Privacy-preserving Image Classification with Deep Learning and Double Random Phase Encoding

Falu Yi¹,², Ongee Jeong², Inkyu Moon², Member, IEEE

¹Department of Clinical Science, University of Texas Southwestern Medical Center, Dallas, TX, 75390, USA
²Department of Robotics Engineering, DGIST, Daegu 42988, South Korea

Corresponding author: Inkyu Moon (e-mail: inkyu.moon@dgist.ac.kr).

1Co-first authors; these authors contributed equally to this work. This work was supported by the National Research Foundation of Korea (NRF) grant funded by the Korean government (MSIT) (No. 2020R1A2C3006234).

ABSTRACT With the emergence of cloud computing, large amounts of private data are stored and processed in the cloud. On the other hand, data owners (users) may not want to reveal data information to cloud providers to protect their privacy. Therefore, users may upload encrypted data to the cloud or third-party platforms, such as Google Cloud, Amazon Web Service, and Microsoft Azure. Conventionally, data must be decrypted before being analyzed in the cloud, which raises privacy concerns. Moreover, decryption of big data such as images requires enormous computation resources, which is unsuitable for energy-constrained devices, particularly Internet of Things (IoT) devices. Data privacy in most popular applications, such as image query or classification, can be preserved if encrypted images can be directly classified on the cloud or IoT devices without decryption. This paper proposes a high-speed double random phase encoding (DRPE) technique of encrypting images into white-noise images. DRPE-encrypted images are then uploaded and stored in the cloud. Images that are encrypted without being decrypted are classified using deep convolutional neural networks in the cloud. The simulation results indicated the feasibility and good performance of the proposed approach. The proposed privacy-preserving image classification method can be useful in data-sensitive fields, such as medicine and transportation.

INDEX TERMS Double random phase encoding, information security, image encryption, deep learning, privacy preserving

I. INTRODUCTION

With the development of the Internet of Things (IoT), several wearable devices, home appliances, agriculture and transportation tools, and other devices are connected to the Internet. These devices generate large amounts of data every day, which are primarily stored in the cloud [1–5]. These data are also mostly processed in the cloud using cloud computing services from third-party platforms, such as Google, Amazon, and Microsoft because they can provide sufficient storage space and processing power. Although IoT and cloud computing techniques have made it easier to work and live, studies have shown that most consumers lack confidence in the data security of IoT devices and cloud computing [6–9]. An individual may not want unauthorized entities to have access to their tour photos, including family information. A patient does not want their medical records and diagnostic reports to be shared with others. A driver may not want to leak their private information, such as license plate, location, and driving habits, to outsiders. Security is a significant concern for data generated by IoT devices or data stored in the cloud because there is a high probability that some personal information would be included in this digital information [6–10]. Therefore, many privacy-preservation approaches, such as data encryption, have been proposed to make data transmission, storage, and extraction safe [11–14]. On the other hand, data encryption in the cloud can decrease the data processing efficiency for some applications, such as image classification and retrieval, because decrypting and then classifying or querying a large number of encrypted images may necessitate considerable computation resources. In addition, decryption processing within the cloud will reveal original images to unauthorized parties, such as...
cloud provider companies. Consequently, data processing based on encrypted information is promising and has become critical [15–24]. Image data are typically large, and image encryption with methods, such as data encryption standard (DES) or advanced encryption standard (AES), may be time-consuming and unsuitable for low-computing devices, with limited power and computation capability, in IoT systems. By contrast, the double random phase encoding (DRPE) algorithm, which was proposed in [25], is an optical encryption algorithm with inherent parallel computing ability that be implemented efficiently. The DRPE can encrypt an input image into a white noise image that will not reveal any information about the original data. The original DRPE technique is achieved in the Fourier domain, whereas several other variations are implemented in different domains, such as the Fresnel domain, fractional Fourier domain, and gyrator transform domain [26–28]. The DRPE algorithm has been studied extensively and used widely in image encryption, authentication, and watermarking [29–32]. A previous study [33] claimed that the DRPE algorithm could be a good encryption algorithm in an IoT system for energy-constrained devices, while a secure key exchange scheme is proposed for image cryptography. Therefore, the DRPE approach would be a suitable scheme for image encryption for large-scale image datasets. The efficiency of image encryption is crucial, particularly for large data transmitted between devices and cloud servers. With advances in computing power, such as graphic processing units, big datasets, such as ImageNet [34], and advanced training schemes, deep learning (DL) methods have become a hot research field in recent years [35]. Because AlexNet [36] achieved good performance in the ImageNet Large-Scale Recognition Challenge, several DL architecture variants have been proposed. For example, ResNet [37], DenseNet [38], and Inception Net [39], are good classification convolutional neural networks (CNNs). U-Net [40], DeepLab [41], and Gated-SCNN [42] are DL models proposed for semantic segmentation. The Faster RCNN [43], single-shot detector [44], and You Only Look Once [45] are robust neural networks (NNs) for object detection. BERT [46] and GPT [47] are robust natural language processing (NLP) models. Traditional ML approaches typically require users to design and discover useful features themselves. Thus, domain knowledge may be critical for feature extraction. By contrast, DL methods can automatically extract the relevant features from data by optimizing a target function, e.g., minimizing a loss function [48]. Moreover, DL models are more robust to images with illumination variations, color differences, and target location offset. DL models are used widely in image analysis, audio processing, and NLP [48, 49]. They are also used extensively in the classification of encrypted images because deep NNs have a great ability to automatically extract good features from data, even encrypted data, whereas manual extraction of the good features from encrypted data is difficult because encryption may cause the encrypted data to have few discriminative patterns to identify a specific category.

