CONTINUAL LEARNING ON FACIAL RECOGNITION USING CONVOLUTIONAL NEURAL NETWORKS

Jingjing FENG[1*#], Valentina GOMEZ[2#]

The field of Neural Networks has been rapidly advancing in recent years, however, the issue of continual learning remains a problem that has not yet been fully resolved. Continual learning involves the ability to learn a second task without losing the information obtained from the first task. A recent study proposed a method called Elastic Weight Consolidation (EWC) which addresses this issue by slowing down the learning procedure on certain weights that are important for those tasks, thereby extending the memory lifetime. In contrast, the present report introduces a purely computational method to achieve continual learning. Specifically, the approach involves performing two consecutive tasks: the first task is to distinguish the identity of a person (task A), and the second task is to distinguish the gender of a person (task B). The method reclassifies images for different tasks using different labels in a supervised learning setting. The innovation in this report lies in the use of fully connected layers to classify features with two different labels. The key to the success of this method lies in training the entire Neural Network when performing task A, and then training only the fully connected layers for task B to classify the features using different labels. Therefore, regardless of the number of tasks, this network only needs to be trained once. This approach can be considered a significant contribution to the field of continual learning as it presents a practical solution that does not require the use of sophisticated algorithms or complicated techniques.

Keywords: Convolutional Neural Network, face recognition, continual learning, gender recognition, fully connected layers

1. Introduction

In the field of artificial intelligence, continual learning [1], which involves learning tasks in succession without forgetting previously trained tasks, is a longstanding challenge in the field of artificial intelligence. Artificial neural networks tend to forget old concepts when learning new ones, and thus, continual learning aims to learn from multiple streams of data and store the knowledge for future use.

[1] Assistant Professor, Shanghai Institute for Science of Science, Fudan University, Shanghai China, e-mail: fjj@siss.sh.cn

[2] Department of Computer Science, The University of Texas at Austin, Austin, Texas, USA.

[#]The two authors share the joint first authorship.

In recent years, alternative solutions have emerged to address this problem, such as the Elastic Weight Consolidation (EWC) method [2, 3], which extends the memory lifetime and allows multiple tasks to be performed successively by slowing down learning on certain weights that are important for those tasks. In EWC method, when learning a task, the multi-layer perceptron adjusts the set of parameters (weights and bias) to optimize performance. When learning task B, EWC maintains the performance in task A by fixing the parameters to stay close to the value in task A. However, such method only extends the memory lifetime instead of preventing loss of memory.

This report proposes an alternative method for achieving continual learning using a purely computational approach based on Convolutional Neural Networks (CNNs) [4, 5, 6] built upon TensorFlow. The CNNs are used to perform two consecutive tasks: identity recognition (task A) and gender recognition (task B) using different labels for each task. The Olivetti Face Database [7] is used as the data source, consisting of forty people's faces, with ten pictures for each person with different facial expressions. Task A is labeled by each person's name, while task B is labeled by gender, resulting in 40 possible outputs and 2 possible outputs, respectively.

The key to achieving continual learning in this method is in how fully connected layers [8, 9, 10, 11] are used to classify features with two different labels. During the first task, the entire Neural Network is trained, while only the fully connected layers are trained during the second task using different labels. Specifically, the network first undergoes convolution and pooling, which is a common operation for both tasks, and this operation needs to be done only once, regardless of the number of tasks. The next step is to train the fully connected layers to perform task A by labeling each person's face by their name. When training task B, only the fully connected layers are retrained using a different set of labeling to classify the images by gender.

This report is organized in the following way. Section 2 presents the implementation of continual learning using CNNs and demonstrates how to tune the parameters in the python code to improve accuracy for both tasks. Besides, the advantages and disadvantages of different approaches are discussed. In Section 3, the report concludes by discussing the future improvements that can be made in this area.

2. Implementation

For both task A (identity recognition) and task B (gender recognition), we utilized the Olivetti database [7] (formerly known as 'The ORL Database of Faces' from ATT and Cambridge University Computer Laboratory [15]). This database contains grayscale images of 40 individuals' faces, including 36 males and 4 females, with 10 pictures per person featuring different facial expressions and angles, for a total of 400 images. We selected the Olivetti database as it was easier to generate gender labels.

The data was randomly divided in two parts for the process of training and testing, respectively. For each person's face, there are in total 300 pictures being used for training (7-8 pictures per person) and 100 pictures for testing and cross validation (the rest 2-3 pictures per person). Please refer to Fig. 1 for the complete database.

