IJASCE : Int. J. of Adv. Sci. Comp. and Eng, 6(2) - August 2024 64-69

International Journal of
Advanced Science Computing and Engineering

Journal Homepage www.ijasce.org/index.php/ijasce

Impact the Classes' Number on the Convolutional Neural Networks Performance for Image Classification

Amna Kadhim Ali [a,*], Abdulhussein Mohsin Abdullah [b], Sabreen Fawzi Raheem [c]

[a] *Veterinary Public Health Branch, College of Veterinary Medicine, University of Basrah, Basrah, 61, Iraq*
[b] *Computer Technology Engineering Department, Alkunooze University College, Basrah, 61, Iraq*
[c] *Basrah Technical Institute, Southern Technical University, Basrah, 61, Iraq*

Corresponding author: [*]amna.kadhim@uobasrah.edu.iq

***Abstract*—Deep learning was developed as a realistic artificial intelligence technique that takes in numerous layers of information and produces the best results in various classes. Deep learning has demonstrated excellent performance in several areas, particularly image classification, object detection, and recognition. The convolutional neural network (CNN) is one of the algorithms that relies on deep learning in its work. It has proven its effectiveness in classifying images with high efficiency in various fields, including medical images and their diagnoses, as well as face recognition. In this paper, the focus was on images to alert new researchers to their effects on the performance of CNN in terms of the number of classes that existed within the database, in addition to the impact of incorrect classification of images by the source on the classification result and the necessity of adopting reliable and correct sources of data to avoid inaccurate results. A group of face images has been used, and three experiments on them were conducted using all existing classes with reduction. The results showed a significant improvement in the performance of the algorithm whenever the number of classes was reduced. The best result was when only two classes were chosen for classification, reaching a validation accuracy of 85%.**

***Keywords*—CNN; deep learning; facial expression; image classification; image preprocessing.**

Manuscript received 4 Feb. 2024; revised 19 Mar. 2024; accepted 28 Apr. 2024. Date of publication 31 Aug. 2024.

I. INTRODUCTION

Since the 1950s, a small portion of artificial intelligence, known as machine learning, has revolutionized various areas. A neural network is a subfield of machine learning, and it was from this topic that Deep Learning (DL) began [1]. Since its beginnings, DL has caused ever-increasing disruptions, demonstrating exceptional performance in practically every application sector [2]. A Convolutional Neural Network (CNN) is one of the DL algorithms that typically analyzes visual input like images and movies. CNNs have transformed the area of computer vision by achieving cutting-edge performance in a variety of applications [3].

The key idea behind CNN is to exploit the spatial structure of the input data. Traditional neural networks process the entire input data as a flat vector, ignoring any spatial relationships. CNN, on the other hand, takes advantage of the local correlations present in the data by using a special type of layer called a convolutional layer [4]. It is widely used in various real-life problem-solving scenarios, particularly in the field of computer vision to solve real-world problems like image classification [5], where CNN excels at image classification tasks because it can accurately identify and categorize objects within images. It has been used in applications such as autonomous driving (to detect pedestrians, traffic signs, and other vehicles) [6], medical imaging (to diagnose diseases from scans) [7], [8], [9], and facial recognition systems [10].

Also, it can perform pixel-level segmentation for semantic segmentation, where each pixel in an image is classified into specific classes or categories. This technique is proper in medical imaging (for tumor segmentation) [11], autonomous vehicles (to identify drivable areas and obstacles) [12], and image editing tools (for background removal and image manipulation) [13]. While CNN is primarily used in computer vision, it can also be applied to Natural Language Processing (NLP) tasks such as text classification and sentiment analysis, especially for tasks involving text that has a grid-like structure, such as character recognition in handwriting or document analysis [14], [15], in addition to other fields.

The CNN algorithm is applied to databases that contain a different number of classes. In research [16], the researcher employed deep learning technology to identify melanoma on a three-class dataset. Melanoma is a form of skin malignancy. The suggested model differentiates between benign lesions, superficial spread, and nodular melanoma. This enables early viral detection and the prompt isolation and treatment required to prevent future spread. Deep learning (DL) and non-standard machine learning techniques are exemplified in the deep layer structure of the CNN's neural network algorithms, demonstrating the CNN algorithm's performance with 88% accuracy compared to other algorithms.

