P2SLR: A Privacy-Preserving Sign Language Recognition as-a-Cloud Service Using Deep Learning For Encrypted Gestures

This paper was downloaded from TechRxiv (https://www.techrxiv.org).

LICENSE

CC BY 4.0

SUBMISSION DATE / POSTED DATE

25-01-2022 / 01-02-2022

CITATION

Tanwar, Vishesh Kumar; sharma, gaurav; Raman, Balasubramanian; Bhargava, Rama (2022): P2SLR: A Privacy-Preserving Sign Language Recognition as-a-Cloud Service Using Deep Learning For Encrypted Gestures. TechRxiv. Preprint. https://doi.org/10.36227/techrxiv.19064063.v1

DOI

10.36227/techrxiv.19064063.v1
P2SLR: A Privacy-Preserving Sign Language Recognition as-a-Cloud Service Using Deep Learning For Encrypted Gestures

Vishesh Kumar Tanwar*, Gaurav Sharma, Senior Member, IEEE
Balasubramanian Raman, Senior Member, IEEE, Rama Bhargava

* Center for Research in Computer Vision, University of Central Florida, USA
† Department of Mathematics, Indian Institute of Technology Roorkee, India

Abstract—Cloud-based services have revolutionized data storage and processing tasks. However, these services raise security concerns as service providers may misuse the user’s stored data. Privacy loss is particularly problematic for hearing and speech impaired individuals that may need to use cloud infrastructure for sign language recognition (SLR). Addressing these challenges, this paper presents a privacy-preserving sign language recognition (P2SLR) as a cloud-service that operates over cloud infrastructure without revealing the individual’s visual information to the cloud service provider (CSP). The proposed P2SLR system is realized through two innovations: (a) block-based probabilistic image encryption scheme that combines fractional-order chaotic system (FOCS) and singular value decomposition (SVD) to obfuscate the visual information in video frames, and (b) a cloud-residing deep convolutional neural network (DCNN) based recognition architecture with a modified classifier to recognize gestures from encrypted video. The proposed scheme is validated for American, Argentinian, and German sign languages. The proposed scheme achieved recognition accuracy in the range 90.76 – 98.09%, comparable to existing state-of-the-art SLR techniques in the plain domain (PD). The proposed image encryption scheme is secure under standard cryptographic image attacks, protecting the individual’s identity. P2SLR is the first move towards developing a secure SLR system over the cloud.

Index Terms—Cloud-services, Sign language recognition, Singular value decomposition, Secure computations, Encryption

I. INTRODUCTION

Individuals with hearing and speech impairments face communication challenges in their daily lives. These people use non-verbal visual gestures, known as sign language (SL), to communicate and express their thoughts. These gestures are different for different regions. For instance, Argentine SL is used in Argentina, German SL in Germany and Belgium and, American SL in the United States of America. For decades, numerous feature-based SLR methods have been proposed [1][2], and recently, the D-CNNs based SLR frameworks have achieved remarkable recognition performance [3][4]. However, these data-driven deep learning (DL) techniques require demanding computational resources such as GPUs and storage space [5]. The setting-up and maintaining of these platforms are expensive and require significant time and effort. In contrast, cloud computing has revolutionized the computing structure allowing an individual/organization to develop virtual infrastructure-resources for storage and processing datasets. These services unshackle users from the burdensome tasks of collecting, assembling, and maintaining the expensive computing resources, instead allowing them to get these services in a pay-as-you-go business model. Microsoft Azure, Amazon Web Services, Google Cloud and IBM Cloud are a few instances of the CSPs.

Cloud computing’s key advantages are mobility via smart devices, quality control, automatic software updates, and data loss prevention in the event of hardware damage. Despite the benefits, the cloud-based services inherit some substantial drawbacks. Third parties develop and maintain these cloud platforms, and users cannot be assured of confidentiality and integrity of data transmitted to the cloud servers. Therefore, many users do not utilize cloud-services. In this paper, we address an SL individual’s security and privacy concerns, which may occur with the leakage of visual information of the individual while using cloud-services for an automatic SLR system. Protecting the SL users’ visual information from an adversary (present at CSP), such as face and background environmental details, before transmitting SL data to CSP, is a primary requisite as the leakage of SL user’s information may increase the risk of violence with them[6]. For instance, an adversary may misuse this visual information for a security breach to curate illegal activities against SL individuals, such as kidnapping and human trafficking. These privacy issues restrict the organizations to outsource SL individual’s visual data with CSPs to train a D-CNN SLR model. An ideal solution to address the data “privacy-utility” trade-off is to encrypt the individual’s identity-related information present in the video database before transmitting to the CSP and then train the D-CNN model for the encrypted data. The fully homomorphic encryption (FHE) schemes can be used to train the SLR model for FHE-encrypted data to achieve the state-

https://www.csoonline.com/article/2130877/the-biggest-data-breaches-of-the-21st-century.html
of-the-art recognition accuracy as that of unencrypted data [6][7]. However, the FHE schemes exponentially increase the data size, sometimes 1MB image is encrypted to 1GB ciphertext, making them unsuitable for real-time applications. In contrast with FHE, most of the computationally efficient image encryption schemes shuffle the pixel-locations to significantly randomize the image’s visual attributes [8] [9]. On the other hand, the D-CNNs models are susceptible to the pixel’s interlocations to generate a robust representation of the input video and thus these pixel shuffling based schemes cannot be utilized with the D-CNNs frameworks. Therefore, a new encryption scheme is required for an SL image that can significantly obfuscate the global features (individual’s visual information) while preserving local features (regions depicting gesture information) with low computation and storage overheads. Such a scheme can be used to encrypt the user’s SL video before transmitting to CSP to train a D-CNNs based SLR model for encrypted data without compromising the SL individual’s privacy.

In this paper, we utilized the chaos-based bit-plane approach [10][11] to obfuscate the SL image information by adding probabilistic noise to each pixel intensity value without altering the location of the pixels. This approach aims to preserve the relevant information such as facial expression and hand movement in an SL image as much as possible while encrypting the regions indicating SL individual identity. This feature preserving property will benefit the D-CNN model for robust parameter learning and achieves state-of-the-art recognition accuracy for encrypted data. From a security perspective, these chaotic systems are sensitive to initial parameters, inherit orbital unpredictability, ergodicity, and the random dynamic phenomena. In particular, FOCS shows higher non-linearity, and more degree-of-freedom than the integer-order chaotic systems [12][13]. Combining the FOCS and SVD, we proposed an encryption scheme for an SL image $I$ by partitioning it into non-overlapping blocks, followed by adding random noise to each SVD components of bit-planes of the blocks using the FOCS. We observe that the approach of inserting randomness to in-depth modules such as SVD components of a bit-plane achieves high-security capability of image’s global features than directly randomizing the bit-plane. The major contributions of the paper are:

1. A scheme to train an end-to-end privacy-preserving SLR system, namely $P2SLR$ using DL techniques over the cloud server is developed. The scheme aims to train a D-CNN-based recognition model for the encrypted SL dataset while protecting the user’s visual information, the person who is performing the gestures in the input video-data, from adversaries.

2. A probabilistic block-based bit-plane image encryption scheme without altering the pixels’ location is proposed to preserve the visual information. Each bit-plane of a block is obfuscated by encrypting its respective SVD components via adding a pseudo-random noise.

3. The performance of $P2SLR$ is reported over American, Argentinian, and German SL datasets for varying block-sizes. The recognition results over the encrypted data with varying block-sizes lie in the range of 90.76-98.09%, comparable with existing SLR schemes in PD.

4. The qualitative and quantitative security analysis indicates that the proposed scheme significantly protects the user’s visual information. Further, the scheme is comparable with the existing image encryption methods under various cryptographic attacks.

Organization: Section [II] provides an overview of the related privacy preserving schemes. The proposed encryption scheme and the DL-framework are presented in Sections [III]. The experimental recognition results are reported in Section [IV]. The security and performance analysis are analyzed in Section [V] and Section [VI] respectively. Finally, Section [VII] concludes the paper with future applications.

II. RELATED WORK

Some instances of cloud-based privacy-preserving services are secure photo-sharing over the social network [14], secure outsourced biometric identification [15], privacy-preserving augmented reality-based virtual cloth try-on system [16], secure and trusted e-healthcare services for social media health users [17], [18] and secure location-based services [19], [20].

This section briefly reviews the previous privacy-preserving recognition methods for image and video datasets [21][25].

