Convolutional Neural Network Model Layers Improvement For Segmentation And Classification On Kidney Stone Images Using Keras And Tensorflow

Waliu Olalekan Apena
(1) Department of Computer / Electrical and Electronics Engineering, The Federal University of Technology, Akure. Nigeria.
(2) Biomedical Computing and Engineering Technology, Applied Research Group, Coventry University, Coventry, United Kingdom
woapena@futa.edu.ng, apenaw@coventry.ac.uk

Orobosa Libert Joseph
Department of Computer / Electrical and Electronics Engineering, The Federal University of Technology, Akure. Nigeria.
josephorobosa@yahoo.com

***Abstract*—Convolutional neural network (CNN) models are beneficial to image classification algorithms training for highly abstract features and work with less parameter. Over-fitting, exploding gradient, and class imbalance are CNN major challenges during training; with appropriate management training, these issues can be diminished and enhance model performance. The models are 128 by 128 CNN-ML and 256 by 256 CNN-ML training (or learning) and classification. The results were compared for each model classifier. The study of 128 by 128 CNN-ML model has the following evaluation results consideration of error: (i) without validation, the accuracy was 85.1%; (ii) with validation, the accuracy was 86.8%; (iii) Absolute Error (AE) of 0.0696 and (iv) Relative Error (RE) of ± 0.0897; while, the 256 by 256 CNN-ML model has the following evaluation results: (i) without validation, the accuracy was 86.3%; (ii) with validation, the accuracy was 85.6%; (iii) Absolute Error (AE) of 0.0257 and (iv) Relative Error (RE) of ± 0.0320. These values show that the higher the nodes in the layer, the better the performance. Node scaling could be deplored in biomedical decision support system and patients' management.**

Keywords— *CNN nodes; layers; Keras; TensorFlow and Performance*

I. INTRODUCTION

The application of artificial intelligence (AI) to real life situation is limitless, as it were in decision support system for positive healthcare outcomes. Convolutional neural network (CNN) is deployed for image analysis with limited initial parameters and its gaining popularity in the field of biomedical engineering research and healthcare patient's management. The process of extracting useful knowledge from huge data is known as Data Mining. Data Mining is an exploration and analysis, by automatic or semiautomatic means, of large quantities of data in order to discover meaningful patterns [1,2]. The domains of data mining include: image mining, opinion mining, web mining, text mining, and graph mining and so on. Some of its applications include anomaly detection, financial data analysis, medical data analysis, social network analysis, market analysis [3]. Recent progress in deep learning using CNN machine learning (CNN-ML) has been helpful in decision support and contributed to positive outcome, significantly. The application of CNN-ML to diverse areas of soft computing is adapted diagnosis procedure to enhance time and accuracy [4]. The study investigated CNN-ML layer nodes improvement with adapted Keras and Tensor Flow for kidney stone image classification.

II. LITERATURE REVIEW

A. Deep learning

Deep learning could be adapted into computational models that are composed of multiple processing layers to learn representations of data with multiple levels of abstraction [5]. This could be deployed in speech recognition, visual object detection and recognition with respect to initial input and internal parameter (weight). Deep learning was firstly introduced by [6] for a class of deep probabilistic generative models called Deep Belief Networks (DBNs) [7].

B. Neural Network

In 1943, Warren McCulloch and Walter Pitts developed the first mathematical model of a neuron. In their research paper "A logical calculus of the ideas immanent in nervous activity", they described the simple mathematical model for a neuron, which represents a single cell of the neural system that takes inputs, processes those inputs, and returns an output. This model is known as the McCulloch-Pitts neural model [8].

C. Convolutional Neural Network

The name “convolutional neural network” indicates that the network employs a mathematical operation called convolution. Convolution is a specialized kind of linear operation. Convolutional networks are simply neural networks that use convolution in place of general matrix multiplication in at least one of their layers [9]. Convolutional networks were inspired by biological processes [10].

D. Convolutional layer

The main building block of a CNN model are the convolutional layers; when programming a CNN, the input is a tensor [11] with shape (number of images) x (image height) x (image width) x (image depth). Then after passing through a convolutional layer, the image becomes abstracted to a feature map, with shape (number of images) x (feature map height) x (feature map width) x (feature map channels). Convolutional layers convolve the input and pass its result to the next layer [12, 13]. Figure (1) shows a typical CNN architecture.

