Tony Blair Institute for Global Change
Trinity University, Lagos, Nigeria
Harmarth Global Educational Services
FAIR Forward – Artificial Intelligence for All
Deutsche Gesellschaft für Internationale Zusammenarbeit (GIZ) GmbH
Society for Multidisciplinary & Advanced Research Techniques (SMART)

32nd Accra Multidisciplinary Cross-Border Conference (AMCBC)

Classification Model For Iris Images Using Convolutional Neural Network (CNN)

Aranuwa F.O & Fawehinmi, O.B
Department of Computer Science
Adekunle Ajasin Uiversity, Akungba – Akoko, Ondo State, Nigeria
E-mail: felix.aranuwa@aaua.edu.ng; **Phone:** +2347031341911

ABSTRACT

learned features, a down-sampling method (pooling) was used to reduce the size of the images. Performance metrics such as sensitivity, specificity and accuracy were used in evaluating the performance of the model. Experimental results show that the model performed well with classification accuracy of 98.57% which is relatively an improvement over the model that was used as a benchmark with 93. 35%. The confusion matrix reported 0 false positives (FP) and 14 false negatives (FN) on the test set. This implies that the model correctly predicted all images belonging to the right category as right irises, while it wrongly predicted 14 images belonging to the left category. Connotionally, it revealed that the right iris recorded higher accuracy compared to that of the left iris. Based on this performance, the research work establishes that the CNN based model is an improvement over existing models in the domain. Therefore, the model in no small measure will increase the efficiency of general biometric systems, security applications and will be of great assistance to the eye specialists. Researcher in future work shall consider the case of multiple classes and larger database.

Keywords: Iris image, Classification model, Deep Learning, Convolutional Neural network, Features

Proceedings Citation Format
Aranuwa F.O & Fawehinmi, O.B (2022): Classification Model For Iris Images Using Convolutional Neural Network (CNN). Proceedings of the 32nd Accra Multidisciplinary Cross-Border Conference. University of Ghana/Academic City University College, Ghana. 2th June-1st July, 2022. Pp 7-22
www.isteams.net/ghanabespoke2022.
dx.doi.org/10.22624/AIMS-/AMCBC2022P2

1. BACKGROUND TO THE STUDY

Iris recognition is a biometric method of identifying people based on unique patterns within the ring-shaped region surrounding the pupil of the eye. Every iris is unique to an individual, making it an ideal form of biometric verification (NEC, 2019).

According to George (2017), the iris is the colored ring component of the eye situated between the pupil and white sclera within the human eye. Figure 1 shows the anatomy of the human eye.

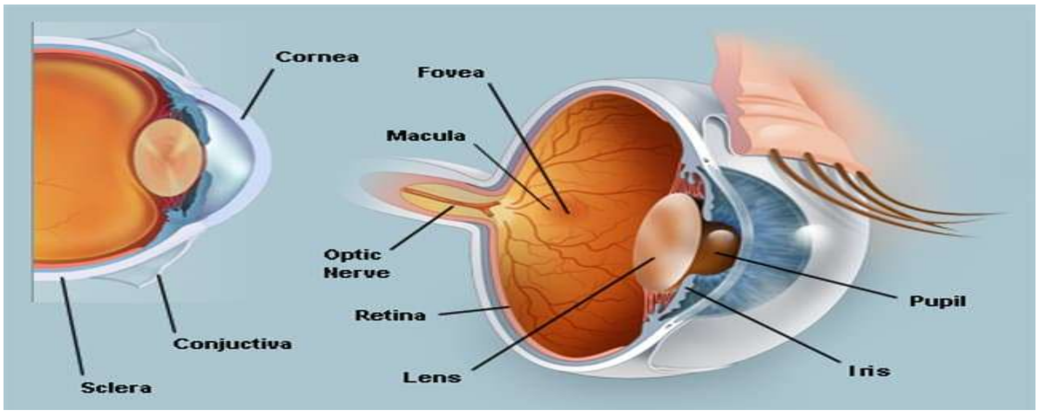

Figure 1: Anatomy of the eye (WebMD, 2015)

In biometrics, the iris belongs to the physiological traits of human being, and its function is to control the amount of light entering through the pupil. Major applications of iris recognition systems include criminal investigation, eye tracking, immigration and border controls, access and security applications, and so on. (Daugman, 2009). Like other biometric recognition system, iris recognition system is composed of the image acquisition, feature extraction, template formation and database, matching and decision-making stages (Minaee *et al*., 2016). According to Schmidhuber (2015), the classification of image data is largely based on the description, texture or similarity of the trait in question. Different robust machine learning techniques have been used for many classification tasks such as: Linear Regression, Random Forest, Support Vector Machine (SVM), Linear regression, Logistic regression, Naive Bayes, Linear discriminant analysis, Decision trees, K-nearest neighbor algorithm and so on.

However, recently deep learning approach have demonstrated remarkable success in multiple domains including computer vision, natural language processing, speech processing and in various biometric systems (Nguyen *et al*, 2018). According to Favio (2022), deep learning is a subfield of machine learning, a new approach on learning representations from data that puts emphasis on learning successive “layers” of increasingly meaningful representations. It is a modern update to Artificial Neural Networks that are concerned with building of large and complex neural networks. Deep learning techniques are effective in learning complex features from data, such as images, texts, audio, and videos. Its learning process can be supervised, semi-supervised or unsupervised depending on the data structure. The technique allows computational models that are composed of multiple processing layers to learn representations of data with multiple levels of abstraction. These layered representations are learned via models called “neural networks”, structured in literal layers stacked one after the other. Figure 2 depicts the simple neural and deep learning neural networks respectively.

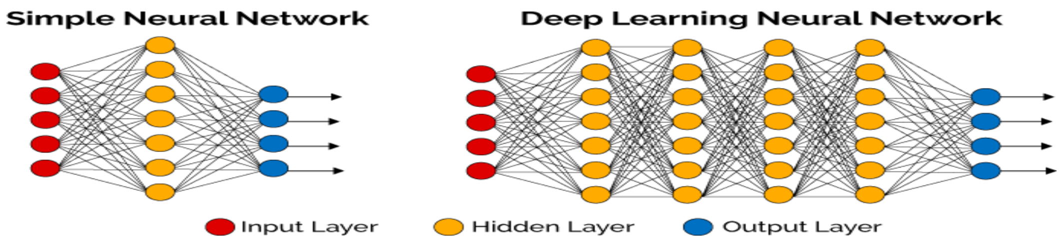

Figure 2: Showing the simple neural and deep learning neural networks
Source: (Favio, 2022)

Deep learning algorithms have been used to solve classical artificial intelligence problems with the main goal of learning high level abstractions from data (Sa *et al*., 2016; Khalifa *et al*., 2019). Popular deep learning techniques include: Artificial Neural Network (ANN), Convolutional Neural Network (CNN), Recurrent Neural Networks (RNNs), Long Short-Term Memory Networks (LSTMs), Stacked Auto-Encoders, Deep Boltzmann Machine (DBM), and Deep Belief Networks (DBN). Meanwhile, much attention is been given to Convolutional Neural Networks (CNN) because of its effectiveness in computer vision task such as image analysis and classifications. It has become one of the most appealing approaches in the recent time. It is very suitable for this work, hence it is considered.