Bost et al. [6] proposed an FHE-based scheme for naive Bayes, decision trees, and hyperplane decision classifiers. The authors combined FHE without bootstrapping, Quadratic Residuosity, and Paillier cryptosystems [26] for data encryption. Rahulamathavan et al. [21] proposed a privacy-preserving system for recognizing facial expressions as a user-CSP service. For encrypting image’s information, the authors presented a randomization technique based lightweight encryption algorithm using local fisher discriminant analysis and the Paillier cryptosystem [26]. The reported classification accuracies are 94.37% and 95.24% over JAFFE and MUG facial expression databases, respectively. Wang and Chang [24] proposed a two-party privacy-preserving image classification scheme by perturbing the image information using local differential privacy (LDP) [27]. They analyzed perturbation’s effect satisfying $\epsilon$-LDP on data utility regarding distance and count-based machine learning algorithms. Chen et al. [25] presented a secure multi-classification scheme to address the privacy-leakage in robot system using DL. The authors adopted two pairs of activation and cost functions using HE, namely softmax plus log-likelihood function and sigmoid plus cross-entropy function, along with secure calculation protocols. The scheme reduced the computation and communication overheads. Xie et al. [28] presented a HE-based theoretical protocol for implementing an image classification algorithm in the encrypted domain (ED) but not validated over any dataset. Gilad et al. [29] proposed an FHE scheme for encrypting gray-scale images to train a basic neural network architecture and validated for the MNIST dataset only. For more privacy-preserving data processing schemes, the reader is advised to refer to Li et al. [30], and Ding et al. [31].

Dai et al. [32] proposed a secure identification method using extremely low resolution (LR) images to perform human action recognition. The LR cameras are utilized to capture
real-time video frames and process them pixel-by-pixel to recognize the human activity. They evaluated the scheme over a 3D-animated room scenario containing five cameras and avatars describing four actions for 12 users. The reported results are close to the state-of-the-art methods in PD. Ryoo et al. [33] presented an inverse super-resolution scheme addressing unstable decision boundaries in low-resolution images, which may affect the classification robustness. The authors augmented the training samples by transforming high-resolution (HR) images to multiple LR images followed by sub-pixel transformation features of these LR images using the Siamese model. Ryoo et al. [34] proposed a privacy-preserving human action recognition scheme by transforming the high-resolution image into an extremely low-resolution image to protect the performer’s identity in input data. Further, Ren et al. [22] addressed the privacy concerns of an individual’s identity by anonymizing face only rather than the complete image without affecting the action information. In contrast with datasets utilized in these schemes, the only relevant information in SL datasets is the gestures expressed through hand kinematics and facial features (reveals individual’s identity). This relevant information is less in amount than the irrelevant information such as the unused body parts, the background of the gesture performer, and light variations, as shown in Fig. 6. Therefore, it is required to develop an encryption scheme for SL image that can obfuscate the irrelevant information, revealing individual’s identity and other related information while preserving the gesture’s features in the encrypted image.

III. PROPOSED FRAMEWORK

This section introduces an end-to-end framework to develop P2SLR. We first define the role of different entities considered in the threat model. Then, the proposed block-based image encryption scheme is presented in which we obfuscate the image’s visual information by encrypting each block’s bit-plane using the FOCS and SVD. Finally, the D-CNN-based SLR architecture, namely ResNet with the modified classifier is described. The overview of the proposed framework P2SLR is presented in Fig. 1.

A. Threat model

In P2SLR, two entities are involved: a user $U$ who wishes to outsource the SL dataset $D$, containing the video-clips of SL individuals performing the gestures, for developing an SLR model but without compromising with the visual information of the SL individuals available in the video-clips and, an “honest-but-curious” CSP $C$ who provides a D-CNN based SLR architecture $\mathcal{M}$ along with the computational and storage resources in a pay-as-you-go business model. A general user-CSP protocol to develop D-CNNs based cloud model is presented in Fig. 2. The visual information of the gesture performer in each video-clip of $D$ is obfuscated using the image encryption scheme proposed in Section III-B which partially preserves the gesture-related features in an encrypted video-clip. The encrypted dataset is denoted by $D^{Enc}$ which is further transmitted by $U$ to the cloud-server for training $\mathcal{M}$ (defined in Section III-C) over this encrypted data. $\mathcal{M}$ utilizes the inter-correlated pixels intensities to generate robust representation of input video to achieve high recognition accuracy. It is important to note that the proposed image encryption scheme preserves the inter-correlated pixels representing gesture regions only and de-correlates the other non-gesture regions. Therefore, $\mathcal{M}$ will learn only the preserved gesture features (local features) in the encrypted video-clip during training-phase which benefits $\mathcal{M}$ to obtain state-of-the-art gesture recognition accuracy. The non-gesture information (global features) will be drastically different in each encrypted video-clip of same gesture, thus this information is considered
as irrelevant by $\mathcal{M}$. We assume that $\mathcal{U}$ will not have any control on the computations performed by $\mathcal{C}$ over $D^{Enc}$. Since, $\mathcal{C}$ train $\mathcal{M}$ over $D^{Enc}$, resulting the model’s weights and hyper-parameters in encrypted form and thus, the trained model say $\mathcal{M}^{Enc}$, is also in the encrypted form. If a practical-adversary attempts to extract the visual information of the gesture performer from the encrypted video-clip in $D^{Enc}$, the obtained information is meaningless and completely random-like structures. Finally, $\mathcal{C}$ obtained information is meaningless and completely random.

**B. Proposed image encryption scheme**

This section proposes an image encryption scheme required to obfuscate the visual information in a gray-scale SL image $I$, for instance Fig. 3 (a). The encryption objective is to randomize the visual and another relevant information present in $I$ (global features), revealing the performer’s identity and related confidential information. We aimed to preserve the gesture attributes (local features) in an encrypted SL image as much as possible to benefit the D-CNN SLR model $\mathcal{M}$ for robust recognition accuracy. As discussed in Section [1] the chaos-based image encryption schemes [9][11], [33] significantly obfuscate both local and global features of $I$ simultaneously through permuting the location of the pixels to break the pixel inter-correlation. Thus, these approaches cannot be utilized to fulfill our requirements of preserving the regions representing the gesture in $I$. To achieve our objectives, we propose a block-based bit-plane encryption scheme for an SL image to encrypt the global features (performer visual information) in $I$ and partially preserves the local features describing gesture attributes. We accumulate the randomness and confusing-noise in each pixel intensity value without altering location of the pixels and thus preserving gesture features. The encryption scheme is explained in two phases. In first phase, we describe the technique to encrypt a bit-plane $BP$ using FOCS and SVD, defined in Section [III-B2]. In the second phase, we incorporate the proposed bit-plane encryption technique (first phase) to each non-overlapping blocks of $I$ with an additional layer of block-encryption as explained in Section [III-B3]. Note that an 8-bit image can be partitioned into eight binary-valued bit-planes of dimension equal to image as shown in Fig. 3.

1) **Generation of noise-vectors:** We generate the noise vectors using the solution-matrix of Chen’s chaotic system defined in [36]. Suppose the solution-matrix of system (??) is $S$ of dimension $num_{sol} \times 3$, where each column indicates $x$, $y$ and $z$ variable and, $num_{sol}$ is total number of mesh-points obtained as fraction of simulation-time and step-size. For initial noise-vectors $N_1$ and $N_2$, we generate two random-numbers using pseudo-random generator over the range $[1, num_{sol}]$ and then consider the corresponding solutions in $S$ namely $N_1 = (n_1^1, n_1^2, n_1^3)$ and $N_2 = (n_2^1, n_2^2, n_2^3)$. These vectors are utilized to perturbate the SVD-components of a bit-plane $BP$ as explained in Section [III-B2].

2) **Encrypting the bit-plane:** The intensity values in a bit-plane $BP$ of dimension $BP_{rows} \times BP_{cols}$ are either 0 or 1. We accumulate the probabilistic random-noise in $BP$ by obfuscating each numeric-values in $BP$’s SVD-components without changing the location of the pixels. The objective is to encrypt $BP$ in a bottom-to-top approach, i.e., generate the randomness to the components of the least significant decomposition of $BP$, for instance, SVD decomposition, so that a substantial amount of random-noise can be incorporated in the encrypted bit-plane obtained after the reconstruction of the randomized components of the decomposition. In the literature, we found that a small perturbation in the SVD-components of an image occurs a large variance within the intensity values of the SVD-reconstructed image. Moreover, the reconstructed bit-plane, after perturbing its SVD-components, significantly decorrelates the pixel’s inter-correlation. In the SL dataset, we
observed that the region/pixels representing the gesture are more-or-less are at the same location irrespective of the same gesture’s video-clip. Thus, the proposed bit-plane encryption accumulates a similar type of noise-like structure at gesture representing regions to each image of the same gesture and drastically different noise-structure for non-gesture regions. The SVD decomposition of $BP$ is defined as -

$$BP = BP^{U} \times BP^{S} \times [BP^{V}]^{T} \quad (1)$$

where $BP^{U}$ and $BP^{V}$ are the orthogonal matrices of dimensions $BP_{rows} \times BP_{rows}$ and $BP_{cols} \times BP_{cols}$ respectively, $T$ indicates matrix transpose. $BP^{S} = diag(a_{1}, a_{2}, ... a_{r})$ represents a diagonal matrix consisting eigenvalues of $BP$ with $a_{i} \geq a_{i+1}$, $i = 1, 2, ..., r - 1$ and $r = min(BP_{rows}, BP_{cols})$.