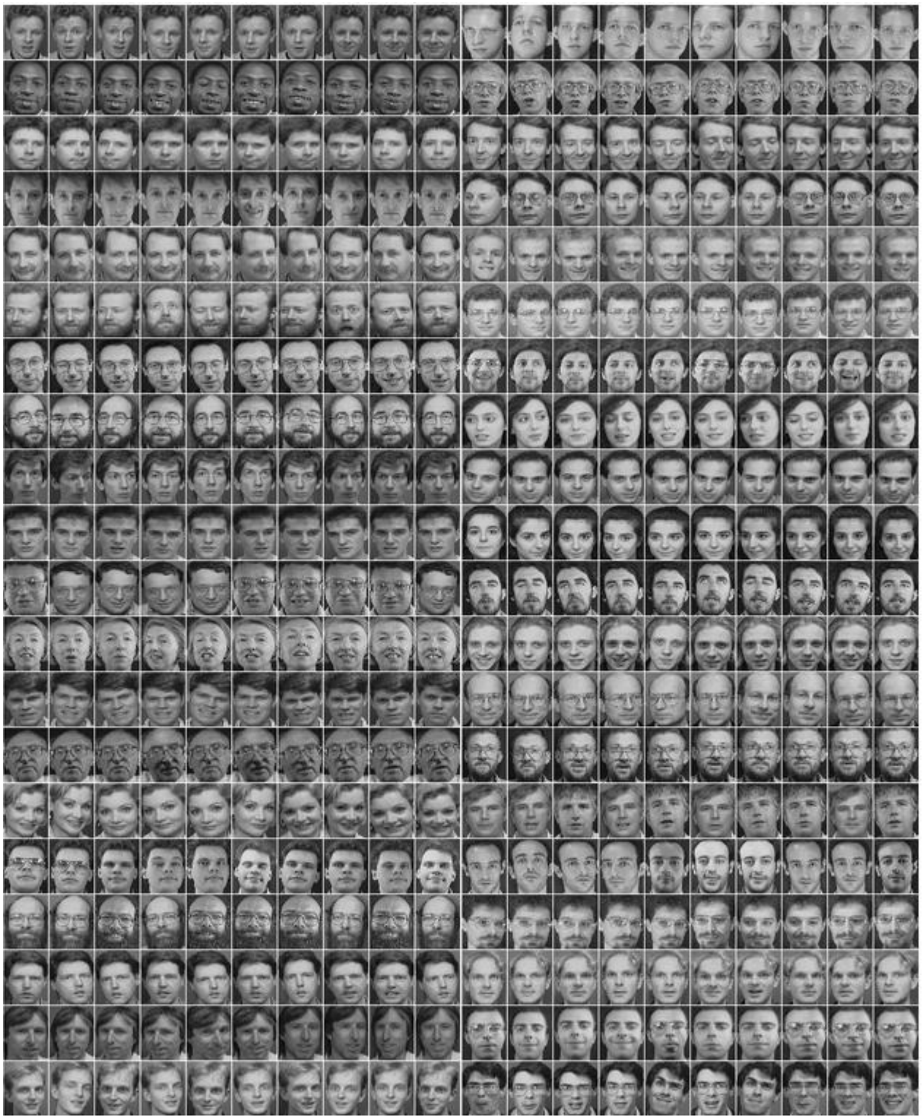

Fig. 1. Olivetti face database [7].

To create the CNN, we employed Keras and designed four layers, starting with a convolutional layer that links to a pooling layer, followed by another convolutional layer and another pooling layer. After several rounds of tuning, we established the best layer design, based on previous work [11] that utilized the Olivetti database for face recognition with CNN. The optimal model concludes with a flattened layer that serves as input to the fully connected layers of the Neural Network, consisting of three hidden layers and one output layer. Please see Fig. 2 for a detailed depiction of the final Neural Network.

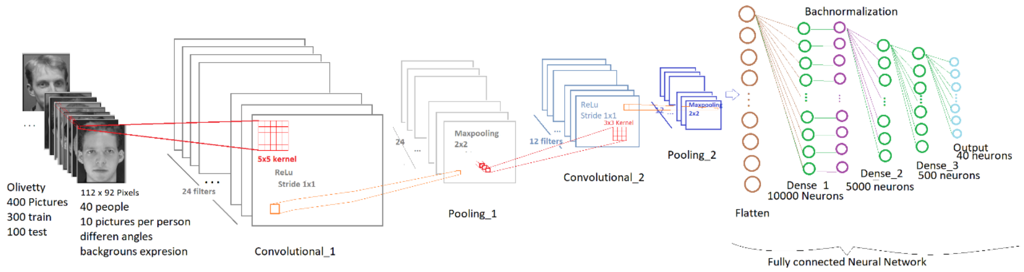

Fig. 2. Face identity and gender recognition Neural Network.

