Verifiable Privacy-Enhanced Rotation Invariant LBP Feature Extraction in Fog Computing

Mingyun Bian, Joseph Liu, *Senior Member, IEEE*, Shifeng Sun, Xinpeng Zhang, and Yanli Ren

***Abstract*—Rotation invariant local binary pattern (RI-LBP) features have been applied in diverse scenarios with the advantages of gray-scale and rotation invariance. Secure fog computing has become an emerging paradigm for enterprises or individuals with a huge volume of private data, but limited computing power for feature extraction. Prior secure outsourcing protocols based on LBP and RI-LBP simply focus on local data privacy, which can only resist ciphertext-only attack, and also make extracted features exposed to the cloud. This work focuses on how to effectively ensure data confidentiality and feature integrity. We propose a verifiable privacy-enhanced protocol for RI-LBP feature extraction (VRLBP) based on the fog computing paradigm, which mitigates the aforementioned challenges by involving the proposed symmetric cryptographic scheme where local data and extracted features are proven secure against chosen plaintext attack. Meanwhile, the stage of verification can check the correctness of outsourced features with an overwhelming probability and constant computational complexity. The security analysis and computational costs demonstrate that VRLBP can reduce the computation overhead to around 30% of original feature extraction in a privacy-preserving manner. To exhibit the practical utility, VRLBP is implemented for deepfake detection on five public datasets. Extensive evaluations indicate that VRLBP achieves almost the same accuracy as the original RI-LBP algorithm and outperforms the state-of-the-art protocols.**

***Index Terms*—Deepfake detection, feature extraction, rotation invariant local binary pattern (RI-LBP), secure fog computing.**

Manuscript received 22 August 2022; revised 16 December 2022 and 20 January 2023; accepted 15 February 2023. Date of publication 22 February 2023; date of current version 18 October 2023. This work was supported in part by the National Natural Science Foundation of China under Grant U22B2047, and in part by the Natural Science Foundation of Shanghai under Grant 20ZR1419700 and Grant 22ZR1481000. Paper no. TII-22-3598. *(Corresponding author: Yanli Ren.)*

Mingyun Bian, Xinpeng Zhang, and Yanli Ren are with the School of Communication and Information Engineering, Shanghai University, Shanghai 200444, China (e-mail: bmy0537@shu.edu.cn; xzhang@shu.edu.cn; renyanli@shu.edu.cn).

Joseph Liu is with the Faculty of Information Technology, Monash University, Clayton, VC 3800, Australia (e-mail: joseph.liu@monash.edu).

Shifeng Sun is with the School of Cyber Science and Engineering, Shanghai Jiao Tong University, Shanghai 200240, China (e-mail: shifeng.sun@sjtu.edu.cn).

Color versions of one or more figures in this article are available at https://doi.org/10.1109/TII.2023.3246992.

Digital Object Identifier 10.1109/TII.2023.3246992

I. Introduction

IN the era of the Internet of Things (IoT), huge amounts of data are processed from IoT systems with limited computing and defense capabilities [1]. Centralized cloud computing infrastructure demands extensive costs for consistent implementation, and the rapid development of fog computing brings considerable advantages in public resource usage efficiency and economic productivity. Efficient and flexible fog computing facilitates public services with sufficient data storage and computing power to reduce time costs. Fog computing achieves decentralized computing by the IoT devices and cloud platform, and the hierarchical architecture is shown in Fig. 1. In the meantime, fog nodes are intended for processing part of the applications' workload locally to the clients' devices. Therefore, the computationally weak end-user tends to transport big data with confidential information to a local fog server, and then deposit it in a cloud. Such practices seem intriguing, yet raise critical security and privacy concerns in smart IoT systems. To ensure individuals' privacy, the confidentiality of data and the query results should not be disclosed in a fog-cloud environment. Rotation invariant local binary pattern (RI-LBP) [2] has attracted considerable attention in different research areas.

The advance of modern digital image processing tools and deep learning has resulted in a flood of forgery images or videos on the Internet, such as deepfake, which synthesizes and manipulates real videos of humans based on deep learning, to create fake videos with high fidelity. Several algorithms based on LBP focus on deepfake detection to uncover deceived images [3], [4], [5]. Arini et al. [3] applied an LBP descriptor for image feature preprocessing to filter the image processing effect of deepfake generation. Remya et al. [4] exploited an efficient deepfake detection model incorporating deep RI-LBP features sourced from chrominance space. To achieve faster execution, Suganthi et al. [5] utilized a fusion of the fisherface-LBP histogram algorithm for reducing the dimensionality of features and execution of detection time.

Fog computing, as a decentralized computing infrastructure, complements cloud computing with the ability to store and analyze data temporarily. In an architecture of fog computing, more owners are willing to delegate the heavy computational workloads of image feature extraction to fog-cloud servers. There exist some potential risks in cyberspace and some researchers have paid increasing attention for privacy-preserving computation to overcome potential security risks in various scenarios. Revathi et al. [6] developed a secure healthcare data publishing protocol by integrating adaptive fractional brain storm with

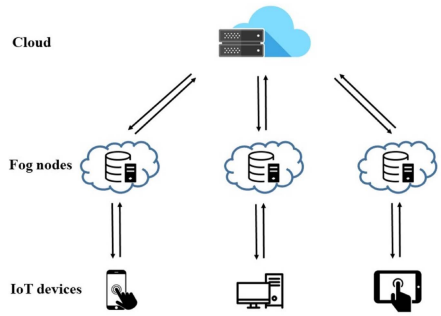

Fig. 1. Architecture of fog computing.

whale optimization algorithm to protect the privacy of healthcare data while keeping data utility. Liu et al. [7] proposed the first dynamic searchable symmetric encryption protocol and achieved a secure index based on the hash chain and trapdoor updates to resist file injection attacks and protect patients' privacy. Deebak et al. [8] provided an identity-based seamless secure protocol based on cryptography and hash-centered rank for resisting quantum-computer attack. To guarantee the validity of cloud data efficiently, Gao et al. [9] put forward an integrity auditing protocol by utilizing a private keyword without exposing any identity of data containing the queried keyword.

Most privacy-preserving works for feature processing based on homomorphic encryption scheme, multiparty computation, secret sharing, and order-preserving encryption (OPE) [10], [11], [12], [13], [14], [15]. There exist tradeoff requirements between utility and privacy preservation without uniform standards. Recently, Hsu et al. [10] conducted a pioneer study of secure scale-invariant feature transform (SIFT) computation in the ciphertext domain by using Paillier's homomorphic encryption, but its deficiencies of intractable and insecure computation on the server side could not be ignored. To achieve image global feature detection algorithms in the ciphertext domain, Qin et al. [11] made it feasible by deploying somewhat homomorphic encryption to convert feature descriptor detection in the plaintext domain into circuit-level operations with a security guarantee. Hu et al. [12] applied the same encryption method as [11] to secure computation on SIFT based on additive secret sharing among two servers.

In recent years, there have been growing research endeavors on building LBP-based privacy-preserving protocols [13], [14], [15]. Xia et al. [13], [14] designed secure image feature extraction protocols on LBP descriptors and proposed special image encryption algorithms, including image block shuffling and pixel substitution operations, in the foundation of OPE, but it can only resist the ciphertext-only attack (COA). Similar to [14], Wang et al. [15] proposed a privacy-preserving rotation invariant LBP feature computation protocol including block shuffling, pixel shift, and diffusion. Although the content of original image sets is protected by OPE [13], [14], [15], it cannot offer strong security to ciphertext statistic information, which may be divulged to semihonest adversaries. Currently, there exist the following security challenges in the privacy-preserving RI-LBP feature extraction. First, the statistic information of outsourced features is exposed to the adversaries in an untrusted environment. Second, the outsourced results may be deceived by the cloud for intentional or unintentional purposes. Third, the ciphertext of data can only resist ciphertext-only attack. To address these security challenges, a verifiable privacy-enhanced RI-LBP protocol is proposed, which can not only protect the content of sensitive data and outsourced features against chosen plaintext attack, but also check the integrity of outsourced results effectively. The main contributions are listed as follows.

1) To pave the way for secure RI-LBP feature extraction in a fog-cloud environment, we propose a lightweight cryptographic scheme based on the approximate greatest common divisor problem (GCD) [16], which supports basic building blocks of the proposed protocol.
2) We design a verifiable privacy-enhanced RI-LBP feature extraction protocol (VRLBP) in fog computing. It not only protects the privacy of both the local data and outsourced results, but also checks the integrity of outsourced RI-LBP features with an overwhelming probability and constant computational complexity.
3) Based on the proposed cryptographic scheme, we provide detailed security proof and prove that VRLBP achieves a higher security strength against chosen plaintext attack than related works [13], [14], [15]. Simultaneously, the constant computational complexity in verification is superior to [17], [18], [19], and [20] with linear computational complexity.
4) Experiments on the computational costs on the client side show that VRLBP decreases around 70% of the original computation overhead. In the meantime, VRLBP gains time efficiency in verification compared with [17] and [18]. To evaluate the performance of secure feature extractions, we apply VRLBP to deepfake detection on five datasets. The utility is almost identical to the original RI-LBP algorithm and outperforms the related works [13], [14], [15].

