TEST Engineering & Management

May- June 2020
ISSN: 0193-4120 Page No. 6323 - 6330

A Framework for A Secure Brain Image Classification Using Deep Learning and Residue Number System

Usman Opeyemi Lateef[1*], Ravie Chandren Muniyandi[2]
[1,2]Research Centre for Cyber Security, Faculty of Information Science and Technology, University Kebangsaan Malaysia,43600 UKM Bangi, Selangor, Malaysia.

Article Info
Volume 83
Page Number: 6323 - 6330
Publication Issue:
May- June 2020

Article History
Article Received: *19 November 2019*
Revised: *27 January 2020*
Accepted: *24 February 2020*
Publication: *18 May 2020*

Abstract:
Increasing availability of medical images generated via different imaging techniques necessitates the need for their remote analysis and diagnosis. This development has made privacy and security of patients' medical records to be extremely important. In this paper, we present a brain image classification framework using deep learning model and concept of residue number system (RNS). Special moduli set of RNS will be used to conceal 8-bit binary value of each pixel present in the training and testing image dataset before the usage of a convolutional neural network (CNN) to classify the encrypted images. As part of the classification procedure, image segmentation and data augmentation procedure will be performed in order to identify the region of interest (ROI) and to avoid overfitting respectively. Specifically, this research will attempt to explore the potencies of CNN to classify cases of dyslexia from control subjects using MRI-generated image dataset. This kind of research becomes expedient due to the educational and medical importance of dyslexia learning disability.

Keywords *Deep learning, Medical imaging, CNN, RNS, MRI, Dyslexia*

Introduction:

An important source of diagnostic information for clinicians and medical experts is the analysis and interpretations of medical images. Medical images have been taken through various medical imaging techniques, such as X-ray machine, mammography, ultrasound, magnetic resonance imaging (MRI), computed tomography (CT), and positron emission tomography (PET) [1]. These images have variously been used for early detection and treatment of diseases. MRI tools allow medical doctors and researchers to visualize and analyze alterations in the anatomy of the brain [2], and are available in three different types: functional MRI (fMRI), structural MRI (sMRI), and diffusion tensor imaging (DTI) [3][4]. These tools generate the best brain soft-tissue resolutions and have been used to capture and analyze different regions of the brain [3][5], with information provided by them used for diagnosis of various brain diseases and learning disabilities such as Alzheimer's, dementia, schizophrenia, Williams syndrome, Landau-Kleffner syndrome, autism, attention deficit hyperactivity disorder (ADHD), and dyslexia etc., [5].

Identifying and classifying learning difficulties has economical implications on a nation. Dyslexia as a learning disability, affect child's academic life and self-esteem into adulthood [6], hence its early diagnosis through brain imaging data is a crucial step towards the provision of appropriate technological interventions [7].

Development in machine learning has made classification of dyslexia easy with promising accuracy. The use of machine learning classifiers in form of neural networks and support vector machine (SVM) [8][9] for the classification of dyslexia conditions, particularly from the brain imaging dataset have variously been demonstrated by early

May- June 2020
ISSN: 0193-4120 Page No. 6323 - 6330

studies however, studies are rare on the application of deep learning models for this kind of scenario. Deep learning techniques are advanced artificial neural networks. Deep learning model such as convolutional neural network (CNN) has shown state-of-the-art performance in computer vision and image analysis, for example Image Net Large-Scale Visual Recognition Challenge (ILSVRC) [10]. The general advantages of deep learning over traditional machine learning is that deep models automatically learn hierarchical feature representations directly from data, thereby eliminating the feature engineering steps inherent in machine learning model [1][2].

With increasing availability of medical imaging dataset, cloud deployment of deep leaning use with these dataset becomes expedient has attracted great attention lately. In this situation, privacy of patients' information is extremely important, and needs urgent attention [11], particularly, when learning disability such as dyslexia is involved. To address the privacy issue in medical image, Al-Haj et al.[12] developed crypto-based algorithms capable of providing safe exchange of medical images along the transmission channel. These algorithms are based on cryptographic function and internally generated primary keys. In similar manner, chaotic map cryptographic algorithm has been proposed by Gatta and Al-Lateef [13] based on pixel confusion and diffusion processes.