Subsection 2.2 will show the tuning process and the selection of hyperparameters. The hyperparameters include activation functions, kernels sizes, pooling sizes, stride sizes, learning rates, number of neurons in any fully connected layers, and numbers of epoch and batches.

2.1. Learning Process

The first problem is how to label two tasks, respectively. we designated the label as a vector of 40 zeros (0,0,0,0…0) to signify a male and a vector of 40 ones (1,1,1,1…1) to indicate a female label. However, the neural network (NN) was incapable of learning this task using Categorical Crossentropy [16, 17, 18], which necessitates at least one of the outputs to be 1 simultaneously. Therefore, we transformed the gender labels to (1,0,0,0…0) for males and (0,0,0…,1) for females to keep the output layer the same as the identity recognition task. Nevertheless, this led to more transfer learning and reduced accuracy in the first task when learning the second task, making continuous learning unfeasible. We require a method to deactivate all the unnecessary neurons in the first task not required for the second task. However, this method has not yet been implemented in Keras. Therefore, a new output layer with just two neurons needs to be added for the second task.

We experimented with three different methods to learn the second task of gender recognition without losing learning from the first task of identity recognition. The first method involved starting with a model that learns identity first and then gender using the full neural network for the training process of both tasks. However, the results were catastrophic, losing complete accuracy of the first task after learning the second task.

The second method entailed training only the output layer for the second task of gender recognition after the first task of identity recognition is completed. We saved the weights after training the first task and then loaded them to perform the second task. However, this led to substantial learning loss (1.88) after learning just a little gender recognition. In contrast, if the gender recognition task was learned first, it was impossible to learn the identity recognition task since the accuracy was only 3%. This is because the gender recognition task demands a smaller classifier (two useful neurons) compared to the identity recognition task (40 useful neurons).

The third method is parallel learning (learning both tasks at the same time). In parallel learning, two different outputs and the same neural network is being used. By comparing the results of parallel learning and the first two methods, the last Neural Network model was a combination of the re-training of the last layer in parallel for the output of the first identity task and the second task of gender identification. After running parallel learning, both tasks were learned simultaneously with a very good value of loss function (identity task 1.37, gender task 1.24) and a high accuracy (identity task 94%, gender task 95%).

The success of parallel learning lies in the way of labeling. The ideal solution for continuous learning is to re-train the entire output layer when training the second task and to deactivate neurons that are not in need. This could be the next step in the learning process, implementing these features in Keras.

2.2. Tuning Hyperparameters

The purpose of the Input Layer is to receive 300 training pictures and to shape them in a way that can be used for the first Convolutional layer.

There are 24 filters in Convolutional Layer 1. In the first method, we use only 16 filters, and we found that the loss and accuracy was slightly better with 24 filters. Given that the Olivetti database contains complex data, we retained the 24 filters. The Kernel is a 5 x 5 dimensions matrix, and we experimented using 3 x 3 dimensions, which did not yield any significant improvement in accuracy, although it helped to reduce the dimension of the data. The Stride is 1x1 dimensions matrix, which provides the best accuracy. The Rectified Linear Unit (ReLU) activation function was used, as it can reduce the error loss (We also tried the Scaled Exponential Linear Unit (SeLU) and Sigmoid functions, but they did not yield better results).

The Pooling Layer 1 is to reduce the dimension of data, in which a 2x2 dimension matrix with Max function is being used.

There are 12 filters in Convolutional layer 2. We initially used 8 filters but found that accuracy improved when we used 12 filters. We found that using a Kernel of 3x3 dimension and a Stride of 1x1 dimension produced the best accuracy. The ReLU activation function was used again.

The Pooling layer 2 is used to further reduce the dimensionality of the data, using a 2x2 dimension matrix with a Max function.

We experimented with adding a third Convolutional layer and Pooling layer, but found that it did not result in any significant improvement in accuracy for either task.

The Flatten layer comprises 26,208 neurons and serves as the starting point of the fully connected Neural Network.

For Dense Layer 1, we experimented with different numbers of neurons (7000, 6000, 5000, 10000), but found that the layer produced similar accuracy when completing the task of identity recognition after completing the gender task. A Sigmoid activation function is being used, as it produced the best loss value due to the non-linearity of the data.