In paper by [17], a database of brain tumors consisting of four classes was used, namely (GLIOMA, MENINGIOMA, NO-TUMOR, and PILUITARY). A three-step preprocessing method is presented, as well as a novel Deep Convolutional Neural Network (DCNN) architecture for glioma, meningioma, and pituitary tumor identification. The approach employs (batch normalization) to allow for faster training with a greater learning rate and to simplify weight initialization. A few convolutional layers, max-pooling layers, and training iterations are included in the suggested design. The proposed designs were compared to the other models discussed in this paper. The overall competitive accuracy is 98.22% when evaluated on a dataset of (3,394) MRI pictures, with 99% recognizing glioma, 99.13% detecting meningioma, 97.3% detecting pituitary, and 97.14% detecting normal images.

In research by [18], the researchers used two facial expression databases and described a unique technique based on hierarchical DL, with the first base (CK+) consisting of seven expressions and six emotions (anger, disgust, fear, happiness, sorrow, and surprise) being used as experimental data. Furthermore, the (JAFFE) collection includes gray-scale frontal facial expression images of ten women, each with a distinct facial emotion (anger, disgust, fear, joy, sadness, surprise, and neutral). The feature produced from the appearance feature-based network is fused with the geometric feature in a hierarchical structure. The proposed method was compared to other current algorithms for the 2 datasets, and the ten-fold cross-validation results show that the CK+ dataset is 96.46% accurate.

In a study by [19] researchers created a computationally efficient and scalable deep learning model utilizing CNN for autonomously identifying diabetic retinopathy (DR). It is a diabetic eye complication that causes impaired vision or blindness. For autonomously diagnosing DR, the researcher employed a computationally efficient and scalable deep learning model based on CNN. To boost accuracy, several preprocessing methods are employed, and a transfer learning strategy is utilized to accelerate the process. The investigation made use of the online fundus picture collection. Kaggle datasets are divided into five categories (none, mild, moderate, severe, or proliferative). The computer simulation generated a comparatively high F1 score of 93.2% for stage-based DR categorization as the conclusion of relevant performance criteria.

In this paper, the effect of the number of classes in the database on the classification accuracy of the CNN algorithm was studied. A database containing a large number of classes specific to facial expressions was chosen. The algorithm was applied to all 8 classes, then reduced the number to 4, and finally to 2. The results showed a clear difference in the performance of the algorithm.

II. Materials and Method

The proposed method involves creating a database of images that encompasses a wide range of classes, thereby achieving the research aim. The second step consists of processing these images for ease of use and then classifying them using the CNN algorithm. The working method is shown in (Fig. 1), and the steps below explain the mechanism in detail.

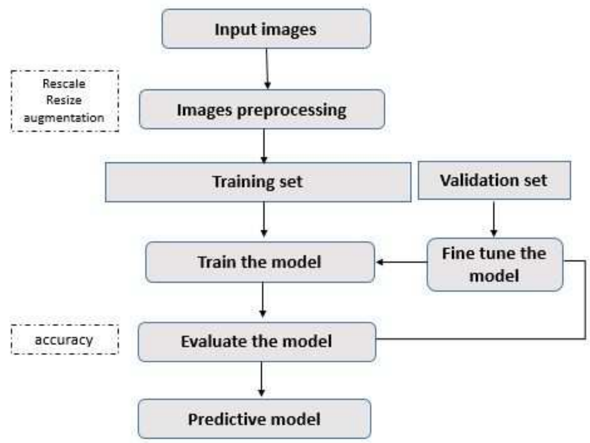

Fig. 1 Flowchart of the proposed work.

A. Dataset

The dataset obtained from AffectNet [20], The AffectNet collection comprises images gathered from the Internet by searching three search engines with 1250 emotion-related keywords in 6 languages. The existence of 8 distinct facial expressions (categorical model) and the strength of valence and arousal (dimensional model) are manually annotated in about half of the recovered pictures. The remaining images are automatically labeled with an average accuracy of 65% using a ResNext Neural Network trained on all hand-annotated training set samples.