Let $N_{1} = (n_{11}, n_{21}, n_{31})$ and $N_{2} = (n_{12}, n_{22}, n_{32})$ be two noise-vectors generated using the method defined in Section III-B1. These vectors are used to incorporate noise to each SVD-component of $BP$ say $[BP^{U}, BP^{S}, BP^{V}]$. For each pixel $(i, j)$, we define a probabilistic noise-vector say $[noise_{ij}^{U}, noise_{ij}^{S}, noise_{ij}^{V}]$ corresponding to each SVD-component of $BP$. The noise is incorporated to $(i, j)^{th}$ pixel-intensity value of the respective component, resulting the encrypted components denoted by $[BP_{enc}^{U}, BP_{enc}^{S}, BP_{enc}^{V}]$. The mathematical formulations are -

$$noise_{ij}^{U} = n_{1i}^{1} + (n_{2}^{1} - n_{1i}^{1}) \times rand_{ij}^{1}$$

$$BP_{enc}^{U}(i, j) = BP^{U}(i, j) + noise_{ij}^{U} \quad (2)$$

where $rand_{ij}^{k}$ is a randomly generated floating-point number over the range $[0, 1]$. It is important to note that $rand_{ij}^{k}$ is different for each $(i, j)$ in every SVD-component and we are adding noise to each pixel $(i, j)$ without altering the position. Now, the encrypted components $[BP_{enc}^{U}, BP_{enc}^{S}, BP_{enc}^{V}]$ of SVD are reconstructed, as defined in Eq. 1 to form the encrypted plane say $BP_{temp}^{enc}$. Further, we normalized $BP_{temp}^{enc}$ over the range $[0, 1]$ followed by assigning the bit-value 0 at $(i, j)$ if the intensity value $BP_{temp}^{enc}(i, j)$ is greater than the mean value ($\mu$) of $BP_{temp}^{enc}$ and 1 otherwise. It gives the encrypted bit-plane say $BP_{enc}$ for unencrypted bit-plane $BP$. The pseudo-code for bit-plane encryption is presented in Algorithm 1 Moreover, a drastic difference between the visual information exhibiting the bit-planes of an SL image and its encrypted form can be observed in Fig. 3.

3) Encryption of a gray-scale image: After successfully proposing bit-plane encryption scheme, we define a block-based approach for encrypting a gray-scale image $I$ of dimension $I_{rows} \times I_{cols}$. The pictorial representation of the proposed encryption scheme for a gesture image $I$ of dimension 210 is depicted in Fig. 4 Initially, $I$ is divided into $T$ non-overlapping blocks say $B = \{I_{1B}, I_{2B}, ..., I_{9B}\}$ each of dimension $\rho_{1} \times \rho_{2}$ where $\rho_{1}$ and $\rho_{2}$ are multiples of $I_{rows}$ and $I_{cols}$ respectively and $T = \frac{I_{rows} \times I_{cols}}{\rho_{1} \times \rho_{2}}$. For instance, we partitioned a image of dimension $210 \times 210$ into nine blocks each of dimension $70 \times 70$. As discussed above, most of the gesture information are contained in second and fifth blocks with a small amount of information in eighth block, rest of the blocks contain the non-gesture or irrelevant information for an SLR system. Thus, we aim to preserve information of second, fifth and eighth blocks (local features) in the encrypted image while completely obfuscating the information in rest of the blocks (global features).

Next, each $i^{th}$ block $I_{B}$ in $B$ is considered as a separate image of dimension $\rho_{1} \times \rho_{2}$ and its bit-planes say $\{b_{1}, b_{2}, ..., b_{8}\}$ are extracted, as shown in Fig. 4 after step (ii). It can be observed that each bit-value in a bit-plane inherits different amount of information based upon its pixel-position. For instance, consider the gesture image (8-bit gray scale) depicted in Fig. 4 The number “1” in a 8-bit binary representation expresses $2^{0} = 1$ at first-bit of the pixel and $2^{7} = 128$ at eight-bit of the same pixel. Experimentally, the least four bit-planes $(b_{1}, b_{2}, b_{3}, b_{4})$ of image in Fig. 4 contain nearly 6-7% and the highest-four bit-planes $(b_{5}, b_{6}, b_{7}, b_{8})$ carry 93-94% of total image information. It can easily be observed through the respective bit-planes in Fig. 4. The percentage of pixel information say $P(i)$ is computed as -

$$P(i) = \frac{2^{i}}{\sum_{j=0}^{7} 2^{j}} \quad \forall \quad i = 0, 1, ..., 7 \quad (3)$$

To reduce the computational complexity without much compromising the security efficiency of the proposed image encryption scheme as a cloud-service, it is sufficient to obfuscate the least-four bit-planes information (as they contain minimal amount of information) by location-wise adding random-noise say $n_{3i}^{2}, n_{3i}^{3}, n_{3i}^{4}, n_{3i}^{5}$ where $n_{3i}^{e} \in [0, 1]$ directly to each pixel value to get encrypted bit-planes denoted by $\{b_{l}^{Enc}\}_{l=1}^{4}$. However, the four most significant bit-planes are encrypted using the scheme defined in Section III-B2 to obtain encrypted bit-plane say $\{b_{l}^{Enc}\}_{l=5}^{8}$. Then, these encrypted bit-planes are reconstructed with their respective positions to obtain the encrypted block say $I_{B}^{Enc}$. After adding noise to bit-planes,
Algorithm 1 Encryption of bit plane

1: Input: A bit plane $B$ of dimension $B_{rows} \times B_{cols}$
2: Input: Noise vectors $N_1 = (n_1^1, n_1^2, n_1^3)$ and $N_2 = (n_2^1, n_2^2, n_2^3)$ as defined in Section III-B1
3: Output: Encrypted bit plane say $B_{enc}$

4: Decomposition of $B$ into SVD form

5: $[B^U, B^S, B^V] = svd(B)$
6: for $i \leftarrow 1$ to $B_{rows}$ do
7:     for $j \leftarrow 1$ to $B_{cols}$ do
8:         $\text{noise}^U_{i,j} = n_1^1 + (n_1^2 - n_1^1) \times \text{rand}_{i,j}^1$
9:         $\text{noise}^S_{i,j} = n_1^2 + (n_2^2 - n_1^2) \times \text{rand}_{i,j}^2$
10:        $\text{noise}^V_{i,j} = n_1^3 + (n_2^3 - n_1^3) \times \text{rand}_{i,j}^3$
11:        $B^U_{enc}(i,j) = B^U(i,j) + \text{noise}^U_{i,j}$
12:        $B^S_{enc}(i,j) = B^S(i,j) + \text{noise}^S_{i,j}$
13:        $B^V_{enc}(i,j) = B^V(i,j) + \text{noise}^V_{i,j}$
14:     end for
15: end for

16: Reconstruction of encrypted bit plane from SVD form

17: $B_{temp}^{enc} = B_{enc}^U \ast B_{enc}^S \ast [B_{enc}^V]^T$
18: Compute maximum $(\max)$, minimum $(\min)$ and mean $(\mu)$ of $B_{temp}^{enc}$
19: Normalizing $B_{temp}^{enc}$ over the range $[0, 1]$
20: $B_{temp}^{enc} = \frac{(B_{temp}^{enc} - \min)}{\max - \min}$
21: for $i \leftarrow 1$ to $rows$ do
22:     for $j \leftarrow 1$ to $cols$ do
23:         $B_{enc}(i,j) = \begin{cases} 
         1, & B_{temp}^{enc}(i,j) \leq \mu \\
         0, & B_{temp}^{enc}(i,j) > \mu 
         \end{cases}$
24:     end for
25: end for

we further enhance the randomness to global features by encrypting each $t^{th}$ encrypted block $[I_{B_{temp}}^1]$ to get final encrypted block denoted by $[I_{B_{temp}}^2]$. In this layer, we add a random value say $\text{rand}_{i,j}$ to each $(i,j)^{th}$ intensity-value $[I_{B_{temp}}^1(i,j)]$ of encrypted block $[I_{B_{temp}}^1]$ as -

$$[I_{B_{temp}}^2](i,j) \leftarrow [I_{B_{temp}}^1](i,j) + \text{rand}_{i,j}$$

(4)

where $1 \leq i \leq \rho_1, 1 \leq j \leq \rho_2$ and $\text{rand}_{i,j} \in [0, \phi]$ in which $\phi$ is given as -

$$\phi = \frac{\max([I_{B_{temp}}^1]) - \min([I_{B_{temp}}^1])}{2}$$

(5)

It is important to note that the range of random-value for each $t^{th}$ block $[I_{B_{temp}}^1]$ is bounded with the maximum and minimum intensity values of the same block. Therefore, the gesture features in an encrypted SL image will get obfuscated but remain at their same pixel-locations as shown in Fig. 4 (after operating (iv) and (v)). The pseudo-code for encrypting a gray-scale image is presented in Algorithm 2. We analyzed that the security-efficiency of global features of an SL image is directly proportional to the block-size $(\rho_1 \times \rho_2)$ to be considered for partitioning the image. For instance, consider an image of dimension $200 \times 200$, then its encrypted form with block-size $\rho_1 \times \rho_2 = 40 \times 40$ will be more secure than the encrypted image with block-size $\rho_1 \times \rho_2 = 4 \times 5$. An RGB-color image can be encrypted by performing image encryption (Algorithm 2) to each red-green-blue color channel simultaneously.

