXiv preprint arXiv:1809.11096* (2018).

[11] Alon Brutzkus, Ran Gilad-Bachrach, and Oren Elisha. 2019. Low latency privacy preserving inference. In *International Conference on Machine Learning*. PMLR, 812–821.

[12] Guopeng Cai, Xiaochao Wei, and Yao Li. 2022. Privacy-preserving CNN feature extraction and retrieval over medical images. *International Journal of Intelligent Systems* 37, 11 (2022), 9267–9289.

[13] Zhenkun Cai, Xiao Yan, Yidi Wu, Kaihao Ma, James Cheng, and Fan Yu. 2021. DGCL: an efficient communication library for distributed GNN training. In *Proceedings of the Sixteenth European Conference on Computer Systems*. 130–144.

[14] Xiuli Chai, Zhihua Gan, Yiran Chen, and Yushu Zhang. 2017. A visually secure image encryption scheme based on compressive sensing. *Signal Processing* 134 (2017), 35–51.

[15] Xiuli Chai, Haiyang Wu, Zhihua Gan, Daojun Han, Yushu Zhang, and Yiran Chen. 2021. An efficient approach for encrypting double color images into a visually meaningful cipher image using 2D compressive sensing. *Information Sciences* 556 (2021), 305–340.

[16] Chen Chen, Chenyu Wang, Bin Liu, Ci He, Li Cong, and Shaohua Wan. 2023. Edge intelligence empowered vehicle detection and image segmentation for autonomous vehicles. *IEEE Transactions on Intelligent Transportation Systems* (2023).

[17] Guobin Chen, Wongun Choi, Xiang Yu, Tony Han, and Manmohan Chandraker. 2017. Learning efficient object detection models with knowledge distillation. *Advances in neural information processing systems* 30 (2017).

[18] Hao Chen, Wei Dai, Miran Kim, and Yongsoo Song. 2019. Efficient multi-key homomorphic encryption with packed ciphertexts with application to oblivious neural network inference. In *Proceedings of the 2019 ACM SIGSAC Conference on Computer and Communications Security*. 395–412.

[19] Hanxiao Chen, Hongwei Li, Guishan Dong, Meng Hao, Guowen Xu, Xiaoming Huang, and Zhe Liu. 2020. Practical membership inference attack against collaborative inference in industrial IoT. *IEEE Transactions on Industrial Informatics* 18, 1 (2020), 477–487.

[20] Jialuo Chen, Jingyi Wang, Tinglan Peng, Youcheng Sun, Peng Cheng, Shouling Ji, Xingjun Ma, Bo Li, and Dawn Song. 2022. Copy, right? a testing framework for copyright protection of deep learning models. In *2022 IEEE symposium on security and privacy (SP)*. IEEE, 824–841.

[21] Junjie Chen, Wendy Hui Wang, Hongchang Gao, and Xinghua Shi. 2021. PAR-GAN: improving the generalization of generative adversarial networks against membership inference attacks. In *Proceedings of the 27th ACM SIGKDD Conference on Knowledge Discovery & Data Mining*. 127–137.

[22] François Chollet. 2017. Xception: Deep learning with depthwise separable convolutions. In *Proceedings of the IEEE conference on computer vision and pattern recognition*. 1251–1258.

[23] Xiangxiang Chu, Liang Li, and Bo Zhang. 2024. Make repvgg greater again: A quantization-aware approach. In *Proceedings of the AAAI Conference on Artificial Intelligence*, Vol. 38. 11624–11632.

[24] Tianshuo Cong, Xinlei He, and Yang Zhang. 2022. Sslguard: A watermarking scheme for self-supervised learning pre-trained encoders. In *Proceedings of the 2022 ACM SIGSAC Conference on Computer and Communications Security*. 579–593.

[25] Bin Dai, Chen Zhu, Baining Guo, and David Wipf. 2018. Compressing neural networks using the variational information bottleneck. In *International Conference on Machine Learning*. PMLR, 1135–1144.

[26] Xin Dong, Shangyu Chen, and Sinno Pan. 2017. Learning to prune deep neural networks via layer-wise optimal brain surgeon. *Advances in neural information processing systems* 30 (2017).

[27] Rong Du, Qingqing Ye, Yue Fu, Haibo Hu, Jin Li, Chengfang Fang, and Jie Shi. 2023. Differential aggregation against general colluding attackers. In *2023 IEEE 39th International Conference on Data Engineering (ICDE)*. IEEE, 2180–2193.

[28] Jiawei Duan, Qingqing Ye, and Haibo Hu. 2022. Utility analysis and enhancement of LDP mechanisms in high-dimensional space. In *2022 IEEE 38th International Conference on Data Engineering (ICDE)*. IEEE, 407–419.

[29] Ege Erdoğan, Alptekin Küpçü, and A Ercüment Çiçek. 2022. Unsplit: Data-oblivious model inversion, model stealing, and label inference attacks against split learning. In *Proceedings of the 21st Workshop on Privacy in the Electronic Society*. 115–124.

[30] Han Fang, Yupeng Qiu, Kejiang Chen, Jiyi Zhang, Weiming Zhang, and Ee-Chien Chang. 2023. Flow-based robust watermarking with invertible noise layer for black-box distortions. In *Proceedings of the AAAI conference on artificial intelligence*, Vol. 37. 5054–5061.

[31] Ayman M Fouad, RM Sharkawy, and Ahmed Onsy. 2019. Fixed obstacle detection for autonomous vehicle. In *2019 IEEE Conference on Power Electronics and Renewable Energy (CPERE)*. IEEE, 217–221.

[32] Jie Fu, Qingqing Ye, Haibo Hu, Zhili Chen, Lulu Wang, Kuncan Wang, and Ran Xun. 2023. DPSUR: Accelerating Differentially Private Stochastic Gradient Descent Using Selective Update and Release. *arXiv preprint arXiv:2311.14056* (2023).

[33] Guanhao Gan, Yiming Li, Dongxian Wu, and Shu-Tao Xia. 2023. Towards robust model watermark via reducing parametric vulnerability. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*. 4751–4761.

[34] Dawei Gao, Xiaoxi He, Zimu Zhou, Yongxin Tong, Ke Xu, and Lothar Thiele. 2020. Rethinking pruning for accelerating deep inference at the edge. In *Proceedings of the 26th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*. 155–164.