The rest of the article is organized as follows. In Section II, we present the background including the symmetric encryption schemes, RI-LBP algorithm, and system model. In Section III, we introduce the proposed encryption scheme and verifiable privacy-enhanced RI-LBP protocol in detail. We carry out the security analysis in Section IV, and evaluate the proposed protocol in Section V. Finally, Section VI concludes this article.

II. Background

In this section, the related background of research topic and the system model structure with security goals are demonstrated briefly.

A. Security Notions of Symmetric Encryption Schemes

Based on the approximate GCD problems [16], a specific assumption is made to describe symmetric encryption schemes, which is constructed as follows.

According to [16], there is a corresponding assumption to be stated. λ is the security parameter used to output secret key p. For a λ-bit odd positive integer p, the distribution over

TABLE I
TABLE OF MAIN NOTATIONS

Notation	Description
$\mathcal{E}$	Proposed encryption scheme
(λ, p)	Security parameter, secure key
(r, q)	Random integers in encryption scheme
$[\![\cdot]\!]$	A cipher generated by $\mathcal{E}$
$[\![\cdot]\!]_p$	The modular arithmetic operation on the divisor of p
κ	The number of image patches and fog nodes
D/D^e	The plaintext/ciphertext of input data
(x_j, y_j)	The coordinates of the j-th neighbor
$M(\cdot)$	Mapping the coordinates into values one-to-one
V	Local neighborhood of an image patch
σ	Private permutation matrix
$q_p(z)/r_p(z)$	The quotient/remainder of z to p
T	The total number of images
PF	Secure RI-LBP feature histograms over an image patch
SF	Secure RI-LBP feature histograms over an image
R_ξ^o/R_ξ^s	Original/secure RI-LBP feature histograms
τ	Threshold value in verification
$\mathscr{A}$	An adversary
$\mathscr{C}$	A challenger

γ-bit integers equals to $\mathscr{D}_{\gamma,\rho}(p) = \{q \xleftarrow{R} \mathbb{Z} \bigcap [0, 2^\gamma/p), r \xleftarrow{R} \mathbb{Z} \bigcap (-2^\rho, 2^\rho)$: Output $\varrho = pq + 256r\}$, where q and r are the random integers constrained in the intervals of $[0, 2^\gamma/p)$ and $(-2^\rho, 2^\rho)$, respectively. It should be noted that the $256r$ is substantially smaller than $p/2$ in absolute value. In short, (λ, γ, ρ)-Approximate $-$ GCD Problem is the hardness of recovering p based on some polynomially samples derived from $\mathscr{D}_{\gamma,\rho}(p)$ [16], [29].

IND-CPA security: The indistinguishability of ciphertexts against chosen plaintext attack (IND-CPA) is represented by the upcoming adversarial procedures of a cryptographic game between a challenger $\mathscr{C}$ and an adversary $\mathscr{A}$.

1) $\mathscr{C}$ first generates public parameters PP and the secret key p based on the initial security parameter λ, then broadcasts PP to $\mathscr{A}$ while secret key p is kept by $\mathscr{C}$ privately.
2) $\mathscr{A}$ submits message m for encryption queries. $\mathscr{C}$ performs encryption operations to return the ciphertext of message m to $\mathscr{A}$.
3) $\mathscr{A}$ generates two chosen plaintexts m_0, m_1 with the same length and transmits them to $\mathscr{C}$. $\mathscr{C}$ randomly picks m_α where $\alpha \in \{0, 1\}$ and encrypts it into c_α based on p. Eventually, $\mathscr{C}$ sends c_α back to $\mathscr{A}$.
4) $\mathscr{A}$ completes its guess $\alpha' \in \{0, 1\}$. If $\alpha' = \alpha$, and $\mathscr{A}$ wins the game.

In the above game, $\mathscr{A}$ is defined as an IND-CPA adversary and the advantage is defined as $|\Pr[\alpha' = \alpha] - \frac{1}{2}|$. (u, ϵ)-IND-CPA secure means all u-time IND-CPA adversaries with the advantage of no more than ϵ to win the game. For a probabilistic encryption system, it is regarded as IND-CPA secure if any IND-CPA adversary wins the game with a negligible advantage.

B. Feature Extraction on RI-LBP

RI-LBP descriptor was first proposed in [2] with the advantages of sample theory and effective computation, and rotation invariance property derived from LBP descriptor, it has been proved to be applicable in different fields. For a better understanding, the process of feature extraction on the RI-LBP descriptor is elaborated in Fig. 2 based on a 3×3 neighborhood with a radius of 1. The whole process of monochrome texture image feature extraction on RI-LBP consists of three parts, i.e., 1) local neighborhood initialization; 2) toward gray-scale invariance; and 3) toward rotation invariance.

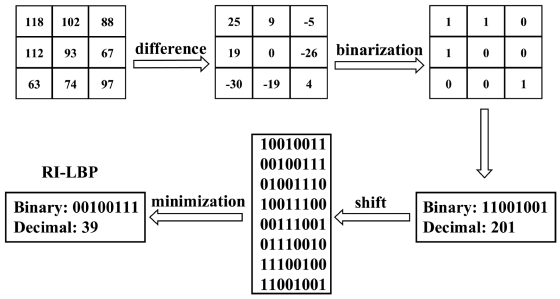

Fig. 2. RI-LBP descriptor with 3×3 neighborhood.

To initialize a local neighborhood, a neighbor set is defined as $V = \{v_c, v_0, v_1, v_2, \ldots, v_{N-1}\}$, where v_c denotes the gray value to the center pixel of a local neighborhood, N ($N > 1$) denotes the maximum number of neighbors on a circle of radius r, and v_i ($i = 0, 1, \ldots, N-1$) denotes gray values corresponding to coordinates of a local neighborhood, which are calculated by

$$x_i = x_c + r\cos\left(\frac{2\pi i}{N}\right),$$
$$y_i = y_c - r\sin\left(\frac{2\pi i}{N}\right) \quad (1)$$

if x_i and y_i are two float point numbers, p_i can be figured out by bilinear interpolation computation.

In order to achieve gray-scale invariance, we calculate the differences of pixel values $v_i - v_c$ to get $\{\delta_i\}_{i=0}^{N-1} = \{v_0 - v_c, v_1 - v_c, \ldots, v_{N-1} - v_c\}$. The neighbors are signed as "1" when the neighborhood pixel value are bigger than the center pixel value, otherwise "0". The calculation of gray-scale invariant features is

$$G = \sum_{i=0}^{N-1} s(v_i - v_c)2^n, \quad (2)$$

$$s(x) = \begin{cases} 1, & x \geq 0 \\ 0, & \text{otherwise.} \end{cases} \quad (3)$$

To make improvements on gray-scale invariance and achieve the rotation variance, the results of RI-LBP features LBP^{RI} are calculated by

$$\text{LBP}^{RI} = \min(\{ROR(G, i) | i = 0, 1, \ldots, N-1\}) \quad (4)$$

where $ROR(G, i)$ means the i times bit-wise operation of the circular right shift in the clockwise direction on a candidate number G, and the function of $\min(\cdot)$ finds the smallest item among a set of inputs. As shown in Fig. 2, the decimal conversion of the binary number $(00100111)_2$ is the final result. We summarize the main notations in Table I.

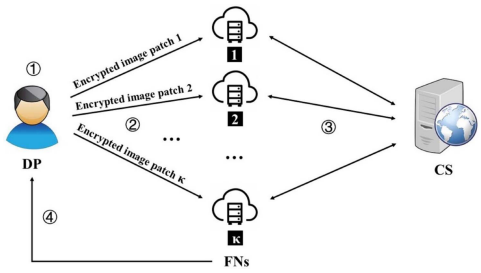

Fig. 3. System model.

C. System Model

Inspired by three hierarchies of the fog computing architecture, as shown in Fig. 3, system model consists of three kinds of entities, including data provider (DP), fog nodes (FNs), and cloud server (CS). They are as follows.

1) *DP:* With the massive multimedia image data collected from the Internet, DP sends several queries to CS via the local FNs to extract RI-LBP features for personal needs (e.g., deepfake detection). Meanwhile, DP can verify the integrity of outsourced results.
2) *FNs:* FNs bridge DP and CS, which have relative advantages in data storage with low network latency consumption for DP. They will store the outsourced data and cooperate with CS to provide the secure feature extraction services for DPs.
3) *CS:* CS intends to get a deep insight into the multimedia image patches stored in every FN, and communicates with multiple FNs asynchronously to solve DP's queries. Suppose that both FNs and CS will confirm to the proposed protocol in principle.

D. Security Model

In this section, the threat model and security requirements of VRLBP are presented as below.

1) Threat Model: As illustrated in Fig. 3, the adversaries have divided into three categories: 1) untrusted CS; 2) malicious FNs; and 3) the third-party adversaries.

Untrusted CS: CS is untrusted in VRLBP, which means it not only may try to compromise the private knowledge of DP from their ciphertext data, but also may feedback incorrect computational results to DP. During the protocol execution, it may collude with some malicious FNs to infer the private intermediate results and deceive DPs with error results. VRLBP requires that at most $\kappa - 1$ malicious FNs can collude with CS for security considerations, where κ denotes the total number of FNs.

Malicious FNs: In VRLBP, malicious FNs may analyze the sensitive knowledge of DPs from their ciphertext data. They may also collude with the untrusted CS to obtain the private information of outsourced features and generate meaningless outsourced results deliberately for illegal uses.