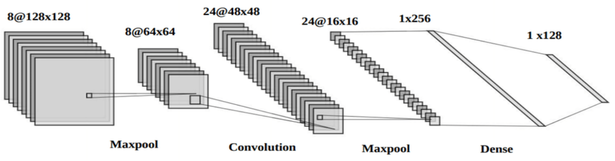

Figure 1: A configuration of typical CNN layer [14]

E. TensorFlow Backend Platform

TensorFlow is an open-source artificial intelligence library, using data flow graphs to build models. It allows developers to create large-scale neural networks with many layers. TensorFlow is mainly used for: Classification, Perception, Understanding, Discovering, Prediction and Creation [15, 16].

F. Keras Library

In 2017, Google's TensorFlow team decided to support Keras in TensorFlow's core library [17]. Keras is an open-source neural-network library written in Python language. It is capable of running on top of TensorFlow, Microsoft Cognitive Toolkit, R, Theano, or PlaidML. Designed to enable fast experimentation with deep neural networks, it focuses on being user-friendly, modular, and extensible [17, 18].

G. Related Works

In their work [19], adopted neural network for kidney stone detection diagnosis, the study was based on empirical data to develop an algorithm using two neural network algorithms viz Radial basis function and Learning vector quantization. The work used many complex dependant variables with few specimens of 5 instances and each having 7 attributes such as age, sex, Lymphocytes Monocytes, Neutrophil, S.Creatinine, and Eosinophis; and an unusually low amount of data.

The paper [7], worked on deep learning-based classification of focal liver lesions with contrast-enhanced ultrasound, it focused on important classification of liver masses to early diagnosis of patients. The study proposed a diagnostic system of liver disease classification based on contrast enhanced ultrasound (CEUS) imaging. In the proposed system, the dynamic CEUS videos of hepatic perfusion are firstly retrieved. Secondly, time intensity curve (TICs) is extracted from the dynamic CEUS videos using sparse non-negative matrix factorizations. Finally, deep learning was employed to classify benign and malignant focal liver lesions based on these TICs, which makes the work a near too robust to adopt.

H. CNN Model Classifier Evaluation

To evaluate the performance of the Neural Network classifiers for kidney stone and non-kidney stone classes, the Accuracy (AC), Sensitivity (SE), Precision (P), Specificity (SP) and Effectiveness (E) parameters have been made use of [20, 21, 22].

III. Method

The study implemented and trained segmentation and classification algorithm; performed on an AMD Quad-Core 1.7GHz processor on a 64-bit windows 7 operating system with 6.0 GB RAM using Python and Jupyter Notebook. The trained data acquired was of size 120 by 100 pixels. The image slices were resized to 50 by 50 pixel size before extracting features from the images. All the images supplied were passed through OpenCV [23] preprocessing steps to enhance its contrast as pure gray scale. The two different Convolutional Neural Network Machine Learning (CNN-ML) models were: 128 by 128 CNN-ML and 256 by 256 CNN-ML models respectively. The study compared the training (or learning) and classification result for each classifier.

A. Neural Network Model of the Training Phase

The study deplored artificial neural network model for the image operation as expressed below. The image input are in vectors matrix as revealed in equation (1) and (2), given set of input vectors $\overrightarrow{x_k}$, (which represents the image matrix) with its corresponding transpose as expressed in equations (1) and (2).

$$\overrightarrow{x_k} = (x_{k1}x_{k2}x_{k3} \dots x_{km}) \tag{1}$$

$$\overrightarrow{x_k} = [x_{k1}x_{k2}x_{k3} \dots x_{km}]^T \tag{2}$$

Where *k = 1, 2, 3, …, m; and m = number of interconnected neurons at the input,*

The corresponding set of output vectors, $\overrightarrow{y_k}$, with its transpose are expressed in (3) and (4).

$$\overrightarrow{y_k} = (y_{k1}y_{k2}y_{k3} \dots y_{km}) \tag{3}$$

$$\overrightarrow{y_k} = [y_{k1}y_{k2}y_{k3} \dots y_{km}]^T \tag{4}$$

And the associated weight vector, $\overrightarrow{w_{jk}}$, to the input vector $\overrightarrow{x_k}$, is expressed in (5)

$$\overrightarrow{w_{jm}} = [w_{j1}(k) \quad w_{j2}(k) \quad \dots \quad w_{jm}(k)]^T \quad (5)$$