Unlike binary and decimal number systems, residue number system (RNS) is a non-weighted number system based on modular arithmetic. Its carry-free computation and parallelism properties have been exploited in various digital signal processing (DSP) applications such as filtering, discrete transformations and cryptography [14]. It has widely been used for either singly[15] or in combination [16] with other methods to encryption of text-based and digital image dataset [17][18][19]. Using RNS concept along with deep learning on medical images will present a novel research in machine learning era as well as an important breakthrough in patients' information privacy preservation in medicine. In this paper, medical image classification and encryption scheme is proposed. RNS with special moduli set will be used to encrypt MRI sourced brain images before classifying them using CNN. The purpose of our secure classification is to seek the possibility of predicting the case of dyslexia from the information embedded. To prevent overfitting problem, our approach will augmentation, though, rarely applied in image encryption [20], improve the performance of the proposed deep model. Data augmentation increases the number of data points used for training deep model in order to avoid overfitting.

The entire paper has been organized into five sections. Section 2 and 3 provides an overview of convolutional neural network (CNN) and RNS-based encryption scheme for brain images respectively. The proposed framework is presented in section 4 while section 5 concludes the paper and provides future direction.

2. AN OVERVIEW OF CNN

Figure 1 shows the building block of the proposed CNN classifier for brain images. CNNs are special type of artificial neural networks which learn hierarchical representations from spatial information contained in digital images. It was originally design to process multi-dimensional (2D and 3D) arrays of high-resolution input dataset such as images and videos using very few connections between the layers [1][2]. Inspired by the visual cortex of a cat, its origin dates back to the *Neocognitron* proposed by Fukushima in 1980 [21] while the first architecture was given by LeCun et al.[22] in 1998 (LeNet-5). CNNs are able to form extremely well-organized representation of input images useful for image-oriented tasks e.g., classification. A CNN possesses several layers of convolutions and activations often intertwined by pooling (or subsampling) layers, and trained using backpropagation and gradient descent algorithms similar to that of a feed-forward neural network. In addition, CNN typically has fully connected layers at

May- June 2020
ISSN: 0193-4120 Page No. 6323 - 6330

the end, which compute the final output. The layers of CNN are briefly described below:

i. **Convolutional Layer:** A convolutional layer is a set of small parameterized *filters* that operates on the input domain. In this research, inputs are raw brain images and encrypted brain images. The purpose of convolutional layers is to learn abstract features from the data [23]. Each filter is a $n \times n$ matrix called a *stride*. In this case, our $n=3$. We convolve the pixels in the image and compute the dot product of the filter values and related values in the neighbour of the pixel, called *feature maps*. The stride is a pair of numbers, for example (3, 3), in which we slide a filter three units to the left or down in each step.

ii. **Activation Layer:** The feature maps from convolutional layers are inputted through a nonlinear activation functions to produce another stride called *feature maps* [2]. After each convolutional layer, we will use a nonlinear activation function. Every activation function performs a certain fixed mathematical operations on a single number it accepts as input. There are several activation functions from where one could choose from in practice. These include ReLU ($ReLU(z)=max(0, z)$), Sigmoid, Tanh functions and several other variants of ReLU such as leaky ReLU and parameter ReLU [2][23]. ReLU is n acronym for rectified linear unit.

iii. **Pooling Layer:** After an activation layer, a pooling layer (or sub-sampling layer) is next. Pooling layer takes small grid regions as input and performs operations on them to produce single number for each region. Different kind of pooling layers have been introduced [2][23], two most popular ones are *max-pooling* and *average pooling*. The pooling layers give CNN some translational invariance because a slight shift of the input image may results in slight change in activation maps.