[35] Xinben Gao and Lan Zhang. 2023. {PCAT}: Functionality and Data Stealing from Split Learning by {Pseudo-Client} Attack. In *32nd USENIX Security Symposium (USENIX Security 23)*. 5271–5288.

[36] Jonas Geiping, Hartmut Bauermeister, Hannah Dröge, and Michael Moeller. 2020. Inverting gradients-how easy is it to break privacy in federated learning? *Advances in Neural Information Processing Systems* 33 (2020), 16937–16947.

[37] Craig Gentry. 2009. Fully homomorphic encryption using ideal lattices. In *Proceedings of the forty-first annual ACM symposium on Theory of computing*. 169–178.

[38] Robin C Geyer, Tassilo Klein, and Moin Nabi. 2017. Differentially private federated learning: A client level perspective. *arXiv preprint arXiv:1712.07557* (2017).

[39] Amir Gholami, Kiseok Kwon, Bichen Wu, Zizheng Tai, Xiangyu Yue, Peter Jin, Sicheng Zhao, and Kurt Keutzer. 2018. Squeezenext: Hardware-aware neural network design. In *Proceedings of the IEEE conference on computer vision and pattern recognition workshops*. 1638–1647.

[40] Ran Gilad-Bachrach, Nathan Dowlin, Kim Laine, Kristin Lauter, Michael Naehrig, and John Wernsing. 2016. Cryptonets: Applying neural networks to encrypted data with high throughput and accuracy. In *International conference on machine learning*. PMLR, 201–210.

[41] Xueluan Gong, Ziyao Wang, Shuaike Li, Yanjiao Chen, and Qian Wang. 2023. A gan-based defense framework against model inversion attacks. *IEEE Transactions on Information Forensics and Security* (2023).

[42] Francisco-Javier González-Serrano, Ángel Navia-Vázquez, and Adrián Amor-Martín. 2017. Training support vector machines with privacy-protected data. *Pattern Recognition* 72 (2017), 93–107.

[43] Ian Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, and Yoshua Bengio. 2020. Generative adversarial networks. *Commun. ACM* 63, 11 (2020), 139–144.

[44] Yiwen Guo, Anbang Yao, and Yurong Chen. 2016. Dynamic network surgery for efficient dnns. *Advances in neural information processing systems* 29 (2016).

[45] Prajjwal Gupta, Krishna Yadav, Brij B Gupta, Mamoun Alazab, and Thippa Reddy Gadekallu. 2023. A Novel Data Poisoning Attack in Federated Learning based on Inverted Loss Function. *Computers & Security* 130 (2023), 103270.

[46] Song Han, Huizi Mao, and William J Dally. 2015. Deep compression: Compressing deep neural networks with pruning, trained quantization and huffman coding. *arXiv preprint arXiv:1510.00149* (2015).

[47] Song Han, Jeff Pool, John Tran, and William Dally. 2015. Learning both weights and connections for efficient neural network. *Advances in neural information processing systems* 28 (2015).

[48] Yefei He, Luping Liu, Jing Liu, Weijia Wu, Hong Zhou, and Bohan Zhuang. 2024. Ptqd: Accurate post-training quantization for diffusion models. *Advances in Neural Information Processing Systems* 36 (2024).

[49] Yang He, Ping Liu, Ziwei Wang, Zhilan Hu, and Yi Yang. 2019. Filter pruning via geometric median for deep convolutional neural networks acceleration. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*. 4340–4349.

[50] Yihui He, Xiangyu Zhang, and Jian Sun. 2017. Channel pruning for accelerating very deep neural networks. In *Proceedings of the IEEE international conference on computer vision*. 1389–1397.

[51] Paul Henman. 2020. Improving public services using artificial intelligence: possibilities, pitfalls, governance. *Asia Pacific Journal of Public Administration* 42, 4 (2020), 209–221.

[52] Byeongho Heo, Minsik Lee, Sangdoo Yun, and Jin Young Choi. 2019. Knowledge transfer via distillation of activation boundaries formed by hidden neurons. In *Proceedings of the AAAI Conference on Artificial Intelligence*, Vol. 33. 3779–3787.

[53] Ehsan Hesamifard, Hassan Takabi, Mehdi Ghasemi, and Rebecca N Wright. 2018. Privacy-preserving machine learning as a service. *Proc. Priv. Enhancing Technol.* 2018, 3 (2018), 123–142.

[54] Geoffrey Hinton, Oriol Vinyals, and Jeff Dean. 2015. Distilling the knowledge in a neural network. *arXiv preprint arXiv:1503.02531* (2015).

[55] Jiahui Hou, Huiqi Liu, Yunxin Liu, Yu Wang, Peng-Jun Wan, and Xiang-Yang Li. 2021. Model Protection: Real-time privacy-preserving inference service for model privacy at the edge. *IEEE Transactions on Dependable and Secure Computing* 19, 6 (2021), 4270–4284.

[56] Andrew Howard, Mark Sandler, Grace Chu, Liang-Chieh Chen, Bo Chen, Mingxing Tan, Weijun Wang, Yukun Zhu, Ruoming Pang, Vijay Vasudevan, et al. 2019. Searching for mobilenetv3. In *Proceedings of the IEEE/CVF international conference on computer vision*. 1314–1324.

[57] Chuang Hu, Wei Bao, Dan Wang, and Fengming Liu. 2019. Dynamic adaptive DNN surgery for inference acceleration on the edge. In *IEEE INFOCOM 2019-IEEE Conference on Computer Communications*. IEEE, 1423–1431.

[58] Hao Huang, Yongtao Wang, Zhaoyu Chen, Yuze Zhang, Yuheng Li, Zhi Tang, Wei Chu, Jingdong Chen, Weisi Lin, and Kai-Kuang Ma. 2022. Cmua-watermark: A cross-model universal adversarial watermark for combating deepfakes. In *Proceedings of the AAAI Conference on Artificial Intelligence*, Vol. 36. 989–997.

[59] Kai Huang, Ximeng Liu, Shaojing Fu, Deke Guo, and Ming Xu. 2019. A lightweight privacy-preserving CNN feature extraction framework for mobile sensing. *IEEE Transactions on Dependable and Secure Computing* 18, 3 (2019), 1441–1455.