The output $\overrightarrow{y_k}$, can thus be expressed as:

$$\overrightarrow{y_{kj}} = \sum_{i=1}^{m} w_{ji}(k) x_{ki} \qquad (6)$$

In an expanded form, (6) can be written as:

$$\overrightarrow{y_{kj}} = [w_{j1}(k) \quad w_{j2}(k) \quad \dots \quad w_{jm}(k)] \begin{bmatrix} x_{k1} \\ x_{k1} \\ \vdots \\ x_{km} \end{bmatrix} \qquad (7)$$

Where *j = 1, 2, 3, …, m;*

The matrix form of (7) gives the association between the input vector $\overrightarrow{x_k}$ and the output vector $\overrightarrow{y_k}$

$$\begin{bmatrix} y_{k1} \\ \vdots \\ y_{km} \end{bmatrix} = \begin{bmatrix} w_{11}(k) & \cdots & w_{1m}(k) \\ \vdots & \ddots & \vdots \\ w_{m1}(k) & \cdots & w_{mm}(k) \end{bmatrix} \begin{bmatrix} x_{k1} \\ \vdots \\ x_{km} \end{bmatrix} \qquad (8)$$

Or, in a more compact form, (8) can be expressed as:

$$\overrightarrow{y_k} = \overrightarrow{w(k)}\overrightarrow{x_k} \qquad (9)$$

The neural network model is illustrated in fig. 2., show graphical representation of (9)

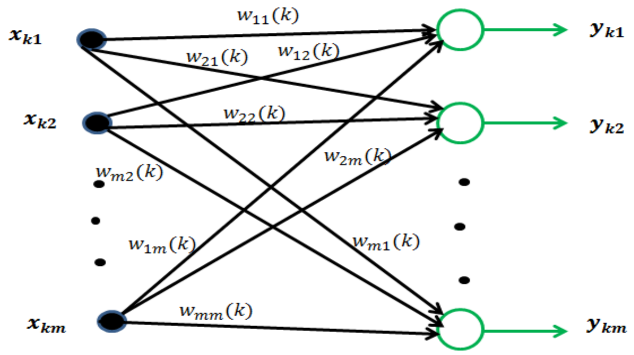

Figure 2: Neural Network model of the training phase.

B. Condition for Perfect Recall (Testing Phase)

Upon training of the network, the condition for perfect recall can be expressed as in (9) (Jain, *et. al*. 2018) as follows:

$$\vec{y} = \overrightarrow{y_j} + \sum_{\substack{k=1 \\ k \neq j}}^{m} (\overrightarrow{x_k^T} . \overrightarrow{x_j}) \overrightarrow{y_k} \qquad (10)$$

Equation (9) may be expressed in a more compact form as revealed in (10).

$$\vec{y} = \overrightarrow{y_j} + \overrightarrow{v_j} \qquad (11)$$

Where $\overrightarrow{v_j} = \sum_{\substack{k=1 \\ k \neq j}}^{m} (\overrightarrow{x_k^T} . \overrightarrow{x_j}) \overrightarrow{y_k}$.

For $\overrightarrow{v_j}$= 0, (or approximately),

$$\vec{y} = \overrightarrow{y_j} \qquad (12)$$

Equation (12) is the condition for perfect recall. $\overrightarrow{y_j}$ is the desired or expected output of the system and $\overrightarrow{v_j}$ is the noise associated with recalling a pattern from memory.

C. Machine Learning Model with Keras and TensorFlow

Two different Convolutional Neural Network Machine Learning (CNN-ML) models were adopted in this work, namely are 128x128 CNN-ML and 256x256 CNN-ML models.

D. Nodes and Layer Training configuration for the 128x128 CNN-ML Model

This model is made up of four layers comprising of an input layer, two (2) hidden layers of 128 units or nodes per layer and a final output layer. Figure 3 shows the actual layer configuration of the 128x128 CNN-ML model while figure 4 shows the training block with Keras and TensorFlow library tools.