Third-party adversaries: Apart from untrusted CS and malicious FNs, some potential adversaries outside the VRLBP are defined as the third-party adversaries, which can eavesdrop, investigate, and manipulate the messages transmitted by the insecure public channels.

2) Security Goals: As mentioned above, for such a malicious model, the security goals are listed below as follows.

1) *Data privacy:* FNs and CS should not gain any content of local image sets at any time for illegal purposes. In the same time, it should not exposed to the third-party adversaries during data transmission.
2) *Feature privacy:* The intermediate results and final outputs should not be disclosed to FNs and CS in cleartexts while protocol executes. The third-party adversaries have no way to infer the content of extracted features.
3) *Feature integrity:* DP will receive correct results if FNs and CS conform to the protocol. Under a malicious setting, the correctness of outsourced results needs to be verified by DP.
4) *Efficiency:* Outsourcing computation should solve the tasks towards limited client resources in that the computational burden of DP in the privacy-preserving protocol should be significantly lower than the original RI-LBP algorithm.

III. Proposed Privacy-Enhanced Feature Extraction Protocol of RI-LBP

In this section, the proposed symmetric encryption scheme and privacy-enhanced protocol for RI-LBP feature extraction are described in detail.

A. Proposed Symmetric Encryption Scheme

The proposed symmetric encryption scheme $\mathcal{E}$ consists of three main components.

1) *KeyGen(λ):* Generate the λ-bit secret key p under the input of security parameter λ.
2) *Enc(p, m):* Select two random integers r, q in $(-2^{\rho}, 2^{\rho})$ and $[0, 2^{\gamma}/p)$, respectively, where $256r$ is substantially smaller than $p/2$ in absolute value. Given a plaintext message $m \in [0, 255]$ and the private key p, the ciphertext of m is calculated
$$c = pq + 256r + m. \tag{5}$$
3) $Dec(p, c)$: Output m based on the input of p and c
$$m = (c \bmod p) \bmod 256 \tag{6}$$
where the results of the modular arithmetic operation on the divisor of p and 256 range from $[0, p-1]$ and $[0, 255]$, respectively.

It should be pointed out that the symmetric encryption scheme turns out to be "somewhat homomorphic" if the noise r is small enough compared with the secret key p. It supports the evaluation of low-degree polynomials over ciphertexts. In other words, the symmetric encryption scheme provides finite arithmetic operations of addition and multiplication over ciphertexts [21].

Correctness: As mentioned before, the noise of $Encrypt$ is substantially smaller than p in absolute value, (i.e., $|256r| \ll p$).

Then, the ciphertext output of $c = pq + 256r + m$ with $|256r + m| < p$. Therefore, we obtain $[\![c]\!]_p = 256r + m$, and $m = [\![[\![c]\!]_p]\!]_{256}$, where $[\![\cdot]\!]_p$, $[\![\cdot]\!]_{256}$ mean the modular arithmetic operation on the divisor of p and 256, respectively.

More importantly, the proposed protocol is built based on the addictive homomorphic properties of $\mathcal{E}$

$$Enc_p(m_1) \oplus Enc_p(m_2) = Enc_p(m_1 + m_2), \tag{7}$$

$$a \cdot Enc_p(m) = Enc_p(a \cdot m) \tag{8}$$

where a denotes a positive scalar integer and the results of $(a \cdot m)$ belong to $[0, 255]$.

Algorithm 1: Query Construction.

Input:
DP : The image data set with identity codes $(D = \{d_i\}_{i=1}^{T}, ID = \{id_i\}_{i=1}^{T})$, the symmetric encryption algorithm $\mathcal{E}$ and secret key p.

Output:
The encrypted image patches with identity codes $(D^e = \{\{[\![d_{ij}]\!]\}_{i=1}^{T}\}_{j=1}^{\kappa}, ID)$.

1: **for** each $i \in [1, T]$ **do**
2: $[\![d_i]\!] = \mathcal{E}.Enc_p(d_i)$, and $[\![d_i]\!]$ is splitted into $\{[\![d_{ij}]\!]\}_{j=1}^{\kappa}$.
3: Send $\{[\![d_{ij}]\!]\}_{j=1}^{\kappa}$ to *FN*s with randomly shuffling.
4: **end for**

B. Proposed Privacy-Enhanced Protocol

Now, the VRLBP is introduced in detail, where the data provider offloads feature extraction tasks to fog servers. The system framework of VRLBP is comprised of the following four stages.

1) *Initialization:* The shared secret key of the symmetric encryption scheme is established by mutual agreement between DP and CS.
2) *Query construction:* DP first encrypts raw image sets by the secret key and splits every encrypted image into κ $(\kappa > 1)$ patches without overlapping. Then, encrypted image patches are transferred to all of FNs as queries, and let every FN receive one patch per encrypted image randomly.
3) *Feature extraction:* FNs and CS extract secure features interactively according to the protocol, and outsourced results are obtained by FNs.
4) *Feature verification:* DP decrypts and verifies the correctness of outsourced features responded by FNs.

The concrete construction of four stages are described in the next paragraphs.

Initialization: In the first stage, each entity implements initialization preparation. The initialization phase consists of the following two parts.

1) DP generates and shares a symmetric key p of $\mathcal{E}$ with CS, which is kept privately by two parties for data protection and feature analysis.

2) There are κ FNs for data collection and the system parameter κ is broadcast to DP.

Query construction: For a dataset D, DP performs the following two steps to form queries for outsourcing computation services.

1) In a period of time, DP collects a dataset $D = \{d_i | i \in [1, T]\}$ with T images from the Internet, and encrypts them by p.

2) DP splits every encrypted image into κ patches, and delivers them to FNs with randomly shuffling, which makes every FN obtain one image patch per encrypted image.

Feature extraction: On receiving the queries from DP, every FN picks an image patch to calculate RI-LBP features with CS through Algorithm 2–4.

As shown in Fig. 2, there exist four kinds of operations in the plaintext algorithm, where four main subprotocols over a neighborhood are summarized in Algorithm 2–4 and the intact feature statistics over an image patch are obtained by Algorithm 5.

1) *Secure difference:* FN first selects an image patch and initiates a local neighborhood as V. The center coordinates are noted as (x_c, y_c), the another coordinates of eight neighbors are figured out by $(\{x_j\}_{j=0}^{7} = \{x_c + \cos(\frac{2\pi j}{8})\}_{j=0}^{7}$, $\{y_j\}_{j=0}^{7} = \{y_c - \sin(\frac{2\pi j}{8})\}_{j=0}^{7})$. For some floating-point numbers in $\{x_j, y_j\}_{j=0}^{7}$, $[\![v_j]\!]$ is calculated by

$$\begin{aligned}[[v_j]] = &(\lceil x_j \rceil - x_j) \cdot (\lceil y_j \rceil - y_j) \cdot M(\lfloor x_j \rfloor, \lfloor y_j \rfloor) \\ &\oplus (\lceil x_j \rceil - x_j) \cdot (y_j - \lfloor y_j \rfloor) \cdot M(\lceil x_j \rceil, \lfloor y_j \rfloor) \\ &\oplus (x_j - \lfloor x_j \rfloor) \cdot (\lceil y_j \rceil - y_j) \cdot M(\lfloor x_j \rfloor, \lceil y_j \rceil) \\ &\oplus (x_j - \lfloor x_j \rfloor) \cdot (y_j - \lfloor y_j \rfloor) \cdot M(\lceil x_j \rceil, \lceil y_j \rceil)\end{aligned} \tag{9}$$

where $M(\cdot)$ is a function that mapping the possible coordinates into values one-to-one. $\lceil \cdot \rceil$ returns a number rounded up to a input number of decimal places, $\lfloor \cdot \rfloor$ returns a number rounded down to an input number of decimal places.

To satisfy the encryption mechanism $\mathcal{E}$, the floating-point number in $\{[\![v_j]\!]\}_{j=0}^{7}$ is transformed into the integer by

$$[\![v_j]\!] = \begin{cases} \lceil [\![v_j]\!] \rceil, & [\![v_j]\!] \ominus \lfloor [\![v_j]\!] \rfloor > \lceil [\![v_j]\!] \rceil \ominus [\![v_j]\!] \\ \lfloor [[v_j]] \rfloor, & \text{otherwise.} \end{cases} \tag{10}$$

Then, the set $\{[\![\delta_k]\!]\}_{k=0}^{7}$ is calculated by subtracting the center pixel from neighbor pixels. To guarantee the security, FN masks $\{[\![\delta_k]\!]\}_{k=0}^{7}$ by a private and random permutation matrix σ with size of 8×8, where permutation matrix is a binary matrix with only one entry of 1 in each row and each column.

2) *Secure binarization:* As CS owns the secret key p, the set of $\{[\![\hat{\delta}_k]\!]\}_{k=0}^{7}$ can be binarized to B with the threshold of 0, where values greater than or equal to the given limit map to 1; otherwise values map to 0.

3) *Secure shift:* To restore the original order of $\hat{B}$, FN utilizes their own σ^{-1} to get B, and calculate eight results through at most seven times bit-wise operations of the circular right shift in clockwise direction based on homomorphic addition.

Algorithm 2: Secure Difference and Binarization.

Input:
FN: An encrypted image patch.
CS: The secret key p of $\mathcal{E}$.
Output:
The results of secure difference and binary operation $\hat{B}$.