This paper proposed an image classification method based on encrypted image data. The proposed method can be crucial for cloud computing and IoT systems. First, the images are encrypted using the DRPE method. The encrypted images are then transmitted to the cloud owned by a third-party company. DRPE-encrypted images are white-noise images that will not reveal the clients’ sensitive information in the cloud. Without decrypting the images in the cloud to train an ML algorithm, DL models have been developed and trained directly using the encrypted data, increasing computation efficiency because the decryption of many images in the cloud is highly time and resource consuming. This study developed two types of DL models for encrypted data classification. One is a CNN, with a similar architecture to the conventional CNN, which is used mainly for classification. The other is an encoder–decoder structure with a CNN as an additional branch. The second model can achieve image classification and decryption simultaneously. The remaining part of this paper is organized as follows. Section II reports the related works. Section III describes DRPE. Section IV presents the procedure of the encrypted image classification. Section V presents experimental results, and the conclusions are drawn in Section VI.

II. RELATED WORKS

Data processing based on encrypted information provides a good way for privacy preservation. Several studies have evaluated information retrieval and classification using encrypted data stored in the cloud to protect sensitive information. In [17], images were captured using roadside units, and the vehicles in those images were segmented through edge detection. Furthermore, segmented images are encrypted using a suitable algorithm using a selected mode of operation, and the encrypted data were classified based on the convolutional neural networks. As a result, the encrypted image provides a way of protecting the drivers’ sensitive information in intelligent transportation systems. In [22], a deep learning model was designed to automatically extract the useful features from encrypted traffic data for traffic identification and classification to preserve the users’ privacy. In [18], they proposed a privacy-preserving algorithm for classifying images, which are encrypted using a pixel-based image encryption method. This algorithm could achieve image augmentation in the encryption domain during the algorithm-training phase. In [15], a block encryption algorithm, such as DES and AES [23], was used for image encryption, and encrypted images were classified using a trained multilayer extreme learning machine to achieve data security. In [24], the researchers proposed deep CNNs with a novel activation function to classify encrypted image data generated using a homomorphic encryption method. Their results highlighted the robustness of the approach proposed.
in the study over encrypted data. In [16], image data are also encrypted using a homomorphic encryption algorithm, and encrypted data were classified using a non-linear support vector machine. The original data were encrypted, and machine learning (ML) algorithms were trained and inferred based on encrypted images, providing a good way to protect sensitive information among these methods. On the other hand, encrypted data based on optical encryption approaches with inherent parallel properties have not been researched. This paper reports the feasibility of encrypted image classification based on DRPE-encrypted data.

Figure 1. Schematic diagram of double random phase encoding in the Fourier domain

III. DOUBLE RANDOM PHASE ENCODING

DRPE, as a popular optical security approach, has been researched extensively because of the easy configuration and parallel processing properties [26–28]. DRPE has also been widely used in image authentication, information hiding, and watermarking [29–33]. Figure 1 presents a graphic diagram of a DRPE method in the Fourier domain. In the DRPE scheme, the input image \( I(x,y) \) is encoded into a stationary white-noise image \( E(\xi,\eta) \) using two random phase masks, \( m_1 = \exp(j2\pi m(x,y)) \) and \( m_2 = \exp(j2\pi n(u,v)) \), where each element within \( m(x,y) \) and \( n(u,v) \) is distributed uniformly between 0 and 1. \( \exp(j) \) represents the natural exponential function, and \( j \) represents the imaginary unit. For DRPE implementation in the Fourier domain, the first random phase mask \( m_1 \) is located in the input image plane. The second random phase mask, \( m_2 \), is positioned in the Fourier domain. For DRPE implemented in other domains, such as the Fresnel, Gyrator, and Fractional Fourier domains, the random phase mask \( m_1 \) remains in the input image plane, whereas the second random phase mask, \( m_2 \), is placed at the corresponding domain [30]. The computational implementation of DRPE in the Fourier domain can also be written mathematically as:

\[
E(\eta,\xi) = FT^{-1}[FT[O(x,y)\exp(j2\pi m(x,y))]\times\exp(j2\pi n(u,v))],
\]

