Multi-Label Multi-Class Classification Of ESRGAN Enhanced Retinal Fundus Images Of Diabetic Retinopathy

Dr. A. V. Sri Harsha[1*], B. Dilli Babu[2], K. Srinath[3], J. Amarnath [4], A. Rithwik[5], Penchala Praveen Vasili[6]

[1*] Professor, Dept. of CSE(DS), Mohan Babu University(Erstwhile Sree Vidyanikethan Engineering College), Tirupati.India
avsreeharsha@gmail.com
[2,3,4,5] UG Scholar, Department of Computer Science and Systems Engineering, Sree Vidyanikethan Engineering College, Tirupati, India
[6]Product Manager,Wellsfargo Inc. Charlotte, USA
penchalapraveen@gmail.com

Abstract. Diseases of eye have the potential to cause blindness in the sufferers. There have been many kinds of diseases those exist in the eyes of human like Myopia, Diabetic Retinopathy, Hypermetropia and so on.Fundus images help doctor to see the amount of eye infected due to diabetes and indicate the suitable prescription whether an immediate action is required or nothing to worry.The research is carried to identify what stage is the diabetes infection of the eye, based on the stage of danger the patient can be informed the further proceedings in the treatment. It helps the doctor to easily identify the disease stage of the patient and proceed with the treatment process.This classification model is implemented on ESRGAN enhanced retinal fundus images which makes the process simple by increasing the quality of DR images. It can be further used to classify the kinds of diseases in the human eye and also the stages of DR.

Keywords: Image processing, Image Acquisition, ESRGAN ,Neural networks, Activation Function and Pooling Layers, Diabetic Retinopathy.

1 Introduction

Diabetic retinopathy is a primary reason why permanent visual reduction in this age individuals (20–65). Approximately one out of every three diabetics have some degree of DR, and one in ten will progress the illness to the point that it threatens their vision. According to the Worldwide Organization for the Initiative to Prevent Blindness projections, 45 M persons experienced vision-threatening DR in 2021, out of 145 million people with DR in general. An abrupt increase in diabetes can be used to predict an increasing burden of DR. It is crucial to identify and treat DR as soon as possible. Fundus fluorescein angiography, optical coherence tomography, and retinal fundus photography are prominent diagnostic techniques for diabetic retina (DR).
Given that DR predominantly impacts the retinal vessels, FFA is essential in recognizing malignant alterations like vascular leakage, non-perfusion regions, & some tiny aneurysms that are missed by other approaches. The components of the neuroglia in DR are NPAs with compromised structure and function. Prior to the usual and successful treatment for DR, laser photocoagulation,

K. R. Madhavi et al. (eds.), *Proceedings of the International Conference on Computational Innovations and Emerging Trends (ICCIET 2024)*, Advances in Computer Science Research 112,
https://doi.org/10.2991/978-94-6463-471-6_38

FFA is required to detect NP and retinal neuro-vascularization. Nevertheless, the accuracy of the manual method of DR lesion detection relies on the experience of the specialists and is a time-consuming process. The progress in artificial intelligence (AI) for medical picture interpretation has caused the creation of many autonomous diagnosis algorithm. 2004 saw Zhang & co. made an effort to use machine learning to categorize and recognize exudates and woolly cotton patches in color fundus images. Since deep learning became widely available and demonstrated high performance in various medical domains such as skin cancer, cardiovascular disease risk prediction, lung cancer detection, and breast cancer histopathology analysis, there has been a great deal of interest in applying DL to support DR diagnosis worldwide.

Researchers demonstrated that a deep learning system could detect referable DR on the APTOS data set, where each training sample is linked to multiple labels because a sample can be described in more than one way, possessing a 0.980 region beneath the operating characteristic curve of the receiver and sensibility & specificity of 87.0% & 96.8% respectively. To detect DR lesions and blindness, we created a ML-MC classification system based on deep learning of ESRGAN -enhanced retinal fundus pictures.