iv. **Fully Connected Layer:** Fully connected layer has the similar structure as hidden layers of a classical feed-forward network. This layer is so-called because each neuron in this layer is linked to all neurons in the preceding layer, where each link represents a value called *weight*. The output of each neuron is the dot product of two vectors i.e., output of neurons in the preceding layers and the associated weight for each neuron.

v. **Dropout Layer:** This layer is also called *dropout regularization*. On many occasions, a model often gets biased to the training dataset, and produces high error when the testing dataset is introduced. In this situation, a problem of overfitting has occurred. To avoid overfitting during the training process, we use dropout layer. In this layer, we dropout a set of connections at random by setting them to zero in each iteration. This dropping of values prevents overfitting from happening and the final model will not be entirely fit to the training dataset.

For the architecture of our CNN, we will adopt similar architecture as [24] which gives excellent performance on the ImageNet Dataset. The architecture comprises five convolutional layers, each followed by ReLU, brightness normalization and overlapping pooling. Classification will be done using two additional fully connected layers and a dropout layer to avoid overfitting.

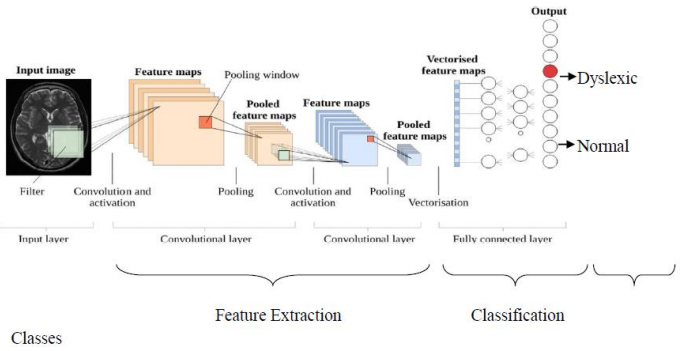


Figure 1: Convolutional neural network [2].

May- June 2020
ISSN: 0193-4120 Page No. 6323 - 6330

3. BACKGROUND OF RNS AND IMAGE ENCRYPTION

In modular arithmetic, a residue number system (RNS) represents a large integer using a set of smaller integers called *residues*, so that computation may be performed more efficiently [15]. Formally, RNS is defined in terms of *n*-tuple of pairwise relatively prime moduli.

$M = m_1 \times m_2 \times \ldots \times m_n = \prod_{i=1}^{n} m_i$If *S* denotes the moduli set, then *S* = $\{m_1, m_2, \ldots, m_n\}$ such that $GCD(m_i, m_j) = 1$ for $i \neq j$ and *GCD* means greatest common divisor. The dynamic range *M* of this system is defined as:

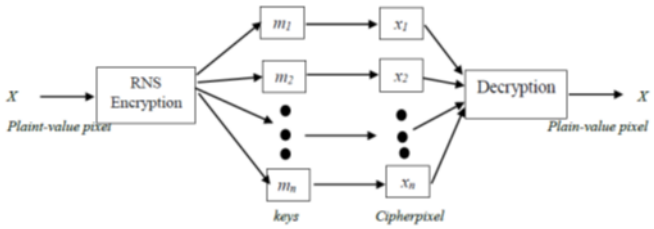
$x_i = |X|_{m_i} = X \bmod m_i X \rightarrow (x_1, x_2, \ldots, x_n)$Any integer $X \in Z_M$ can be represented in the RNS using equation below,

where x_i represents residue, *X* is a large integer, m_i is a module and *M* represent system dynamic range which must be sufficiently large enough for *i=1, 2, …, n*. Z_M ranges from *[0, M)* called the legitimate range of *X*.

Residue number system (RNS) possesses the capabilities to support parallel, carry-free addition, borrow-free subtraction and a single step multiplication without partial product. These characteristics make RNS useful in Digital Signal Processing (DSP) applications such as digital filtering, convolution, Fast Fourier transform, Discrete Fourier transform, image processing and cryptography etc [25]. Also, ordered significance among the residue digits is not important implying that removal some of the residue digits has no effect on the result except reduction in dynamic range [18].