C. D-CNN based SLR framework

After successfully defining the encryption scheme to obfuscate the local and global features in an SL gesture video-clip in the previous section, we will define the D-CNNs based recognition architecture denoted by $M$ which is trained for encrypted data $D^{Enc}$. For experiments, we utilized the variants of Residual neural network (ResNet) architecture [37]. In literature, we found that the ResNet architecture significantly boosted-up the recognition accuracy with low computational and storage overheads than other D-CNNs architectures in numerous applications such as face recognition [38], object detection [39], and SLR system [40]. The ResNet architecture has two parts - feature extractor and ImageNet classifier. The feature extractor takes encrypted gesture video-clip as input and produces a robust representation of the clip. It is important to note that the representation inherits only the local features preserved during the encryption scheme. The classifier is used to classify the input video-clip into pre-defined gestures (class labels). In this work, we do not make any changes in the feature extractor but modified the classifier by incorporating two adaptive pooling layers, namely maximum and average over the last batch-normalization layer of feature-extractor in Fig. 5. The output features of these pooling layers are concatenated and flattened to a one-dimensional global feature descriptor of the input image. Further, we incorporate layers as BatchNorm-Dropout-FullyConnected-Activation(ReLU)-BatchNorm-Dropout-FullyConnected-LogSoftmax (output layer). Moreover, the class-labels associated with the gesture video-clips are uniquely transformed into numeric numbers (one for each class-label) before transmitting to the cloud server. In this manner, the CSP receives the supervised encrypted data $D^{Enc}$ with the encoded class labels rather than the actual text-labels.

IV. RECOGNITION EXPERIMENTS

This section demonstrates the recognition performance for the variants of ResNet namely ResNet18, ResNet34, ResNet50 and VGG16 recognition models with feature extractor and classifier settings defined in Section in III-C over three SL datasets given in Section IV-A in which each video-frame is encrypted using the image encryption scheme proposed in Section III-B3.
A deep-learning architecture (ResNet) without ImageNet classifier
BatchNorm2D Layer
Adaptive Average Pooling Layer
Adaptive Max Pooling Layer
(1D)
Dropout Layer
Fully-connected Layer
Log Softmax Layer
(Output Layer)
Classifier

Fig. 5. The schematic representation of the modified classifier incorporating
the adaptive average and max pooling layers.

Algorithm 2 Encryption of gray-scale image

1. **Input:** A gray-scale image say \( I \) and Block-size \( \rho_1 \times \rho_2 \)
2. **Output:** Encrypted image say \( I_{Enc} \)
3. **Assume:** \( [I_{rows}, I_{cols}] \leftarrow \) number of rows and columns in \( I \)
4. **Assume:** \([I_{Max}, I_{Min}] \leftarrow \) maximum and minimum intensity in \( I \)
5. Partition \( I \) into blocks (sub-images) of dimension \( \rho_1 \times \rho_2 \) in row-roster form say \([I_{B1}^1, I_{B2}, ..., I_{B_T}^T]\)
6. for each block \( I_{B}^l \) do
7. extract all the bit-planes of \( I_{B}^l \) say \([b_1, b_2, ..., b_8,]\)
8. \( N_3 = [n_1^1, n_2^1, n_3^1, n_2^2] \)
9. for \( l \leftarrow 1 \) to 4 do
10. \( b_l^{Enc} = b_l + n_l \)
11. end for
12. for \( l \leftarrow 5 \) to 8 do
13. Encrypt \( b_l \) using Algorithm 1 to get \( b_l^{Enc} \)
14. end for
15. Reconstruct the encrypted block \( [I_{B}^{Enc}]_{temp} \) using encrypted bit planes \([b_{Enc}^l]_{temp}\]
16. # Add a noise to each intensity value of encrypted block \( [I_{B}^{Enc}]_{temp} \)
17. for \( i \leftarrow 1 \) to \( \rho_1 \) do
18. for \( j \leftarrow 1 \) to \( \rho_2 \) do
19. \( [I_{B}^{Enc}](i,j) \leftarrow [I_{B}^{Enc}](i,j) + rand_{i,j} \) (Eqs. 4 and 5)
20. end for
21. end for
22. end for
23. Concatenate the encrypted blocks \([I_{B}^{Enc}]_{temp}\) to obtain final encrypted image \( I_{Enc} \) of dimension \( I_{rows} \times I_{cols} \)

A. Datasets

1. **American Sign Language (ASL):** The dataset is provided by Kaggle consisting of 200 \times 200-dimensional RGB-color images for 29 signs namely 26 alphabetic letters A-Z along with three class as DELETE, NOTHING and SPACE. We consider the training set containing a

2. **Argentinean Sign Language (LSA64):** The dataset is collected by Ronchetti et al. [41] namely LSA64 consisting 3200 video-sequences of \( I_{rows} \times I_{cols} \)-dimension for 64 signs (22 double-handed and 42 single-handed) performed by 10 subjects. Some instances of these datasets are depicted in Fig. 6 (b).

3. **RWTH German finger spelling:** The dataset [42] consists of RGB-color video sequences for 35 gestures namely the 26 alphabets A-Z, SCH, 5 number digits Eins, Zwei, Drei, Vier, Fuenf and 3 German umlauts A, Ù and Ô. It contains 1400 videos performed by 20 subjects using webcam and camcorder. In this paper, we consider webcam videos of dimension 320 \times 240 as depicted in Fig. 6 (c). The visual representation of all the sign-gestures contained in each dataset are provided in the supplementary material.

Implementation details: Codes are implemented using Jupyter-notebook running on 64-bit Ubuntu 18.04 OS, over HP workstation - Intel Xeon(R) Gold 5120 CPU \( \times 2.20GHz \) \times 56 and Nvidia Quadro P5000 GPU. All recognition architectures are trained end-to-end for rescaled video-frames of dimension 200 \times 200 with batch-size of 128, 100 epochs, Adam optimizer (lr = \( 1e^{-3} \)), Cosine Annealing lr scheduler and negative log likelihood loss. Each dataset is partitioned into training-validation-testing data as 75%-10%-15% respectively.

B. Gesture recognition analysis for the encrypted datasets

Here, we analyze the recognition accuracy for the encrypted forms of the above-defined SL datasets. Each recognition architecture is evaluated for five encrypted variants of ASL, LSA64 and RWTH SL datasets obtained by varying block-sizes \( \rho_1 \times \rho_2 \) namely 5 \times 5, 10 \times 10, 20 \times 20, 25 \times 25 and, 40 \times 40 in the proposed image encryption scheme. For these encrypted variants, the achieved recognition accuracies for testing dataset lie in the range 91-98%, 94-99%, and 81-91% for ASL, LSA64, and RWTH datasets respectively as depicted in Tables II-III. Also, we observed a inverse trade-off between the recognition accuracy and the block-size, considered for image encryption. For instance, the recognition accuracies of
ResNet50 for encrypted ASL dataset is 98.09% with block-size $5 \times 5$ and 90.76% for block-size $40 \times 40$ as shown in Table I. A similar trade-off can be analyzed with other datasets and block-sizes. It can be noted that the ResNet50 outperforms the other frameworks for all three encrypted SL datasets. Further, we trained ResNet50 end-to-end with the same implementation setting for unencrypted ASL, LSA64, and RWTH datasets and achieved recognition accuracies for testing datasets as 98.89%, 99.73%, and 95.84% respectively. It can be noticed that the error-difference in accuracies of unencrypted and encrypted datasets is not more than 7-8% only. However, a small recognition loss can be acceptable by the society with the significant increment of SL users privacy and security benefits. Next, we compare the recognition accuracies of ResNet50 for each of encrypted ASL, LSA64 and RWTH datasets with block-sizes $5 \times 5$ and $40 \times 40$ with the existing permutation-based image encryption schemes.