(1)

where \( FT \) and \( FT^{-1} \) represent the 2D Fourier and inverse Fourier transforms, respectively. By contrast, the implementation of DRPE decryption in the Fourier domain is the inverse operation of DRPE in encryption processing. DRPE decryption can be expressed mathematically as follows:

\[
D(x,y) = \left| FT^{-1}[FT[E(\eta,\xi)\text{conj}(\exp(j2\pi n(u,v)))]] \right|,
\]

(2)

where \( D(x,y) \) is the decrypted image from DRPE, and \( |\cdot| \) is the modulus operation, \( \text{conj} \) is the complex conjugate.

Equation (2) does not include the first random phase mask, \( m_1 \), because the intensity of the decrypted image \( D(x,y) \) will not be affected because of the modulus operation [25].

Figure 2. Procedure of encrypted image classification

IV. CLASSIFICATION OF DOUBLE RANDOM PHASE ENCRYPTED IMAGES

The first step in the encrypted image classification process is to encrypt input images before transmitting them to the cloud. Consequently, in the cloud, there are only encrypted images, and no image information is revealed. In this study, all images were encrypted with DRPE algorithms. The elements of the DRPE-encrypted images are complex values, and both real and imaginary parts are used as inputs in training a DL model. Figure 2 presents the procedure for DRPE-encrypted image classification. As illustrated in Figure 2, the trained model was tested for encrypted image classification after training the DL model. In the testing phase, the same random phase mask keys as those used in the training step were adopted in the DRPE algorithm, and the encrypted images with both real and imaginary parts are the inputs of the trained model for prediction.

Figure 3. CNN architecture for the DRPE-encrypted image classification

This paper proposed two DL models for DRPE-encrypted image classification. The first is a CNN that is referred to as the CNN algorithm in the following description. The second one is a fully connected CNN with an auxiliary branch for encrypted image classification. The second DL structure is expressed as FCN_{Aux} in the following context. Figure 3 shows the CNN method used here. In this CNN approach, the input image size was \( 32 \times 32 \times 2 \), and there were 10 categories in the output layer. In Figure 3, the Conv 3 x 3 is a convolutional operation with a kernel size of \( 3 \times 3 \). BN represents batch normalization. ReLu is the rectified linear unit activation function. Max pooling \( 2 \times 2 \) means max-
pooling operation with a stride value of 2 in both the x and y directions in the feature map. The max-pooling layer may extract the largest element from the 2 × 2 regions in the feature map, and the resolution was reduced by a factor of 2 × 2 at each max-pooling 2 × 2 block. FC denotes the fully connected operation, whereas Softmax is a normalized exponential function that can normalize the output of an NN to a probability distribution over the predicted output classes [49, 50]. The number above each block in Figure 3 is the size of the feature map of that layer. Typically, the convolutional operation in NNs can preserve the spatial relationship among pixels and learn the image features within the receptive field [49, 50]. An activation function, such as ReLu, can introduce non-linearity into the models and enhance the ability to extract complex features [49, 50]. Max pooling can reduce feature map dimensionality and is beneficial to the models’ computation. This also makes the DL models more robust to target translation within the images [49, 50]. BN can mitigate the internal covariate shift problem and reduce the gradient vanishing problem during algorithm training [51].

![FCNAux architecture for DRPE-encrypted image classification](image)

**FIGURE 4.** FCNAux architecture for DRPE-encrypted image classification

Figure 4 presents the FCNAux architecture. Compared to the DL structure in Figure 3, there is an additional decoder part in the FCNAux structure in Figure 4. The decoder operation in FCNAux can be considered a decryption operation and can help modulate the NNs to learn features that are helpful for both classification and decryption. In Figure 4, the Up-Conv 2 × 2 means an up-pooling operation, which can increase the feature map resolution by a factor of 2 × 2. The Sigmoid function is an activation function that can scale the output values to a range of 0–1. For both CNN and FCNAux models, the cross-entropy (CE) loss is used for encrypted image classification and is expressed as the following equation [48, 50]:

\[
CE = -\frac{1}{N} \sum_{i=1}^{M} y_{oc} \log(p_{oc}),
\]

where N is the size of the dataset, and M is the number of classes; \(y_{oc}\) is the binary indicator (0 or 1), which is 1 when the class label \(c\) is the correct classification for the observation sample \(o\); otherwise, \(y_{oc}\) is 0. \(p_{oc}\) is the predicted probability for observation \(o\) with class \(c\). The mean absolute error (MAE) is used as the regression loss function for the decoder branch in the FCNAux model, as shown in Figure 4, and it is mathematically given as follows:

\[
MAE = \frac{1}{MN} \sum_{i=1}^{M} \sum_{j=1}^{N} |P(i, j) - G(i, j)|.
\]

where \(M\) and \(N\) are the sizes of an image along the \(x\) and \(y\) directions, respectively. \(G(i, j)\) is the ground truth pixel value at the image location \((i, j)\), and \(P(i, j)\) is the predicted image pixel value at the image location \((i, j)\). The \(|\cdot|\) denotes the absolute operation.

**V. EXPERIMENTAL RESULTS**

The Fashion-MNIST dataset [52] is used to demonstrate the proposed approach. The dataset contained 70,000 images, of which 50,000 are used for training; 10,000, for validation; and 10,000, for testing. The original size of the images in the Fashion-MNIST dataset was 28 × 28, and they were resized to 32 × 32, and then encrypted using the DRPE algorithm. There are 10 categories within this dataset: T-shirt, Trouser, pullover, dress, coat, sandal, shirt, sneaker, bag, and ankle boot. The models were developed on a server with 48 central processing units (CPUs) (Inter(R) Xeon(R) CPU E5-2650, Ubuntu 16.04 operating system, and one P100 NVIDIA GPU). The DL algorithms were implemented with PyTorch, a Python DL framework [53]. Both training and testing were performed through GPU parallel computing. Figure 5 presents a DRPE-encrypted image. There are real and imaginary parts of the DRPE-encrypted image in the figure because the element of the encrypted image is a complex number. The real and imaginary parts of the encrypted image were concatenated to form a 32 × 32 × 2 tensor, which serves as the input to the DL models.

![Illustration of a DRPE-encrypted image. (a) Original image, (b) real part of the encrypted image, and (c) imaginary part of the encrypted images](image)

**FIGURE 5.** Illustration of a DRPE-encrypted image. (a) Original image, (b) real part of the encrypted image, and (c) imaginary part of the encrypted images

![Epoch number versus loss value for the CNN algorithm](image)

**FIGURE 6.** Epoch number versus loss value for the CNN algorithm
To train the CNN algorithm, the batch size was set to 64; the learning rate was given as 0.01, which decreased by a factor of 10 every 40 epochs; the number of epochs was set to 120. L2 regularization [50] was used in this study, and its value was given as 0.0005. The momentum gradient descent method was used as the optimization approach, which was set to 0.9. An image augmentation technique was used in algorithm training, including image flipping, random image rotation, random noise addition, and random pixel exclusion. The overfitting problem was mitigated by performing the early-stop technique on the training and validation datasets.

Figure 6 presents the epoch number versus loss value curve on the training and validation datasets. As shown in Figure 6, the loss difference between the training and validation datasets begins to enlarge at the epoch number of 82. Therefore, the trained model at epoch number 82 was used for testing.

After training the CNN algorithm, the trained model was evaluated using 10,000 test images. The accuracy of the CNN algorithm for DRPE-encrypted image classification was 0.8972. Table I lists the corresponding confusion matrix.

### TABLE I

| Predicted Class | T-shirt | Trouser | Pullover | Dress | Coat | Sandal | Shirt | Sneaker | Bag | Ankle boot |
|-----------------|---------|---------|----------|-------|------|--------|-------|---------|-----|------------|
| T-shirt         | 820     | 1       | 15       | 20    | 3    | 1      | 130   | 0       | 10 | 0          |
| Trouser         | 0       | 978     | 4        | 16    | 0    | 2      | 0     | 0       | 0   | 0          |
| Pullover        | 15      | 1       | 815      | 8     | 82   | 0      | 76    | 0       | 3  | 0          |
| Dress           | 15      | 9       | 18       | 894   | 36   | 0      | 24    | 0       | 4  | 0          |
| Coat            | 1       | 1       | 80       | 23    | 847  | 0      | 43    | 0       | 5  | 0          |
| Sandal          | 0       | 0       | 0        | 0     | 967  | 0      | 23    | 2       | 8  | 0          |
| Shirt           | 105     | 4       | 56       | 18    | 64   | 1      | 746   | 0       | 6  | 0          |
| Sneaker         | 0       | 0       | 0        | 0     | 13   | 0      | 959   | 0       | 28 | 0          |
| Bag             | 5       | 0       | 0        | 3     | 4    | 3      | 3     | 4       | 978| 0          |
| Ankle boot      | 0       | 0       | 0        | 0     | 4    | 1      | 27    | 0       | 968| 0          |