FN initializes a set $V = \{[\![v_c]\!], [\![v_0]\!], [\![v_1]\!], \ldots, [\![v_7]\!]\}$ as a local neighborhood of an image patch where $[\![v_c]\!]$ is the center value in ciphertext.
for each $i \in [0, 7]$ **do**
  $[\![\delta_i]\!] = [\![v_i]\!] \textcircled{-} [\![v_c]\!]$.
end for
Choose a random permutation matrix σ to get $\{[\![\hat{\delta}_i]\!]\}_{i=0}^{7} = \{[\![\delta_i]\!]\}_{i=0}^{7} \cdot \sigma$.
Send $\{[\![\hat{\delta}_i]\!]\}_{i=0}^{7}$ to CS.
for each $i \in [0, 7]$ **do**
  CS binarizes the plaintext of $[\![\hat{\delta}_i]\!]$ with a threshold of 0 to get $\hat{b}_i$.
  Encrypt $\hat{b}_i$ to get $[\![\hat{b}_i]\!] = \mathcal{E}.Enc_p(\hat{b}_i)$.
end for
Send $\hat{B} = \{[\![\hat{b}_i]\!]\}_{i=0}^{7}$ to FN.

Algorithm 3: Secure Shift.

Input:
FN :$\hat{B}$, the inverse matrix σ^{-1}.
Output:
The results of secure shifting operation $\{Tc_i\}_{i=0}^{7}$.

Cancel permutation and get $B = \hat{B} \cdot \sigma^{-1} = \{[\![b_i]\!]\}_{i=0}^{7}$.
for each $i \in [0, 7]$ **do**
  Convert $\{[\![b_i]\!]\}_{i=0}^{7}$ to the decimal with eight results:
  $[\![Td]\!]_i = [\![b_0]\!] \cdot 2^{(7-i) \bmod 8} \oplus [\![b_1]\!] \cdot 2^{(6-i) \bmod 8} \oplus [\![b_2]\!] \cdot 2^{(5-i) \bmod 8} \oplus [\![b_3]\!] \cdot 2^{(4-i) \bmod 8} \oplus [\![b_4]\!] \cdot 2^{(3-i) \bmod 8} \oplus [\![b_5]\!] \cdot 2^{(2-i) \bmod 8} \oplus [\![b_6]\!] \cdot 2^{(1-i) \bmod 8} \oplus [\![b_7]\!] \cdot 2^{(0-i) \bmod 8}$.
end for
for each $i \in [0, 7]$ **do**
  if $i < 7$ **then**
    for each $j \in [i + 1, 7]$ **do**
      $Tc_i = \{'Tc_i, '[\![Td_i]\!] \textcircled{-} [\![Td_j]\!]\}$.
    end for
  else
    $Tc_7 = \{'Tc_7, '[\![Td_7]\!] \textcircled{-} [\![Td_6]\!]\}$.
  end if
end for
Send $\{Tc_i\}_{i=0}^{7}$ to CS.

4)*Secure minimization:* To find the minimum value in a neighborhood, CS finds the first subset that only contains nonpositive elements in $\{Tc_i\}_{i=0}^{7}$. Based on the predefined label set $\{[\![Td_i]\!]\}_{i=0}^{7}$, FN will find the minimum cipher.

Suppose there are t neighborhoods in an image patch, a ciphertext feature set $\{[\![c_i]\!]\}_{i=1}^{t}$ is extracted by Algorithm 2–4, the principal task is to count the feature values in total. Followed by

Algorithm 4: Secure Minimization.

Input:
CS :$\{Tc_i\}_{i=0}^{7}$, secret key p of $\mathcal{E}$.
Output:
The label of minimum value in a neighborhood $'Tc_i'$.

for each $i \in [0, 7]$ **do**
  Decrypt $\{Tc_i\}$ to get a difference set in plaintext $\{Tp_i\} = \mathcal{E}.Dec_p(\{Tc_i\})$.
  if all elements in $\{Tp_i\} \leq 0$ **then**
    Store the corresponding label $'Tc_i'$ and break.
  else
    Continue.
  end if
end for
Send the label of minimum value $'Tc_i'$ to FN.

Algorithm 5: Secure Feature Set Over an Image Patch.

Input:
FN: Given a feature set of ciphertext $\{[\![c_i]\!]\}_{i=1}^{t}$.
CS: Secret key p of $\mathcal{E}$.
Output:
The secure RI-LBP feature set PF over an image patch.

for each $i \in [1, t-1]$ **do**
  foreach $j \in [j+1, t]$ **do**
    $TC_i = \{'TC_i, '[\![c_i]\!] \textcircled{-} [\![c_j]\!]\}$.
  end for
end for
$TC_t = \{'TC_t, 'TC_t = [\![c_t]\!] \textcircled{-} [\![c_{t-1}]\!]\}$.
Send $\{TC_i\}_{i=1}^{t}$ to CS.
for each $i \in [1, t]$ **do**
  CS gets $TP_i = \mathcal{E}.Dec_p(TC_i)$.
end for
for each $i \in [1, t]$ **do**
  Count the numbers of value 0 in TP_i, which is noted as tc_i, and store it with corresponding label $'TC_i'$.
end for
Send $\{'TC_i, 'tc_i\}_{i=1}^{t}$ to *FN*.
FN gets a secure feature set $PF = \{\{[\![c_i]\!], tc_i\}_{i=1}^{t}, id\}$ via the label sets $\{'TC_i'\}_{i=1}^{t}$.
return PF.

Algorithm 5, FN takes the differences in the encrypted domain, which can be visualized by an upper triangular matrix

$$\begin{pmatrix} [\![c_1]\!] \textcircled{-} [\![c_2]\!] & [\![c_1]\!] \textcircled{-} [\![c_3]\!] & \cdots & [\![c_1]\!] \textcircled{-} [\![c_t]\!] \\ & [\![c_2]\!] \textcircled{-} [\![c_3]\!] & \cdots & [\![c_2]\!] \textcircled{-} [\![c_t]\!] \\ & & \ddots & \vdots \\ & 0 & & [\![c_{t-1}]\!] \textcircled{-} [\![c_t]\!] \end{pmatrix}.$$

FN allocates a label set $\{'TC_i'\}_{i=1}^{t-1}$ to each row of the upper triangular matrix and attach $TC_t = [\![c_t]\!] \textcircled{-} [\![c_{t-1}]\!]$ with the label of $\{'TC_t'\}$. Then, CS decrypts $\{TC_i\}_{i=1}^{t}$ into $\{TP_i\}_{i=1}^{t}$, which

can be visualized by a matrix

$$\begin{pmatrix} m_1-m_2 & m_1-m_3 & \cdots & m_1-m_t \\ & m_2-m_3 & \cdots & m_2-m_t \\ & & \ddots & \vdots \\ & 0 & & m_{t-1}-m_t \end{pmatrix}$$

and checks the numbers of zero elements in each subset. Finally, FN gets the intact feature statistics over an image patch.

Feature verification: Upon receiving FNs' feedback, DP first merges some *PF*s with the same *id* to obtain *SF*, thus the total secure feature set is noted as $\{SF_i, id_i\}_{i=1}^{T}$. Then, a random sample SF_ξ ($\xi \in [1, T]$) is selected and decrypted to get the secure feature histogram R_ξ^{s}. At the same time, the original image feature histogram R_ξ^{o} is matched by id_ξ. Eventually, DP measures the similarity between R_ξ^{o} and R_ξ^{s} to check the correctness of outsourced results. The verification is successful if the score f is no more than the empirical threshold τ ($\tau = 0.4$); otherwise, the outsourced results are invalid and rejected.

$$(f \cup \perp) \leftarrow \mathrm{Verify}(\tau, \sqrt{\sum (R_\xi^{\mathrm{o}} - R_\xi^{\mathrm{s}})^2}). \tag{11}$$

Remark: To get the empirical threshold τ, it is necessary to calculate errors between outsourced and original RI-LBP features over all samples involved in experiments. As the process of RI-LBP feature extraction involves floating-point calculation and the proposed symmetric encryption scheme only deals with integers, thus there exists rounding-off errors between outsourced and plaintext features. The maximum calculation error is set as the threshold τ.

IV. Analysis of the Proposed Protocol

In this section, the proof of the proposed symmetric encryption scheme and security analysis of VRLBP are provided. Moreover, computational complexity and advantages compared to the related works are demonstrated.

A. Security Proof of Symmetric Encryption Scheme

In this section, the proposed symmetric cryptographic scheme is proved to be IND-CPA secure based on the (λ, γ, ρ)-approximate $-$ GCD problem, where λ denotes the security parameter, γ, ρ are separately set to restrict the range of q and r, which are public parameters in (5).

Lemma IV.1: The proposed symmetric probabilistic encryption scheme is (u, ϵ)-IND-CPA secure if $(\lambda, \gamma, \rho, u', \epsilon')$-$Approximate - GCDProblem$ holds, where $u' = u$ and $\epsilon' = \frac{\epsilon}{256}$.

Proof: $q_p(z)$ and $r_p(z)$ are referred to as the quotient and remainder of z to p, respectively, therefore, $z = q_p(z) \cdot p + r_p(z)$, which is an integer belonging to the set $[0, 2^\gamma)$. Provable security can be evaluated by a game between an adversary $\mathscr{A}$ and a challenger $\mathscr{C}$. ■

Suppose that $\mathscr{A}$ can output correct plaintext bit with the advantage of at least ϵ and has access to adequate numbers of samples from $\mathscr{D}_{\gamma,\rho}(p)$, the target is to get the encryption key p with a probability of ϵ' and its procedures are presented as follows.