Researchers demonstrated that a deep learning system could detect referable DR on the APTOS data set, where each training sample is linked to multiple labels because a sample can be described in more than one way, with sensitivity and specificity of 87.0% & 96.8% respectively, and region beneath the operating characteristic curve of the receiver of 0.980. To detect DR lesions and blindness, we created a ML-MC classification system based on deep learning of ESRGAN-enhanced retinal fundus pictures.
—>A Neural Network is developed for MLMC classification.
—>Collecting a complex dataset of retinal images.

2 DATASET

The collection is made up of retinal images that have been Gaussian filtered in order to identify diabetic retinopathy. APTOS 2019 Blindness Detection is where you may access the actual dataset. To make them compatible with a variety of pre-trained deep-learning models, these photos have been scaled to 224 × 224 pixels. Every shot has already been categorized into the correct folders according to the degree or stage of diabetic retinopathy using the train.csv file that is included. Five directories with corresponding photos can be found here:
0 - No DR
1 - Mild
2 - Moderate
3 - Severe
4 - Proliferate DR.
The collected pictures are enhanced by using an image processing technique GAN. ESRGAN made the images more clear by increasing the size and resolution of DR images to a size of 896x896 ie, quadruple the size of each image.
No_DR is a stage in blindness where there is no effect of diabetics on the sight of a person, Mild is a condition where the patient is in the starting stage of losing sight and has no need to worry.

Proliferation is the final stage of human eye blindness which occurs when the diabetic human eye is completely lost and is about to loss its ability to sight.

3 Methodology

The suggested process for creating a model that applies multi-label and multi-class categorization is depicted in the figure 1. Two networks—one for classification and the other for labeling—make up the model. The data is first split in 2 groups, the train_set and the val_set, with 0.8% and 0.2% respectively. Next, the data is cleaned up using appropriate preprocessing techniques, and finally, the images are quadrupled from 224x224 to 896x896. The classifier's capacity to extract high features and traits is correlated with its prediction power. Scaling and cropping are used to enhance the data and increase the training dataset's diversity. Particularly in situations where there is a shortage of data, data augmentation assisted in enhancing the model's capacity for generalization and lowering overfitting. To improve picture collections for the diagnosis of diabetic retinopathy, transfer learning is applied ESR-GAN is the main technique we employ for image enhancement.

A. ESR-GAN Architecture:

ESR-GAN's complex architecture enables high capacity and effective training. Figure depicts the fundamental building unit, known as the Residual in Residual Dense unit (RRDB). A residual structure with dense blocks can be found in RRDB. Within the Dense blocks, a further layer of residual learning is incorporated to boost network capacity without becoming overly complicated. Compared to conventional Dense blocks, this novel architecture, known as ESRGAN+, produces images with higher perceptual quality by enhancing both feature exploitation and exploration.

B. Generator training objective:

Train the generator network, denoted GθG, to generate high- resolution and super-resolution (ISR) images from low resolution (ILR) inputs.
Loss function: Use a super-resolution specific loss function, referred to as lSR, designed to guide the process of training by quantifying the distinction between the generated super-resolution images and the corresponding high-resolution ground truths.
Optimization: Minimize the average loss across a dataset of N pairs of training images (I LR, I H R), where I HR represents the corresponding high-resolution ground truth. The optimization process involves adjusting the parameters (θG) of the generator to increase its ability to accurately reconstruct high-frequency details in super-resolution images.

C. Discriminator training objective:

To train a discriminator network, denoted DθD , to distinguish between generated super-resolution images (ISR) and true high-resolution (IHR) images.