### TABLE II

| Predicted Class | T-shirt | Trouser | Pullover | Dress | Coat | Sandal | Shirt | Sneaker | Bag | Ankle boot |
|-----------------|---------|---------|----------|-------|------|--------|-------|---------|-----|------------|
| T-shirt         | 843     | 1       | 18       | 17    | 3    | 0      | 112   | 0       | 6  | 0          |
| Trouser         | 3       | 983     | 0        | 11    | 1    | 0      | 1     | 0       | 1  | 0          |
| Pullover        | 12      | 1       | 828      | 6     | 100  | 0      | 53    | 0       | 0  | 0          |
| Dress           | 12      | 2       | 7        | 902   | 35   | 0      | 39    | 0       | 2  | 1          |
| Coat            | 2       | 0       | 34       | 21    | 913  | 0      | 30    | 0       | 0  | 0          |
| Sandal          | 0       | 0       | 0        | 0     | 1    | 0      | 984   | 0       | 14 | 0          |
| Shirt           | 93      | 1       | 54       | 26    | 101  | 0      | 722   | 0       | 3  | 0          |
| Sneaker         | 0       | 0       | 0        | 0     | 4    | 0      | 983   | 0       | 13 | 0          |
| Bag             | 3       | 0       | 6        | 2     | 0    | 1      | 2     | 1       | 985| 0          |
| Ankle boot      | 0       | 0       | 1        | 0     | 6    | 0      | 38    | 0       | 955| 0          |

**FIGURE 7.** Illustration of partial pixel loss attack. (a) An original image in the Fashion-MNIST test dataset, (b) real part of a DRPE-encrypted image, (c) encrypted image with 20% of pixels excluded, (d) an encrypted image with 40% of pixels excluded, and (e) encrypted image with 60% of pixels excluded.

For comparison, the same CNN structure was used to classify the original images without DRPE encryption. That is, the CNN algorithm was trained and tested using the original images without applying any encryption algorithm. In this case, the input image size for the CNN algorithm was $32 \times 1$. An accuracy of 0.9098 was obtained on the 10,000 test images. Table I lists the corresponding prediction confusion matrix. Prediction results in Tables I and II have indicated that the trained CNN algorithm obtained similar classification accuracy for both DRPE-encrypted images and the original images (i.e., without DRPE encryption).

Some attacks were used to assess the robustness of the CNN algorithms. Data can easily suffer from noise contamination or pixel value loss during data transmission on the Internet. Therefore, the ability of the proposed CNN architecture for encrypted image classification was tested in terms of partial pixel loss and noise attack. Partial pixel loss attack was simulated by randomly excluding 20%, 40%, and 60% of pixels from an encrypted image. Figure 7 presents the
encrypted image with some pixels being excluded randomly. Only the real part image in the encrypted image is presented in Figure 7, whereas the same operation was performed on the imaginary part. The excluded pixel locations were given a value of zero in the partial pixel loss attack. Table III lists the encrypted image classification accuracy of the proposed CNN model for images with some pixels excluded randomly. The simulation for the noise attack is based on additive noise. The adding noise operation is expressed in the following equation [54]:

\[ E_{\text{noise}} = E \times (1 + w \times m), \]

where \( E \) and \( E_{\text{noise}} \) are the DRPE-encrypted image and DRPE-encrypted image with additive noise, respectively; \( w \) is the weight of noise; \( m \) is the matrix with the same size as the encrypted image \( E \), with the element value chosen at random between 0 and 1. The additive noise attack test was performed with three different weight values: 0.25, 0.5, and 1. Figure 8 illustrates the encrypted image with the addition of the noise attack. Similarly, Figure 8 shows only the real part of the DRPE-encrypted image. The CNN classification results for DRPE-encrypted images under the noise attack are presented in Table III. As shown in Table III, the proposed CNN algorithm can achieve good DRPE-encrypted image classification performance under partial pixel loss and noise attacks.