According to Muhammad et al, (2022), CNN, also known as ConvNet, is one of the best neural networks for classification, segmentation, natural language processing (NLP), and video processing. Its application has become most demanding due to its ability to learn features from images automatically, involving massive amount of training data and high computational resources like graphic processing units (GPUs). According to Senan et al., (2020), the reason behind better performance of CNN is due to its ability to work on raw data without having prior knowledge of the data. The architecture of CNN is very similar to the ordinary Neural Networks, they are made up of neurons that have learnable weights and biases. Acording to Biswal (2022), a convolutional neural network is a feed-forward neural network (FNN) that is generally used to analyze visual images by processing data with grid-like topology. When using CNN algorithm for image classification, the image is usually fed directly to the convolutional neural networks, then the algorithm extracts the best features of the image.

CNNs are regularized versions of multilayer perceptron. Each neuron in one layer is connected to all neurons in the next layer. This full connection of these networks makes them prone to over-fitting. However, pooling process is a down-sampling operation that reduces the dimensionality of the feature map. According to Mandal (2021), the CNN consists of multiple hidden layers of structural parameters that help in extracting information from an image. The architecture is divided into four important layers, namely: the convolution layers, ReLU layer, pooling layers, and fully connected layers (O'Shea and Nash, 2015; Muhammad et al., 2022). Figure 3 shows CNN architecture.

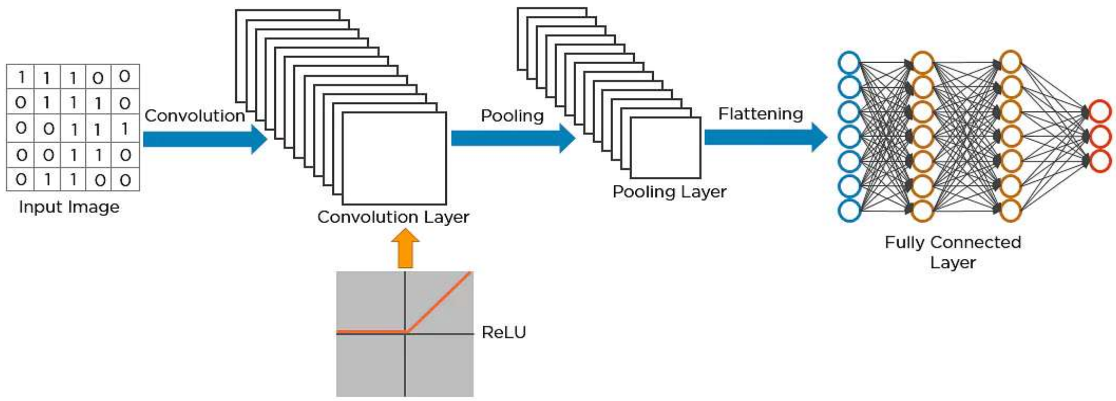

Figure 3: Architecture of CNN (Fully Connected Layers)
Source: (Sultana et al, 2018).

2. STATEMENT OF PROBLEM

Iris recognition has been adjudged one of the most reliable forms of biometric technology, as a result of its unique patterns, stringency to forgery and robustness against spoof attacks (Sundaram *et al*., 2011; Elgamal *et al*., 2013; Shah and Shrinath, 2014). Iris image is well known for high precision and possesses a good resistance to replication (Kong *et al*., 2010; Shah and Shrinath, 2014). Its unique patterns do not change over time, thus making it a perfect and reliable means of recognition (Dehkordi and Abu-Bakar, 2015). However, studies have revealed that inadequate classification sorts and methods leading to inaccurate matching and classification has characterized its processes in many authentication and identification applications. Various methods proposed to address these challenges as reported in (Hollingsworth *et al*., 2011; Thiyaneswaran & Padma, 2012; Oyedotun & Khashma, 2016; Arsalan *et al*., 2017; Zhao & Kumar, 2017; Tapia & Aravena, 2017; Bobeldyk & Ross, 2019; Khalifa *et al*., 2019) are majorly based on single iris image classification, and major drawbacks of these approaches are insufficient learning input for proper classifications. To the best of our knowledge, few works that considered both left and right iris classification using traditional neural networks suffers a trade-off between accuracy and computational complexity coupled with high error rate. Hence, this work presents an enhanced classification model based on convolutional neural network (CNN) to proffer solutions to the challenges of the convetional approaches.

3. METHODOLOGY

3.1 The Research Design

To achieve the objectives of the work, the architecture of the classification model was first designed as depicted in figure 4. The model was designed based on the CNN, composed of five major components: the image processing, the convolution layer feature maps, the rectifier linear unit, pooling layer features maps and the fully connected layers (classification layers). The image preprocessing in the model architecture involves feature extraction from the image pixels.

In the convolusion layer, the convolution operations are performed by combining the image with a feature detector or learnable filter in order to extract data from neighbouring pixels.

The rectified linear unit (ReLU) is used to increase the non-linearity of the output image in the convolution layer after the required features have been filtered. Pooling is applied to the output from the ReLU to extract the most meaningful information. In order to avoid over fitting of the learned features, a down-sampling method (max pooling) was used to reduce the size of the pooled image, and the number of values as input are forwarded to the classification layer containing fully connected layer of neurons for image classification.