$X \oplus Y \leftrightarrow (x_1 \oplus y_1, \ldots, x_i \oplus y_i, \ldots, x_n \oplus y_n), where\ operator \oplus conotes$)In image security, cryptography conceals information often, referred to as plain image and involves three important processes: keys generation, encryption and decryption. Image encryption algorithms distort the original arrangement of pixels in an image, scramble and make them appear disorganized. RNS has been used to improve the performance of other traditional cryptographic algorithms such as Rivest Shamir and Adleman (RSA)[26] and Data Encryption Standard (DES) [15]. RNS encryption scheme has been proven to be homomorphic with respect to *addition, subtraction and multiplication*. In other words, given two integers *X* and *Y*, it is possible to express the operation that maps their respective residues as [27]:

As illustrated in Figure 2, RNS creates multiple shares of a data in encrypted form and allows homomorphic encryptions to be performed on those shares.



(a): Multiple shares for integer X with respect to m_i

X	:	x_1	x_2	…	x_i	$x_{i+1} \cdots x_{n-1}$	x_n
Y	:	y_1	y_2	…	y_i	$y_{i+1} \cdots y_{n-1}$	y_n
$X \oplus Y$:		$x_1 \oplus y_1$	$x_2 \oplus y_2$	….	$x_i \oplus y_i$	… … …	$x_n \oplus y_n$

(b): RNS Encryption scheme with multiple share for integer X and Y

Figure 2: Homomorphic property of RNS

Like all other digital images, brain image is an array of pixel, sometimes called *voxel*. Each pixel corresponds to any numerical value in between 0 and 225, where 0 represents block colour and 225 represents white colour. In an 8-bit gray scale image,

May- June 2020
ISSN: 0193-4120 Page No. 6323 - 6330

the value of a pixel at any point corresponds to the intensity of the light photons striking at that point.

In this research, special moduli set of RNS will be used to encrypt brain images before subjecting them to deep learning using CNN architecture discussed in section 2. The aim of our secure classification is the possibility of predicting dyslexic cases form the brain image dataset. Our proposed methodology is detailed in the subsequent section. However, RNS cryptosystem requires the design of binary-to-residue (BR) converter circuit to handle the encryption part and design of residue-to-binary (RB) converter circuit to handle the decryption part. The latter can be implemented using variants of Chinese Remainder Theorem algorithm.

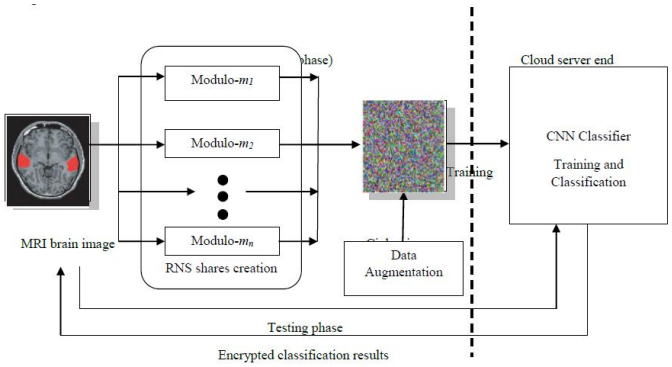


Figure 3: Proposed framework

4.1 RNS Encryption

For the RNS encryption phase, moduli set $\{2^n-1, 2^n, 2^{n+1}-1\}$ will be used for creation of multiple shares of each pixel's binary-value in the brain image, where $m_1=2^n-1$, $m_2=2^n$ and $m_3=2^{n+1}-1$ representing the channel-order of the moduli set with value of $n\geq 3$. In this scenario, the order of the moduli represents the public key (*pk*) while the value of *n* represents the secret key (*sk*) which must be kept as confidential as possible. Illustrative case of how the shares are created from 8-bit sequence of a pixel is given in Example 1. The shares binary sequences are combined to generate an encrypted pixel. The procedure is repeated for all pixels making up the image.