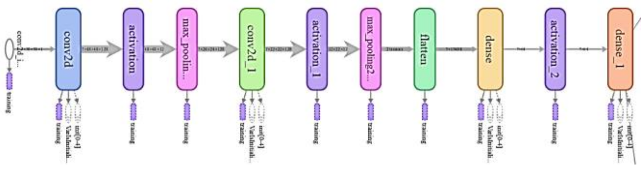

Figure 3: TensorFlow Layer configuration for the 128x128 CNN-ML model

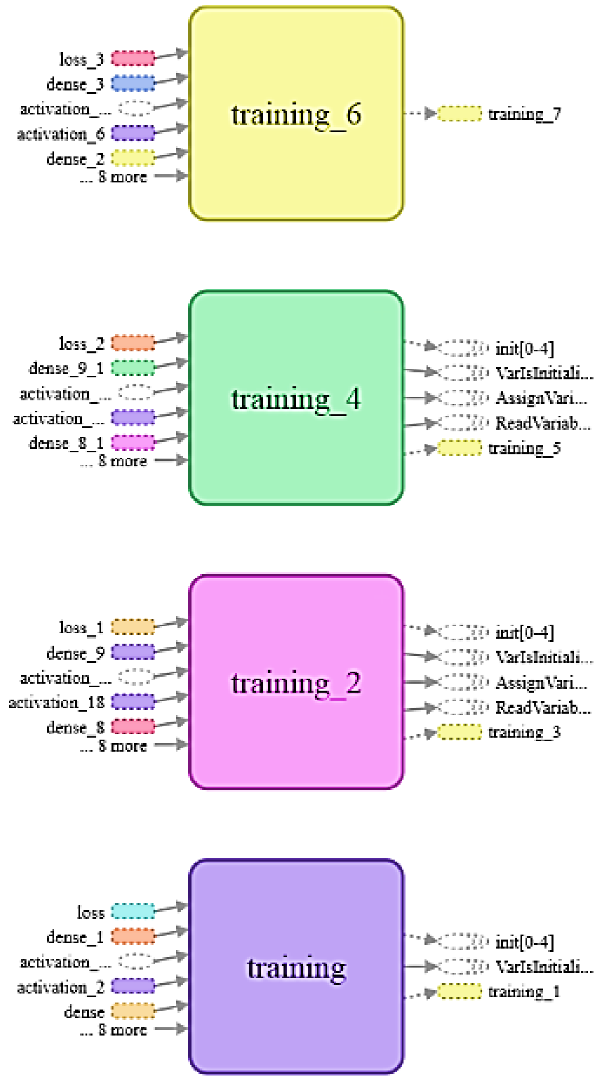

Figure 4: TensorFlow Training Block for the 128x128 CNN-ML model

E. Training configuration for Nodes and Layer for the 256x256 CNN-ML Model

This model is a little modification of the 128x128 CNN-ML model. Also, this model is made up of four layers comprising of an input layer, two (2) hidden layers of 256 units or nodes per layer and a final output layer. Figure 5 shows the actual layer configuration of the 256x256 CNN-ML model while figure 6 shows the training block.

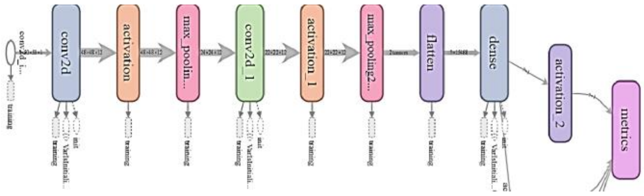

Figure 5: TensorFlow Layer configuration for the 256x256 CNN-ML model

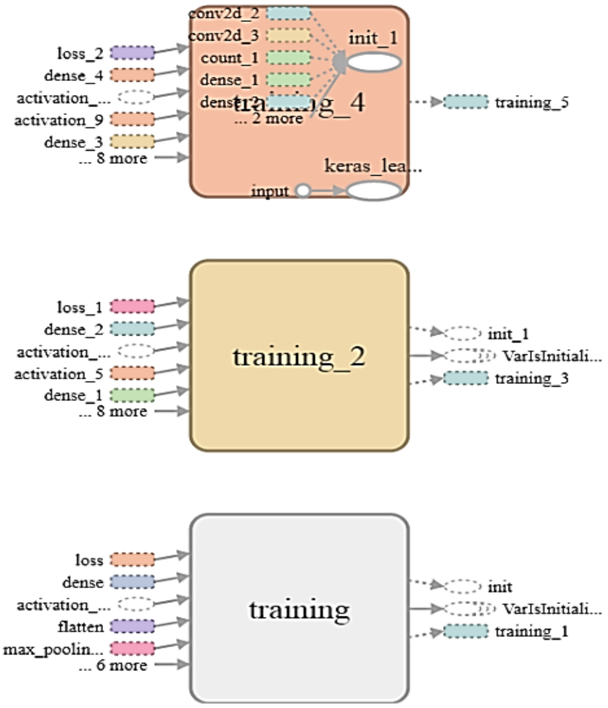

Figure 6: TensorFlow Training Block for the 256x256 CNN-ML model

IV. RESULTS AND DISCUSSIONS

The study evaluates the performance of the models through error analysis as follows:

A. Performance Error Analysis

The study investigated database measurement taken from the models (128x128 and 256x256 CNN-ML) as revealed in Tables 1 and Table 2 respectively. Error analysis was performed in Tables 1 and Table 2 including inferences as shown below using equation (13) and (14).