The data consists of 8 labels: 1. Neutral, 2. Happy, 3. Sad, 4. Surprise, 5. Fear, 6. Disgust, 7. Anger, 8. Contempt. And the total of these pictures is approximately 27,000, divided into categories according to the following Table 1, and Figure 2 shows a sample of the categories.

TABLE I
Categories of Face Expressions.

Facial Expression	Number of images	Percentage of Total (%)
Neutral	5,132	18.6
Happy	5,043	18.3
Sad	3,430	12.4
Surprise	4,296	15.6
Fear	3,622	13.1
Disgust	2,660	9.6
Anger	3,638	13.2
Contempt	3,179	11.5
Total	27,000	100

The table presents the number of images for each facial expression category in a dataset of 27,000 images. The most common expressions are Neutral (5,132 images) and Happy

(5,043), each making up about 19% of the dataset. Disgust has the fewest images (2,660), followed by Contempt (3,179), which may indicate class imbalance. Other emotions like Sad, Surprise, Fear, and Anger are moderately represented, ranging from about 12% to 15%. This distribution suggests a relatively balanced dataset with slight underrepresentation in certain emotional categories.

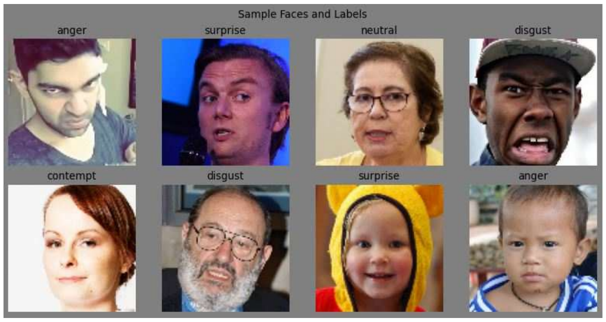

Fig. 2 Sample of the face categories.

B. Image Preprocessing

Image preprocessing improves the quality of input data and provides more helpful information for training models. Consequently, it can enhance the performance of deep learning models and reduce training time [21]. It enables several operations to be performed on an image before it is fed into a deep learning model. Among the main benefits of preprocessing are: noise reduction, dimensionality reduction to reduce the image size and therefore reduce training time and increase analysis speed, edge enhancement to improve the quality of the image's edges and improve the differentiation of objects and features within the image, color conversion to improve the appearance of the image, Increased data diversity by making changes to the input images, such as rotating, flipping, resizing, adjusting contrast, and distorting them in a known way [22]. This helps improve the model's ability to recognize different objects and features in the images [23].

In the proposed work, the train_data object was configured to apply various transformations to the training images, including rescaling, rotation, horizontal and vertical shifts, shearing, and zooming. These transformations can help the model learn to be more robust to variations in the training data.The validation_data object was configured only to rescale the validation images since data augmentation is not typically applied to the validation set.

C. Convolutional Neural Network (CNN)

CNN is a type of artificial neural network that is particularly well-suited for image processing and analysis tasks, it is based on the idea of using a series of convolutional layers to extract and transform features from input images, which are then used to perform classification or other tasks [24]. The basic architecture of a CNN consists of several layers, shown in (Fig. 3), including convolutional layers, pooling layers, and fully connected layers. The input to the network is a 2-dimensional array of pixel values that represents an image, and the output is a prediction or classification of the image based on its features [25].

A convolutional layer is often the first layer in a CNN, and it applies a collection of filters to the input picture to extract important information. Each filter performs a convolution operation on the input image, sliding over it in a process known as "convolution". The output of this layer is a set of feature maps, which represent different aspects of the image [26]. The next layer in a CNN is typically a pooling layer, which decreases the spatial dimensions of the feature maps by subsampling them. This reduces the number of parameters in the network and can help to prevent overfitting [27].