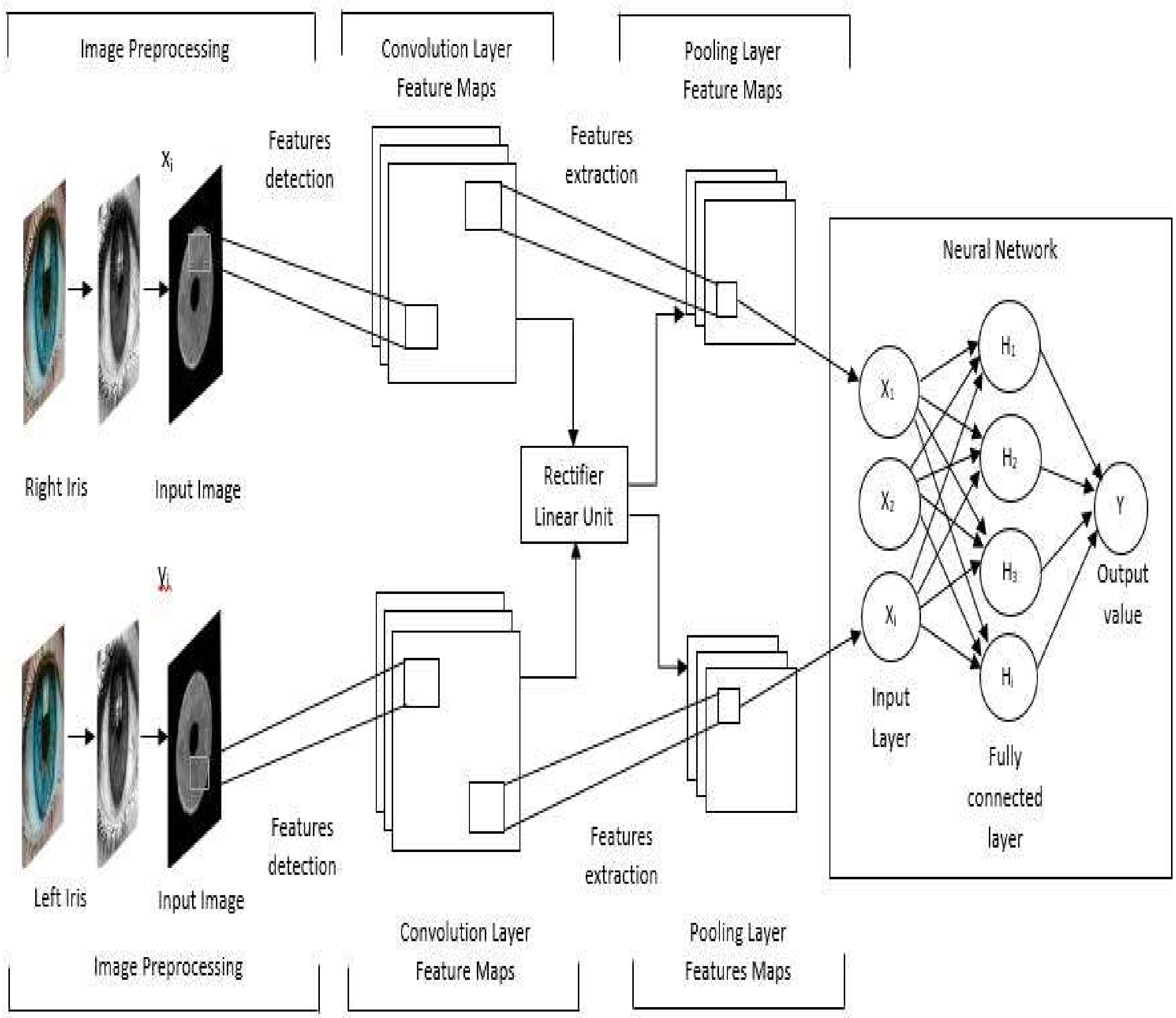

Figure 4 The Architecture of the Classification Model

The process is mathematically presented in equation 1:

$$Preprocessed\ image \rightarrow (Conv \rightarrow ReLU \rightarrow Pool) * M \rightarrow (FC \rightarrow ReLU) * K \rightarrow FC \rightarrow Classification \quad (1)$$

Where:

1. Conv: is the Convolutional Layer
2. ReLU: is the Rectified Linear Unit
3. FC: is the Fully Connected Layer
4. M and K: are numbers representing the number of times each operation is performed.

3.2 Image Acquisition

The data for the work was acquired from the CASIA-Iris-Lamp dataset (http://biometrics.idealtest.org, 2010). The CASIA-Iris-Lamp dataset is a robust iris feature representation. The iris images are 8-bit gray-level JPEG files, collected under near infrared illumination. A total of 16,163 iris data was considered in the research work. The training set contained 12,931 iris images, while the test set contained 3,232 data. 20 iterations were passed on the model to determine the accuracy of the model.

3.3 Image Processing

The image processing involves salient feature extraction, segmentation and normalization of the acquired images. The image acquired was enhanced through segmentation of the required region of the trait, followed by normalization process to form a fixed pattern in polar coordinates. At the segmentation phase, the separation of the iris from the whole eye area took place. At this stage, the position of the upper and lower eyelids was determined, as well as the exclusion of areas covered by the lashes. Iris segmentation acts directly on the image of the iris, seeking the maximum normalized standard circle along the path, a partial derivative of the blurred image relating to the increase of the circle radius.

The center of the coordinates and the radius of the circle were sought for, and edge of the iris was determined. The purpose of the segmentation stage is to enhance the image in order to produce high clarity iris image that serves as input to the classification model. The enhancement reduces computational complexity of the model with a goal to avoid decline of matching performance. For the segmentation process, circular Hough Transform was applied to detect the boundaries of the iris and the pupil. This also involves performing Canny Edge detection to generate an edge map (Sangwan & Rani, 2015). The Canny Edge detection contains five stages: Smoothing, Finding gradients, Non-maximum suppression, double thresholding, and edge tracking through hysteresis (Jayachandra & Reddy, 2013). The normalization process is the transformation of the localized iris to a defined format in order to allow comparisons with other iris codes into an iris template (binary code). The number of values as input are forwarded to the classification layer.

3.4 Matching Process

This process establishes correspondences between the set of the images acquired. The approach involves detecting a set of interest points each associated with image descriptors from iris image. The performance of the matching methods is based on the properties of the underlying interest points and the choice of associated iris image descriptors (Tyagi, 2019). Table 1: shows the sampled interest point scores.

Table 1: Sampled interest point scores

Identity	Right iris		Left iris		Tx	Ty
	x_i (mm)	xi (mm)	yi (mm)	y (mm)	(mm)	(mm)
1	10.1	10.3	10. 3	10.4	10.2	10.35
2	10.2	10.5	11.5	11.6	10.35	11.55
3	11.0	11.4	11.5	12.0	11.2	11.75
4	12.2	12.3	12.5	12.8	12.25	12.65
5	11.5	12.0	12.1	12.6	11.75	12.35

3.5 Performance Evaluation

Sensitivity, specificity and accuracy were used to evaluate the performance of the classification model coupled with the confusion matrix report.