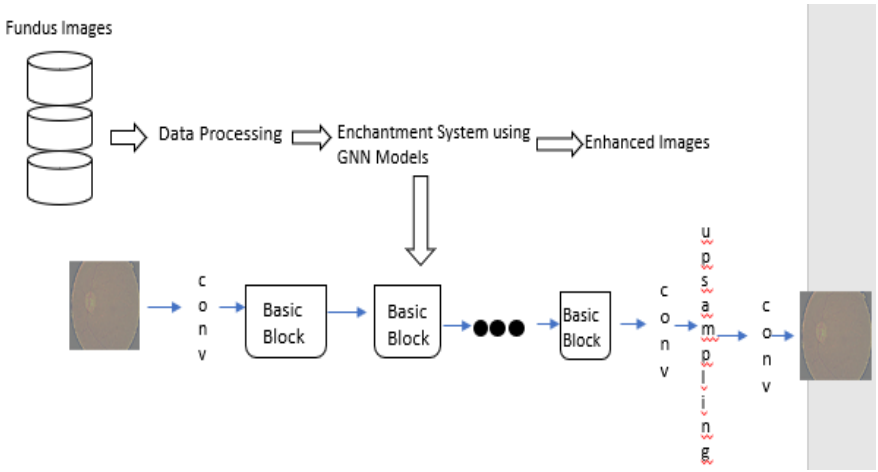

Fig. 1. ESRGAN MODEL ARCHITECTURE

Adversary Training: Implement an adversary training strategy that includes an adversary loss term to encourage the generator (GθG) to produce high-resolution photographs.

Optimization: Use an alternating optimization process, where the discriminator is optimized to maximize the adverse loss, in order to better discern from both real and produced images. Additionally, the generator is tuned to minimize adverse losses, resulting in the generation of high-resolution images that closely resemble real high-resolution images.

D. Data Preprocessing:

The datasets are preprocessed by caching, shuffling, and prefetching to optimize data retrieval during training. This improves training performance by reducing data loading time. They are cached, randomized, and prefetched using methods like. cache(), .shuffle(), and .prefetch(). Caching keeps a dataset in memory after it is loaded for faster access. Shuffling ensures that the order of examples is random to avoid any bias during training. Preloading overlays data preprocessing and model execution to improve training performance. Before feeding the images into the CNN model, they are resized to a uniform size and rescaled to a range between 0 and 1. This is achieved using the tf.keras. Sequential() model named resize_and_rescale. The layers.Resizing() layer resizes images to the specified IMAGE_SIZE. Layers.Rescaling() scales the pixel values by dividing them by 255.0 so that they are in the range [0, 1].

E. Data Augmentation:

Data_augmentation is a different tf. keras. Sequential() model that is used to design data augmentation strategies. By randomly transforming photos, data augmentation contributes to the training dataset's increased diversity. This code makes use of two extension techniques:

- strata.Images are randomly flipped both horizontally and vertically with

RandomFlip("horizontal_and_vertical").

• strata. Images can be rotated randomly up to 20 degrees with Random Rotation(0.2).

Overall, these data preprocessing steps and data augmentation ensure that the dataset is properly formatted, partitioned, and augmented to facilitate efficient training of the CNN model. Preprocessing steps also help improve model generalization and performance.

The convolutional layers are used for feature extraction. These layers detect patterns in input images using convolutions with learnable filters. Max-pooling layers are used to resample feature maps obtained from convolutional layers, reducing computational complexity and focusing on the most important features. Dense (fully connected) Layers are employed in classification.These layers take the extracted elements and map them to output classes.

F. Convolution Layers Algorithm:

Convolutional Neural Networks(CNN) for Prediction:

This study employs Convolutional Neural Networks (CNNs), shown in figure 2, as the operational framework for disease prediction using DR images. It incorporates multiple convolutional layers and pooling layers, alongside a singular fully connected layer, to train the image dataset. The process involves several steps:

- Initially, the Sequential model is initialized, enabling the sequential addition of layers.
- Conv2D layers are added to capture features from input images. Key parameters like filter size, activation function ('relu'), and stride are defined.
- MaxPooling2D layers are inserted after some Conv2D layers to down sample the data and manage model complexity.
- Following the convolutional and pooling layers, a Flatten layer is introduced to reshape the data into a one-dimensional array. Dense layers are employed for classification tasks, utilizing the extracted features. The final Dense layer utilizes 'softmax' activation to produce class probabilities for accurate classification[12].
- During compilation, the model is configured with the 'Adam' optimizer and employs the 'categorical cross-entropy' loss function. Additional evaluation metrics are designated to evaluate the model's performance throughout the training phase.
- Training the model with the fit() method involves learning from the training data generator (train_gen) and validating using the validation data generator (val_gen).
- Parameters like steps_per_epoch and validation_steps are set to control the number of batches processed during each epoch for training and validation, respectively.