To test the performance of the FCNAux model, the same hyperparameters, as used to test the CNN model, were configured, and the same early stop technique was used to reduce the overfitting problem. Figure 9 shows the epoch number versus loss value curve. The final epoch number selected was 81 because the loss value begins to increase for the validation dataset, whereas that for the training dataset decreases. Consequently, the trained model at epoch number 81 evaluated the DRPE-encrypted image classification performance of the model. The accuracy of the FCNAux algorithm on 10,000 test images was calculated to be 0.8944. Table IV lists the confusion matrix. The FNCAux model can achieve classification results and output regression images. Figure 10 presents the regressed images generated by FCNAux. As illustrated in Figure 10, the FCNAux model can generate images similar to the original images, even though some details in the regressed images are missing.

**TABLE III**

| ENCRYPTED IMAGE CLASSIFICATION OF CNN BASED ON AN ATTACK |
|----------------------------------------------------------|
| Partial pixel exclusion attack | Adding noise attack |
| 20% exclusion | 40% exclusion | 60% exclusion | Weight | Weight | Weight | Weight |
| Accuracy | 0.8855 | 0.8755 | 0.8729 | 0.8929 | 0.8814 | 0.8262 |

**TABLE IV**

| CONFUSION MATRIX OF FCNAUX PREDICTION RESULTS BASED ON DRPE-ENCRYPTED IMAGES |
|--------------------------------------------------------------------------------|
| Predicted Class | T-shirt | Trouser | Pullover | Dress | Coat | Sandy | Shirt | Sneaker | Bag | Ankle boot |
| True class | 864 | 4 | 22 | 1 | 0 | 0 | 137 | 0 | 6 | 1 |
| T-shirt | 1 | 16 | 33 | 2 | 3 | 0 | 74 | 0 | 0 | 0 |
| Trouser | 976 | 0 | 14 | 3 | 0 | 3 | 0 | 10 | 0 | 0 |
| Pullover | 836 | 10 | 55 | 0 | 71 | 0 | 4 | 0 | 0 | 0 |
| Dress | 1 | 34 | 775 | 0 | 68 | 0 | 2 | 0 | 0 | 0 |
| Coat | 921 | 27 | 0 | 9 | 0 | 8 | 0 | 0 | 0 | 0 |
| Sandy | 115 | 0 | 0 | 0 | 978 | 0 | 16 | 0 | 0 | 0 |
| Shirt | 30 | 43 | 0 | 709 | 0 | 10 | 0 | 0 | 0 | 0 |
| Sneaker | 0 | 0 | 0 | 0 | 19 | 0 | 961 | 0 | 20 | 0 |
| Bag | 6 | 4 | 4 | 6 | 1 | 3 | 972 | 0 | 0 | 0 |
| Ankle boot | 1 | 0 | 0 | 0 | 9 | 0 | 38 | 0 | 952 | 0 |
The same attacks, partial pixel exclusion and noise attack, are applied to the FCNAux to test its robustness. For partial pixel loss attack, the exclusion pixel percentages were set to 0.2, 0.4, and 0.6. The weight value for the noise attack was 0.25, 0.5, and 1.0. Figure 11 shows the predicted images from the decoder branch in the FCNAux algorithm under the partial pixel loss attack, whereas those under the noise attack are given in Figure 12. The corresponding prediction accuracy values are included in Table V. Table V indicates that the FCNAux model is robust to partial pixel loss and noise attacks.

The proposed CNN and FCNAux algorithms were also trained and tested on the MNIST dataset [55], which was encrypted using the DRPE technique. The accuracy for the proposed CNN and FCNAux methods based on 10,000 MNIST test images was 0.9237 and 0.9278, respectively. The accuracy of the CNN algorithm using the original images (i.e., without encryption) in the MNIST dataset is calculated to be 0.9303. These results using the MNIST dataset also confirmed that the proposed CNN and FCNAux models could perform well for DRPE-encrypted image classification.

The proposed methods were also compared with those in [15,16,18], where images are encrypted with AES [15], homomorphic encryption [16], and a pixel-based encryption approach [18]. Here, 16 pixels which have a total of 128 bits are viewed as a block to be used as the input of AES and homomorphic encryption. Table VI lists the encrypted image classification results from these approaches based on 10000 testing Fashion-MNIST datasets. The method in [18] showed similar classification accuracy in the present and much better than those in [15] and [16]. This study did not focus on the classification accuracy and mainly demonstrated that encrypted data from optical encryption methods, such as DRPE, can be classified using a deep learning approach so that privacy can be preserved in the cloud. Compared to other encryption methods, such as AES in [15], homomorphic encryption in [16], and pixel-based encryption in [18], DRPE as an optical encryption method has a potential advantage in parallel computing. The computational complexity of the optical encryption algorithm was O(1) in Big-O notation [33]. They were O(n^3) for AES in [15], homomorphic encryption in [16], and pixel-based encryption in [18] for encrypting an n x n image when only “+”,”-”, “x”, “/”, and modulo operations were considered.