1) $\mathscr{C}$ initiates a random sample set $\{\varrho_1, \varrho_2, \ldots, \varrho_\phi\} \in \mathscr{D}_{\gamma,\rho}(p)$ and sends the security parameter λ to $\mathscr{A}$.
2) $\mathscr{A}$ submits the plaintext message m for encryption queries. After obtaining it, $\mathscr{C}$ randomly selects a subset $S \subseteq \{1, 2, \ldots, \phi\}$ and figures out $c = m + \sum_{l \in S} \varrho_l$ as the ciphertext of m. Finally, c is sent to $\mathscr{A}$.
3) $\mathscr{A}$ submits two chosen plaintexts m_0, m_1 with the same length and $\mathscr{C}$ randomly chooses a plaintext m_α where $\alpha \in \{0, 1\}$, two integers $r \in (-2^\rho, 2^\rho), z \in [0, 2^\gamma)$, a subset $S' \subseteq \{1, 2, \ldots, \phi\}$ and calculates
$$c' = z + m_\alpha + 256r + 256 \sum_{l \in S'} \varrho_l. \tag{12}$$
4) $\mathscr{A}$ completes its guess $\alpha' \in \{0, 1\}$.

The above procedures of the game are conducted interactively by $\mathscr{A}$ and $\mathscr{C}$ many times. Based on the assumption that $\mathscr{A}$ has the advantage of ϵ to win the game, thus c' will be a correct ciphertext of m_α or $m_{1-\alpha}$. Simultaneously, the plaintext of c' equals to $m_{\alpha'} = [\![r_p(z)]\!]_{256} + m_\alpha$, considering

$$[\![[\![c']\!]_p]\!]_{256} = [\![[\![z]\!]_p]\!]_{256} + m_\alpha = [\![r_p(z)]\!]_{256} + m_\alpha. \tag{13}$$

Take an operation of modular 2 on c'

$$[\![m_{\alpha'}]\!]_2 = [\![[\![r_p(z)]\!]_{256}]\!]_2 \oplus [\![m_\alpha]\!]_2 = [\![r_p(z)]\!]_2 \oplus [\![m_\alpha]\!]_2. \tag{14}$$

Besides

$$[\![r_p(z)]\!]_2 = [\![m_{\alpha'}]\!]_2 \oplus [\![m_\alpha]\!]_2. \tag{15}$$

As the prime p is large enough, therefore

$$[\![z]\!]_2 = [\![q_p(z)]\!]_2 \oplus [\![r_p(z)]\!]_2. \tag{16}$$

From (15) and (16)

$$[\![z]\!]_2 = [\![q_p(z)]\!]_2 \oplus [\![m_{\alpha'}]\!]_2 \oplus [\![m_\alpha]\!]_2. \tag{17}$$

Eventually, $\mathscr{C}$ gains

$$[\![q_p(z)]\!]_2 = [\![z]\!]_2 \oplus [\![m_{\alpha'}]\!]_2 \oplus [\![m_\alpha]\!]_2. \tag{18}$$

As shown in [16] and [22], $\mathscr{C}$ can recover p with a probability of $\frac{1}{2}$ if $\mathscr{C}$ gets access to multiple $[\![q_p(z)]\!]_2$ corresponding to different z. Consequently, the algorithm with above procedures could recover p under the conditions of the probability of $\mathscr{A}$ distinguishing ciphertext is larger than $\frac{1}{2}$.

Now, we give a probability analysis to illustrate the success probability for $\mathscr{C}$ to recover p. If $[\![r_p(z)]\!]_{256} + m_\alpha = m_\alpha$ or $m_{1-\alpha}$ holds, accordingly, $[\![r_p(z)]\!]_{256} = 0$ or $m_{1-\alpha} - m_\alpha$, and the probability is $\frac{1}{128}$. In the aforementioned analysis, $\mathscr{C}$ can recover p with a probability of $\frac{1}{2}$ if $\mathscr{C}$ gets access to multiple $[\![q_p(z)]\!]_2$ corresponding to the different z. In total, $\mathscr{C}$ can recover p with the probability of $\epsilon' = \frac{\epsilon}{256}$.

Remark: The secret key length is set to at least 128 bits to obtain enough key space for resisting brute force and exhaustive attacks. In experimental settings, λ is set as 150 in the proposed encryption scheme for security consideration.

B. Security Analysis of VRLBP

In this section, the security of VRLBP is analyzed from two aspects of data privacy and feature privacy as follows.

1) *Data privacy:* The raw image patches are encrypted before exposing to FNs, the security of the symmetric encryption algorithm has been analyzed before, and the outsourced results achieve IND-CPA security. While the owners of the secret key are limited to DP and CS, the confidentiality of complete original data is well protected against corrupted parties. Even if $\kappa - 1$ FNs are malicious and colluding with CS, it is still secure as the partial image patches that the honest FN holds are well protecting. In addition, randomly shuffling in image patches transferring from DP to FNs will make it hard for data restoration.
2) *Feature privacy:* For the whole process of feature extraction in an untrusted fog-cloud environment, VRLBP guarantees the security of extracted features. During the process of *secure difference and binarization*, FNs obtain the results of *secure difference* over a neighborhood, as the intermediate results $\{[\![\delta_i]\!]\}_{i=0}^{7}$ are shuffled, which guarantees Algorithm 2 with a security strength of 8! per local neighborhood. Upon gaining the results of *secure binarization* that CS encrypts, FNs will not decipher it without the secret key. In the phase of *secure shift*, eight results of seven times bit-wise of circular right shifting clockwise are achieved through addition, scalar multiplication operation in ciphertext based on the homomorphic properties, which could prevent it from being exposed to CS. In a period of the whole secure features generation of a patch, FNs finally own the features statistics results in ciphertext without the secret key, and CS obtains the total difference results without corresponding labels; therefore, the plaintext of complete RI-LBP features cannot be divulged in an untrusted fog-cloud environment.

C. Probability Analysis for Verification

In this section, the security of the verification phase is analyzed in the following paragraphs. After interactive computation between FNs and CS, FNs feedback the outsourced feature sets $\{SF_i, id_i\}_{i=1}^{T}$ to DP, which contains T ciphertext feature vectors with corresponding identity codes. Under the malicious setting, we declare that DP is intended to choose a secure feature histogram randomly from $\{SF_i\}_{i=1}^{T}$ to verify. It is secure for deceptive attacking while $T - 1$ ciphertext feature sets are fabricated and one real feature set is left in probability.

The success probability of verification is

$$Pr = \frac{T-1}{T} = 1 - \frac{1}{T}. \tag{19}$$

It has a negligible advantage with the number of images increasing, therefore the phase of verification is effective with an overwhelming probability.

As the prior works [13], [14], [15] have not deployed verification modules in their LBP-based privacy-preserving protocols, thus we analyze VRLBP with verifiable protocols [17], [18], [19], [20] applied in different scenarios. In detail, VPMLP [17] and GuardLR [18] verify the correctness of machine learning predictions based on discrete logarithm problem and message authentication code, respectively. FEncKV [19] realize verification by a self-designed trapdoor permutation function, and Ge et al. [20] designed a verifiable keyword search scheme based on bilinear pairing. Note that the schemes of [17], [19], and [20] could guarantee results with the probability of 1 and [18] gives a near 100% success rate, but the computational complexity of [17], [18], [19], and [20] was linearly related to the total number of data/records. Only a random sample is involved in the verification of VRLBP such that the computational complexity is independent of the number of samples. Thus, the verification mechanism of VRLBP has the competitive advantages compared with [17], [18], [19], and [20]. The theoretical analysis on different verification approaches is listed in Table II and VRLBP achieves relative advantages in computational complexity and success probability.

TABLE II
COMPUTATIONAL COMPLEXITY AND SUCCESS PROBABILITY OF VERIFICATION

Work	VPMLP [17]	GuardLR [18]	FEncKV [19]	Ge et al. [20]	VRLBP
Complexity	$O(T)$	$O(T)$	$O(T)$	$O(T)+O(1)$	$O(1)$
Probability	1	≈ 1	1	1	≈ 1

TABLE III
COMPUTATIONAL COMPLEXITY OF DIRECT AND OUTSOURCING FEATURE EXTRACTION

Operation	Add.	Sub.	Mult.	Div.	Mod.	Exp.
RI-LBP	6.0×10^8	3.8×10^8	1.0×10^9	1.3×10^8	0	5×10^8
DP	1.6×10^7	3.8×10^4	1.6×10^7	1.3×10^4	1.6×10^7	5×10^4
FNs	1.6×10^9	4.2×10^9	2.0×10^9	1.3×10^8	5×10^8	5.0×10^9
CS	1.3×10^8	0	1.0×10^8	0	6.3×10^9	0

D. Comparisons of Computational Complexity With Direct Feature Extraction

To exhibit the efficiency of computational complexity at DP, Table III compares direct feature extraction and outsourcing computation complexity on 10 000 JPEG images with the size of 28×28, where "Add., Sub., Mult., Div., Mod., Exp." means the arithmetic operations of addition, subtraction, multiplication, division, modulo, and exponential, respectively.