Example 1: Assuming a pixel from a grayscale brain image has a value of 156 (10011100 in 8-bit) and $n=3$. Using the proposed moduli set, $m_1=7(111)$, $m_2=8(1000)$ and $m_3=15(1111)$. Let plain pixel, X=10011100 and cipher pixel, X'=? The shares (x_i) are created in RNS using equation above as follows: $x_1=10$, $x_2=100$, and $x_3=110$. Combining the sequence of shares gives cipher pixel X'=10100110 in 8-bit (166 in decimal). Inadvertently, a pixel with value of 156 has been encrypted to another pixel with value 166 and the procedure is repeated for all other pixels making up image.

4.2 Image Segmentation

Before RNS encryption, there is the need for image segmentation. Segmentation is a process sketches out a region of interest (ROI) from a region of background (ROB) in a digital image [28]. Dividing the medical image into ROI and ROB is very important in medical image diagnosis [11]. In our own case, ROI will be segmented from the brain image dataset in order to identify the pixels that contain crucial information for the classification task.

4.3 Image Augmentation

To solve complex task using CNN model, a large quantity of image dataset is needed to train the classifier. The objective of data augmentation is to enlarge the number of data points used for training and enables deep models dealt with overfitting problem [20]. Many augmentation techniques exist e.g., horizontal/vertical flip, random crop random rotation, cut-out, and random erasing [20][29]. Despite the useful of data augmentation at increasing the accuracy of deep learning models, it is rarely considered in the existing image encryption methods. For the above reason, we consider the use of random erasing augmentation technique for training image dataset after RNS encryption. Random erasing technique randomly selects a rectangle-shaped ROI in an image and erases its pixels with random values. In this process, training images with various levels of blockages are produced, which reduces the risk of overfitting. This technique is parameter learning-free, easy to

May- June 2020
ISSN: 0193-4120 Page No. 6323 - 6330

implement, and can easily be incorporated with most of the deep models [29].

4.4 CNN Training and Classification

As stated in section 2, the architecture of the proposed CNN shall comprise of five convolutional layers, each followed by ReLU, brightness normalization and overlapping pooling. Training and classification will be done using two fully connected layers and a dropout layer to avoid overfitting. The network will be trained by using stochastic gradient descent (SGD) with momentum for 500 epochs. The learning rate will be initialized to 0.1 and later be decreased by a factor of 10 at 150, 225, 350 and 450 epochs respectively. We shall maintain a weight decay value of 0.0005, a momentum of 0.9, and a batch size of 120. Implementation of the model shall be done on *tf.keras* Library of Python programming on a GPU-based processor. *tf.keras* is TensorFlow's implementation of the which provides a high-level application programming interface (API) to create and train deep models support TensorFlow-specific functionality.

4.5 Brain Image Acquisition

A Philips 3.0-T Achieva scanner with a 32-channel coil will be used to obtain fMRI and DTI whole-brain images of 100 samples comprising both normal and dyslexic subjects. The acquisition procedure shall be based on the standard acquisition protocol of the Spinoza Centre for NeuroImaging in Amsterdam [8].

4.6. Proposed Simplified Algorithm

Algorithm below simplifies the implementation of the proposed framework for a secure classification of brain images:

Algorithm 1: Secure classification of MRI brain images into dyslexic and control

Let C denotes classification result, I_{pi} de denotes plain pixels from grayscale brain image of dimension 32×32 each, I_{ci} denotes encrypted pixels, I_c is concatenated I_{ci}, n is total number of images to be classified, m demotes moduli set and *i, j, k* are counters.