| DL model | Block-size | 5 × 5 | 10 × 10 | 20 × 20 | 25 × 25 | 40 × 40 |
|----------|------------|-------|---------|---------|---------|---------|
| VGG16    |            | 85.03%| 81.79%  | 78.55%  | 74.17%  | 67.79%  |
| ResNet18 |            | 92.13%| 91.57%  | 86.24%  | 82.67%  | 78.55%  |
| ResNet34 |            | 96.74%| 93.19%  | 88.73%  | 85.03%  | 81.55%  |
| ResNet50 |            | 98.09%| 96.84%  | 94.11%  | 93.47%  | 90.76%  |

| DL model | Block-size | 5 × 5 | 10 × 10 | 20 × 20 | 25 × 25 | 40 × 40 |
|----------|------------|-------|---------|---------|---------|---------|
| VGG16    |            | 94.75%| 93.89%  | 91.14%  | 89.74%  | 84.07%  |
| ResNet18 |            | 95.76%| 93.41%  | 91.30%  | 88.60%  | 85.91%  |
| ResNet34 |            | 98.28%| 97.29%  | 95.13%  | 94.28%  | 92.89%  |
| ResNet50 |            | 98.88%| 97.99%  | 95.89%  | 95.08%  | 94.17%  |

| DL model | Block-size | 5 × 5 | 10 × 10 | 20 × 20 | 25 × 25 | 40 × 40 |
|----------|------------|-------|---------|---------|---------|---------|
| VGG16    |            | 74.17%| 72.29%  | 69.88%  | 64.08%  | 61.20%  |
| ResNet18 |            | 87.13%| 84.07%  | 81.97%  | 79.07%  | 77.89%  |
| ResNet34 |            | 90.55%| 88.67%  | 85.43%  | 83.99%  | 79.17%  |
| ResNet50 |            | 93.10%| 91.76%  | 87.70%  | 85.09%  | 81.46%  |

D. Comparison with state-of-the-art SLR methods

Since P2SLR is the first work to perform SLR in the ED; therefore we compare the gesture recognition accuracies of ResNet50 for each of the encrypted SL datasets using the proposed encryption scheme with the existing SLR methods in the PD utilizing classical feature and D-CNNs based approaches. As the ASL dataset is provided in Kaggle’s SLR challenge; therefore, P2SLR is compared with the results reported by participants, namely Dan, Rohit, and Jeffy. These participants utilized variants of D-CNNs frameworks in their end-to-end SLR methods to achieve the recognition accuracies in the range 85.49-99% whereas P2SLR reported 90.76-98.09%. For the LSA64 dataset, we consider the features based SLR methods proposed by Ronchetti et al. [41] and Tanwar et al. [1] with reported accuracies as 95.95% and 82.57% respectively. Ronchetti’s method constituted the hand tracking and segmentation of colored gloves, followed by the Hidden Markov Model (HMM) and Gaussian Mixture Model (GMM) based classifier. In contrast, Tanwar’s model utilized the dense trajectories for tracing hand-movement and GMM for hand segmentation with Random Forest and support vector machine as classifiers. Further, Javed et al. [4] generated three deep motion templates for each gesture-video and train three different D-CNNs architectures followed by their fusions using a Kernel-based extreme learning machine to obtain the final class-label. Javed’s model reported recognition accuracies of 97.81% and 85.86% for LSA64 and RWTH datasets, respectively. The comparable results are shown in Table IV. It is observed that the P2SLR outperforms the existing schemes on LSA64 and RWTH datasets and achieves close recognition accuracy for the ASL dataset.

| TABLE IV | ACCURACY COMPARISON WITH EXISTING ENCRYPTION SCHEMES. |
|----------|-----------------------------|
| Dataset | ASL | LSA64 | RWTH |
| Original | 98.89% | 99.73% | 95.84% |
| Zhu et al. [43] | 90.11% | 12.73% | 10.83% |
| Zhang & Xiao [44] | 11.97% | 15.10% | 12.61% |
| Ping et al. [45] | 09.33% | 13.64% | 11.04% |
| P2SLR | Block-size - 5 × 5 | 98.09% | 98.88% | 93.10% |
| P2SLR | Block-size - 40 × 40 | 90.76% | 94.17% | 81.46% |

C. Comparison with image encryption schemes

The image encryption scheme proposed in Section III-B3 is aimed to partially preserve the gesture features in an encrypted image without permuting the location of the pixels. Therefore, it is desirable to compare this scheme with the pixels location permutation image encryption schemes. For experiments, we consider the chaos-based encryption schemes proposed by Zhu et al. [43], Zhang et al. [44], and Ping et al. [45]. Each of the ASL, LSA64, and RWTH datasets are encrypted using these schemes, and ResNet50 is trained end-to-end for these encrypted datasets. The obtained recognition accuracy lies in the range of 9 − 15% only as reported in Table IV whereas the accuracy reported by the proposed scheme is close to 81 − 99%. Hence, it is worthwhile to mention that the proposed encryption scheme fulfills the desired objective of preserving gesture features while efficiently secure an individual’s visual information.

V. Security analysis

This section presents the proof-of-security of the proposed encryption scheme and various standard cryptographic attacks practiced by an adversary to extract the original image’s visual information.
information from its encrypted form to obtain the individual’s identity.

A. Proof-of-security

Consider a gray-scale gesture image $I$ of dimension $M \times N$ which is encrypted using the proposed encryption scheme with block-size $\rho_1 \times \rho_2$. In other words, a total of $T = \frac{M \times N}{\rho_1 \times \rho_2}$ blocks each of dimension $\rho_1 \times \rho_2$ are encrypted. Since, the encryption is performed to all 8 bit-planes of $T$ blocks, therefore, a total of $8 \times T$ bit-planes of dimension $\rho_1 \times \rho_2$ are encrypted. As discussed in Section 4.4 the information in least-significant four bit-planes are obfuscated by additive random-noise whereas the most-significant four bit-planes are encrypted through the proposed Algorithm 1. Thus, half of total $8 \times T$ bit-planes i.e., $4 \times T$ bit-planes are obfuscated through former method and rest $4 \times T$ bit-planes using later method. The total number of random-values required for least significant $4 \times T$ bit-planes are $4 \times T \times \rho_1 \times \rho_2$.

Now, we calculate the total number of random-values for highest-significant $4 \times T$ bit-planes. Here, each bit-plane is initialized with two noise-vectors i.e., solution of Chen’s chaotic system. So, for $4 \times T$ bit-planes, it requires $4 \times T \times 2$ solutions. Also, each bit-plane is decomposed into three SVD-components where each component is of dimension $\rho_1 \times \rho_2$ and all the entries in each component are obfuscated through a random-value (Eq. [2][2]), which requires $3 \times \rho_1 \times \rho_2$ random-values for a single bit-plane and therefore, $4 \times T \times 3 \rho_1 \times \rho_2$ random-values for $4 \times T$ bit-planes. For bit-plane encryption, the total number of required random-values denoted by $RV_{bit}$ are -

$$RV_{bit} = (4 \times T \times \rho_1 \times \rho_2) + (4 \times T \times 2 + 4 \times T \times 3 \times \rho_1 \times \rho_2)$$

$$= 8 \times T \times (2 \times \rho_1 \times \rho_2 + 1)$$

In Eq. 4, a random-value is added to each pixel-intensity value of the encrypted block (as a second-security layer) whose range is different for each block that further increases $M \times N$ more random-values to $RV_{bit}$. Thus, the total number of random-values denoted by $RV_{total}$ become -

$$RV_{total} = RV_{bit} + M \times N$$

$$= 8 \times T \times (2 \times \rho_1 \times \rho_2 + 1) + M \times N$$

It is important to note that the $RV_{bit}$ random-values vary uniformly over the range $[0, 1]$ i.e., if it is partitioned into $L$ values uniformly then, the probability for getting a random-value rand is $\frac{1}{T}$. For instance, if $L = 10^7$, then the probability for each random-value is $0.0000001 \approx 0$. Moreover, these random-values are generated through a cryptographic-secure pseudo-random generator $G$ and we assume that an adversary is unable recover the output of $G$ with any condition. Further, the last $M \times N$ random-values (Eqs. [6] - [7]) varies for $T$ different intervals, one for each block (Eq. 5). With above analysis, we claim that an adversary is unable to extract the original pixel-intensity value form that of encrypted pixel-value. For a RGB-color image, total number of random-values becomes 3-times (one for each color channel) of $RV_{total}$ (Eq. 7). For the visualization of total incorporated randomness and confusion in the encrypted images obfuscating the global features, please refer to the supplementary material.

B. Visual comparison with varying block-size

After theoretically analyzing the total number of random-values in an encrypted image, this section qualitatively compares the encrypted images with varying encryption scheme parameters such as block-size. The difference between different encrypted images for a gesture image can be visualized in Fig. 7. A trade-off between the block-size and level of randomness in an encrypted image can easily be observed. For instance, an encrypted image obtained with block-size $5 \times 5$ reveals edges and hand-shape (local features) features, whereas an encrypted image with block-size $40 \times 40$ does not leak any original image information (local and global features) and depicts only noisy blocks. It is worth mentioning that an adversary cannot extract sensitive information from the encrypted images for large block-sizes and vice-versa.