As described in Section II, the original RI-LBP algorithm composes of local neighborhood initialization, toward gray-scale invariance, and toward rotation invariance. First, we analyze the computational complexity of a single pixel required in RI-LBP. In the stage of local neighborhood initialization, it takes 20 addition, 40 subtraction, 64 multiplication, and 16 division operations. In the next stage, it takes 8 subtraction operations to achieve gray-scale invariance over a local neighborhood. For the advanced improvement of rotation variance, it costs 56 addition, 64 multiplication, and 64 exponential operations per pixel. Therefore, it requires 76 addition, 48 subtraction, 128 multiplication, 16 division, and 64 exponential operations for a single pixel in the original RI-LBP in total. Then, the total costs for 10 000 images per 784 pixels have been reckoned, and it needs around 6.0×10^8 addition, 3.8×10^8 subtraction, 1.0×10^9 multiplication, 1.3×10^8 division, and 5.0×10^8 exponential operations.

In VRLBP, as two addition and two multiplication operations are required in one image encryption, DP first encrypts 10 000 images by taking 2×10^5 addition, and 2×10^5 multiplication

TABLE IV
COMPARISON OF LBP-BASED PRIVACY-PRESERVING WORKS

Work	Data privacy	Feature privacy	Verifiability	CPA-security
Xia et al. [13]	√	×	×	×
Xia et al. [14]	√	×	×	×
PPRILBP [15]	√	×	×	×
VRLBP	√	√	√	√

operations. Then, FNs and CS jointly extract ciphertext features by performing around 1.7×10^9 addition, 4.2×10^9 subtraction, 2.1×10^9 multiplication, 1.3×10^8 division, 6.8×10^9 modulo, and 5×10^9 exponential operations in total following the Algorithm 2–5. Lastly, two modulo operations are needed in one image decryption, thus DP decrypts the ciphertext features by taking around 1.6×10^7 modulo operations in feature verification.

In VRLBP, DP can reduce the computational complexity apparently in five terms except for the modulo operations in comparison with original RI-LBP algorithm. It should be noted that the modulo operations that DP executes for verification are constant with the variable number of images. Thus, it ensures efficient implementation of DP.

E. Comparisons of the Prior Works

As VRLBP is a crypto-based algorithm to protect confidential information for RI-LBP descriptor, the comparison of the existing LBP-based secure outsourcing protocols is listed in Table IV to show meaningful advantages in different security properties.

Although all of the related works [13], [14], [15] can ensure data confidentiality for resisting ciphertext-only attack, VRLBP based on an IND-CPA encryption scheme can achieve higher security towards local data than the previous ones. More importantly, the confidentiality of extracted features is guaranteed in a fog-cloud environment, at the same time, the correctness of outsourced features can be checked. In total, VRLBP can resist chosen plaintext attack while guarding the privacy of user data with extracted features and the integrity of outsourced results to defend against potential deceptive attacks. In summary, VRLBP has the significant advantages in a higher secure guarantee than prior works.

V. EXPERIMENTAL EVALUATIONS

In this section, VRLBP is evaluated in terms of the computational efficiency of clients, the utility of outsourced RI-LBP features, and the efficiency of verification. The experiments are implemented on two machines, a desktop computer with Intel(R) Core(TM) i7-10700 CPU at 2.90 GHz and 16-GB memory (CS+NPs), and a MacBook Pro laptop with Intel Core i5 CPU@1.4 GHz and 8-GB memory (DP). The number of FNs κ is set as 4. The programming language version is Python3.6, using the NumPy library. In addition, the facial images are normalized to 28×28 and each FN obtains image patches with 14×14.

As far as we know, there are three LBP-based privacy-preserving works [13], [14], [15], serving as backbones for utility evaluation. Based on the OPE encryption technique, Xia et al. [13], [14] and Wang et al. [15] separately design privacy-preserving protocols for extracting LBP and RI-LBP features. All of prior arts only enable data privacy with COA secure. The encryption approaches of prior arts mainly focus on shuffling image blocks and neighbors inside image blocks. Its utility degradation cannot be ignored because of the randomly shuffling operation for neighbors inside image blocks, which irreversibly disturb the original feature values of local neighborhoods. More importantly, the plaintext of extracted features can be obtained by an untrusted server, it is vulnerable to feature privacy and potential deceptive attacks. Thus, we design a privacy-enhanced RI-LBP protocol to solve the abovementioned issues effectively.

A. Communication Overhead

Now the communication costs between DP and FNs are calculated as follows. As every pixel is encrypted by (5), the bit length of a pixel in ciphertext is $\gamma > \lambda \geq 128$. We set $\lambda = 150$, and the total number of images $T = 10\,000$; thus, the total size of the encrypted image patches is

$$10\,000 \times (150 \times 28 \times 28 + 32) \approx 1.176 \times 10^9 \text{bit} \approx 140.2\,\text{M}. \tag{20}$$

Noted that each integer of image label occupies 32 bits of space in label sets.

Assume that the transmission rate is 10 M/s, the communication time of transferring the above ciphertext image patches from DP to FNs is calculated by

$$140.2/10 \approx 14\text{s} \tag{21}$$

where "s" means "second."

FNs and CS extract RI-LBP features over ciphertext image patches and transmit the results of outsourced features to DP. Finally, $\{SF_i, id_i\}_{i=1}^{T}$ are sent back to DP. The total length is

$$(150 + 32) \times 784 \times 10000 \approx 1.43 \times 10^9 \text{bit} \approx 170.5\,\text{M}. \tag{22}$$

The communication time is

$$170.5/10 \approx 17.1\ \text{s}. \tag{23}$$

From (21) and (23), the total communication costs between DP and FNs are

$$14 + 17.1 = 31.1\ \text{s}. \tag{24}$$

B. Computation Overhead

DP encrypts the local image data in pixel-wise by utilizing the proposed encryption algorithm and decrypts the result of extracted features to obtain the complete outsourced features in plaintext. FNs and CS extract the secure features over the ciphertext images. We compare the computational overheads of DP with direct RI-LBP feature extraction, including the computational time of raw data encryption, results decryption, and communication latency. In specific, facial images are cropped from Faceforensic++ [23] which is a dataset of facial forgeries.

To highlight the efficiency at DP in VRLBP, the comparison results of computational time on different approaches are shown in Fig. 4.

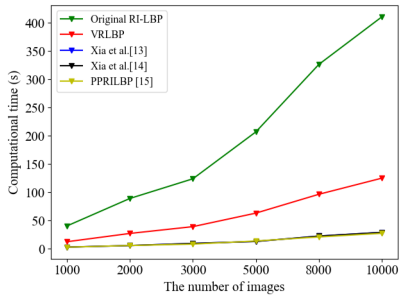

Fig. 4. Comparison of computational time.

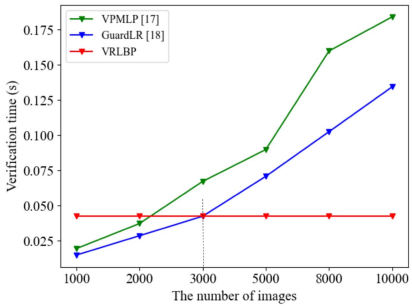

Fig. 5. Comparison of verification time.

For DP, the computational overheads of VRLBP are decreased to around 70% of direct feature extraction such that it is efficient to delegate large volumes of data to FNs for feature extraction with sufficient computational burden savings. In addition, we also compare the computational costs of DP with those of related works [13], [14], [15] based on the permutation encryption, and VRLBP requires higher computational costs by three types of arithmetic operation in a pixel-wise manner. But it achieves higher security with CPA secure and better prediction accuracy based on five public datasets as shown in Section V-C.

Next, to present the efficiency at DP in the verification stage, the comparison results of VRLBP with VPMLP [17] and GuardLR [18] in verification time are shown in Fig. 5. There are insurmountable gaps in research objectives between VRLBP and [19] and [20], which are oriented to secure keyword search; thus, the verification time of [19] and [20] are absent in Fig. 5. As analyzed in Section IV-C, the computational complexity of VPMLP is linearly related to the total number of samples, while the time costs of VRLBP (≈ 0.0427 s) are constant under the different number of samples. We conduct fair simulations to evaluate verification time under the same experimental settings. From simulation results, VRLBP gains the competitive advantage in the verification time when the total number of samples is approximately larger than 3000. Based on the above analysis, the proposed verification mechanism has the significant advantages in computational complexity and time consumption.

C. Performance Evaluation

To illustrate the utility of VRLBP by comparing it with the prior arts [13], [14], [15] and the original RI-LBP algorithm, we implement VRLBP for deepfake detection in two scenarios, namely, 1) binary classification; and 2) multiclass classification. In the meantime, we evaluate the verification time with other verifiable protocols.

1) *Datasets:* We examine VRLBP on five public deepfake datasets including Faceforensics++ [23], HifiFace [24], DeepfakeTIMIT (TIMIT) [25], Deepfake Detection Challenge (DFDC) [26], and MegaFS [27].

Faceforensics++ [23]: The dataset consists of 1000 original video sequences and four kinds of automated face manipulation methods.

HifiFace [24]: The dataset is a high-fidelity face swapping dataset by preserving the face identity and attributes of source face and target face. As we know, it totally contains 1000 swapped face videos sourced from real face videos in Faceforensics++ [23].