Values of TN, TP, FN and FP were used to calculate the performance metrics considered.

Sensitivity = $\frac{TP}{TP+FN} = \frac{1602}{1602+14} = 0.9913$ (99.13%) *equation 2*

Specificity = $\frac{TN}{TN+FP} = \frac{1616}{161} = 1.0000$ (100%) *equation 3*

Accuracy = $\frac{TP+TN}{TP+TN+FP+FN} = \frac{1602+161}{1602+1616+0+} = 0.9957$ (99.57%) *equation 4*
(11)

Where,

1. TP represents the True Positives: These are cases in which the model predicted YES (the images are left irises), and they are actually left irises.
2. TN represents the True Negatives: The model predicted NO for left irises, and they are actually NOT left irises.
3. FP represents False Positives: The model predicted YES for left irises, but they are actually NOT left irises.
4. FN represents False Negatives: The model predicted NO for left irises, but they are actually left irises. Table 2 showed the confusion matrix report for the classification of the iris images considered.

Table 2: Confusion Matrix Report for Classification of Iris Images

Actual Class	Predicted Class	
	Left Iris	Right Iris
Left Iris	*TN = 1,616*	*FP = 0*
Right Iris	*FN = 14*	*TP = 1,602*

4. DATA PRESENTATION

4.1 Experimental Setup

The derived pooled feature map in equation (1) is flattened into a long vector to serve as input layer for the neural network with fully connected neuron layers for the image classification. During the experiment, the dataset was fed into the Anaconda platform installed for the purpose of the work. The software is an open-source application for management of multiple Python versions on one computer. Anaconda provides a large collection of highly optimized, commonly used data science libraries to carry out experiments. It has a powerful IDE with advanced editing, interactive testing, debugging and introspection features called Spyder. Figure 5 presents the Anaconda Navigator (Individual Edition (Distribution) Interface).

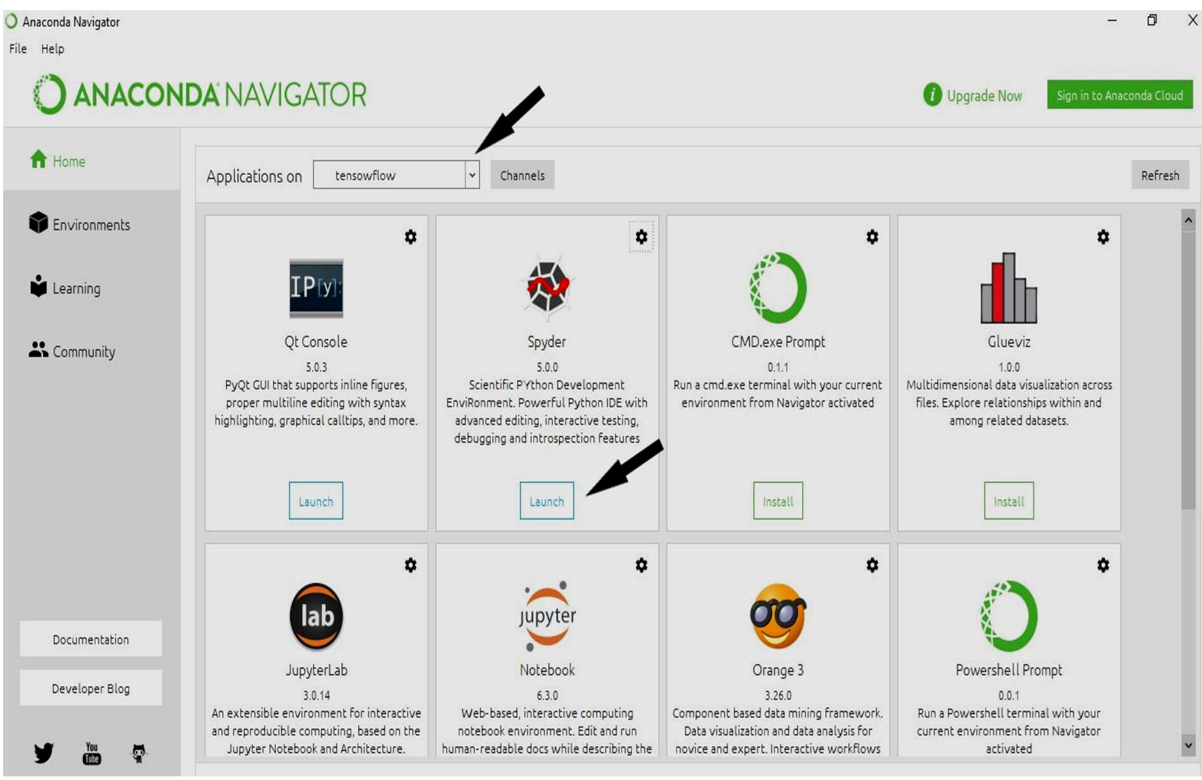

Figure 5: Anaconda Navigator (Individual Edition (Distribution) Interface)

A specific internal environment (tensorflow) was created to manage the IDE and python libraries (Theano, TensorFlow, Keras). Figure 6 presents the IDE and Python Libraries Environment