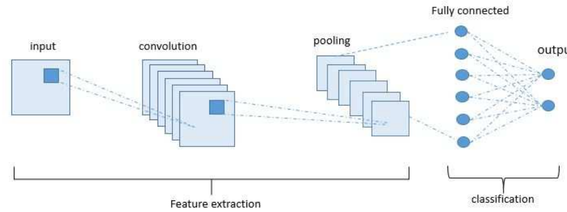

Fig. 3 The basic architecture of CNN algorithm

During training, the weights of the network are adjusted to minimize the error between the predicted output and the true output. This is typically done using a technique called backpropagation, which computes the gradients of the loss function with respect to the weights of the network and updates them accordingly [28], [29].

In this paper, 2 convolutional layers, 2 max pooling layers, and 2 dense layers were used as shown in (Fig. 4) and trained using the ImageDataGenerators train_generator and validation_generator created previously.

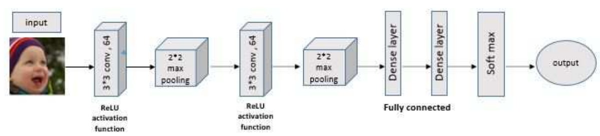

Fig. 4 Architecture of the proposed CNN

The architecture of the model is defined using the Sequential API of Keras. The first 2 layers are convolutional layers with 64 filters each, a filter size of (3x3), and a ReLU activation function. The input shape of the first layer is specified as (224, 224, 3). After each convolutional layer, a max pooling layer with a 2x2 pool size is added. The output of the max pooling layer is flattened and fed into a fully connected (dense) layer with 128 neurons and a ReLU activation function. Finally, a dense layer with output neurons and a softmax activation function is implemented to output the predicted class probabilities.

The Adam optimizer with a learning rate of 0.0001 and a categorical cross-entropy loss function is used to build the model. The accuracy metric is also supplied to track the model's performance during training. The model is trained using the fit method of the model object with 100 steps per epoch, 30 epochs, and a batch size of 64. The train_generator and validation_generator are used as input data for the model, and the dataset was divided into a training set of 75% and a validation set of 25%.

To achieve the goal of the research and to demonstrate the effect of increasing the number of classes on the resulting accuracy of the CNN algorithm, three experiments with the same settings were conducted. The first experiment included the use and classification of all eight classes in the database, which are (Happy, Sad, Neutral, Surprise, Fear, Disgust, Anger, Contempt), and extracted the accuracy. In comparison, in the second experiment 4 classes (Anger, Surprise, Sad, Happy) were chosen, and in the last experiment only 2 classes (Happy, Sad).

D. Evaluation Stage

Accuracy is a performance metric used to measure the effectiveness of a machine learning model [30]. It reflects the proportion of correct predictions made to total predictions produced by the model. In other words, accuracy is the fraction of properly identified occurrences in the dataset out of all instances.

III. Results and Discussion

This part discusses the research results and provides a thorough discussion. Google Colab was used to run all experiments, which is a Python-based integrated development environment used for deep learning, data analysis, and scientific programming based on the Jupyter Notebook environment. The results showed a significant difference in accuracy between all experiments in terms of training and evaluation. As shown in Table 2 and Fig. 5, all experiments were done on the same settings related to image processing and algorithm, using only 30 epochs.

TABLE II
PERFORMANCE OF THE CNN MODEL WITH DIFFERENT CLASSES.

Dataset	Training loss	Training accuracy	Validation loss	Validation accuracy
Dataset with 8 classes	1.784	35%	1.884	30%
Dataset with 4 classes	1.019	55%	1.319	44%
Dataset with 2 classes	0.3047	87%	0.3262	85%

Fig. 5 Visualization of the accuracy between the three experiments

When the model is trained on a dataset with 8 classes, the training accuracy and validation accuracy are relatively low at 35% and 30%, respectively. This suggests that the model is struggling to learn and distinguish between the various classes. However, when the number of classes in the dataset is reduced to 4, the accuracy of the model improves significantly, with a training accuracy of 55% and a validation accuracy of 44%. This suggests that reducing the number of classes in the dataset makes it easier for the model to distinguish between them.