TIMIT [25]: The dataset consists of 32 subjects with 640 videos where faces are swapped by the generative adversarial network-based (GAN-based) approach, which originates from the autoencoder-based Deepfake algorithm.

DFDC [26]: The dataset is an "in-the-wild" face swap video dataset generated by deep learning based, and nonlearned methods, which contains more than 100 000 videos with 3426 subjects.

MegaFS [27]: The dataset is designed for the research of forgery detection and face swapping. For forgery detection, a face-swapped dataset MegaFS-FF++ is generated with 889 subjects based on Faceforensics++. Besides, there are three kinds of datasets for face swapping research, called MegaFS-IFL for short.

2) *Experimental implementation:* In experiments, we are intended to compare VRLBP with direct feature extraction and prior works to test the utility of outsourced features. Based on [28], deepfake detection is deemed a classification task where classifiers are utilized to identify the authentic images or the fake ones. Support vector machine (SVM) is generally used for two-group classification problems with the advantages of high speed and good performance on a limited number of samples. Thus, secure RI-LBP features are fed to the SVM classifier to detect deepfake images.

To eliminate biases from data, VRLBP is fitted to two scenarios of binary and multiclass classification on five different datasets, in which HifiFace, TIMIT, DFDC, and MegaFS-FF++ are intended for binary classification, Faceforensics++ and MegaFS-IFL are intended for multiclass classification. In addition, 70% of images are evenly allocated to the training set, while the rest of images are regarded as the test set.

3) *Results and analysis:* First, to validate the utility, the experimental results of VRLBP with the original RI-LBP algorithm, and prior protocols [13], [14], [15] in binary classification are presented in Fig. 6. It is obvious that the classification accuracy of VRLBP is nearly same as that of the original RI-LBP algorithm. In principle, the goal of secure outsourcing protocol is that computationally weak clients can obtain outsourced features with similar utility compared with the plaintext features. Thus, VRLBP achieves the goal and its performance is similar to that of the direct RI-LBP. In addition, VRLBP outperforms [13], [14],

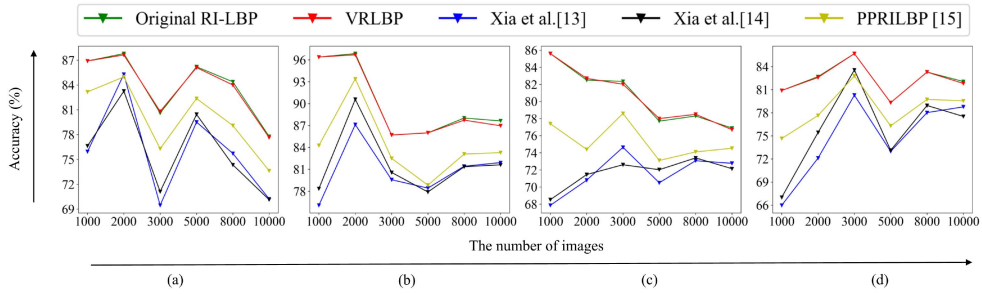

Fig. 6. Comparison of binary classification performance. (a) The performance of HiFiFace. (b) The performance of TIMIT. (c) The performance of DFDC. (d) The performance of MegaFS-FF++.

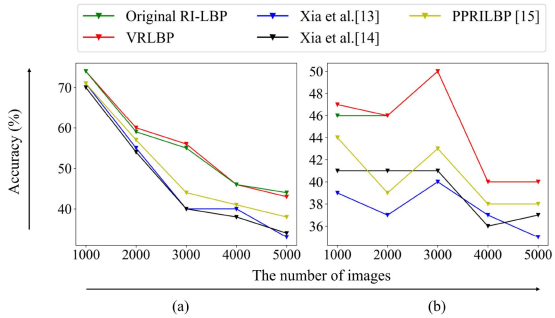

Fig. 7. Comparison of multiclass classification performance. (a) The performance of Faceforensics++. (b) The performance of MegaFF-IFL.

[15] with an apparent accuracy gap. The phenomenon of utility degradation stems from the operation of "intrablock pixels shuffling permutations," which disturbs the original LBP/RI-LBP feature values of the local neighborhoods. Therefore, VRLBP gains better performance than related works [13], [14], [15].

Then, multiclass classification on Faceforensic++ [23] and MegaFS-IFL [27] are conducted in Fig. 7 to explore diverse scenarios. Similar with binary classification, the experimental results also reflect the outperformance of VRLBP over various images. Overall, the performance of VRLBP is almost the same as the original RI-LBP algorithm and superior to the state-of-the-art works [13], [14], [15]. In the meantime, VRLBP is executed based on integers, and there may exist somewhat accuracy errors in both of VRLBP and the original RI-LBP algorithm owing to rounding operation in the local neighborhood computation. Fortunately, the rounding errors are acceptable from perspective of experimental results.

In addition, SVM classifier models the posterior probability directly from the input features, i.e., probability of class given inputs. Note that unbalanced and noisy data without proportional support vectors will degrade prediction accuracy significantly. To show data diversity, five different datasets are involved and the trends of accuracy curves are varied which depends on the quality of datasets.

Based on the above experimental results, VRLBP can not only check the integrity of outsourced RI-LBP features with an overwhelming probability and constant computational complexity, but also decrease the computational overhead at the client side greatly. Based on the complexity analysis, the verification complexity of VRLBP is superior to [17], [18], [19], and [20], and gains time efficiency compared with [17] and [18] based on simulation results. In addition, we implement secure RI-LBP features to deepfake detection in two different scenarios to show practical utility. Note that deepfake detection as an auxiliary task serves to validate the effectiveness of utility. Besides, VRLBP can be extended to more privacy-preserving research, such as secure image retrieval [30], [31], [32]. Intrinsically, the schemes of [30], [31], and [32] extract features over the plaintext images and focus on how to protect the confidential features in secure similarity measurement. To make secure image retrieval possible, a specific privacy-preserving protocol is required to measure the similarity of query and database images in ciphertexts.

VI. Conclusion

In summary, we have proposed a verifiable privacy-enhanced protocol for RI-LBP feature extraction. The privacy of local data and outsourced RI-LBP features were well preserved based on the proposed probabilistic encryption scheme. At the same time, the correctness of outsourced RI-LBP features can be checked with an overwhelming probability and constant computational complexity. Based on security analysis and experimental performance, the proposed protocol showed superiority concerning the prior works while keeping the local image data and extracted features secure. By applying outsourced features to deepfake detection over binary and multiclass classification tasks on five open-source datasets, extensive evaluations demonstrated that the proposed protocol could save the computational costs of clients greatly while gaining practical performance.

References

[1] S. Yu, G. Wang, X. Liu, and J. Niu, "Security and privacy in the age of the smart Internet of Things: An overview from a networking perspective," *IEEE Commun. Mag.*, vol. 56, no. 9, pp. 14–18, Sep. 2018.

[2] T. Ojala, M. Pietikainen, and T. Maenpaa, "Multiresolution gray-scale and rotation invariant texture classification with local binary patterns," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 24, no. 7, pp. 971–987, Jul. 2002.

[3] A. Arini, R. Broer Bahaweres, and J. Al Haq, "Quick classification of xception and resnet-50 models on deepfake video using local binary pattern," in *Proc. Int. Semin. Mach. Learn., Optim., Data Sci.*, 2022, pp. 254–259.

[4] R. K. Remya and M. Wilscy, "Image forgery detection using deep textural features from local binary pattern map," *J. Intell. Syst. Fuzzy Syst.*, vol. 38, no. 5, pp. 6391–6401, 2020.

[5] S.T. Suganthi et al., "Deep learning model for deep fake face recognition and detection," *PeerJ Comput. Sci.*, vol. 8, 2022, Art. no. e881.

[6] S. Thanga Revathi, A. Gayathri, J. Kalaivani, M. S. Christo, D. Pelusi, and M. Azees, "Cloud-assisted privacy-preserving method for healthcare using adaptive fractional brain storm integrated whale optimization algorithm," *Secur. Commun. Netw.*, vol. 2021, pp. 1–10, 2021.