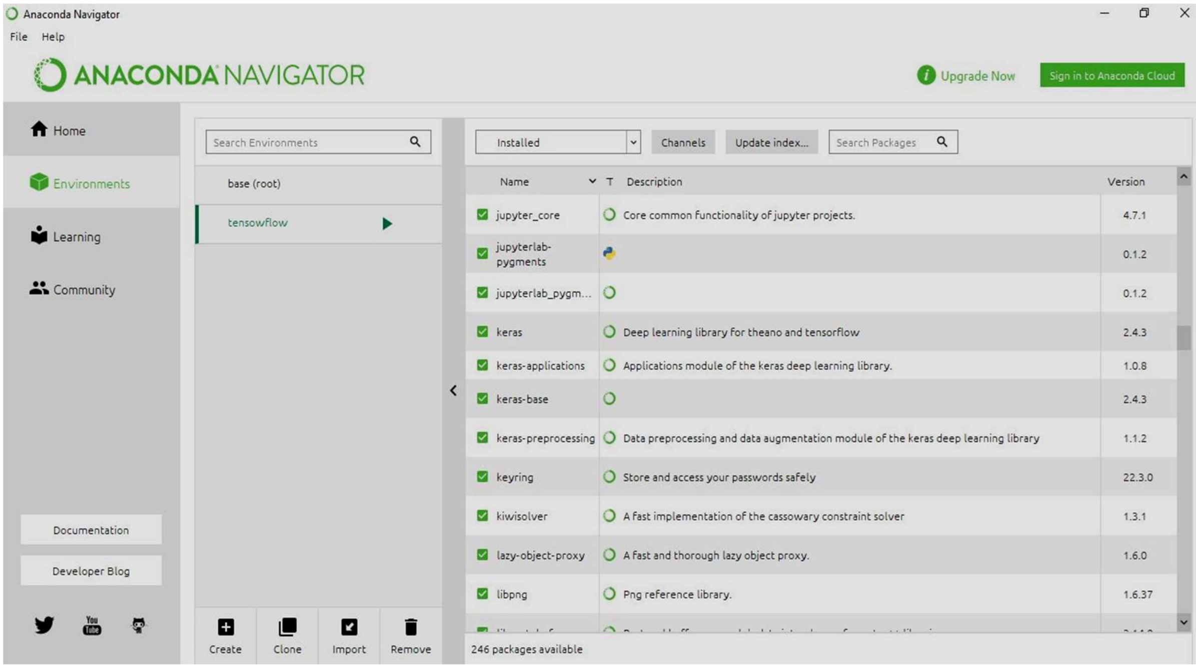

Figure 6: IDE and Python Libraries Environment

Figure 7 presents the implementation in Spyder IDE and describes the following three major aspects: the dataset was fed into the platform by setting it as working directory. The dataset was splitted into two folders (*training_set*, *test_set*) within the working directory using 80/20 ratio (that is 80% of the total dataset for training the model and the remaining 20%) for validating the model. Since this work focuses on binary classification of the left and right iris image, two sub-folders (*left*, *right*) were created within the *training_set* and *test_set* folders to hold the two classes of labelled iris images (left irises, right irises). Secondly, the CNN model was built using Python programming language which supports scientific/numeric computing and effective integration of systems, and Thirdly, a console in figure 7 shows the classification report as the model is trained and validated with the dataset.