Backnormalization [19, 20] was applied to reduce the range of the values in the layer and speed up the training process of the first task without losing accuracy. Even by adding more of this type of layer to any dense layer, there was little to no improvement in accuracy.

For Dense Layer 2, more weights are assigned to prevent forgetting the first task of identity recognition. There are 5000 neurons in Dense Layer 2. Besides, using 1000 neurons did not result in high accuracy for the first task. A Sigmoid activation function was used.

For Dense Layer 3, more weights are assigned to prevent forgetting the first task. Dense Layer 3 comprises a layer of 500 neurons. A Sigmoid activation function was used.

The Identity Layer serves as the output layer for face recognition of 40 people, with 40 neurons and a Softmax activation function that produces values of the output as 0 or 1. This layer is used in both models.

The layer for gender recognition is an output layer of 2 neurons that classifies male and female, using a Softmax activation function to produce values of the output as 0 or 1.

The network uses the Categorical Crossentropy function as its Loss Function

$$E(f(x)) = -y \cdot \log(f(x)) - (1-y) \cdot (1-f(x)) \qquad \text{Eq. (1)}$$

Equation. (1) helps the network to learn both tasks at the same time by turning the learning rates of some layers to 0 in order to freeze values. The weight needs to be kept same as it was learned in the first task.

The optimizer function is rmsprop [8]

$$W(t) = meansquatr(W(t-1) \cdot 0.9 + 0.1 \cdot (\frac{\partial E}{\partial w})^2 \quad \text{Eq. (2)}$$

The purpose to tune Epochs is to tune the number of epochs that can help improve accuracy (see Fig. 6 and Fig. 7).

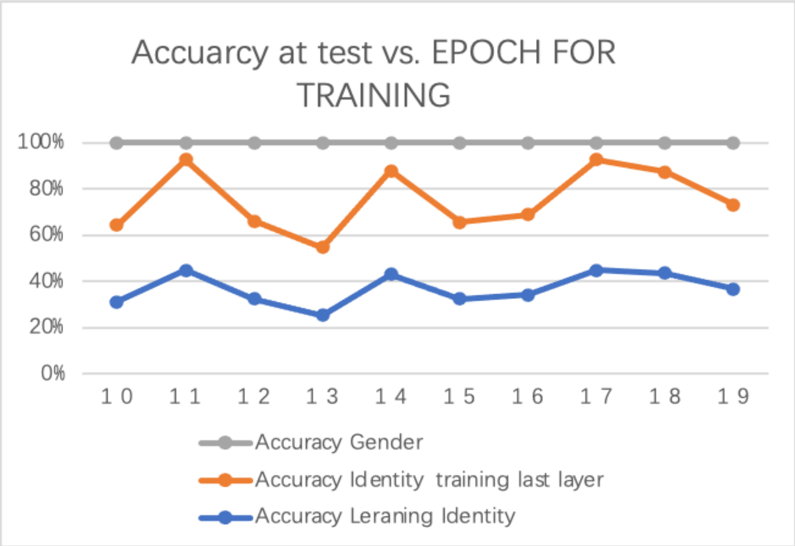

Fig. 6. Accuracy vs. Epochs at first training task needed to continue the learning after the parallel training of the identity and gender task.

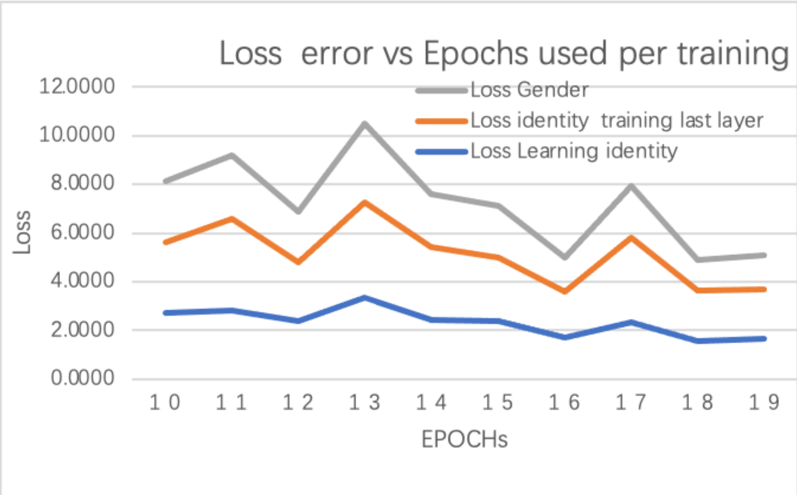

Fig. 7. Loss vs. Epochs at first training task needed to continue the learning after the parallel training of the identity and gender task.