[60] Wenke Huang, Mang Ye, Zekun Shi, He Li, and Bo Du. 2023. Rethinking federated learning with domain shift: A prototype view. In *2023 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*. IEEE, 16312–16322.

[61] Forrest N Iandola, Song Han, Matthew W Moskewicz, Khalid Ashraf, William J Dally, and Kurt Keutzer. 2016. SqueezeNet: AlexNet-level accuracy with 50x fewer parameters and< 0.5 MB model size. *arXiv preprint arXiv:1602.07360* (2016).

[62] Zuzana Jelčicová and Marian Verhelst. 2022. Delta keyword transformer: Bringing transformers to the edge through dynamically pruned multi-head self-attention. *arXiv preprint arXiv:2204.03479* (2022).

[63] Zhaoyang Jia, Han Fang, and Weiming Zhang. 2021. Mbrs: Enhancing robustness of dnn-based watermarking by mini-batch of real and simulated jpeg compression. In *Proceedings of the 29th ACM international conference on multimedia*. 41–49.

[64] Linzhi Jiang, Chunxiang Xu, Xiaofang Wang, Bo Luo, and Huaqun Wang. 2017. Secure outsourcing SIFT: Efficient and privacy-preserving image feature extraction in the encrypted domain. *IEEE Transactions on Dependable and Secure Computing* 17, 1 (2017), 179–193.

[65] Xiaoqian Jiang, Miran Kim, Kristin Lauter, and Yongsoo Song. 2018. Secure outsourced matrix computation and application to neural networks. In *Proceedings of the 2018 ACM SIGSAC conference on computer and communications security*. 1209–1222.

[66] Yimin Jiang, Yibo Zhu, Chang Lan, Bairen Yi, Yong Cui, and Chuanxiong Guo. 2020. A unified architecture for accelerating distributed {DNN} training in heterogeneous {GPU/CPU} clusters. In *14th USENIX Symposium on Operating Systems Design and Implementation (OSDI 20)*. 463–479.

[67] Xiao Jin, Pin-Yu Chen, Chia-Yi Hsu, Chia-Mu Yu, and Tianyi Chen. 2021. Cafe: Catastrophic data leakage in vertical federated learning. *Advances in Neural Information Processing Systems* 34 (2021), 994–1006.

[68] Chiraag Juvekar, Vinod Vaikuntanathan, and Anantha Chandrakasan. 2018. {GAZELLE}: A low latency framework for secure neural network inference. In *27th USENIX Security Symposium (USENIX Security 18)*. 1651–1669.

[69] Hyeong-Ju Kang. 2019. Accelerator-aware pruning for convolutional neural networks. *IEEE Transactions on Circuits and Systems for Video Technology* 30, 7 (2019), 2093–2103.

[70] Tero Karras, Samuli Laine, and Timo Aila. 2019. A style-based generator architecture for generative adversarial networks. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*. 4401–4410.

[71] Tero Karras, Samuli Laine, Miika Aittala, Janne Hellsten, Jaakko Lehtinen, and Timo Aila. 2020. Analyzing and improving the image quality of stylegan. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*. 8110–8119.

[72] Hyesung Kim, Jihong Park, Mehdi Bennis, and Seong-Lyun Kim. 2019. Blockchained on-device federated learning. *IEEE Communications Letters* 24, 6 (2019), 1279–1283.

[73] Sehoon Kim, Coleman Hooper, Amir Gholami, Zhen Dong, Xiuyu Li, Sheng Shen, Michael W Mahoney, and Kurt Keutzer. 2023. Squeezellm: Dense-and-sparse quantization. *arXiv preprint arXiv:2306.07629* (2023).

[74] Serkan Kiranyaz, Turker Ince, Alexandros Iosifidis, and Moncef Gabbouj. 2017. Generalized model of biological neural networks: Progressive operational perceptrons. In *2017 International Joint Conference on Neural Networks (IJCNN)*. IEEE, 2477–2485.

[75] Serkan Kiranyaz, Turker Ince, Alexandros Iosifidis, and Moncef Gabbouj. 2017. Progressive operational perceptrons. *Neurocomputing* 224 (2017), 142–154.

[76] Serkan Kiranyaz, Turker Ince, Alexandros Iosifidis, and Moncef Gabbouj. 2020. Operational neural networks. *Neural Computing and Applications* 32, 11 (2020), 6645–6668.

[77] Serkan Kiranyaz, Junaid Malik, Habib Ben Abdallah, Turker Ince, Alexandros Iosifidis, and Moncef Gabbouj. 2021. Exploiting heterogeneity in operational neural networks by synaptic plasticity. *Neural Computing and Applications* 33 (2021), 7997–8015.

[78] Serkan Kiranyaz, Junaid Malik, Habib Ben Abdallah, Turker Ince, Alexandros Iosifidis, and Moncef Gabbouj. 2021. Self-organized operational neural networks with generative neurons. *Neural Networks* 140 (2021), 294–308.

[79] Serkan Kiranyaz, Junaid Malik, Mehmet Yamac, Mert Duman, Ilke Adalioglu, Esin Guldogan, Turker Ince, and Moncef Gabbouj. 2023. Super neurons. *IEEE Transactions on Emerging Topics in Computational Intelligence* (2023).

[80] Lan Lan, Lei You, Zeyang Zhang, Zhiwei Fan, Weiling Zhao, Nianyin Zeng, Yidong Chen, and Xiaobo Zhou. 2020. Generative adversarial networks and its applications in biomedical informatics. *Frontiers in public health* 8 (2020), 164.

[81] Christian Ledig, Lucas Theis, Ferenc Huszár, Jose Caballero, Andrew Cunningham, Alejandro Acosta, Andrew Aitken, Alykhan Tejani, Johannes Totz, Zehan Wang, et al. 2017. Photo-realistic single image super-resolution using a generative adversarial network. In *Proceedings of the IEEE conference on computer vision and pattern recognition*. 4681–4690.

[82] Namhoon Lee, Thalaiyasingam Ajanthan, and Philip HS Torr. 2018. Snip: Single-shot network pruning based on connection sensitivity. *arXiv preprint arXiv:1810.02340* (2018).