[7] Y. Liu, J. Yu, J. Fan, P. Vijayakumar, and V. Chang, "Achieving privacy-preserving DSSE for intelligent IoT healthcare system," *IEEE Trans. Ind. Informat.*, vol. 18, no. 3, pp. 2010–2020, Mar. 2022.
[8] B. D. Deebak, F. H. Memon, S. A. Khowaja, K. Dev, W. Wang, and N. M. F. Qureshi, "In the digital age of 5G networks: Seamless privacy-preserving authentication for cognitive-inspired Internet of Medical Things," *IEEE Trans. Ind. Informat.*, vol. 18, no. 12, pp. 8916–8923, Dec. 2022.
[9] X. Gao, J. Yu, Y. Chang, H. Wang, and J. Fan, "Checking only when it is necessary: Enabling integrity auditing based on the keyword with sensitive information privacy for encrypted cloud data," *IEEE Trans. Dependable Secure Comput.*, vol. 19, no. 6, pp. 3774–3789, Nov./Dec. 2022.
[10] C. Hsu, C. Lu, and S. Pei, "Image feature extraction in encrypted domain with privacy-preserving SIFT," *IEEE Trans. Image Process.*, vol. 21, no. 11, pp. 4593–4607, Nov. 2012.
[11] Z. Qin, J. Yan, K. Ren, C. W. Chen, C. Wang, and X. Fu, "Privacy-preserving outsourcing of image global feature detection," in *Proc. IEEE Glob. Commun. Conf.*, 2014, pp. 710–715.
[12] S. Hu, Q. Wang, J. Wang, Z. Qin, and K. Ren, "Securing SIFT: Privacy-preserving outsourcing computation of feature extractions over encrypted image data," *IEEE Trans. Image Process.*, vol. 25, no. 7, pp. 3411–3425, Jul. 2016.
[13] Z. Xia, X. Ma, Z. Shen, X. Sun, N. N. Xiong, and B. Jeon, "Secure image LBP feature extraction in cloud-based smart campus," *IEEE Access*, vol. 6, pp. 30392–30401, 2018.
[14] Z. Xia, L. Jiang, X. Ma, W. Yang, P. Ji, and N. N. Xiong, "A privacy-preserving outsourcing scheme for image local binary pattern in secure industrial Internet of Things," *IEEE Trans. Ind. Informat.*, vol. 16, no. 1, pp. 629–638, Jan. 2020.
[15] S. Wang, M. Jiang, J. Qin, H. Yang, and Z. Gao, "A secure rotation invariant LBP feature computation in cloud environment," *Comput., Mater. Continua*, vol. 68, no. 3, pp. 2979–2993, 2021.
[16] M. Van Dijk, C. Gentry, S. Halevi, and V. Vaikuntanathan, "Fully homomorphic encryption over the integers," in *Proc. Annu. Int. Conf. Theory Appl. Cryptology Technol.*, 2010, pp. 24–43.
[17] X. Li, J. He, P. Vijayakumar, X. Zhang, and V. Chang, "A verifiable privacy-preserving machine learning prediction scheme for edge-enhanced HCPSs," *IEEE Trans. Ind. Informat.*, vol. 18, no. 8, pp. 5494–5503, Aug. 2022.
[18] Z. Ma et al., "Verifiable data mining against malicious adversaries in Industrial Internet of Things," *IEEE Trans. Ind. Informat.*, vol. 18, no. 2, pp. 953–964, Feb. 2022.
[19] K. Wang, C.-M. Chen, Z. Tie, M. Shojafar, S. Kumar, and S. Kumari, "Forward privacy preservation in IoT enabled healthcare systems," *IEEE Trans. Ind. Informat.*, vol. 18, no. 3, pp. 1991–1999, Mar. 2022.
[20] X. Ge, J. Yu, R. Hao, and H. Lv, "Verifiable keyword search supporting sensitive information hiding for the cloud-based healthcare sharing system," *IEEE Trans. Ind. Informat.*, vol. 18, no. 8, pp. 5573–5583, Aug. 2022.
[21] C. Gentry, "A fully homomorphic encryption scheme," Ph.D. dissertation, Dept. Comput. Sci., Stanford Univ., Stanford, CA, USA, 2009.
[22] Y. Ren, N. Ding, T. Wang, H. Lu, and D. Gu, "New algorithms for verifiable outsourcing of bilinear pairings," *Sci. China Inf. Sci.*, vol. 59, no. 9, pp. 246–248, 2016.
[23] A. Rössler, D. Cozzolino, L. Verdoliva, C. Riess, J. Thies, and M. Nießner, "Faceforensics: Learning to detect manipulated facial images," in *Proc. IEEE/CVF Int. Conf. Comput. Vis.*, 2019, pp. 1–11.
[24] Y. Wang et al., "HiFiFace: 3D shape and semantic prior guided high fidelity face swapping," in *Proc. Int. Joint Conf. Artif. Intell.*, 2021, pp. 1136–1142.
[25] P. Korshunov and S. Marcel, "DeepFakes: A new threat to face recognition? Assessment and detection," *CoRR*, vol. abs/1812.08685, 2018.
[26] B. Dolhansky, R. Howes, B. Pflaum, N. Baram, and C. C. Ferrer, "The deepfake detection challenge dataset," *CoRR*, vol. abs/2006.07397, Oct. 2020.
[27] Y. Zhu, Q. Li, J. Wang, C. Xu, and Z. Sun, "One shot face swapping on megapixels," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, 2021, pp. 4834–4844.
[28] Y. Mirsky and W. Lee, "The creation and detection of deepfakes: A survey," *ACM Comput. Surv.*, vol. 54, no. 1, pp. 1–41, 2021.
[29] Y. Ren, X. Zhang, G. Feng, Z. Qian, and F. Li, "How to extract image features based on co-occurrence matrix securely and efficiently in cloud computing," *IEEE Trans. Cloud Comput.*, vol. 8, no. 1, pp. 207–219, Jan.–Mar. 2020.
[30] Y. Li, J. Ma, Y. Miao, L. Liu, X. Liu, and K.-K. R. Choo, "Secure and verifiable multikey image search in cloud-assisted edge computing," *IEEE Trans. Ind. Informat.*, vol. 17, no. 8, pp. 5348–5359, Aug. 2021.
[31] L. Song, Y. Miao, J. Weng, K.-K. R. Choo, X. Liu, and R. H. Deng, "Privacy-preserving threshold-based image retrieval in cloud-assisted Internet of Things," *IEEE Internet Things J.*, vol. 9, no. 15, pp. 13598–13611, Aug. 2022.
[32] Y. Li et al., "DVREI: Dynamic verifiable retrieval over encrypted images," *IEEE Trans. Comput.*, vol. 71, no. 8, pp. 1755–1769, Aug. 2022.

Mingyun Bian received the M.S. degree in computer applied technology from Hunan University of Technology, Hunan, China, in 2020. He is currently working toward the Ph.D. degree in information and communication engineering from Shanghai University, Shanghai, China.

His research interests include applied cryptography, secure outsourcing computation and privacy-preserving machine learning.

Joseph Liu (Senior Member, IEEE) received the Ph.D. degree in information engineering from The Chinese University of Hong Kong, Hong Kong, in 2004.

He is currently an Associate Professor with the Faculty of Information Technology, Monash University, Melbourne, Australia. Prior to joining Monash University in 2015, he was a Research Scientist with the Institute for Infocomm Research (I2R), Singapore, for more than seven years. He is currently the lead of the Monash Cyber Security Group. He established the Monash Blockchain Technology Centre in 2019, and serves as the founding Director. He has received more than 5700 citations and his H-index is 43, with more than 170 publications in top venues such as CRYPTO, ACM CCS. His research interests include cyber security, blockchain, IoT security, applied cryptography, and privacy enhanced technology.

Shifeng Sun received the Ph.D. degree in computer science from Shanghai Jiao Tong University, Shanghai, China, in 2016.

He is currently an Associate Professor with the School of Cyber Science and Engineering, Shanghai Jiao Tong University (SJHU), Shanghai, China. Prior to joining SJTU, he was a Lecturer with the Department of Software Systems and Cybersecurity, Monash University, Melbourne, Australia. Before that, he was a Research Fellow with Monash University, and CSIRO, Herston, Australia, and a Visiting Scholar with the Department of Computing and Information Systems, the University of Melbourne, Melbourne, Australia, during his Ph.D. study. He has authored or coauthored more than 50 quality papers, including publications in ACM CCS, USENIX SEC, NDSS, EUROCRYPT, PKC, ESORICS, AsiaCCS, FC, IEEE TRANSACTIONS ON DEPENDABLE AND SECURE COMPUTING, IEEE TRANSACTIONS ON KNOWLEDGE AND DATA ENGINEERING, etc. His research interests include cryptography and data privacy, particularly on provably secure cryptosystems against physical attacks, data privacy-preserving technology in cloud storage, and privacy-enhancing technology in blockchain.

Xinpeng Zhang received the B.S. degree in computational mathematics from Jilin University, Changchun, China, in 1995, and the M.S. and Ph.D. degrees in communication and information system from Shanghai University, Shanghai, China, in 2001 and 2004, respectively.

Since 2004, he has been with the Faculty of the School of Communication and Information Engineering, Shanghai University, where he is currently a Professor. From January 2010 to January 2011, he was a Visiting Scholar with the State University of New York at Binghamton, Binghamton, NY, USA, and from March 2011 to May 2012, was an experienced Researcher sponsored by the Alexander von Humboldt Foundation with Konstanz University, Konstanz, Germany. He has authored or coauthored more than 200 papers in the areas of his interest. His research interests include multimedia security, image processing, and digital forensics.

Yanli Ren received the M.S. degree in applied mathematics from Shaanxi Normal University, Xi'an, China, in 2005, and the Ph.D. degree in computer science and technology from Shanghai Jiao Tong University, Shanghai, China, in 2009.

She is currently a Professor with the School of Communication and Information Engineering, Shanghai University, Shanghai, China. She has authored or coauthored more than 80 quality papers, including publications in IEEE TRANSACTIONS ON DEPENDABLE AND SECURE COMPUTING, IEEE TRANSACTIONS ON INDUSTRIAL INFORMATICS, IEEE TRANSACTIONS ON INFORMATION FORENSICS AND SECURITY, IEEE TRANSACTIONS ON CLOUD COMPUTING, IEEE TRANSACTIONS ON MULTIMEDIA, AsiaCCS, etc. Her research interests include applied cryptography, secure outsourcing computation, blockchain security, AI security, and network security.