Convolutional layers are in charge of taking input images and extracting their features. With 32 3x3 filters, the Conv2D employs the ReLU activation function. MaxPooling2D that lowers a spatial dimensions of feature maps comes next. Two more sets of max-pooling (MaxPooling2D) and convolution (Conv2D) layers follow the first layers. ReLU activation function and 64 3x3 filters are used in the second and third Con2D layers. Each Con2D layer is complied with a MaxPooling2D layer towards resample on maps with feature.

The formula for the convolution operation at one spatial position is:

$$z_{ij} = \sum_{m=0}^{F} \sum_{n=0}^{F} x_{(i+m)(j+n)} \cdot w_{mn} + b$$

Softmax activation is used in the last dense layer for multi-class classification. The formula for the softmax function for a class I is:

$$softmax(x_i) = \frac{e^{x_i}}{\sum_{j=1}^{C} e^{x_j}}$$

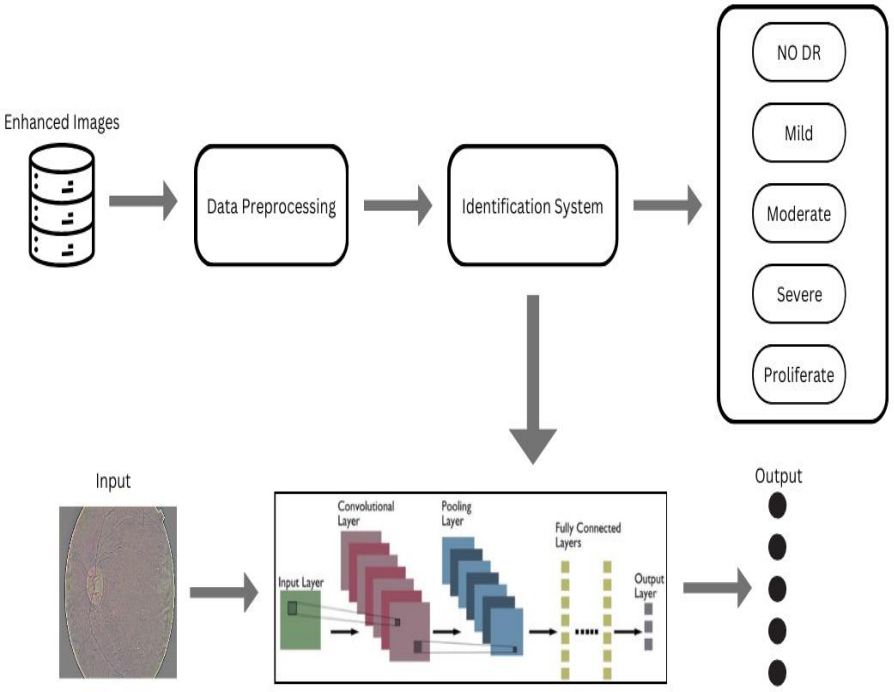

Fig. 2. CONVOLUTION NERUAL NETWORK

G. Training & validation loss:

The machine learning model's training and validation losses are plotted on a graph over multiple epochs in the above figure. One full run through the whole training data set is called an epochs. The training is shown by the blue line loss. This row shows how the model's functionality on the training data improves over time. We see that the training loss drops sharply at the beginning and then flattens out, which is typical when the model starts to converge to a solution.

Training and Validation loss shown in figure 3, the orange line represents the loss of authentication. This row shows how the model operates on a separate dataset not used for training, which can provide insight into how effectively the model extrapolates to fresh, untested data.The validation loss initially decreases, but then shows some fluctuations and spikes that could indicate problems such as overfitting or instability in the training process.