[83] Seung Hyun Lee, Dae Ha Kim, and Byung Cheol Song. 2018. Self-supervised knowledge distillation using singular value decomposition. In *Proceedings of the European conference on computer vision (ECCV)*. 335–350.

[84] Taegyeong Lee, Zhiqi Lin, Saumay Pushp, Caihua Li, Yunxin Liu, Youngki Lee, Fengyuan Xu, Chenren Xu, Lintao Zhang, and Junehwa Song. 2019. Occlumency: Privacy-preserving remote deep-learning inference using SGX. In *The 25th Annual International Conference on Mobile Computing and Networking*. 1–17.

[85] Xinyu Lei, Hongguang Pan, and Xiangdong Huang. 2019. A dilated CNN model for image classification. *IEEE Access* 7 (2019), 124087–124095.

[86] Anran Li, Jiahui Huang, Ju Jia, Hongyi Peng, Lan Zhang, Luu Anh Tuan, Han Yu, and Xiang-Yang Li. 2023. Efficient and Privacy-Preserving Feature Importance-based Vertical Federated Learning. *IEEE Transactions on Mobile Computing* (2023).

[87] Haoyang Li, Qingqing Ye, Haibo Hu, Jin Li, Leixia Wang, Chengfang Fang, and Jie Shi. 2023. 3dfed: Adaptive and extensible framework for covert backdoor attack in federated learning. In *2023 IEEE Symposium on Security and Privacy (SP)*. IEEE, 1893–1907.

[88] Lixiang Li, Xiaohui Mu, Siying Li, and Haipeng Peng. 2020. A review of face recognition technology. *IEEE access* 8 (2020), 139110–139120.

[89] Minghui Li, Sherman SM Chow, Shengshan Hu, Yuejing Yan, Chao Shen, and Qian Wang. 2020. Optimizing privacy-preserving outsourced convolutional neural network predictions. *IEEE Transactions on Dependable and Secure Computing* 19, 3 (2020), 1592–1604.

[90] Pengzhen Li, Erdem Koyuncu, and Hulya Seferoglu. 2021. Respipe: Resilient model-distributed dnn training at edge networks. In *ICASSP 2021-2021 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 3660–3664.

[91] Shen Li, Yanli Zhao, Rohan Varma, Omkar Salpekar, Pieter Noordhuis, Teng Li, Adam Paszke, Jeff Smith, Brian Vaughan, Pritam Damania, et al. 2020. Pytorch distributed: Experiences on accelerating data parallel training. *arXiv preprint arXiv:2006.15704* (2020).

[92] Xiling Li, Rafael Dowsley, and Martine De Cock. 2021. Privacy-preserving feature selection with secure multiparty computation. In *International Conference on Machine Learning*. PMLR, 6326–6336.

[93] Haotian Liang, Youqi Li, Chuan Zhang, Ximeng Liu, and Liehuang Zhu. 2023. EGIA: An External Gradient Inversion Attack in Federated Learning. *IEEE Transactions on Information Forensics and Security* (2023).

[94] Zhuofan Liao, Xiang Pang, Jingyu Zhang, Bing Xiong, and Jin Wang. 2021. Blockchain on security and forensics management in edge computing for IoT: A comprehensive survey. *IEEE Transactions on Network and Service Management* 19, 2 (2021), 1159–1175.

[95] Honggu Liu, Xiaodan Li, Wenbo Zhou, Han Fang, Paolo Bestagini, Weiming Zhang, Yuefeng Chen, Stefano Tubaro, Nenghai Yu, Yuan He, et al. 2023. BiFPro: A Bidirectional Facial-data Protection Framework against DeepFake. In *Proceedings of the 31st ACM International Conference on Multimedia*. 7075–7084.

[96] Jing Liu, Xuesong Hai, and Keqin Li. 2023. TDLearning: Trusted Distributed Collaborative Learning Based on Blockchain Smart Contracts. *Future Internet* 16, 1 (2023), 6.

[97] Jian Liu, Mika Juuti, Yao Lu, and Nadarajah Asokan. 2017. Oblivious neural network predictions via minionn transformations. In *Proceedings of the 2017 ACM SIGSAC conference on computer and communications security*. 619–631.

[98] Ning Liu, Xiaolong Ma, Zhiyuan Xu, Yanzhi Wang, Jian Tang, and Jieping Ye. 2020. Autocompress: An automatic dnn structured pruning framework for ultra-high compression rates. In *Proceedings of the AAAI Conference on Artificial Intelligence*, Vol. 34. 4876–4883.

[99] Xiaoyuan Liu, Hongwei Li, Guowen Xu, Zongqi Chen, Xiaoming Huang, and Rongxing Lu. 2021. Privacy-enhanced federated learning against poisoning adversaries. *IEEE Transactions on Information Forensics and Security* 16 (2021), 4574–4588.

[100] Xiang Liu, Xueli Zhao, Zhihua Xia, Qian Feng, Peipeng Yu, and Jian Weng. 2023. Secure Outsourced SIFT: Accurate and Efficient Privacy-Preserving Image SIFT Feature Extraction. *IEEE Transactions on Image Processing* (2023).

[101] Zechun Liu, Barlas Oguz, Changsheng Zhao, Ernie Chang, Pierre Stock, Yashar Mehdad, Yangyang Shi, Raghuraman Krishnamoorthi, and Vikas Chandra. 2023. Llm-qat: Data-free quantization aware training for large language models. *arXiv preprint arXiv:2305.17888* (2023).

[102] Adriana López-Alt, Eran Tromer, and Vinod Vaikuntanathan. 2012. On-the-fly multiparty computation on the cloud via multikey fully homomorphic encryption. In *Proceedings of the forty-fourth annual ACM symposium on Theory of computing*. 1219–1234.

[103] Jian-Hao Luo, Jianxin Wu, and Weiyao Lin. 2017. Thinet: A filter level pruning method for deep neural network compression. In *Proceedings of the IEEE international conference on computer vision*. 5058–5066.

[104] Ziyuan Luo, Qing Guo, Ka Chun Cheung, Simon See, and Renjie Wan. 2023. Copyrnerf: Protecting the copyright of neural radiance fields. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*. 22401–22411.