C. Pixel-known attack

In this attack, an adversary is known with the frequency of each pixel intensity value in an encrypted gesture image

---

**TABLE V**

| Block–size | ASL | LSA64 | RWTH | Privacy |
|------------|-----|------|------|---------|
| 5 x 5      | -   | 95.95% | - | ✗ |
| 10 x 10    | -   | 82.57% | 73.50% | ✗ |
| 20 x 20    | -   | 97.81% | 85.86% | ✗ |
| 25 x 25    | -   | - | 64.30% | ✗ |
| 40 x 40    | -   | - | 98.09% | ✗ |
| 5 x 5      | 85.49% | - | - | ✗ |
| 10 x 10    | 98.99% | - | - | ✗ |
| 20 x 20    | 99.00% | - | - | ✗ |
| 40 x 40    | 90.76% | 94.17% | 81.46% | ✓ |

*indicates the author who submitted their results to Kaggle ASL challenge https://www.kaggle.com/grassknoted/asl-alphabet/notebooks.*

---

Fig. 7. Comparison of encrypted images with varying block-size. A utility-privacy trade-off of data with the increasing block-size is presented.
The task is to extract the frequency of each pixel intensities of the original gesture image \( I \). It is performed to measure the total amount of diffusion and confusion attributes in \( I_{Enc} \) compared to \( I \). Therefore, the frequencies of \( I_{Enc} \) must be unrelated to \( I \) and uniform as much as possible. These frequencies explain the contrast, brightness, and saturation effects in an image, which are analyzed through their graphical representation. For experimentation, we compare the frequencies of a \( 200 \times 200 \)-dimensional ASL gesture image \( I \) and its encrypted form \( I_{Enc} \) obtained with block-size \( 40 \times 40 \) as depicted in Fig. 8 (a). The channel-wise (red-green-blue) frequencies comparing \( I \) and \( I_{Enc} \) are shown in the first and second columns, respectively. It can easily be observed the histograms of each channel of \( I_{Enc} \) are nearly uniform and drastically different from that of histograms of \( I \).

In a similar attack, an adversary can perform frequency-equalization to the histograms of \( I_{Enc} \) to obtain the equalized histograms and compare them with the existing histograms database in the PD to retrieve the best-similar image. Therefore, we perform histogram-equalization over frequency-histograms of \( I_{Enc} \) as shown in the third column of Fig. 8 and compare with the respective channel histogram of \( I \). It can be noticed that the equalized frequencies of each color-channel are significantly different from the original frequencies, leaking zero image information.

In this section, we evaluate the mean square error (MSE) to analyze the deviation-difference in the pixel-intensity of an original SL image \( I \) and its encrypted form \( I_{Enc} \). The mathematical equation of MSE is:

\[
MSE = \frac{1}{M \times N} \sum_{m=1}^{M} \sum_{n=1}^{N} [I(m,n) - I_{Enc}(m,n)]^2
\] (8)
where \( I(m, n) \) and \( I_{Enc}(m, n) \) denote the intensity value of \( I \) and \( I_{Enc} \) at pixel location \((m, n)\) respectively. Analytically, high MSE indicates the large changes in the global features of an encrypted image than the original image and vice-versa. Experimentally, we considered 100 images from each of the ASL, LSA64, and RWTH datasets and computed the MSE between the original and their encrypted images with varying block-sizes. The average MSE values for each dataset are reported in Table VII.

Since the different encryption levels in the proposed image encryption scheme utilized pseudo-random values to generate noise; therefore, it is worthwhile to report the average MSE between two encrypted images, say \( I_{Enc_1} \) and \( I_{Enc_2} \) of the same original image \( I \), as shown in Table VII. It can be observed that the MSE values are small for small block-sizes and vice-versa, as depicted in Tables VII and VIII. Also, it supports our claim of the inverse trade-off between the block-size and image security in the proposed encryption scheme.

Since the different encryption levels in the proposed image encryption scheme utilized pseudo-random values to generate noise; therefore, it is worthwhile to report the average MSE between two encrypted images, say \( I_{Enc_1} \) and \( I_{Enc_2} \) of the same original image \( I \), as shown in Table VII. It can be observed that the MSE values are small for small block-sizes and vice-versa, as depicted in Tables VII and VIII. Also, it supports our claim of the inverse trade-off between the block-size and image security in the proposed encryption scheme.

TABLE VI
Comparison of MSE between the original and encrypted images.

| Block-size | 5 \times 5 | 10 \times 10 | 20 \times 20 | 25 \times 25 | 40 \times 40 |
|-----------|-----------|-------------|-------------|-------------|-------------|
| ASL       | 12023     | 11035       | 10436       | 9888.2      | 10054       |
| LSA64     | 20663     | 19162       | 17852       | 18011       | 16123       |
| RWTH      | 12901     | 10468       | 8368.9      | 8577.1      | 9309.6      |

TABLE VII
Comparison of MSE between two different encrypted images of same block-size.

| Block-size | 5 \times 5 | 10 \times 10 | 20 \times 20 | 25 \times 25 | 40 \times 40 |
|-----------|-----------|-------------|-------------|-------------|-------------|
| ASL       | 2060.3    | 3728.9      | 6231.4      | 7141.5      | 9781.1      |
| LSA64     | 1051.4    | 2109.3      | 3579.8      | 4081.3      | 6123.8      |
| RWTH      | 2966.1    | 4902.4      | 7811.1      | 8550.1      | 9891.2      |

F. Pixel-wise Information Entropy

Information entropy, proposed by Claude Shannon [46], analyzes the total amount of randomness and the probability distribution of intensity values in an image. The more uniformity and randomness in the distribution of intensity values, the higher the information entropy is. For possible pixel intensity values \( \{0, 1, ..., L-1\} \) in an \( N \)-bit image \( I, L = 2^N \), the entropy denoted by \( H(I) \), is defined as -

\[
H(I) = - \sum_{l=0}^{L-1} P(l) \log_2(P(l)) \tag{9}
\]

where \( P(l) \) is the probability of intensity value \( l \) in \( I \). The entropy-range for an \( N \)-bit gray-scale image is \([0, N]\), where \( H(I) \) close to \( N \) indicates the encryption scheme’s high-security efficiency with low-probability of information leakage and vice-versa. Also, it indicates the amount of confusion generated by the encryption scheme to obfuscate the image visual information.

G. Differential Analysis

The analysis is performed to examine the total change in the encrypted image \( I_{Enc_1} \) and \( I_{Enc_2} \), obtained by performing a small variation in the original SL image \( I \). In other words, we compute the difference between \( I_{Enc_1} \) and \( I_{Enc_2} \) where \( I_{Enc_1} \) and \( I_{Enc_2} \) are obtained before and after changing a pixel intensity value in \( I \), respectively. The mathematical equation to compute difference denoted by \( \mathcal{D} \) is defined as

\[
\mathcal{D}(i, j) = \begin{cases} 
0, & \text{if } I_{Enc_1}(i, j) = I_{Enc_2}(i, j) \\
1, & \text{if } I_{Enc_1}(i, j) \neq I_{Enc_2}(i, j) 
\end{cases} \tag{10}
\]

Then the difference percentage, denoted by \( \mathcal{DP} \), between \( I_{Enc_1} \) and \( I_{Enc_1} \) is evaluated as

\[
\mathcal{DP} = \frac{\sum_{i=1}^{M} \sum_{j=1}^{N} \mathcal{D}(i, j)}{M \times N} \times 100 \tag{11}
\]

We evaluate \( \mathcal{DP} \) for encrypted RGB-color SL image with varying block-sizes and reported the channel-wise \( \mathcal{DP} \) in Table X. The obtained values lie in the range of 90-98%, which signifies that the encrypted images’ intensities vary drastically with a single-pixel variation in the original image. Further, the obtained \( \mathcal{DP} \) values are found to be comparable with existing state-of-the-art encryption schemes, as presented in Table XI. Similar to entropy, the \( \mathcal{DP} \) values present a trade-off which increases with large block-sizes and vice-versa.

TABLE VIII
Entropy comparison for varying block-size.

| Block-size | 10 \times 10 | 20 \times 20 | 25 \times 25 | 40 \times 40 |
|-----------|-------------|-------------|-------------|-------------|
| Red       | 7.7021      | 7.8586      | 7.9025      | 7.9383      |
| Green     | 7.8455      | 7.9504      | 7.9296      | 7.9417      |
| Blue      | 7.7485      | 7.8915      | 7.9396      | 7.9368      |
| Mean      | 7.7654      | 7.8995      | 7.9239      | 7.9389      |

TABLE IX
Entropy comparison with existing encryption techniques.

|          | Original [47] | [48] | [49] | [50] | Ours |
|----------|--------------|------|------|------|------|
| Red      | 7.9996       | 7.9991 | 7.9793 | 7.9213 | 7.9994 | 7.9383 |
| Green    | 7.9997       | 7.9991 | 7.9793 | 7.9231 | 7.9993 | 7.9417 |
| Blue     | 7.9997       | 7.9991 | 7.9793 | 7.9342 | 7.9993 | 7.9368 |
| Mean     | 7.9997       | 7.9991 | 7.9793 | 7.9256 | 7.9994 | 7.9389 |
VI. TIME COMPLEXITY ANALYSIS

Besides the high recognition efficiency, the encryption scheme’s time-complexity is also a vital requirement for cloud-based services, which we will discuss in this section. We report the encryption time for the RGB-color SL image of different dimensions namely $32 \times 32$, $128 \times 128$, $256 \times 256$ and $512 \times 512$ with encryption block-sizes $5 \times 5$, $10 \times 10$, $20 \times 20$, $25 \times 25$ and $40 \times 40$ over the system with configuration as defined in Section IV. The chaotic behaviour of Chen’s system restricts to compute it once for the complete dataset and utilize its solution set $S$ to initialize the random noise-vectors (ref. Algorithm 2) for image encryption.