Furthermore, when the number of classes in the dataset is reduced to 2, the accuracy of the model improves even further, with a training accuracy of 87% and a validation accuracy of 85%. This suggests that the model performs better when there are fewer classes to distinguish between. It is also possible to note the training loss and validation loss, which are metrics used to evaluate the performance of the CNN for the three groups, where the difference was evident, as shown in Figure 6. For the first dataset with eight classes, the training loss was 1.784 and the validation loss was 1.884. This indicates that the model struggled to learn the patterns in the data, and the validation loss was higher than the training loss, which suggests overfitting. For the second dataset with four classes, the training loss was 1.019 and the validation loss was 1.319. These values were lower than those of the first dataset, indicating that the model improved in learning the patterns in the data. However, the validation loss was still higher than the training loss, suggesting overfitting. For the third dataset with two classes, the training loss was 0.3047 and the validation loss was 0.3262. These values were the lowest among the three datasets, indicating that the model effectively learned the patterns in the data and generalized well to new, unseen data.

Overall, these results suggest that the number of classes in the dataset plays a crucial role in determining the accuracy of the CNN model, with a smaller number of classes generally leading to better performance.

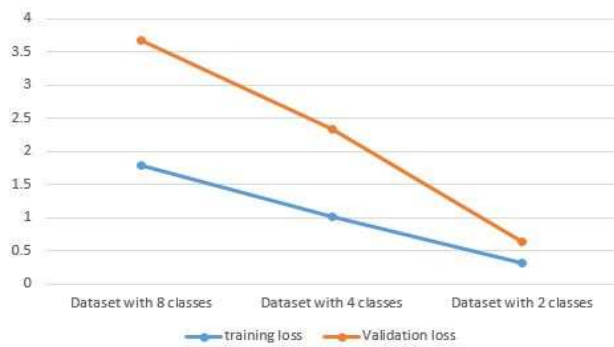

Fig. 6 Visualization of the training and validation loss between the three experiments

In addition to the number of classes, the quality of the dataset itself can be a reason for decreasing accuracy. For example, when looking at a sample of the images of the expression of surprise, as in Figure 7, we notice that not all expressions are clear. The algorithm may misunderstand them, leading to confusion in their performance. When there are more classes in the dataset, it can become harder for the CNN model to effectively learn and distinguish between them, especially if the classes are similar or have overlapping features. This can lead to increased confusion and misclassification, resulting in lower accuracy scores.

On the other hand, when the number of classes in the dataset is reduced, it can become easier for the CNN model to learn and distinguish between the classes, leading to higher accuracy scores, so A high-quality and diverse dataset with well-labeled and representative samples can help the model better learn and generalize patterns across different classes, leading to higher accuracy scores.

Fig. 7 Sample of the expression of surprise

IV. CONCLUSION

Convolutional neural networks are one of the most widely used deep learning algorithms in various fields, offering high efficiency. Most of the previous research focused on image processing methods and adjusting the number of convolutional layers in the algorithm to improve its performance. In this research, the effect of the database itself on the performance of the algorithm was studied in terms of the number of classes within the database. A database with many courses was used, and in three stages, the number of classes was reduced to note how it could affect its performance. The results demonstrate that reducing the number of classes in a dataset can lead to improved accuracy in the CNN algorithm, as the model is better able to distinguish between smaller numbers of classes.

REFERENCES

[1] C. Zuo et al., “Deep learning in optical metrology: a review,” Light: Science & Applications, vol. 11, no. 1, Feb. 2022, doi:10.1038/s41377-022-00714-x.

[2] M. Z. Alom et al., “A State-of-the-Art Survey on Deep Learning Theory and Architectures,” Electronics, vol. 8, no. 3, p. 292, Mar. 2019, doi: 10.3390/electronics8030292.

[3] T. Kattenborn, J. Leitloff, F. Schiefer, and S. Hinz, “Review on Convolutional Neural Networks (CNN) in vegetation remote sensing,” ISPRS Journal of Photogrammetry and Remote Sensing, vol. 173, pp. 24–49, Mar. 2021, doi: 10.1016/j.isprsjprs.2020.12.010.