VI. CONCLUSIONS
This paper proposed a privacy-preserving image classification method based on DRPE and DL. Unstructured data, such as images, were first encrypted using the DRPE method. The encrypted images have a white noise appearance, which reveals no information to third parties. The parallel computing ability of the DRPE approach
improved the speed of big data encryption, which is important for edge devices, such as smartphones, sensors, and other portable devices. Image classification is a popular ML task that has always been performed on original images. On the other hand, this task raises data privacy/security concerns for users in cloud computing and IoT systems. In this study, the classification approach was established, whereby DL models were applied to DRPE-encrypted and unencrypted images. No sensitive information was leaked in training because the models were trained using encrypted images. In addition, the performance of the models in two cases (i.e., DRPE-encrypted and unencrypted images) were compared. The results suggested that the DL models achieved good performance on the DRPE-encrypted images. The proposed method can be useful for storing and processing data in the cloud because it ensures data privacy. This method is also useful in IoT systems.

REFERENCES

[1] Z. Yang, Q. Zhou, L. Lei, K. Zheng, and W. Xiang, “An IoT-cloud based wearable ECG monitoring system for smart healthcare,” Journal of medical electronics, vol. 40, no. 12, pp. 1-11, 2016.

[2] Y. Chen, “IoT, cloud, big data and AI in interdisciplinary domains,” ed: Elsevier, 2020.

[3] L. Jiang, L. Da Xu, H. Cai, Z. Jiang, F. Bu, and B. Xu, “An IoT-oriented data storage framework in cloud computing platform,” IEEE Transactions on Industrial Informatics, vol. 10, no. 2, pp. 1443-1451, 2014.

[4] A. Botta, W. De Donato, V. Persico, and A. Pescapé, “Integration of cloud computing and internet of things: a survey,” Future generation computer systems, vol. 56, pp. 684-700, 2016.

[5] M. S. Mekala and P. Viswanathan, “A Survey: Smart agriculture IoT with cloud computing,” in 2017 international conference on microelectronic devices, circuits and systems (ICMDCS), 2017: IEEE, pp. 1-7.

[6] W. H. Hassan, “Current research on Internet of Things (IoT) security: A survey,” Computer networks, vol. 148, pp. 283-294, 2019.

[7] F. A. Alaba, M. Othman, I. A. T. Hashem, and F. Alotaibi, “Internet of Things security: A survey,” Journal of Network and Computer Applications, vol. 88, pp. 10-28, 2017.

[8] A. Singh and K. Chatterjee, “Cloud security issues and challenges: A survey,” Journal of Network and Computer Applications, vol. 79, pp. 88-115, 2017.

[9] I. M. Khalil, A. Kherishav, and M. Azeem, “Cloud computing security: A survey,” Computers, vol. 3, no. 1, pp. 1-35, 2014.

[10] T. Mather, S. Kumasrawamy, and S. Latif, Cloud security and privacy: an enterprise perspective on risks and compliance. “O’Reilly Media, Inc.”, 2009.

[11] I. Han, J. Qin, and J. Hu, “Secure searches in the cloud: A survey,” Future Generation Computer Systems, vol. 62, pp. 66-75, 2016.

[12] P. Yang, N. Xiong, and J. Ren, “Data security and privacy protection for cloud storage: A survey,” IEEE Access, vol. 8, pp. 131723-131740, 2020.

[13] S. N. Kumar and A. Vajpayee, “A survey on secure cloud: security and privacy in cloud computing,” American Journal of Systems and Software, vol. 4, no. 1, pp. 14-26, 2016.

[14] A. Domingo-Ferrer, O. Farrás, J. Ribes-González, and D. Sánchez, “Privacy-preserving cloud computing on sensitive data: A survey of methods, products and challenges,” Computer Communications, vol. 140, pp. 38-60, 2019.

[15] W. Wang, C.-M. Yong, Y. Yang, and P.-K. Wong, “Encrypted image classification based on multilayer extreme learning machine,” Multidimensional Systems and Signal Processing, vol. 28, no. 3, pp. 851-865, 2017.

[16] A. Barnett et al., “Image Classification using non-linear Support Vector Machines on Encrypted Data,” IACR Cryptol. ePrint Arch., vol. 2017, p. 857, 2017.

[17] V. M. Litdieka, R. Muresan, and A. Al-Dweik, “Convolutional neural network framework for encrypted image classification in cloud-based ITS,” IEEE Open Journal of Intelligent Transportation Systems, vol. 1, pp. 35-50, 2020.