Now, we compute the time required to encrypt an $32 \times 32$-dimensional RGB-color SL image and block-size $5 \times 5$. The total number of $5 \times 5$-dimensional blocks is 49 (using the same padding technique). Each block require 0.011 second to encrypt and thus, 49 blocks require a total of $49 \times 0.011 = 0.539$ seconds. Moreover, the time required for pre-processing and post-processing operations such as partitioning an image into (R, G, B)-channels and generation of $5 \times 5$-dimensional 49 blocks followed by concatenating the encrypted blocks and channels to generate a complete encrypted image of dimension $32 \times 32$ is 1.831 seconds. Therefore, the total encryption time (with above assumptions) is $0.539 + 1.831 = 2.37$ seconds. Also, the same image with block-size $40 \times 40$ requires 0.14 seconds only. It happens because of the large block-size partition of the image into fewer blocks that ultimately require encrypting fewer blocks and vice-versa. Similarly, we compute the encryption time of RGB-color SL for dimensions $128 \times 128$, $256 \times 256$, and $512 \times 512$ with varying block-sizes, as shown in Table XII.

VII. CONCLUSION

In this paper, a privacy-preserving system, namely $P2SLR$, for end-to-end training of a D-CNNs based SLR architecture for the encrypted SL data over the cloud platform is developed. The image’s visual information reveals that the individual’s identity in the original data is protected through a proposed probabilistic block-based bit-plane image encryption scheme without altering the pixel’s location using the FOCS SVD. Additionally, the scheme partially preserves the gesture’s local features such as hand-kinematics and facial expressions in the encrypted form and significantly obfuscated the global features (non-gesture features). For recognition experiments, we trained/learned the variants of ResNet framework with a modified classifier that performs dual adaptive-pooling over the feature-extractor layer. The $ResNet50$ outperformed the other architectures for three encrypted SL gesture datasets, namely ASL, LSA64, and RWTH, and achieved the recognition accuracy in the range of $90.76\%$–$98.09\%$. Moreover, $P2SLR$ is proved to be secure theoretically and various quantitative and interactive measure cryptographic measures. Also, we observed an inversely proportional trade-off between the recognition accuracy and security level with varying block-sizes. To the best of our knowledge, $P2SLR$ is a first-of-its-kind method to develop a secure SLR system as a ‘cloud-service."

ACKNOWLEDGMENT

This research was supported by University Grant Commission (UGC), India reference grant number: 21/12/2014(ii)EU-V, 2121440593.

REFERENCES

[1] V. K. Tanwar, H. Buckchash, B. Raman, and R. Bhargava, “Dense motion analysis of German finger spellings,” Multimedia Tools and Applications, vol. 78, no. 8, pp. 9511–9536, 2019.
[2] H. Wang, X. Chai, and X. Chen, “A novel sign language recognition framework using hierarchical grassmann covariance matrix,” IEEE Transactions on Multimedia, vol. 21, no. 11, pp. 2806–2814, 2019.
[3] R. Cui, H. Liu, and C. Zhang, “A deep neural framework for continuous sign language recognition by iterative training,” IEEE Transactions on Multimedia, vol. 21, no. 7, pp. 1880–1891, 2019.
[4] J. Imran and B. Raman, “Deep motion templates and extreme learning machine for sign language recognition,” The Visual Computer, vol. 36, no. 6, pp. 1233–1246, 2020.
[5] S. K. Nayak and S. Tripathy, “Segdp: Secure and efficient privacy preserving provable data possession in cloud storage,” IEEE Transactions on Services Computing, 2018.
[6] R. Bost, R. A. Popa, S. Tu, and S. Goldwasser, “Machine learning classification over encrypted data.” in NDSS, 2015.
[7] A. Chatterjee and I. Sengupta, “Sorting of fully homomorphic encrypted cloud data: Can partitioning be effective?,” IEEE Transactions on Services Computing, vol. 13, no. 3, pp. 545–558, 2017.
[8] T. Hoang, A. A. Yavuz, and J. G. Merchan, “A secure searchable encryption framework for privacy-critical cloud storage services,” IEEE Transactions on Services Computing, 2019.
[9] Q. Lu, C. Zhu, and X. Deng, “An efficient image encryption scheme based on the lss chaotic map and single s-box,” IEEE Access, vol. 8, pp. 25 664–25 678, 2020.
[10] C. Fu, B.-b. Lin, Y.-s. Miao, X. Liu, and J.-j. Chen, “A novel chaos-based bit-level permutation scheme for digital image encryption,” Optics Communications, vol. 284, no. 23, pp. 5415–5423, 2011.
M. S. Ryoo, K. Kim, and H. J. Yang, “Extreme low resolution activity
recognition based on chaotic maps,” *Optics and Lasers in Engineering*, vol. 78, pp. 17–25, 2016.

D. Chen, R. Zhang, J. Sprott, H. Chen, and X. Ma, “Synchronization between integer-order chaotic systems and a class of fractional-order chaotic systems via sliding mode control,” *Chaos: An Interdisciplinary Journal of Nonlinear Science*, vol. 22, no. 2, p. 023130, 2012.

Z. Wang, J. Liu, F. Zhang, and S. Leng, “Hidden chaotic attractors and synchronization for a new fractional-order chaotic system,” *Journal of Computational and Nonlinear Dynamics*, vol. 14, no. 8, 2019.

L. Xu, T. Bao, L. Zhu, and Y. Zhang, “Trust-based privacy-preserving photo sharing in online social networks,” *IEEE Transactions on Multimedia*, vol. 21, no. 3, pp. 591–602, 2019.

S. Hu, M. Li, Q. Wang, S. S. Chow, and M. Du, “Outsourced biometric identification with privacy,” *IEEE Transactions on Information Forensics and Security*, vol. 13, no. 10, pp. 2448–2463, 2018.

Y. A. Sekhvat, “Privacy preserving cloth try-on using mobile augmented reality,” *IEEE Transactions on Multimedia*, vol. 19, no. 5, pp. 1041–1049, 2017.

W. Tang, J. Ren, and Y. Zhang, “Enabling trusted and privacy-preserving healthcare services in social media health networks,” *IEEE Transactions on Multimedia*, vol. 21, no. 3, pp. 579–590, 2019.

R. Zhang, R. Xue, and L. Liu, “Security and privacy for healthcare blockchains,” *IEEE Transactions on Services Computing*, 2021.

X. Li, Y. Zha, and J. Wang, “Highly efficient privacy preserving location-based services with enhanced one-round blind filter,” *IEEE Transactions on Emerging Topics in Computing*, 2019.

G. Cui, Q. He, F. Chen, H. Jin, Y. Xiang, and Y. Yang, “Location privacy protection via delocalization in 5g mobile edge computing environment,” *IEEE Transactions on Services Computing*, 2021.

Y. Rahulamathavan, R. C.-W. Phan, J. A. Chambers, and D. J. Parish, “Facial expression recognition in the encrypted domain based on local fisher discriminant analysis,” *IEEE Transactions on Affective Computing*, vol. 4, no. 1, pp. 83–92, 2013.

Z. Ren, Y. Jiee Lee, and M. S. Ryoo, “Learning to anonymize faces for privacy preserving action detection,” in *Proceedings of the European Conference on Computer Vision*, 2018, pp. 620–636.

M. Zheng, J. Zhou, Z. Cao, and X. Dong, “Poptim: privacy-preserving shape context based image denoising and matching with efficient outsourcing,” in *International Conference on Information and Communications Security*. Springer, 2018, pp. 215–231.

S. Wang and J. M. Chang, “Privacy-preserving image classification in the local setting,” *arXiv preprint arXiv:2002.03261*, 2020.

Y. Chen, Y. Ping, Z. Zhang, B. Wang, and S. He, “Privacy-preserving image multi-classification deep learning model in robot system of industrial iot,” *Neural Computing and Applications*, pp. 1–18, 2020.

P. Dallier, “Public key cryptosystems based on composite degree residuosity classes,” in *International conference on the theory and applications of cryptographic techniques*. Springer, 1999, pp. 223–238.