[105] Lingjuan Lyu, James C Bezdek, Jiong Jin, and Yang Yang. 2020. FORESEEN: Towards differentially private deep inference for intelligent Internet of Things. *IEEE Journal on Selected Areas in Communications* 38, 10 (2020), 2418–2429.

[106] Ningning Ma, Xiangyu Zhang, Hai-Tao Zheng, and Jian Sun. 2018. Shufflenet v2: Practical guidelines for efficient cnn architecture design. In *Proceedings of the European conference on computer vision (ECCV)*. 116–131.

[107] Rui Ma, Mengxi Guo, Yi Hou, Fan Yang, Yuan Li, Huizhu Jia, and Xiaodong Xie. 2022. Towards blind watermarking: Combining invertible and non-invertible mechanisms. In *Proceedings of the 30th ACM International Conference on Multimedia*. 1532–1542.

[108] Xirong Ma, Chuan Ma, Yali Jiang, and Chunpeng Ge. 2024. Improved privacy-preserving PCA using optimized homomorphic matrix multiplication. *Computers & Security* 138 (2024), 103658.
[109] Junaid Malik, Serkan Kiranyaz, and Moncef Gabbouj. 2020. FastONN–Python based open-source GPU implementation for Operational Neural Networks. *arXiv preprint arXiv:2006.02267* (2020).
[110] Junaid Malik, Serkan Kiranyaz, and Moncef Gabbouj. 2021. Self-organized operational neural networks for severe image restoration problems. *Neural Networks* 135 (2021), 201–211.
[111] Yunlong Mao, Shanhe Yi, Qun Li, Jinghao Feng, Fengyuan Xu, and Sheng Zhong. 2018. Learning from differentially private neural activations with edge computing. In *2018 IEEE/ACM Symposium on Edge Computing (SEC)*. IEEE, 90–102.
[112] Christian Meurisch and Max Mühlhäuser. 2021. Data protection in AI services: A survey. *ACM Computing Surveys (CSUR)* 54, 2 (2021), 1–38.
[113] Yinbin Miao, Ziteng Liu, Hongwei Li, Kim-Kwang Raymond Choo, and Robert H Deng. 2022. Privacy-preserving Byzantine-robust federated learning via blockchain systems. *IEEE Transactions on Information Forensics and Security* 17 (2022), 2848–2861.
[114] Payman Mohassel and Yupeng Zhang. 2017. Secureml: A system for scalable privacy-preserving machine learning. In *2017 IEEE symposium on security and privacy (SP)*. IEEE, 19–38.
[115] Mohammad Naseri, Yufei Han, and Emiliano De Cristofaro. 2023. BadVFL: Backdoor Attacks in Vertical Federated Learning. *arXiv preprint arXiv:2304.08847* (2023).
[116] Milad Nasr, Shuang Songi, Abhradeep Thakurta, Nicolas Papernot, and Nicholas Carlin. 2021. Adversary instantiation: Lower bounds for differentially private machine learning. In *2021 IEEE Symposium on security and privacy (SP)*. IEEE, 866–882.
[117] Rodion Novkin, Florian Klemme, and Hussam Amrouch. 2023. Approximation-and Quantization-Aware Training for Graph Neural Networks. *IEEE Trans. Comput.* (2023).
[118] Xudong Pan, Mi Zhang, Yifan Yan, Jiaming Zhu, and Zhemin Yang. 2022. Exploring the security boundary of data reconstruction via neuron exclusivity analysis. In *31st USENIX Security Symposium (USENIX Security 22)*. 3989–4006.
[119] Divyarajsinh N Parmar and Brijesh B Mehta. 2014. Face recognition methods & applications. *arXiv preprint arXiv:1403.0485* (2014).
[120] Nikolaos Passalis and Anastasios Tefas. 2018. Learning deep representations with probabilistic knowledge transfer. In *Proceedings of the European Conference on Computer Vision (ECCV)*. 268–284.
[121] Nikolaos Passalis, Maria Tzelepi, and Anastasios Tefas. 2020. Heterogeneous knowledge distillation using information flow modeling. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 2339–2348.
[122] Xiong Peng, Feng Liu, Jingfeng Zhang, Long Lan, Junjie Ye, Tongliang Liu, and Bo Han. 2022. Bilateral dependency optimization: Defending against model-inversion attacks. In *Proceedings of the 28th ACM SIGKDD Conference on Knowledge Discovery and Data Mining*. 1358–1367.
[123] Zirui Peng, Shaofeng Li, Guoxing Chen, Cheng Zhang, Haojin Zhu, and Minhui Xue. 2022. Fingerprinting deep neural networks globally via universal adversarial perturbations. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*. 13430–13439.
[124] Ngoc Duy Pham, Alsharif Abuadbba, Yansong Gao, Khoa Tran Phan, and Naveen Chilamkurti. 2023. Binarizing split learning for data privacy enhancement and computation reduction. *IEEE Transactions on Information Forensics and Security* 18 (2023), 3088–3100.
[125] Yuanhang Qi, M Shamim Hossain, Jiangtian Nie, and Xuandi Li. 2021. Privacy-preserving blockchain-based federated learning for traffic flow prediction. *Future Generation Computer Systems* 117 (2021), 328–337.
[126] Zuomin Qu, Zuping Xi, Wei Lu, Xiangyang Luo, Qian Wang, and Bin Li. 2024. DF-RAP: A Robust Adversarial Perturbation for Defending against Deepfakes in Real-world Social Network Scenarios. *IEEE Transactions on Information Forensics and Security* (2024).
[127] Yuhui Quan, Huan Teng, Ruotao Xu, Jun Huang, and Hui Ji. 2023. Fingerprinting Deep Image Restoration Models. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*. 13285–13295.
[128] Shashank Rajput, Hongyi Wang, Zachary Charles, and Dimitris Papailiopoulos. 2019. DETOX: A redundancy-based framework for faster and more robust gradient aggregation. *Advances in Neural Information Processing Systems* 32 (2019).
[129] Keshav Kasturi Rangan, Jihan Abou Halloun, Henrique Oyama, Samantha Cherney, Ilham Azali Assoumani, Nazir Jairazbhoy, Helen Durand, and Simon Ka Ng. 2022. Quantum computing and resilient design perspectives for cybersecurity of feedback systems. *IFAC-PapersOnLine* 55, 7 (2022), 703–708.
[130] Ronald L Rivest, Adi Shamir, and Leonard Adleman. 1978. A method for obtaining digital signatures and public-key cryptosystems. *Commun. ACM* 21, 2 (1978), 120–126.
[131] Adriana Romero, Nicolas Ballas, Samira Ebrahimi Kahou, Antoine Chassang, Carlo Gatta, and Yoshua Bengio. 2014. Fitnets: Hints for thin deep nets. *arXiv preprint arXiv:1412.6550* (2014).