[18] W. Srirotchotudomrong, T. Maekawa, Y. Kinoshita, and H. Kiyama, “Privacy-preserving deep neural networks with pixel-based image encryption considering data augmentation in the encrypted domain,” in 2019 IEEE International Conference on Image Processing (ICIP), 2019: IEEE, pp. 674-678.
[35] S. Pouyanfar et al., “A survey on deep learning: Algorithms, techniques, and applications,” ACM Computing Surveys (CSUR), vol. 51, no. 5, pp. 1-36, 2018.

[36] A. Krizhevsky, I. Sutskever, and G. E. Hinton, “Imagenet classification with deep convolutional neural networks,” Advances in neural information processing systems, vol. 25, pp. 1097-1105, 2012.

[37] 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.

[38] G. Huang, Z. Liu, L. Van Der Maaten, and K. Q. Weinberger, “Densely connected convolutional networks,” in Proceedings of the IEEE conference on computer vision and pattern recognition, 2017, pp. 4700-4708.

[39] C. Szegedy, S. Ioffe, V. Vanhoucke, and A. Alemi, “Inception-v4, inception-resnet and the impact of residual connections on learning,” in Proceedings of the AAAI Conference on Artificial Intelligence, 2017, vol. 31, no. 1.

[40] O. Ronneberger, P. Fischer, and T. Brox, “U-net: Convolutional networks for biomedical image segmentation,” in International Conference on Medical image computing and computer-assisted intervention, 2015: Springer, pp. 234-241.

[41] L.-C. Chen, G. Papandreou, I. Kokkinos, K. Murphy, and A. L. Yuille, “Deeplab: Semantic image segmentation with deep convolutional nets, atrous convolution, and fully connected crfs,” IEEE transactions on pattern analysis and machine intelligence, vol. 40, no. 4, pp. 834-848, 2017.

[42] T. Takikawa, D. Acuna, V. Jampani, and S. Fidler, “Gated-scnn: Gated shape cnns for semantic segmentation,” in Proceedings of the IEEE/CVF International Conference on Computer Vision, 2019, pp. 5229-5238.

[43] S. Ren, K. He, R. Girshick, and J. Sun, “Faster r-cnn: Towards real-time object detection with region proposal networks,” arXiv preprint arXiv:1506.01497, 2015.

[44] W. Liu et al., “Ssd: Single shot multibox detector,” in European conference on computer vision, 2016: Springer, pp. 21-37.

[45] J. Redmon, S. Divvala, R. Girshick, and A. Farhadi, “You only look once: Unified, real-time object detection,” in Proceedings of the IEEE conference on computer vision and pattern recognition, 2016, pp. 779-788.

[46] J. Devlin, M.-W. Chang, K. Lee, and K. Toutanova, “Bert: Pre-training of deep bidirectional transformers for language understanding,” arXiv preprint arXiv:1810.04805, 2018.

[47] A. Radford, K. Narasimhan, T. Salimans, and I. Sutskever, “Improving language understanding by generative pre-training,” 2018.

[48] C. F. Higham and D. J. Higham, “Deep learning: An introduction for applied mathematicians,” SIAM Review, vol. 61, no. 4, pp. 860-891, 2019.

[49] L. Deng, “A tutorial survey of architectures, algorithms, and applications for deep learning,” APSIPA Transactions on Signal and Information Processing, vol. 3, 2014.

[50] Y. LeCun and M. Ranzato, “Deep learning tutorial,” in Tutorials in International Conference on Machine Learning (ICML ’13), 2013: Citeseer, pp. 1-29.

[51] S. Ioffe and C. Szegedy, “Batch normalization: Accelerating deep network training by reducing internal covariate shift,” in International conference on machine learning, 2015: PMLR, pp. 448-456.

[52] H. Xiao, K. Rasul, and R. Vollgraf, “Fashion-mnist: a novel image dataset for benchmarking machine learning algorithms,” arXiv preprint arXiv:1708.07747, 2017.

[53] N. Kettar, “Introduction to pytorch,” in Deep learning with python: Springer, 2017, pp. 195-208.

[54] H. Hai, S. Pan, M. Liao, D. Lu, W. He, and X. Peng, “Cryptanalysis of random-phase-encoding-based optical cryptosystem via deep learning,” Optics express, vol. 27, no. 15, pp. 21204-21213, 2019.

[55] L. Deng, “The mnist database of handwritten digit images for machine learning research [best of the web],” IEEE Signal Processing Magazine, vol. 29, no. 6, pp. 141-142, 2012.