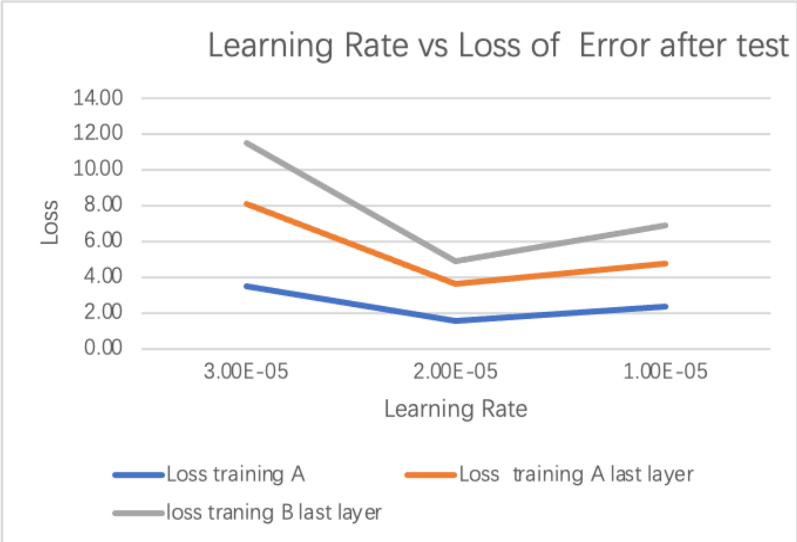

Fig. 8. Loss of error vs. learning rates after testing.

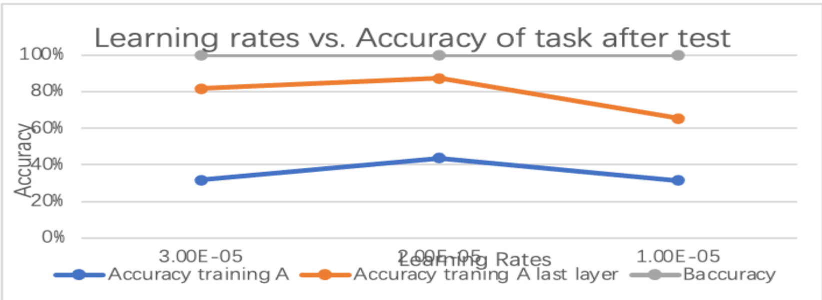

Fig. 9. Learning rates vs. Accuracy of task after testing.

3. Conclusions

In this paper, Continual Learning is achieved with an accuracy of 94% for both tasks. The Convolutional Neural Network uses a flexible loss function, Categorical Crossentropy, and a fully connected Neural Network that continues to learn the first task of identity recognition while simultaneously learning the second task of gender recognition. While the second task cannot be trained with the same output layer as the first task, it can be trained with a different output layer in parallel with the last layer of the first task. Notably, the network not only learns the second task, but also improves accuracy and reduces loss error for the identity recognition (first) task (Fig. 8 and Fig. 9). Besides, it is important to note that the first task has to be the more complex one of the two task (identity recognition is more complex than gender recognition) in order to realize continual learning. When the task of gender recognition is being trained first, the weights of the fully connected neural network were not adequate for learning the identity recognition task by training only the output layers. Keras is a powerful tool for CNN implementation that helps to speed up the network, and while it does have limitations, new functions can be added to enable the achievement of such goals.

Acknowledgment

This study was supported by grant No. 21511104505 from the Science and Technology Commission of Shanghai Municipality (STCSM). This work was carried out under Science Technology and Society Office at Shanghai Institute for Science of Science.