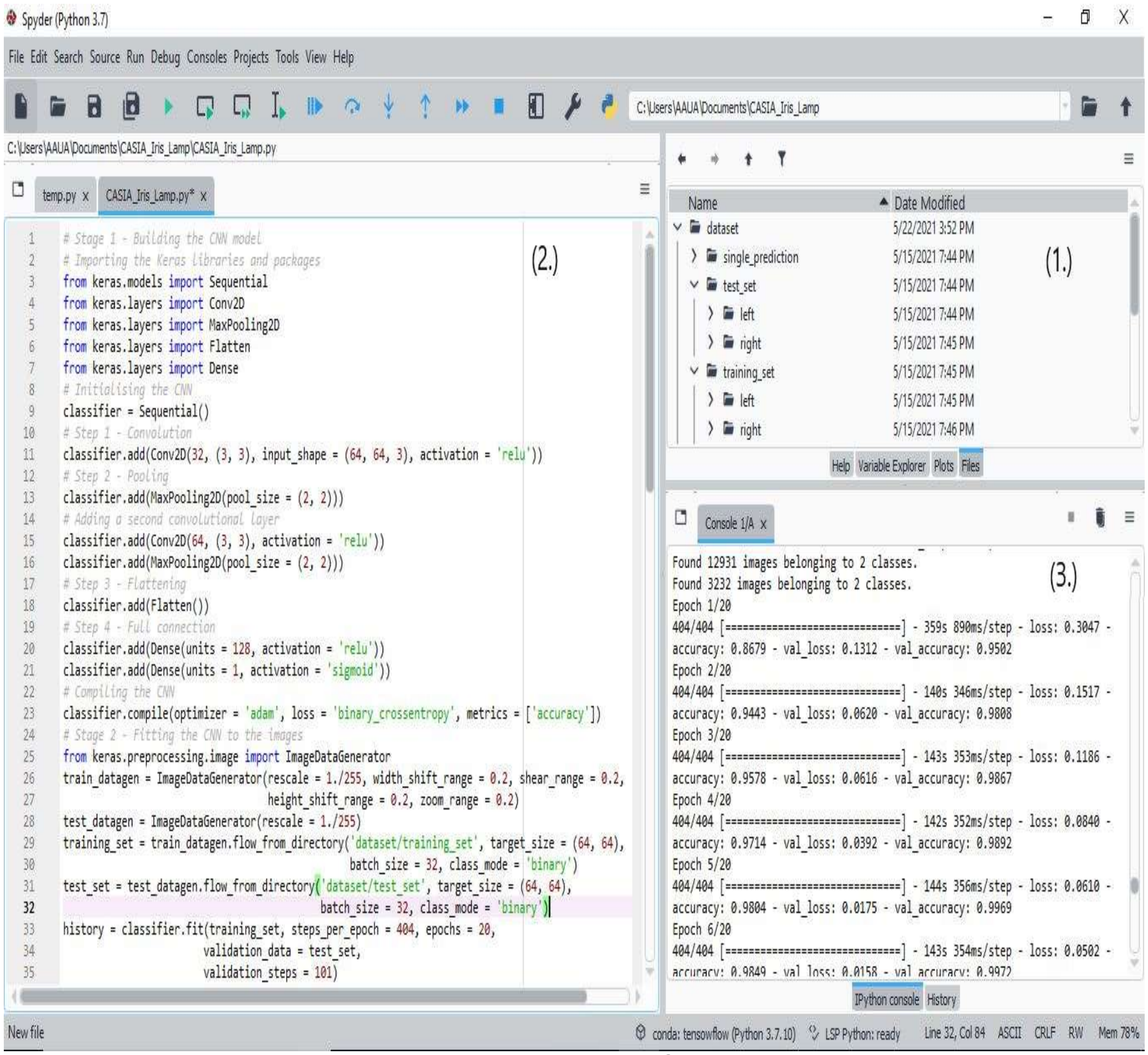

Figure 7: Implementation in Spyder IDE

5. RESULTS AND FINDINGS

This work adopts the scaling law ratio for training set composed of 80% of the total dataset, while the remaining 20% is allocated for the test set. 20 iterations were passed on the model to determine the accuracy of the developed model. The snapshot of the the iteration process is shown in Figure 8.

```
Epoch 8/20
404/404 [==============================] - 143s 355ms/step - loss: 0.0415 - accuracy: 0.9884 - val_loss: 0.0252 - val_accuracy: 0.9966
Epoch 9/20
404/404 [==============================] - 145s 358ms/step - loss: 0.0371 - accuracy: 0.9895 - val_loss: 0.0287 - val_accuracy: 0.9960
Epoch 10/20
404/404 [==============================] - 145s 359ms/step - loss: 0.0338 - accuracy: 0.9905 - val_loss: 0.0202 - val_accuracy: 0.9960
Epoch 11/20
404/404 [==============================] - 144s 356ms/step - loss: 0.0347 - accuracy: 0.9912 - val_loss: 0.0256 - val_accuracy: 0.9935
Epoch 12/20
404/404 [==============================] - 145s 360ms/step - loss: 0.0296 - accuracy: 0.9918 - val_loss: 0.0217 - val_accuracy: 0.9941
Epoch 13/20
404/404 [==============================] - 144s 357ms/step - loss: 0.0255 - accuracy: 0.9942 - val_loss: 0.0204 - val_accuracy: 0.9950
Epoch 14/20
404/404 [==============================] - 146s 362ms/step - loss: 0.0264 - accuracy: 0.9925 - val_loss: 0.0323 - val_accuracy: 0.9947
Epoch 15/20
404/404 [==============================] - 148s 367ms/step - loss: 0.0258 - accuracy: 0.9930 - val_loss: 0.0544 - val_accuracy: 0.9858
Epoch 16/20
404/404 [==============================] - 155s 384ms/step - loss: 0.0248 - accuracy: 0.9929 - val_loss: 0.0249 - val_accuracy: 0.9957
Epoch 17/20
404/404 [==============================] - 168s 417ms/step - loss: 0.0239 - accuracy: 0.9933 - val_loss: 0.0359 - val_accuracy: 0.9913
Epoch 18/20
404/404 [==============================] - 181s 447ms/step - loss: 0.0202 - accuracy: 0.9950 - val_loss: 0.0242 - val_accuracy: 0.9960
Epoch 19/20
404/404 [==============================] - 190s 469ms/step - loss: 0.0214 - accuracy: 0.9940 - val_loss: 0.0195 - val_accuracy: 0.9947
Epoch 20/20
404/404 [==============================] - 234s 579ms/step - loss: 0.0186 - accuracy: 0.9950 - val_loss: 0.0259 - val_accuracy: 0.9957
```

Figure 8: Iteration Process

At the 20th iteration level, the classification report gave a training accuracy of 99.50% with validation accuracy of 99.57% while a training loss of 1.86% was achieved with validation loss of 2.59%. A difference of 0.07% was achieved between the training and validation accuracy which shows that the model was able to overcome over-fitting by effectively closing the gap between the training and validation accuracy.

Table 3 showed the classification report for the training and validation process. Figure 9 presents a graph of the Training and Validation Loss while figure 10 presents the Training and Validation Accuracy using matplotlib in python.

Table 3: Classification Report for Training and Validation

Epoch	Training Loss (%)	Validation Loss (%)	Training Accuracy (%)	Validation Accuracy (%)
1	30.47	13.12	86.79	95.02
2	15.17	6.20	94.43	98.08
3	11.86	6.16	95.78	98.67
4	8.40	3.92	97.14	98.92
5	6.10	1.75	98.04	99.69
6	5.02	1.58	98.49	99.72
7	4.53	1.89	98.56	99.78
8	4.15	2.52	98.84	99.66
9	3.71	2.87	98.95	99.60
10	3.38	2.02	99.05	99.60
11	3.47	2.56	99.12	99.35
12	2.96	2.17	99.18	99.41
13	2.55	2.04	99.42	99.50
14	2.64	3.23	99.25	99.47
15	2.58	5.44	99.30	98.58
16	2.48	2.49	99.29	99.57
17	2.39	3.59	99.33	99.13
18	2.02	2.42	99.50	99.60
19	2.14	1.95	99.40	99.47
20	1.86	2.59	99.50	99.57

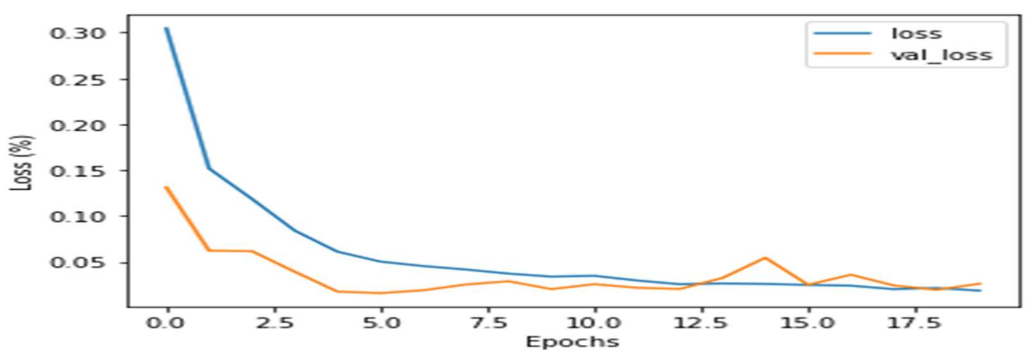

Figure 9: Training and Validation Loss

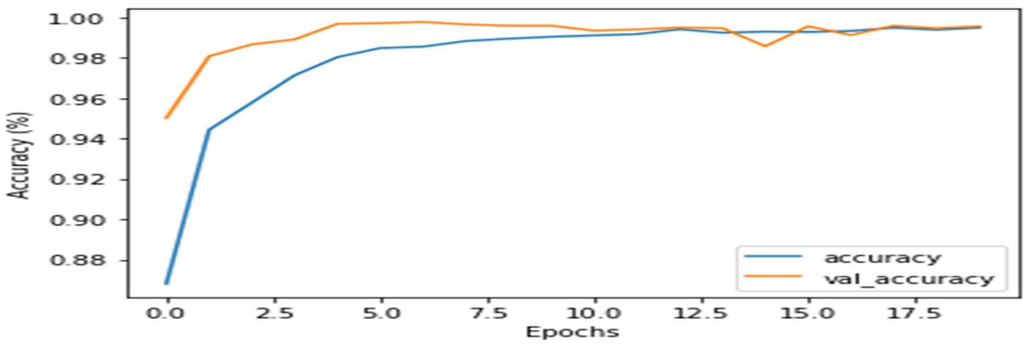

Figure 10: Training and Validation Accuracy

Figure 11 showed the snapshot of confusion matrix report.