1) Absolute Error (AE)

This is the absolute difference between the model value and the manual value

$$AE = |Model\ with\ Validation - Model\ without\ Validation| \qquad (13)$$

The Mean Absolute Error (MAE) of the *128x128 CNN-ML Model* database of Table 1 is *0.0695* while the Mean Absolute Error (MAE) of *256x256 CNN-ML Model* database of Table 2 is 0.02571

2) Relative Error (RE)

This is obtained using equation (14)

$$RE = \frac{AE}{ModelValue} \qquad (14)$$

B. Error Analysis: 128x128 CNN-ML Model

The values for the error performance analysis for this model are shown in Table 1 and expressed using equations (13) and (14) as follows:

Table 1: Performance Error Analysis for the 128x128 CNN-ML Model

Sample Batch Image Number	Model without Validation Accuracy	Model With Validation Accuracy	Absolute Error (AE)	Relative Error (RE)
1	0.5149	0.5644	0.0495	± 0.087704
2	0.5489	0.8713	0.3224	± 0.370022
3	0.8128	0.6337	0.1791	± 0.282626
4	0.9191	0.9208	0.0017	± 0.001846
5	0.9277	0.9604	0.0327	± 0.034048
6	0.9447	0.9604	0.0157	± 0.016347
7	0.9617	0.9703	0.0086	± 0.008863
8	0.9447	0.9505	0.0058	± 0.006102
9	0.9532	0.9505	0.0027	± 0.002841
10	0.9787	0.9010	0.0777	± 0.086238
MEAN	**0.85064**	**0.86833**	**0.06959**	**± 0.089664**

The Table 1 revealed RE values as shown above, this indicates how reliable the model values can be regarded with respect to the Model without Validation. From Table 1, the Mean Relative Error (MRE) is **± 0.0897**. This MRE value is only significant from the third decimal point value and this indicates that the developed system is reliable and could be adopted in decision support. The output could be deplored in patient's management procedure.

C. Error Analysis of the 256x256 CNN-ML Model

Also, the values for the error performance analysis for the (256x256) model are shown in Table 2 and expressed using equations (13) and (14) above. Table 2, considered two types of errors as shown below.

Table 2: Performance Error Analysis for the 256x256 CNN-ML Model

Sample Batch Image Number	Model without Validation Acc.	Model With Validation Accuracy	Absolute Error (AE)	Relative Error (RE)
1	0.5021	0.5149	0.0128	±0.024859
2	0.5957	0.6139	0.0182	±0.029647
3	0.8723	0.7624	0.1099	± 0.14415
4	0.9064	0.9109	0.0045	± 0.00494
5	0.9362	0.9604	0.0242	± 0.025198
6	0.9362	0.9703	0.0341	± 0.035144
7	0.9702	0.9604	0.0098	± 0.010204
8	0.9660	0.9604	0.0056	± 0.005831
9	0.9787	0.9604	0.0183	± 0.019055
10	0.9702	0.9505	0.0197	± 0.020726
MEAN	**0.8634**	**0.85645**	**0.02571**	± **0.031975**

The RE values indicate reliability of the model. From Table 2, the Mean Relative Error (MRE) is ± **0.031975**. This MRE value is only significant from the third decimal point value and this indicates that the developed system is reliable, it could be adapted to decision support in the diagnosis procedure.

D. Comparison of the Developed Models

This subsection accounts for the performance of the two models by way of comparison of the output mean values. The comparison is summarized in Table 3 as shown below.

S/N	**Parameter (Mean values)**	**Model**	
		128x128 CNN-ML Model	**256x256 CNN-ML Model**
1	Without Validation Accuracy	**0.85064**	**0.8634**
2	With Validation Accuracy	**0.86833**	**0.85645**
3	**Absolute Error (AE)**	**0.06959**	**0.02571**
4	**Relative Error (RE)**	± **0.089664**	± **0.031975**

The result of the comparison from Table 3, shows that the 256x256 CNN-ML Model has better overall output performance. This revealed that the model could be adopted for biomedical image analysis and decision system.