[4] J. Gu et al., “Recent advances in convolutional neural networks,” Pattern Recognition, vol. 77, pp. 354–377, May 2018, doi:10.1016/j.patcog.2017.10.013.

[5] P. T. Q. Anh, D. Q. Thuyet, and Y. Kobayashi, “Image classification of root-trimmed garlic using multi-label and multi-class classification with deep convolutional neural network,” Postharvest Biology and Technology, vol. 190, p. 111956, Aug. 2022, doi:10.1016/j.postharvbio.2022.111956.

[6] M. M. Yusro, R. Ali, and M. S. Hitam, “Comparison of Faster R-CNN and YOLOv5 for Overlapping Objects Recognition,” Baghdad Science Journal, vol. 20, no. 3, p. 0893, Jun. 2023, doi:10.21123/bsj.2022.7243

[7] A. K. Sharma et al., “Dermatologist-Level Classification of Skin Cancer Using Cascaded Ensembling of Convolutional Neural Network and Handcrafted Features Based Deep Neural Network,” IEEE Access, vol. 10, pp. 17920–17932, 2022, doi:10.1109/access.2022.3149824.

[8] F. S. Hanoon and A. H. Hassin Alasadi, “A modified residual network for detection and classification of Alzheimer’s disease,” International Journal of Electrical and Computer Engineering (IJECE), vol. 12, no. 4, p. 4400, Aug. 2022, doi: 10.11591/ijece.v12i4.pp4400-4407.

[9] H. S. Das, A. Das, A. Neog, S. Mallik, K. Bora, and Z. J. F. i. G. Zhao, "Breast cancer detection: Shallow convolutional neural network against deep convolutional neural networks based approach,", Frontiers, vol. 13, p. 1097207, 2023.

[10] M. A. H. Akhand, S. Roy, N. Siddique, M. A. S. Kamal, and T. Shimamura, “Facial Emotion Recognition Using Transfer Learning in the Deep CNN,” Electronics, vol. 10, no. 9, p. 1036, Apr. 2021, doi:10.3390/electronics10091036.

[11] Y. Cao, W. Zhou, M. Zang, D. An, Y. Feng, and B. Yu, “MBANet: A 3D convolutional neural network with multi-branch attention for brain tumor segmentation from MRI images,” Biomedical Signal Processing and Control, vol. 80, p. 104296, Feb. 2023, doi:10.1016/j.bspc.2022.104296.

[12] J. Alfred Daniel et al., "Fully convolutional neural networks for LIDAR–camera fusion for pedestrian detection in autonomous vehicle,", Multimedia Tools and Applications, pp. 1-24, 2023.

[13] M. Ahmadi, A. Sharifi, M. Jafarian Fard, and N. Soleimani, “Detection of brain lesion location in MRI images using convolutional neural network and robust PCA,” International Journal of Neuroscience, vol. 133, no. 1, pp. 55–66, Feb. 2021, doi:10.1080/00207454.2021.1883602.

[14] H. Chen, L. Wu, J. Chen, W. Lu, and J. Ding, “A comparative study of automated legal text classification using random forests and deep learning,” Information Processing & Management, vol. 59, no. 2, p. 102798, Mar. 2022, doi: 10.1016/j.ipm.2021.102798.

[15] S. Soni, S. S. Chouhan, and S. S. J. A. I. Rathore, "TextConvoNet: A convolutional neural network based architecture for text classification,", Applied Intelligence, vol. 53, no. 11, pp. 14249-14268, 2023.

[16] V. R. J. I. J. o. C. Allugunti, Programming and D. Management, "A machine learning model for skin disease classification using convolution neural network,", International Journal of Computing, Programming and Database Management, vol. 3, no. 1, pp. 141-147, 2022.

[17] A. S. Musallam, A. S. Sherif, and M. K. Hussein, “A New Convolutional Neural Network Architecture for Automatic Detection of Brain Tumors in Magnetic Resonance Imaging Images,” IEEE Access, vol. 10, pp. 2775–2782, 2022, doi:10.1109/access.2022.3140289.