R E F E R E N C E S

[1]. Mark Bishop Ring, “Continual learning in reinforcement environments,” Ph.D. dissertation, University of Texas at Austin, 1994.
[2]. James Kirkpatrick, Razvan Pascanu, Neil Rabinowitz, Joel Veness, Guillaume Desjardins, Andrei, A. Rusu, Kieran Milan, John Quan, Tiago Ramalho, Agnieszka Grabska-Barwinska, Demis Hassabis, Claudia Clopath, Dharshan Kumaran, Raia Hadsell, “Overcoming Catastrophic Forgetting in Neural Networks,” PNAS, p. 201611835, 2017.
[3]. Matthias De Lange, Rahaf Aljundi, Marc Masana, Sarah Parisot, Xu Jia, Aleš Leonardis, Gregory Slabaugh, Tinne Tuytelaars, “A Continual Learning Survey: Defying Forgetting in Classification Tasks,” IEEE Transactions on Pattern Analysis and Machine Intelligence, July 2022, pp. 3366-3385, vol. 44.
[4]. Alex Krizhevsky, Ilya Sutskever, Geoffrey E. Hinton, “ImageNet Classification with Deep Convolutional Neural Networks” Communications of the ACM, Volume 60Issue 6 June 2017 pp 84–90.
[5]. Christian Szegedy, Wei Liu, Yangqing Jia, Pierre Sermanet, Scott Reed, Dragomir Anguelov, Dumitru Erhan, Vincent Vanhoucke, Andrew Rabinovich, “Going deeper with convolution”, 2015 IEEE Conference on Computer Vision and Pattern Recognition, Electronic ISSN 1063-6919.
[6]. Matthew D. Zeiler, Rob Fergus, “Visualizing and Understanding Convolutional Networks”, European Conference on Computer Vision, ECCV 2014: Computer Vision – ECCV 2014 pp 818–833.
[7]. Olivetti database, November 22, 2017.
[8]. S.H. Shabbeer Basha, Shiv Ram Dubey, Viswanath Pulabaigari, Snehasis Mukherjee, “Impact of fully connected layers on performance of convolutional neural networks for image classification”, Neurocomputing, Volume 378, 22 February 2020, Pages 112-119.
[9]. Xiaohan Ding, Chunlong Xia, Xiangyu Zhang, Xiaojie Chu, Jungong Han, Guiguang Ding, “RepMLP: Re-parameterizing Convolutions into Fully-connected Layers for Image Recognition”, ArXiv, arXiv:2105.01883.
[10]. T.-Y. Wu, Q. Meng, L. Yang, X. Guo, and S. Kumari,“A provably secure lightweight authentication protocol in mobile edge computing environments,” The Journal of Supercomputing, 2022. [Online]. Available: https://doi.org/10.1007/s11227-022-04411-9.
[11]. S. Lawrence, C.L. Giles, Ah Chung Tsoi, A.D. Back, “Face Recognition: A Convolutional Neural Network Approach,” IEEE Transactions on Neural Networks, Volume: 8, Issue: 1, January 1997.
[12]. Zewen Li, Fan Liu, Wenjie Yang, Shouheng Peng, Jun Zhou, “A Survey of Convolutional Neural Networks: Analysis, Applications, and Prospects,” IEEE Transactions on Neural

Networks and Learning Systems, 2021. [Online]. Available: 10.1109/TNNLS.2021.3084827.

[13]. Jiuxiang Gu, Zhenhua Wang, Jason Kuen, Lianyang Ma, Amir Shahroudy, Bing Shuai, Ting Liu, Xingxing Wang, Gang Wang, Jianfei Cai, Tsuhan Chen "Recent advances in convolutional neural networks" Pattern Recognition, Volume 77, May 2018, Pages 354-377.

[14]. Saad Albawi, Tareq Abed Mohammed, Saad Al-Zawi "Understanding of a convolutional neural network" 2017 International Conference on Engineering and Technology (ICET)}, 21-23 August 2017, Accession Number: 17615756.

[15]. ATT Laboratories Cambridge. The Database of Faces. November 24, 2017.

[16]. Yaoshiang Ho, Samuel Wookey, "The Real-World-Weight Cross-Entropy Loss Function: Modeling the Costs of Mislabeling" IEEE Access, Volume 8, 27 December 2019, Page 4806 - 4813.

[17]. Kevin P. Murphy, Machine Learning A Probabilistic Perspective, The MIT Press, 2012.

[18]. A. Rusiecki, "Trimmed categorical cross-entropy for deep learning with label noise" Electronics Letters, Volume55, Issue6, March 2019, Pages 319-320.

[19]. Ademola E. Ilesanmi, Taiwo O. Ilesanmi "Methods for image denoising using convolutional neural network: a review", Complex Intelligent Systems, 7, pages2179–2198 (10 June 2021).

[20]. Bhavesh Suneja, Ashish Negi, Rishav Bhardwaj, "Cloud based Medical Image De-Noising using Deep Convolution Neural Network" International Journal of Computer Applications}, Volume 184– No.20, July 2022.