V. CONCLUSION

This research has adopted two CNN Machine Learning models to evaluate the impact of CNN layer nodes improvement (using Keras with TensorFlow in a Python programming language environment) on segmentation and classification performance. The result shows that with increased CNN layer nodes, performance is also increased in the positive direction. Hence, with more increased nodes in the layer, performance is expected to improve for a CNN model. This study outcome could be adopted in biomedical decision system for positive healthcare outcome and patients' management diagnosis procedure.

REFERENCES

[1] Tan P; Steinbach M. and Kumar V.: *Introduction to Data Mining*, 130-139, (2004).

[2] Perdana, H.: *Integrated Retinal Information System for Analyzing Kidney Condition* (Doctoral dissertation, Universiti Utara Malaysia). (2009).

[3] Sadaf, F., Shafiya, S., Chandini, A. H., Nagana Devi, G. J. and Akshatha, M.: Chronic Kidney Disease and Stage Detection Using Machine Learning Classifiers. *Department of CSE, VidyaVikas Institute of Engineering and Technology, Mysuru, Karnataka.* DOI: https://doi.org/10.21467/proceedings.1.60 (01/04/2020), (2018).

[4] Brattain, L. J., Telfer, B. A., Dhyani, M., Grajo, J. R., and Samir, A. E.: Machine learning for medical ultrasound: status, methods, and future opportunities. *Abdominal Radiology*, *43*(4), 786-799 (2018).

[5] LeCun, Y., Bengio, Y., and Hinton, G.: Deep learning. *Nature*, *521*(7553), 436-444 (2015).

[6] Hinton, G. E., and Salakhutdinov, R. R: Reducing the dimensionality of data with neural networks. *science*, *313*(5786), 504-507 (2006).

[7] Wu K., Chen X., and Ding M: Deep learning based classification of focal liver lesions with contrast-enhanced ultrasound. *Optik*, *125*(15), 4057-4063 (2014).

[8] Warren S. Mcculloch and Walter Pitts Bulletin of Mothemnticnl Biology Vol. 52, No. l/2. pp. 99-115. 1990.

[9] Ian G. and Yoshua B. and Aaron C.: *Deep Learning*. MIT Press. p. 326. (2016).

[10] Fukushia K.:"Neocognitron". Scholarpedia. 2 (1):1717. Bibcode:2007SchpJ...2.1717F. doi: 10.4249/scholarpedia.1717 (2007).

[11] Fridtjov I: "Continuum Mechanics". Springer. ISBN 3-540-74297-2 (2008)

[12] LISA Lab "Convolutional Neural Networks (LeNet) – DeepLearning 0.1 documentation". *DeepLearning 0.1*. LISA Lab. Retrieved 31 August 2013.

[13] Habibi, Aghdam, Hamed: *Guide to convolutional neural networks : a practical application to traffic-sign detection and classification*. Heravi, Elnaz Jahani. Cham, Switzerland. ISBN 9783319575490. OCLC 987790957, 2017

[14] Pandey S; Shekhawat H. and Prasanna, S.: Deep Learning Techniques for Speech Emotion Recognition : A Review. (2019).

[15] Bhutada, S; Mukundha, C; Shreya, G. and Lahari, C.: Convolutional Neural Networks for Automatic Classification of Diabetic Retinopathy. *Networks*, *5*(04). (2018).

[16] Fridtjov I.: "Continuum Mechanics". Springer. ISBN 3-540-74297-2 (2008)

[17] Chollet, François: "Xception: Deep Learning with Depthwise Separable Convolutions". arXiv:1610.02357 (2016).

[18] Gulli A; and Pal S.: *Deep learning with Keras*. Packt Publishing Ltd (2017).

[19] Sneha A. and Chougule S. R.: Neural Network for Kidney Stone Detection. *International Journal of Science and Research (IJSR)*, Volume 5 Issue 4, April 2016

[20] Udekwe, C. N., and Ponnle, A. A.: Application of Cardinal Points Symmetry Landmarks Distribution Model to B-Mode Ultrasound Images of Transverse Cross-section of Thin-walled Phantom Carotid Arteries. *European Journal of Engineering Research and Science*, *4*(12), 96-101. (2019).

[21] Visa, S.; Ramsay, B.; Ralescu, A. L. and Van Der Knaap, E.: Confusion Matrix-based Feature Selection. *MAICS*, *710*, 120-127 (2011).

[22] Shrestha, U.: *Automatic Liver and Tumor Segmentation from CT Scan Images using Gabor Feature and Machine Learning Algorithms* (Doctoral dissertation, University of Toledo). (2018).