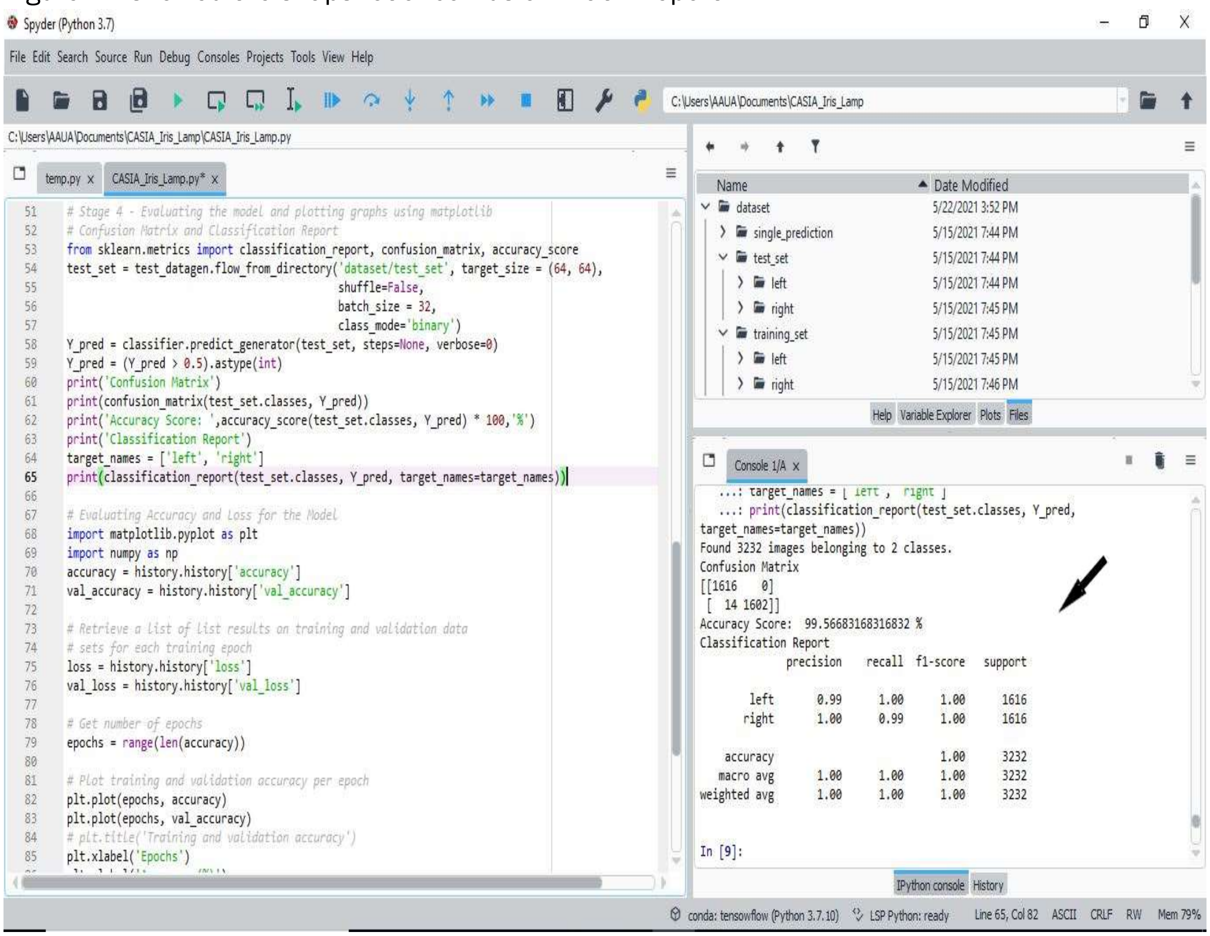

Figure 11: Snapshot of Confusion Matrix Report

Interpreting the confusion matrix report, the model reported 0 false positives (FP) and 14 false negatives (FN) on the test set. This surfice to say the model correctly predicted all images belonging to right category as right irises while it wrongly predicted 14 images belonging to the left category. By implication, there was higher accuracy on the right iris compared to the left iris.

6. CONCLUDING SUMMARY AND CONTRIBUTION TO KNOWLEDGE

This work presented an efficient classification model based on CNN for the classification of both left and right iris images to proffer solution to the inadequacies of the classification sorts and the challenges of the single mode classification techniques. Data for the work acquired was segmented and normalized and were later fed into the Anaconda model building platform installed for the work. The classification model developed was tested and had an accuracy of 99.57% which is relatively an improvement over existing iris feature detectors and classification techniques. The confusion matrix reported 0 false positives (FP) and 14 false negatives (FN) on the test set. The shows that the model correctly predicted all images belonging to right category as right irises while it wrongly predicted 14 images belonging to the left category.

By implication, there was higher accuracy on the right iris compared to the left iris. Based on the performance of the model presented, the research work has established that CNN based model as an improvement compared to other models in the domain. The current model in no small measure will increase the efficiency of biometric security applications and assit eye medical practitioners. Researcher in the future work will consider the case of multiple classes and larger database.

REFERENCES

1. Adekunle, Y. A., and Aiyeniko, O. (2020). Feature Extraction Techniques for Iris Recognition System: A Survey. International Journal of Innovative Research in Computer Science & Technology, ISSN: 2347-5552, 8(2) 56-61. Available at SSRN: https://ssrn.com/abstract=3578701
2. Aranuwa, F. (2020). Information Fusion Schemes for Reliable Biometric System. Journal of Biometrics & Biostatistics. 4(1). 1-005.
3. Arsalan, M., Hong, H. G., Naqvi, R. A., Lee, M. B., Kim, M. C., Kim, D. S., Kim, C. S., and Park, K. R. (2017). Deep Learning-Based Iris Segmentation for Iris Recognition in Visible Light Environment. *Symmetry* 9(263), 1-25, doi:10.3390/sym9110263
4. Bobeldyk, D., and Ross, A. (2019). Analyzing Covariate Influence on Gender and Race Prediction from Near-infrared Ocular Images. IEEE Access 2019, 7(1), 7905–7919.
5. Bilswal, A (2022). Convolutional Neural Network Tutorial. Simplilearn. Retrieved online on 25/05/2022 at: https://www.simplilearn.com/tutorials/deep-learning-tutorial/convolutional-neural-network.
6. CASIA Iris Image Database. Available online: http://biometrics.idealtest.org/dbDetailForUser.do?id=4 (accessed on 13 July, 2020).
7. Dehkordi, A. B., and Abu-Bakar, S. A. (2015). A Review of Iris Recognition System. *Journal Teknologi*, 77(1) 234-245.