Input: I_{pi}

Output: C

1: **for** i=1 to n do
2: **for** j=1 to 32 do
3: **for** k=1 to 3 do
4: $I_{ci} = Enc(I_{pi})$, $I_{ci} \leftarrow I_{pi}$ *mod* m_k //shares of I_{pi} are created using the given moduli set
5: **end for**
6: $I_c \leftarrow I_{c1}I_{c2}\ldots I_{c32}$ //I_c=*concatenate*(I_{ci})
7: $I_c = Aug(I_c)$ //Augment I_c using random erasing algorithm [29]
8: $C \leftarrow classify(I_c)$ //classification is done using CNN model with the given architecture
9: **end for**
10: **end for**

4.7 Performance Evaluation Metrics

To evaluate the performance of CNN model, the following metrics will be used: classification accuracy, sensitivity, specificity and Area under ROC etc. For the evaluation of the encryption scheme, metrics such as histogram analysis, entropy analysis, correction between plain image and encrypted image shall be used.

5. CONCLUSION

In this paper, a framework for a secure brain image classification using convolutional neural network (CNN) and residue number system (RNS) is presented. Special moduli set provided in the paper will be used to scramble the well-organized 8-bit representation of each pixel making up the training image dataset in the encryption phase which follows the segmentation technique that has been applied on the training image in order to identify region of interest (ROI) to the classification task. CNN possesses the ability to represent hierarchical abstract features inherent in image dataset with high accuracy compared to the state-of-the-art. The proposed secure CNN is suitable to classify any form of images, in this case, medical images obtained with the aid of MRI tools. Increasing availability of medical images necessitate the private preservation of patients' records and remote implementation of deep learning models for their analysis, hence the need for this kind of research.

May- June 2020
ISSN: 0193-4120 Page No. 6323 - 6330

Meanwhile, dyslexia is chosen as a case study owing to its implications on the life of affected individual and the society at large. Future work will focus on the implementation of the proposed framework and achievement comparison with the state-of-the-art deep models.