[132] Bita Darvish Rouhani, M Sadegh Riazi, and Farinaz Koushanfar. 2018. Deepsecure: Scalable provably-secure deep learning. In *Proceedings of the 55th annual design automation conference*. 1–6.

[133] Mark Sandler, Andrew Howard, Menglong Zhu, Andrey Zhmoginov, and Liang-Chieh Chen. 2018. Mobilenetv2: Inverted residuals and linear bottlenecks. In *Proceedings of the IEEE conference on computer vision and pattern recognition*. 4510–4520.

[134] Alireza Sarmadi, Hao Fu, Prashanth Krishnamurthy, Siddharth Garg, and Farshad Khorrami. 2023. Privacy-Preserving Collaborative Learning through Feature Extraction. *IEEE Transactions on Dependable and Secure Computing* (2023).

[135] Reza Shokri and Vitaly Shmatikov. 2015. Privacy-preserving deep learning. In *Proceedings of the 22nd ACM SIGSAC conference on computer and communications security*. 1310–1321.

[136] Shivani Singh, Razia Sulthana, Tanvi Shewale, Vinay Chamola, Abderrahim Benslimane, and Biplab Sikdar. 2021. Machine-learning-assisted security and privacy provisioning for edge computing: A survey. *IEEE Internet of Things Journal* 9, 1 (2021), 236–260.

[137] Jinhyun So, Başak Güler, and A Salman Avestimehr. 2020. Byzantine-resilient secure federated learning. *IEEE Journal on Selected Areas in Communications* 39, 7 (2020), 2168–2181.

[138] Jingwei Sun, Ang Li, Binghui Wang, Huanrui Yang, Hai Li, and Yiran Chen. 2020. Provable defense against privacy leakage in federated learning from representation perspective. *arXiv preprint arXiv:2012.06043* (2020).

[139] Li Tang, Qingqing Ye, Haibo Hu, Qiao Xue, Yaxin Xiao, and Jin Li. 2024. DeepMark: A Scalable and Robust Framework for DeepFake Video Detection. *ACM Transactions on Privacy and Security* 27, 1 (2024), 1–26.

[140] Xinyu Tang, Saeed Mahloujifar, Liwei Song, Virat Shejwalkar, Milad Nasr, Amir Houmansadr, and Prateek Mittal. 2022. Mitigating membership inference attacks by {Self-Distillation} through a novel ensemble architecture. In *31st USENIX Security Symposium (USENIX Security 22)*. 1433–1450.

[141] Dat Thanh Tran, Serkan Kiranyaz, Moncef Gabbouj, and Alexandros Iosifidis. 2018. Progressive Operational Perceptron with Memory. *arXiv e-prints* (2018), arXiv–1808.

[142] Dat Thanh Tran, Serkan Kiranyaz, Moncef Gabbouj, and Alexandros Iosifidis. 2019. Heterogeneous multilayer generalized operational perceptron. *IEEE transactions on neural networks and learning systems* 31, 3 (2019), 710–724.

[143] Dat Thanh Tran, Serkan Kiranyaz, Moncef Gabbouj, and Alexandros Iosifidis. 2019. Knowledge transfer for face verification using heterogeneous generalized operational perceptrons. In *2019 IEEE international conference on image processing (ICIP)*. IEEE, 1168–1172.

[144] Marten Van Dijk, Craig Gentry, Shai Halevi, and Vinod Vaikuntanathan. 2010. Fully homomorphic encryption over the integers. In *Advances in Cryptology–EUROCRYPT 2010: 29th Annual International Conference on the Theory and Applications of Cryptographic Techniques, French Riviera, May 30–June 3, 2010. Proceedings 29*. Springer, 24–43.

[145] Esther Villar-Rodriguez, María Arostegi Pérez, Ana I Torre-Bastida, Cristina Regueiro Senderos, and Juan López-de Armentia. 2023. Edge intelligence secure frameworks: Current state and future challenges. *Computers & Security* 130 (2023), 103278.

[146] Yichen Wan, Youyang Qu, Longxiang Gao, and Yong Xiang. 2022. Privacy-preserving blockchain-enabled federated learning for B5G-Driven edge computing. *Computer Networks* 204 (2022), 108671.

[147] Cheng Wang, Zenghui Yuan, Pan Zhou, Zichuan Xu, Ruixuan Li, and Dapeng Oliver Wu. 2023. The security and privacy of mobile edge computing: An artificial intelligence perspective. *IEEE Internet of Things Journal* (2023).

[148] Hui Wang, Di Xiao, Min Li, Yanping Xiang, and Xinyan Li. 2019. A visually secure image encryption scheme based on parallel compressive sensing. *Signal Processing* 155 (2019), 218–232.

[149] Ji Wang, Jianguo Zhang, Weidong Bao, Xiaomin Zhu, Bokai Cao, and Philip S Yu. 2018. Not just privacy: Improving performance of private deep learning in mobile cloud. In *Proceedings of the 24th ACM SIGKDD international conference on knowledge discovery & data mining*. 2407–2416.

[150] Run Wang, Ziheng Huang, Zhikai Chen, Li Liu, Jing Chen, and Lina Wang. 2022. Anti-forgery: Towards a stealthy and robust deepfake disruption attack via adversarial perceptual-aware perturbations. *arXiv preprint arXiv:2206.00477* (2022).

[151] Run Wang, Felix Juefei-Xu, Meng Luo, Yang Liu, and Lina Wang. 2021. Faketagger: Robust safeguards against deepfake dissemination via provenance tracking. In *Proceedings of the 29th ACM International Conference on Multimedia*. 3546–3555.

[152] Tianhao Wang, Yuheng Zhang, and Ruoxi Jia. 2021. Improving robustness to model inversion attacks via mutual information regularization. In *Proceedings of the AAAI Conference on Artificial Intelligence*, Vol. 35. 11666–11673.