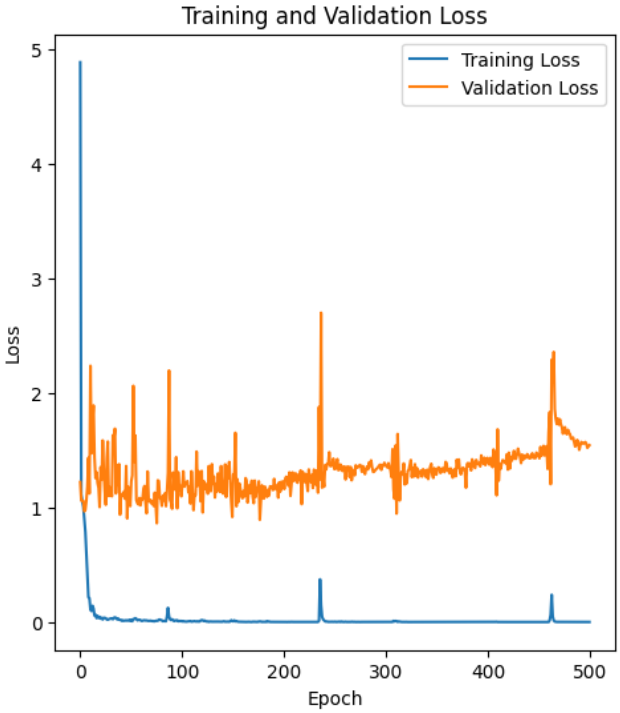

Fig. 3. TRAINING AND VALIDATION LOSS

H. Training & validation accuracy:

The train accuracy is represented by blue line. The model increasing performance the training data set is displayed this row. It is clear that the model performs admirably on the train_set of data because the train accuracy quickly approaches 1.0.

The correctness of the verification is shown by the orange line. This row provide information on the model's generalization ability to new, unknown data by displaying its performance on a different dataset that was not utilized for training. Along with a rise, the validation accuracy eventually falls short of the typical machine learning model's training accuracy.
A graph plotting the train and val accuracy is displayed in figure 4.

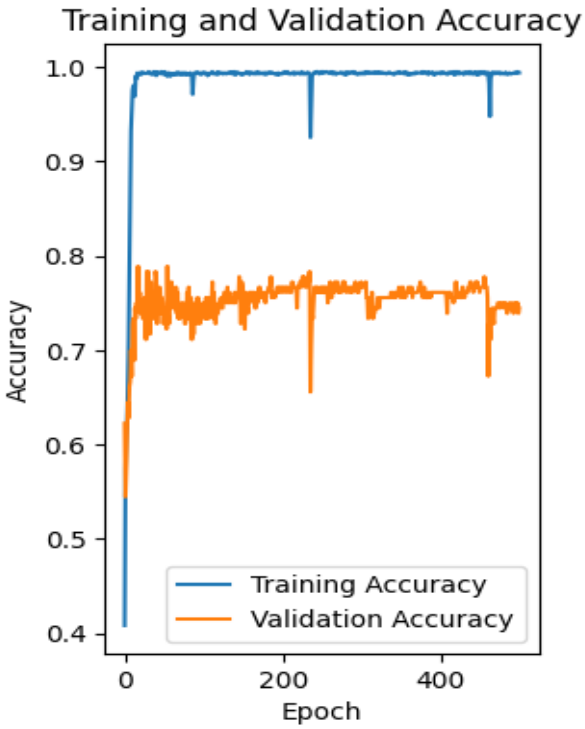

Fig. 4. TRAINING & VALIDATION ACCURACY

4 Results and Discussions

The accuracy of the model that we obtained is 84.15%. Google Colab was used to train the model with divided data of train=80% and test= 20%. The model that is trained on the improved DR pictures by ESRGAN employs random eraser augmentation. The model is trained for over 500 epochs, but after 200 epochs we reached the desired accuracy and there is no further progress in accuracy. Training for the first session took about two hours, and the website provides the class output using the Flask framework. Our approach only categorizes DR images according to their susceptibility to vision loss and the stage at which it is expected to impact human eyes. The accuracy at each label provides information about the model's performance. It is OK to train a single network for many labels, but there are numerous issues to resolve.

ESR-GAN Output:

Because the loss from each component is passed back to the initial network, model optimization is a difficult task in this training. This method has demonstrated a notable improvement in the model's accuracy and performance, assisting ophthalmologists in determining the stage of diabetic retinopathy. It also streamlines the prescription process and increases the doctors' productivity by cutting down on the time spent using some older instruments for the classification of DR images and its output shown in figure 5.