M. Abadi, A. Chu, I. Goodfellow, H. B. McMahan, I. Mironov, K. Talwar, and L. Zhang, “Deep learning with differential privacy,” in *ACM SIGSAC Conference on Computer and Communications Security*, 2016, pp. 308–318.

P. Xie, M. Bilenko, T. Finley, R. Gilad-Bachrach, K. Lauter, and M. Naehrig, “Crypto-nets: Neural networks over encrypted data,” *arXiv preprint arXiv:1412.6181*, 2014.

R. Gilad-Bachrach, N. Dowlut, K. Laine, K. Lauter, M. Naehrig, and J. Wernsing, “Cryptonets: Applying neural networks to encrypted data with high throughput and accuracy,” in *Proceedings of the International Conference on Machine Learning*, 2016, pp. 201–210.

H. Li, K. Wang, X. Liu, Y. Sun, and S. Guo, “A selective privacy-preserving approach for multimedia data,” *IEEE Multimedia*, vol. 24, no. 4, pp. 14–25, 2017.

W. Ding, X. Jing, Z. Yan, and L. Y. Yang, “A survey on data fusion in internet of things: Towards secure and privacy-preserving fusion,” *Information Fusion*, vol. 51, pp. 129–144, 2019.

J. Dai, B. Saghati, J. Wu, J. Konrad, and P. Ishwar, “Towards privacy-preserving recognition of human activities,” in *Proceedings of the IEEE International Conference on Image Processing (ICIP)*, 2015, pp. 4238–4242.

M. S. Ryoo, B. Rothrock, C. Fleming, and H. J. Yang, “Privacy-preserving human activity recognition from extreme low resolution,” *arXiv preprint arXiv:1604.03196*, 2016.

M. S. Ryoo, K. Kim, and H. J. Yang, “Extreme low resolution activity recognition with multi-siamese embedding learning,” *arXiv preprint arXiv:1708.00999*, 2017.

P. Li, J. Xu, J. Mou, and F. Yang, “Fractional-order 4d hyperchaotic memristive system and application in color image encryption,” *EURASIP Journal on Image and Video Processing*, vol. 2019, no. 1, p. 22, 2019.

J. G. Lu and G. Chen, “A note on the fractional-order chen system,” *Chaos, Solitons & Fractals*, vol. 27, no. 3, pp. 685–688, 2006.

K. He, X. Zhang, S. Ren, and J. Sun, “Deep residual learning for image recognition,” in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2016, pp. 770–778.

Z. Lu, X. Jiang, and A. Kot, “Deep coupled reset for low-resolution face recognition,” *IEEE Signal Processing Letters*, vol. 25, no. 4, pp. 526–530, 2018.

J. Dai, Y. Li, K. He, and J. Sun, “R-fcn: Object detection via region-based fully convolutional networks,” *arXiv preprint arXiv:1605.06409*, 2016.

E. K. Kumar, P. Kishore, M. T. K. Kumar, D. A. Kumar, and A. Sastry, “Three-dimensional sign language recognition with angular velocity maps and convolved feature reset,” *IEEE Signal Processing Letters*, vol. 25, no. 12, pp. 1860–1864, 2018.

F. Ronchetti, F. Quirroga, C. A. Estrebou, L. C. Lanzariní, and A. Rosete, "LSA64: an argentian sign language dataset," in XXII Congreso Argentino de Ciencias de la Computación, 2016.

P. Dreuw, T. Deselaers, D. Keysers, and H. Ney, “Modeling image variability in appearance-based gesture recognition,” in *ECCV workshop on statistical methods in multi-image and video processing*, 2006, pp. 7–18.

Z.-J. Zhu, W. Zhang, K.-W. Wong, and H. Yu, “A chaos-based symmetric image encryption scheme using a bit-level permutation,” *Information Sciences*, vol. 181, no. 6, pp. 1171–1186, 2011.

Y. Zhang and D. Xiao, “An image encryption scheme based on rotation matrix bit-level permutation and block diffusion,” *Communications in Nonlinear Science and Numerical Simulation*, vol. 19, no. 1, pp. 74–82, 2014.

P. Ping, J. Fan, Y. Mao, F. Xu, and J. Gao, “A chaos based image encryption scheme using digit-level permutation and block diffusion,” *IEEE Access*, vol. 6, pp. 67 581–67 593, 2018.

C. E. Shannon, “A mathematical theory of communication,” *ACM SIGMOBILE Mobile Computing and Communications Review*, vol. 5, no. 1, pp. 3–55, 2001.

A. Souyah and K. M. Farooun, “Fast and efficient randomized encryption scheme for digital images based on quadtree decomposition and reversible memory cellular automata,” *Nonlinear Dynamics*, vol. 84, no. 2, pp. 715–732, 2016.

X. Wang, X. Zhu, and Y. Zhang, “An image encryption algorithm based on josephus traversing and mixed chaotic map,” *IEEE Access*, vol. 6, pp. 23 733–23 746, 2018.

N. Zhou, Y. Hu, L. Gong, and G. Li, “Quantum image encryption scheme with iterative generalized Arnold transforms and quantum image cycle shift operations,” *Quantum Information Processing*, vol. 16, no. 6, p. 164, 2017.

X. Wang, L. Feng, and H. Zhao, “Fast image encryption algorithm based on parallel computing system,” *Information Sciences*, vol. 486, pp. 340–358, 2019.

Y.-Q. Zhang and X.-Y. Yang, “A symmetric image encryption algorithm based on mixed linear–nonlinear coupled map lattice,” *Information Sciences*, vol. 273, pp. 329–351, 2014.

X. Wang, L. Liu, and Y. Zhang, “A novel chaotic block image encryption algorithm based on dynamic random growth technique,” *Optics and Lasers in Engineering*, vol. 66, pp. 10–18, 2015.

G. Ye and X. Huang, “An efficient symmetric image encryption algorithm based on an intertwining logistic map,” *Neurocomputing*, vol. 251, pp. 45–53, 2017.

Vishesh Kumar Tanwar is a post-doctoral associate in the Center for Research in Computer Vision, University of Central Florida, USA. He received his Ph.D. in Mathematics from the Indian Institute of Technology Roorkee, Roorkee, India. He obtained his Bachelor (2011) and Master (2013) degree in Mathematics from University of Delhi, India. His research interests include Privacy Preserving Computer Vision, Differential Privacy, Federated Learning and Cloud Computing.
Gaurav Shrama (Fellow, IEEE) received the B.E. degree in electronics and communication engineering from the Indian Institute of Technology, Roorkee (formerly, University of Roorkee), the master’s degree in electrical communication engineering from the Indian Institute of Science, Bengaluru, India, and the master’s degree in applied mathematics and the Ph.D. degree in electrical and computer engineering from North Carolina State University, Raleigh, NC, USA. From 1996 to 2003, he was with the Xerox Research and Technology, Webster, NY, USA, first as a member of Research and Technology Staff and then as the Principal Scientist and the Project Leader. From 2008 to 2010, he served as the Director of the Center for Emerging and Innovative Sciences (CEIS), a New York state supported center for promoting joint university-industry research and technology development, University of Rochester. He is currently at the Departments of Electrical and Computer Engineering, Computer Science, and Biostatistics and Computational Biology, University of Rochester. His research interests include data analytics, signal and image processing, computer vision, color imaging, media security, and communications. He is a fellow of SPIE and the Society of Imaging Science and Technology (IS&T). He served as the Editor-in-Chief (EIC) for the IEEE TRANSACTIONS ON IMAGE PROCESSING (TIP), from 2018 to 2020, and the Journal of Electronic Imaging (JEI), from 2011 to 2015.

Balasubramanian Raman is a professor in the Department of Computer Science and Engineering at Indian Institute of Technology Roorkee, obtained his Bachelor and Master Degree in Mathematics in 1994 and 1996 respectively, and Ph.D from Indian Institute of Technology Madras in 2001. He was a Post Doctoral Fellow at University of Missouri Columbia, USA in 2001-2002 and a Post Doctoral Associate at Rutgers, the State University of New Jersey, USA in 2002-2003. He was a Visiting Professor and a member of Computer Vision and Sensing Systems Laboratory in the Department of Electrical and Computer Engineering at University of Windsor, CANADA during May–August 2009. His area of Research includes Digital Watermarking using Mathematical Transformations, Biometrics, Secure Image Transmission over, Content Based Image Retrieval and Hyper-spectral Imaging. He has more than 150 research publications in reputed journals and conference proceedings.

Rama Bhargava was a professor in the Department of Mathematics at Indian Institute of Technology Roorkee from 2001. She obtained her Bachelor and Master degree in Mathematics in 1973 and 1975 respectively. She obtained her Ph.D from University of Roorkee in 1978. She has honored as International Women of the Year in years 2001 and 2007. In 2009–2010, she was honored with INSA Exchange Award for UK by Royal Society of London and INSA. She has published more than 165 research papers in reputed journals and conferences. Her research interests include Computational Fluid Dynamics, Finite Element and Computer Graphics.