[153] Weiyang Wang, Moein Khazraee, Zhizhen Zhong, Manya Ghobadi, Zhihao Jia, Dheevatsa Mudigere, Ying Zhang, and Anthony Kewitsch. 2023. {TopoOpt}: Co-optimizing Network Topology and Parallelization Strategy for Distributed Training Jobs. In *20th USENIX Symposium on Networked Systems Design and Implementation (NSDI 23)*. 739–767.

[154] Xueyu Wang, Jiajun Huang, Siqi Ma, Surya Nepal, and Chang Xu. 2022. Deepfake disrupter: The detector of deepfake is my friend. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 14920–14929.

[155] Xingyuan Wang, Cheng Liu, and Donghua Jiang. 2022. An efficient double-image encryption and hiding algorithm using a newly designed chaotic system and parallel compressive sensing. *Information Sciences* 610 (2022), 300–325.
[156] Yijue Wang, Chenghong Wang, Zigeng Wang, Shanglin Zhou, Hang Liu, Jinbo Bi, Caiwen Ding, and Sanguthevar Rajasekaran. 2020. Against membership inference attack: Pruning is all you need. *arXiv preprint arXiv:2008.13578* (2020).
[157] Zhibo Wang, Hengchang Guo, Zhifei Zhang, Wenxin Liu, Zhan Qin, and Kui Ren. 2021. Feature importance-aware transferable adversarial attacks. In *Proceedings of the IEEE/CVF international conference on computer vision*. 7639–7648.
[158] Merrick S Watchorn and Q QIS. 2022. Quantum Chemistry for Detecting Cybersecurity Threats to Information Systems. (2022).
[159] Hanzhou Wu, Gen Liu, Yuwei Yao, and Xinpeng Zhang. 2020. Watermarking neural networks with watermarked images. *IEEE Transactions on Circuits and Systems for Video Technology* 31, 7 (2020), 2591–2601.
[160] Wen Wu, Mushu Li, Kaige Qu, Conghao Zhou, Xuemin Shen, Weihua Zhuang, Xu Li, and Weisen Shi. 2023. Split learning over wireless networks: Parallel design and resource management. *IEEE Journal on Selected Areas in Communications* 41, 4 (2023), 1051–1066.
[161] Xidong Wu, Feihu Huang, Zhengmian Hu, and Heng Huang. 2023. Faster adaptive federated learning. In *Proceedings of the AAAI Conference on Artificial Intelligence*, Vol. 37. 10379–10387.
[162] Xiaoshuai Wu, Xin Liao, and Bo Ou. 2023. Sepmark: Deep separable watermarking for unified source tracing and deepfake detection. In *Proceedings of the 31st ACM International Conference on Multimedia*. 1190–1201.
[163] Danyang Xiao, Chengang Yang, and Weigang Wu. 2021. Mixing activations and labels in distributed training for split learning. *IEEE Transactions on Parallel and Distributed Systems* 33, 11 (2021), 3165–3177.
[164] Guangxuan Xiao, Ji Lin, Mickael Seznec, Hao Wu, Julien Demouth, and Song Han. 2023. Smoothquant: Accurate and efficient post-training quantization for large language models. In *International Conference on Machine Learning*. PMLR, 38087–38099.
[165] Yinhao Xiao, Yizhen Jia, Chunchi Liu, Xiuzhen Cheng, Jiguo Yu, and Weifeng Lv. 2019. Edge computing security: State of the art and challenges. *Proc. IEEE* 107, 8 (2019), 1608–1631.
[166] Yaxin Xiao, Qingqing Ye, Haibo Hu, Huadi Zheng, Chengfang Fang, and Jie Shi. 2022. Mexmi: Pool-based active model extraction crossover membership inference. *Advances in Neural Information Processing Systems* 35 (2022), 10203–10216.
[167] An Xu, Zhouyuan Huo, and Heng Huang. 2021. Step-ahead error feedback for distributed training with compressed gradient. In *Proceedings of the AAAI Conference on Artificial Intelligence*, Vol. 35. 10478–10486.
[168] Dianlei Xu, Tong Li, Yong Li, Xiang Su, Sasu Tarkoma, Tao Jiang, Jon Crowcroft, and Pan Hui. 2021. Edge intelligence: Empowering intelligence to the edge of network. *Proc. IEEE* 109, 11 (2021), 1778–1837.
[169] Qiao Xue, Qingqing Ye, Haibo Hu, Youwen Zhu, and Jian Wang. 2022. DDRM: A continual frequency estimation mechanism with local differential privacy. *IEEE Transactions on Knowledge and Data Engineering* (2022).
[170] Hongyang Yan, Shuhao Li, Yajie Wang, Yaoyuan Zhang, Kashif Sharif, Haibo Hu, and Yuanzhang Li. 2022. Membership inference attacks against deep learning models via logits distribution. *IEEE Transactions on Dependable and Secure Computing* (2022).
[171] Haomiao Yang, Qixian Zhou, Jianbing Ni, Hongwei Li, and Xuemin Shen. 2020. Accurate image-based pedestrian detection with privacy preservation. *IEEE Transactions on Vehicular Technology* 69, 12 (2020), 14494–14509.
[172] Lei Yang, Can Zheng, Xiaoyuan Shen, and Guoqi Xie. 2023. OfpCNN: On-Demand Fine-Grained Partitioning for CNN Inference Acceleration in Heterogeneous Devices. *IEEE Transactions on Parallel and Distributed Systems* (2023).
[173] Tien-Ju Yang, Yu-Hsin Chen, and Vivienne Sze. 2017. Designing energy-efficient convolutional neural networks using energy-aware pruning. In *Proceedings of the IEEE conference on computer vision and pattern recognition*. 5687–5695.
[174] Xuanang Yang, Jing Chen, Kun He, Hao Bai, Cong Wu, and Ruiying Du. 2023. Efficient Privacy-preserving Inference Outsourcing for Convolutional Neural Networks. *IEEE Transactions on Information Forensics and Security* (2023).
[175] Yang Yang, Ming Cheng, Yingqiu Ding, and Weiming Zhang. 2023. A visually meaningful image encryption scheme based on lossless compression SPIHT coding. *IEEE Transactions on Services Computing* (2023).
[176] Ziqi Yang, Lijin Wang, Da Yang, Jie Wan, Ziming Zhao, Ee-Chien Chang, Fan Zhang, and Kui Ren. 2023. Purifier: defending data inference attacks via transforming confidence scores. In *Proceedings of the AAAI Conference on Artificial Intelligence*, Vol. 37. 10871–10879.
[177] Dayong Ye, Sheng Shen, Tianqing Zhu, Bo Liu, and Wanlei Zhou. 2022. One parameter defense—defending against data inference attacks via differential privacy. *IEEE Transactions on Information Forensics and Security* 17 (2022), 1466–1480.
[178] Seul-Ki Yeom, Kyung-Hwan Shim, and Jee-Hyun Hwang. 2021. Toward compact deep neural networks via energy-aware pruning. *arXiv preprint arXiv:2103.10858* (2021).
[179] Yupeng Yin, Xianglong Zhang, Huanle Zhang, Feng Li, Yue Yu, Xiuzhen Cheng, and Pengfei Hu. 2023. Ginver: Generative Model Inversion Attacks Against Collaborative Inference. In *Proceedings of the ACM Web Conference 2023*.