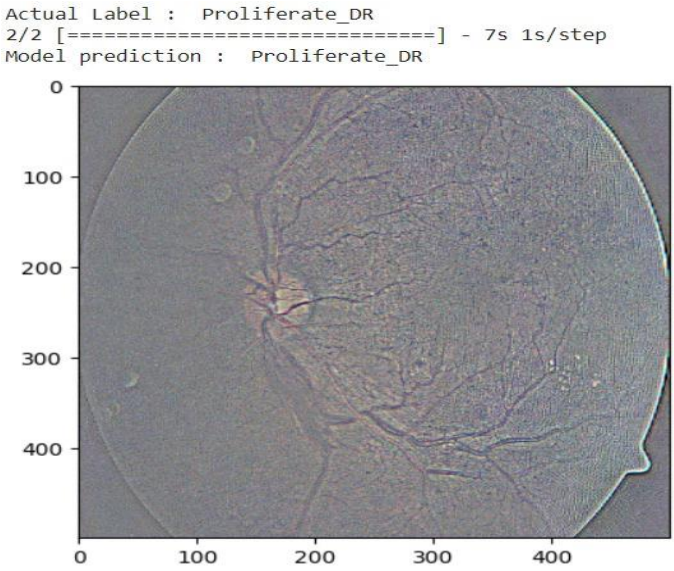

Fig. 5. ESRGAN Model Output

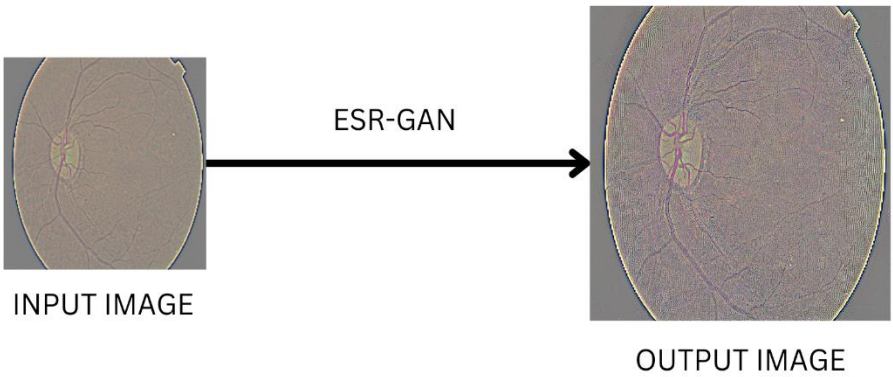

Fig. 6. MCML Output

The figure 6 shows the output of the model by predicting to which stage of Diabetic Retinopathy the input image belongs. As shown above actual label refers to which stage the input image belongs to and the model prediction refers to the output from our model. As the actual label and model prediction both are the same our model is working perfectly.

5 Conclusion

To sum up, this work introduced a deep learning (ML-FEC) model for automated recognition & grouping of various retinal lesions from fundus pictures for five stages of diabetic retinopathy (DR). The model is built on a pre-trained CNN framework. According to experimental findings, the suggested model offered an accuracy of 84.15%. In comparison with the most recent models, our outcomes were superior. These results demonstrated the effectiveness and precision of our fundus image analysis technique, allowing for its integration into routine clinical practice and bolstered by comprehensive, well-designed screening programs for the early identification of DR. Future work will undertake further tests because there aren't many publicly available big training fundus datasets with numerous detected lesion labels that facilitate multi-label training.