[18] J.-H. Kim, B.-G. Kim, P. P. Roy, and D.-M. Jeong, “Efficient Facial Expression Recognition Algorithm Based on Hierarchical Deep Neural Network Structure,” IEEE Access, vol. 7, pp. 41273–41285, 2019, doi: 10.1109/access.2019.2907327.

[19] N. M. A.-M. M. Al and R. S. J. I. Khudeyer, "ResNet-34/DR: a residual convolutional neural network for the diagnosis of diabetic retinopathy,", Informatica, vol. 45, no. 7, 2021.

[20] A. Mollahosseini, B. Hasani, and M. H. Mahoor, “AffectNet: A Database for Facial Expression, Valence, and Arousal Computing in the Wild,” IEEE Transactions on Affective Computing, vol. 10, no. 1, pp. 18–31, Jan. 2019, doi: 10.1109/taffc.2017.2740923.

[21] A. M. Hasan, A. F. Qasim, H. A. Jalab, and R. W. Ibrahim, “Breast Cancer MRI Classification Based on Fractional Entropy Image Enhancement and Deep Feature Extraction,” Baghdad Science Journal, vol. 20, no. 1, p. 0221, Feb. 2023, doi:10.21123/bsj.2022.6782.

[22] R. Sarki, K. Ahmed, H. Wang, Y. Zhang, J. Ma, and K. Wang, “Image Preprocessing in Classification and Identification of Diabetic Eye Diseases,” Data Science and Engineering, vol. 6, no. 4, pp. 455–471, Aug. 2021, doi: 10.1007/s41019-021-00167-z.

[23] R. G, “A Study to Find Facts Behind Preprocessing on Deep Learning Algorithms,” March 2021, vol. 3, no. 1, pp. 66–74, Apr. 2021, doi:10.36548/jiip.2021.1.006.

[24] W. Samek, G. Montavon, S. Lapuschkin, C. J. Anders, and K.-R. Muller, “Explaining Deep Neural Networks and Beyond: A Review of Methods and Applications,” Proceedings of the IEEE, vol. 109, no. 3, pp. 247–278, Mar. 2021, doi: 10.1109/jproc.2021.3060483.

[25] Z. Li, F. Liu, W. Yang, S. Peng, and J. Zhou, “A Survey of Convolutional Neural Networks: Analysis, Applications, and Prospects,” IEEE Transactions on Neural Networks and Learning Systems, vol. 33, no. 12, pp. 6999–7019, Dec. 2022, doi:10.1109/tnnls.2021.3084827.

[26] S. Cong and Y. J. A. I. R. Zhou, "A review of convolutional neural network architectures and their optimizations,", Artificial Intelligence Review, vol. 56, no. 3, pp. 1905-1969, 2023.

[27] A. Khan, A. Sohail, U. Zahoora, and A. S. J. A. i. r. Qureshi, "A survey of the recent architectures of deep convolutional neural networks,", Artificial Intelligence Review, vol. 53, pp. 5455-5516, 2020.

[28] D. Negrov, I. Karandashev, V. Shakirov, Yu. Matveyev, W. Dunin-Barkowski, and A. Zenkevich, "An approximate backpropagation learning rule for memristor based neural networks using synaptic plasticity," Neurocomputing, vol. 237, pp. 193–199, May 2017, doi: 10.1016/j.neucom.2016.10.061.

[29] S. Kiranyaz, O. Avci, O. Abdeljaber, T. Ince, M. Gabbouj, and D. J. Inman, "1D convolutional neural networks and applications: A survey," Mechanical Systems and Signal Processing, vol. 151, p. 107398, Apr. 2021, doi: 10.1016/j.ymssp.2020.107398.

[30] V. H. Phung and E. J. Rhee, "A High-Accuracy Model Average Ensemble of Convolutional Neural Networks for Classification of Cloud Image Patches on Small Datasets," Applied Sciences, vol. 9, no. 21, p. 4500, Oct. 2019, doi: 10.3390/app9214500.