2122–2131.

[180] Ruichi Yu, Ang Li, Chun-Fu Chen, Jui-Hsin Lai, Vlad I Morariu, Xintong Han, Mingfei Gao, Ching-Yung Lin, and Larry S Davis. 2018. Nisp: Pruning networks using neuron importance score propagation. In *Proceedings of the IEEE conference on computer vision and pattern recognition*. 9194–9203.

[181] Xiaoyan Yu and Marin Marinov. 2020. A study on recent developments and issues with obstacle detection systems for automated vehicles. *Sustainability* 12, 8 (2020), 3281.

[182] Xiaohan Yuan, Jiqiang Liu, Bin Wang, Wei Wang, Tao Li, Xiaobo Ma, and Witold Pedrycz. 2023. FedComm: A Privacy-Enhanced and Efficient Authentication Protocol for Federated Learning in Vehicular Ad-hoc Networks. *IEEE Transactions on Information Forensics and Security* (2023).

[183] Zijie Yue, Shuai Ding, Lei Zhao, Youtao Zhang, Zehong Cao, Mohammad Tanveer, Alireza Jolfaei, and Xi Zheng. 2021. Privacy-preserving time-series medical images analysis using a hybrid deep learning framework. *ACM Transactions on Internet Technology (TOIT)* 21, 3 (2021), 1–21.

[184] Dun Zeng, Siqi Liang, Xiangjing Hu, Hui Wang, and Zenglin Xu. 2023. Fedlab: A flexible federated learning framework. *Journal of Machine Learning Research* 24, 100 (2023), 1–7.

[185] Feng Zhang, Xiatian Zhu, and Mao Ye. 2019. Fast human pose estimation. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*. 3517–3526.

[186] Jie Zhang, Dongdong Chen, Jing Liao, Han Fang, Weiming Zhang, Wenbo Zhou, Hao Cui, and Nenghai Yu. 2020. Model watermarking for image processing networks. In *Proceedings of the AAAI conference on artificial intelligence*, Vol. 34. 12805–12812.

[187] Jianqing Zhang, Yang Hua, Hao Wang, Tao Song, Zhengui Xue, Ruhui Ma, and Haibing Guan. 2023. Fedala: Adaptive local aggregation for personalized federated learning. In *Proceedings of the AAAI Conference on Artificial Intelligence*, Vol. 37. 11237–11244.

[188] Lan Zhang, Anran Li, Hongyi Peng, Feng Han, Fan Huang, and Xiang-Yang Li. 2024. Privacy-preserving Data Selection for Horizontal and Vertical Federated Learning. *IEEE Transactions on Parallel and Distributed Systems* (2024).

[189] Sai Qian Zhang, Jieyu Lin, and Qi Zhang. 2020. Adaptive distributed convolutional neural network inference at the network edge with ADCNN. In *Proceedings of the 49th International Conference on Parallel Processing*. 1–11.

[190] Xiangyu Zhang, Xinyu Zhou, Mengxiao Lin, and Jian Sun. 2018. Shufflenet: An extremely efficient convolutional neural network for mobile devices. In *Proceedings of the IEEE conference on computer vision and pattern recognition*. 6848–6856.

[191] Yuheng Zhang, Ruoxi Jia, Hengzhi Pei, Wenxiao Wang, Bo Li, and Dawn Song. 2020. The secret revealer: Generative model-inversion attacks against deep neural networks. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*. 253–261.

[192] Yifeng Zheng, Helei Cui, Cong Wang, and Jiantao Zhou. 2017. Privacy-preserving image denoising from external cloud databases. *IEEE Transactions on Information Forensics and Security* 12, 6 (2017), 1285–1298.

[193] Yifeng Zheng, Huayi Duan, Xiaoting Tang, Cong Wang, and Jiantao Zhou. 2019. Denoising in the dark: Privacy-preserving deep neural network-based image denoising. *IEEE Transactions on Dependable and Secure Computing* 18, 3 (2019), 1261–1275.

[194] Jun Zhou, Meng Zheng, Zhenfu Cao, and Xiaolei Dong. 2020. PVIDM: Privacy-preserving verifiable shape context based image denoising and matching with efficient outsourcing in the malicious setting. *Computers & Security* 88 (2020), 101631.

[195] Zhi Zhou, Xu Chen, En Li, Liekang Zeng, Ke Luo, and Junshan Zhang. 2019. Edge intelligence: Paving the last mile of artificial intelligence with edge computing. *Proc. IEEE* 107, 8 (2019), 1738–1762.

[196] Jiren Zhu, Russell Kaplan, Justin Johnson, and Li Fei-Fei. 2018. Hidden: Hiding data with deep networks. In *Proceedings of the European conference on computer vision (ECCV)*. 657–672.

[197] Jun-Yan Zhu, Taesung Park, Phillip Isola, and Alexei A Efros. 2017. Unpaired image-to-image translation using cycle-consistent adversarial networks. In *Proceedings of the IEEE international conference on computer vision*. 2223–2232.

Received 20 February 2007; revised 12 March 2009; accepted 5 June 2009