References

1. Alyoubi, W. L., Shalash, W. M., & Abulkhair, M. F. (2020). Diabetic retinopathy detection through deep learning techniques: A review. Informatics in Medicine Unlocked, 20, Article 100377. 10.1016/j.imu.2020.100377.
2. Gadekallu, T. R., et al., (2020). Early detection of diabetic retinopathy using pca-firefly based deep learning model. Electron, 9(2), 1–16. 10.3390/electronics9020274.
3. APTOS 2019 Blindness Detection. 2019, Kaggle.: Kaggle.
4. Willis, J. R., et al., (2017). Vision-related functional burden of diabetic retinopathy across severity levels in the United States. JAMA ophthalmology, 135(9), 926–932. 10.1001/jamaophthalmol.2017.2553.
5. Arcadu, F., Benmansour, F., Maunz, A., Willis, J., Haskova, Z., & Prunotto, M. (2019). Deep learning algorithm predicts diabetic retinopathy progression in individual patients. NPJ Digital Medicine, 2(1). 10.1038/s41746-019-0172-3.
6. K. Simonyan, A. Vedaldi, and A. Zisserman, “Deep inside convolutional networks: Visualising image classification models and saliency maps,” 2nd Int. Conf. Learn. Represent. ICLR 2014 - Work. Track Proc., pp. 1–8, 2014.
7. Kuo, J. Z., et al., (2012). Systemic soluble tumor necrosis factor receptors 1 and 2 are associated with severity of diabetic retinopathy in Hispanics. Ophthalmology, 119(5), 1041–1046. 10.1016/j.ophtha.2011.10.040O.
8. Chitteti, Chengamma, and K. Reddy Madhavi. "Taylor African vulture optimization algorithm with hybrid deep convolution neural network for image captioning system." Multimedia Tools and Applications (2024): 1-19.
9. Hamad, R. A., Kimura, M., & Lundström, J. (2020). Efficacy of imbalanced data handling methods on deep learning for smart homes environments. SN Computer Science, 1(4), 1–10. 10.1007/s42979-020-00211-1.H.
10. Heydon, P., et al., (2021). Prospective evaluation of an artificial intelligence-enabled algorithm for automated diabetic retinopathy screening of 30 000 patients. British Journal of Ophthalmology, 105(5), 723–728. 10.1136/bjophthalmol-2020-316594C.

11. Fisher, L., Polonsky, W. H., & Hessler, D. (2019). Addressing diabetes distress in clinical care: A practical guide. Diabetic medicine: a journal of the British Diabetic Association, 36(7), 803–812. 10.1111/dme.13967S.
12. Alwakid, G.; Gouda, W.; Humayun, M. Deep Learning-based prediction of Diabetic Retinopathy using CLAHE and ESRGAN for Enhancement. Preprints 2023, 2023020097 (doi: 10.20944/preprints202302.0097.v1.
13. Y.K. Saheed, A.O. Akanni, and M.O. Alimi, "Influence of discretization in classification of breast cancer disease," Univ. PITESTI Sci. Bull. Electron. Comput. Sci., vol. 18, no. 2, pp. 13–20, 2018.
14. Kuraparthi, Swaraja, Madhavi K. Reddy, C. N. Sujatha, Himabindu Valiveti, Chaitanya Duggineni, Meenakshi Kollati, and Padmavathi Kora. "Brain Tumor Classification of MRI Images Using Deep Convolutional Neural Network." Traitement du Signal 38, no. 4 (2021).
15. Ting, D. S. W., et al., (2017). Development and validation of a deep learning system for diabetic retinopathy and related eye diseases using retinal images from multiethnic populations with diabetes. JAMA - The Journal of the American Medical Association, 318(22), 2211–2223. 10.1001/jama.2017.18152.
16. Prasanth, S. (2021). A detailed survey on Prognostication of diabetes diagnosis on the basis of machine learning techniques and the detection approaches to diabetic retinopathy using Artificial Intelligence. International Journal of Advanced Trends in Computer Science and Engineering, 10(2), 874–887. 10.30534/ijatcse/2021/571022021.
17. Madhavi, K. Reddy, Padmavathi Kora, L. Venkateswara Reddy, Janagaraj Avanija, K. L. S. Soujanya, and Prabhakar Telagarapu. "Cardiac arrhythmia detection using dual-tree wavelet transform and convolutional neural network." Soft Computing 26, no. 7 (2022): 3561-3571.
18. F. Image, K. Xu, D. Feng, and H. Mi, "Deep Convolutional Neural NetworkBased Early Automated Detection of Diabetic Retinopathy Using," 2017, doi: 10.3390/molecules22122054H.