8. Elgamal, M., and Al-Biqami N. (2013). An Efficient Feature Extraction Method for Iris Recognition Based on Wavelet Transformation. *International Journal of Computer and Information Technology,* 2(3) 1-7.
9. Favio, V. (2022). Deep Learning Made Easy with Deep Cognition. BBVA Data & Analytics. Retrieved on 30/3/2022 at: https://www.kdnuggets.com/2017/12/deep-learning-made-easy-deep-cognition.html.
10. George, A. (2017). Image Based Eye Gaze Tracking and its Applications. 10.13140/RG.2.2.33992.88329
11. Hollingsworth, K., Bowyer, K. W., Lagree, S., Fenker, S. P., and Flynn, P. J. (2011). Genetically Identical Irises have Texture Similarity that is not Detected by Iris Biometrics. *Computer Vision and Image Understanding*, 115(1), 1493–1502.
12. Jayachandra, C., and Reddy, H. V. (2013). Iris Recognition based on Pupil using Canny edge detection and KMeans Algorithm. International Journal Of Engineering And Computer Science ISSN:2319-7242 2(1). 221-225.
13. Khalifa, N., Taha, M., Hassanien, A. E., & Mohamed, H. (2019). Deep Iris: Deep Learning for Gender Classification Through Iris Patterns. *Journal of the Society for Medical Informatics of Bosnia and Herzegovina*, 27(2), 96–102. https://doi.org/10.5455/aim.2019.27.96-102.
14. Kong, W. K., Zhang D., and Kamel, M. S. (2010). An Analysis of Iris Code. *IEEE Trans. on Image Processing*, 19(1), 522-532.
15. Mandal, M (2021). Introduction to Convolutional Neural Networks (CNN). An article published as a part of the Data Science Blogathon. Retrived online at https://www.analyticsvidhya.com/blog/2021/05/convolutional-neural-networks-cnn/ on 25/05/2022.
16. Minaee, S., Abdolrashidiy, A., and Wang, Y. (2015). Iris Recognition Using Scattering Transform and Textural Features. *In Signal Processing and Signal Processing Education Workshop (SP/SPE)*, 2015 IEEE, 2015, pp. 37-42.
17. Minaee, S., Abdolrashidiy, A., and Wang, Y. (2017). Face Recognition using Scattering Convolutional Network. *2017 IEEE Signal Processing in Medicine and Biology Symposium (SPMB),* Philadelphia, PA, 2017, pp. 1-6, doi: 10.1109/SPMB.2017.8257025.
18. Muhammad, A. S., Norhalina, S., Fazli, W., Muhammad, A. Ali, S., and Mukhtaj, K (2022). Comparative Analysis of Recent Architecture of Convolutional Neural Network 22(1), 1-9.
19. NEC (2019). What is Iris Recognition and how does it work? Retrieved on 24/05/2022 from https://www.nec.co.nz/market-leadership/publications-media/what-is-iris-recognition-and-how-does-it-work/
20. Nguyen, K., Fookes, C., Ross, A., and Sridharan, S. (2018). Iris Recognition with off-the-shelf CNN Features: A Deep Learning Perspective. *IEEE Access 2018*, Vol. 6, pp. 18848–18855.
21. O'Shea, K., and R. Nash, (2015). An Introduction to Convolutional Neural Networks. CoRR abs/1511.08458: http://arxiv.org/abs/1511.08458.
22. Oyedotun, O., and Khashman, A. (2017). Iris Nevus Diagnosis: Convolutional Neural Network and Deep Belief Network. *Turkish Journal of Electrical Engineering & Computer Sciences*, Vol. 25, pp. 1106-1115, 2017.

23. Sa, I., Ge, Z., Dayoub, F., Upcroft, B., Perez, T., and McCool, C. (2016). DeepFruits: A Fruit Detection System Using Deep Neural Networks. *Sensors 2016*, 16(8), 12-22; https://doi.org/10.3390/s16081222
24. Sangwan, S., and Rani, R. (2015). A Review on: Iris Recognition. International Journal of Computer Science and Information Technologies (IJCSIT), 6 (4) , 3871-3873.
25. Schmidhuber, J. (2015). Deep Learning in Neural Networks: An Overview. *Neural Networks*, 61(1), 85–117.
26. Senan, N. Aamir, M. Ibrahim, R. Taujuddin, N. and Muda, W. W (2020). An efficient convolutional neural network for paddy leaf disease and pest classification," *International Journal of Advanced Computer Science and Applications* 11(1). 27-36.
27. Setman, W (2020). A Picture of the Eye. WebMD, LLC. Retrieved on 23/03/2022 at https://www.webmd.com/eye-health/picture-of-the-eyes
28. Shah, N., and Shrinath, P. (2014). Iris Recognition System – A Review. *International Journal of Computer and Information Technology*, 3(1) 13.20.
29. Sultana, F. Sufian, A.and Dutta, P. (2018). Advancements in image classification using convolutional neural network in *Proceedings of the 2018 4th International Conference on Research in Computational Intelligence and Communication Networks (ICRCICN)*, pp. 122–129, IEEE, Kolkata, India, November 2018.
30. Sundaram, R. M., and Dhara, B. C. (2011). Neural Network Based Iris Recognition System using Haralick Features. *2011 3rd International Conference on Electronics Computer Technology (ICECT 2011)*. DOI: 10.1109/ICECTECH.2011.5941793
31. Tapia, J., & Aravena, C. (2017). Gender Classification from NIR Iris Images Using Deep Learning. In *Advances in Computer Vision and Pattern Recognition* PartF1, 219-239), Springer London, https://doi.org/10.1007/978-3-319-61657-5_9
32. Thiyaneswaran, B., and Padma, S. (2012). Iris Recognition using Left and Right Iris Feature of the Human Eye for Bio-Metric Security System. *International Journal of Computer Applications (0975 - 8887)*, 50(12), 211-218.
33. Tyagi, D. (2019). Introduction To Feature Detection And Matching. In Data Breach. Available online - https://medium.com/data-breach/introduction-to-feature-detection-and-matching-65e27179885d, Accessed 22nd June, 2021.
34. WebMD (2015). A Picture of the Eye. WebMD, LLC. Retrieved on 23/03/2022 at https://www.webmd.com/eye-health/picture-of-the-eyes.
35. Zhao, Z., and Kumar, A. (2017). Accurate Periocular Recognition Under Less Constrained Environment Using Semantics-Assisted Convolutional Neural Network. In *IEEE Transactions on Information Forensics and Security*, 12(5), 1017-1030.