REFERENCES

1. D. Shen, G. Wu, and H. Suk, "Deep Learning in Medical Image Analysis," *Annu. Rev. Biomed. Eng.*, vol. 19, no. 1, pp. 221–248, 2017.
2. A. S. Lundervold and A. Lundervold, "An overview of deep learning in medical imaging focusing on MRI," *Z. Med. Phys.*, vol. 29, no. 2, pp. 102–127, 2019.
3. A. Elnakib, A. Soliman, M. Nitzken, M. F. Casanova, G. Gimel'farb, and A. El-Baz, "Magnetic resonance imaging findings for dyslexia: A review," *Journal of Biomedical Nanotechnology*, vol. 10, no. 10. pp. 2778–2805, 2014.
4. Y. F. Sun, J. S. Lee, and R. Kirby, "Brain imaging findings in dyslexia," *Pediatr. Neonatol.*, vol. 51, no. 2, pp. 89–96, 2010.
5. M. F. Casanova *et al.*, "Corpus callosum shape analysis with application to dyslexia," *Transl. Neurosci.*, vol. 1, no. 2, pp. 124–130, 2010.
6. S. O. Wajuihian and K. S. Naidoo, "Dyslexia: An overview," *African Vis. Eye Heal.*, vol. 70, no. 2, pp. 89–98, 2011.
7. N. A. M. Yuzaidey, N. C. Din, M. Ahmad, N. Ibrahim, R. A. Razak, and D. Harun, "Interventions for children with dyslexia: A review on current intervention methods," *Med. J. Malaysia*, vol. 73, no. 5, pp. 311–320, Oct. 2018.
8. P. Tamboer, H. C. M. Vorst, S. Ghebreab, and H. S. Scholte, "Machine learning and dyslexia: Classification of individual structural neuro-imaging scans of students with and without dyslexia," *NeuroImage Clin.*, vol. 11, pp. 508–514, 2016.
9. P. Płoński *et al.*, "Multi-parameter machine learning approach to the neuroanatomical basis of developmental dyslexia," *Hum. Brain Mapp.*, vol. 38, no. 2, pp. 900–908, 2017.
10. O. Russakovsky *et al.*, "ImageNet Large Scale Visual Recognition Challenge," *Int. J. Comput. Vis.*, vol. 115, no. 3, pp. 211–252, 2015.
11. A. Mahmood, T. Hamed, C. Obimbo, and R. Dony, "Improving the Security of the Medical Images," *Int. J. Adv. Comput. Sci. Appl.*, vol. 4, no. 9, pp. 137–146, 2013.
12. A. Al-Haj, G. Abandah, and N. Hussein, "Crypto-based algorithms for secured medical image transmission," *IET Inf. Secur.*, vol. 9, no. 6, pp. 365–373, 2015.
13. M. T. Gatta and S. T. A. Al-Latief, "Medical image security using modified chaos-based cryptography approach," *J. Phys. Conf. Ser.*, vol. 1003, no. 1, 2018.
14. E. K. Bankas and K. A. Gbolagade, "A New Efficient RNS Reverse Converter for the 4-Moduli Set," vol. 8, no. 2, pp. 328–332, 2014.
15. A. H. Navin, "A Novel Approach Cryptography by using Residue Number System," no. November 2015, 2011.
16. S. Abdul-mumin, "Mixed Radix Conversion based RSA Encryption System," vol. 150, no. 1, pp. 43–47, 2016.
17. S. Alhassan and K. A. Gbolagade, "Enhancement of the Security of a Digital Image using the Moduli Set," vol. 2, no. 7, pp. 2223–2229, 2013.
18. M. I. Youssef, "Multi-Layer Data Encryption using Residue Number System in DNA Sequence," vol. 45, no. 10, pp. 19–24, 2012.
19. M. I. Youssef, A. E. Emam, S. M. Saafan, and M. A. B. D. Elghany, "Secured Image Encryption Scheme Using both Residue Number System and DNA Sequence," no. 6.
20. W. Sirichotedumrong, T. Maekawa, Y. Kinoshita, and H. Kiya, "Privacy-Preserving Deep Neural Networks with Pixel-based Image Encryption Considering Data Augmentation in the Encrypted Domain," 2019.
21. K. Fukushima, "Neocognition: a self," *Biol. Cybern.*, vol. 202, pp. 193–202, 1980.
22. Y. Lecun, Y. Bengio, and G. Hinton, "Deep learning," *Nature*, vol. 521, no. 7553, pp. 436–444, 2015.
23. E. Hesamifard, H. Takabi, and M. Ghasemi, "CryptoDL: Deep Neural Networks over Encrypted Data," pp. 1–21, 2017.
24. M. D. Zeiler and R. Fergus, "Visualizing and understanding convolutional networks," *Lect.*

May- June 2020
ISSN: 0193-4120 Page No. 6323 - 6330

Notes Comput. Sci. (including Subser. Lect. Notes Artif. Intell. Lect. Notes Bioinformatics), vol. 8689 LNCS, no. PART 1, pp. 818–833, 2014.

25. K. A. Gbolagade, S. D. Cotofana, and S. M. Ieee, "An O (n) Residue Number System to Mixed Radix Conversion Technique," no. May 2009, 2014.
26. S. Abdul-mumin and K. A. Gbolagade, "International Journal of Emerging Technologies in Computational and Applied Sciences (IJETCAS) An Improved Residue Number System Based RSA Cryptosystem," pp. 70–74, 2017.
27. M. Gomathisankaran, K. Namuduri, and A. Tyagi, "HORNS : A Semi-perfectly Secret Homomorphic Encryption System," vol. 2, no. 1, pp. 17–23, 2013.
28. K. Kim, H. Lee, and H. Lee, "Spatial Error Concealment Technique for Losslessly Compressed Images Using Data Hiding in Error-Prone Channels," vol. 12, no. 2, pp. 168–173, 2010.
29. Z. Zhong, L. Zheng, G. Kang, S. Li, and Y. Yang, "Random Erasing Data Augmentation."