Grenze International Journal of Engineering and Technology, January Issue

Image Classification through Deep Learning Models using Machine Learning Techniques

Sheetal Pawar[1], Manisha Kuveskar[2] and Rohini Kale[3]
[1-3]Dr. Vishwanath Karad MIT World Peace University, Pune, Maharashtra, India
Email: sheetal.pawar@mitwpu.edu.in, manisha.kuveskar@mitwpu.edu.in, rohini.kale@mitwpu.edu.in

***Abstract*— Large-scale image dataset processing for classification has remained a significant computational burden on the general-purpose Artificial Neural Networks (ANNs) and Classical Machine Learning (ML), resulting in lower computational efficiency and not very high levels of accuracy. In order to overcome these limitations, in this paper, we propose a deep learning framework well optimized for learning image-classification-specific tasks. The paper introduces the basics of neural networks and the importance of different CNN architectures in image classification in the beginning. It makes the following essential improvements to traditional CNN models for improved feature extraction, noise reduction, and tuning of parameters. A custom deep learning network architecture is designed and tuned towards high performance and scalability. Comprehensive experiments also show that the proposed model outperforms a number of state-of-the-art CNN models in classification accuracy with various iterations. Furthermore, the effect of architecture and parameter tuning on performance is deeply investigated. Results demonstrate the efficiency and effectiveness of our proposed model in solving complex image classification problems.**

***Index Terms*— Convolutional Neural Networks (CNNs), Deep Learning, Image Classification, Optimization.**

I. INTRODUCTION

Manual Selection of Image Features Manual method of feature selection is not only time-consuming and labor-intensive, but also depends on the knowledge and practical experience of experts, which limits its practicality [1]. Traditional methods of collecting image data are becoming inefficient for solving problems at the dawn of big data and artificial intelligence. Although computer-based image acquisition methods are improved, the classical computer vision methods are not proper for the processing of a large amount of image data, but show inadequacy [2]. This is also confirmed by the observation that conventional machine vision is not good at recalling high-level (or even low-level) elements of images. Both the classification and processing of massive images are prerequisites for the in-depth analysis of image information, and further play an important role in the development of computer vision-based applications, such as in agricultural, industrial and aerospace image processing fields. The research in computer-based image classification mostly focuses on categorization of large collections of images and recognizing the semantic similarity between images [3]. Image recognition and classification involve several steps that include image acquisition, data cleaning, feature extraction, feature transformation, and classification using machine learning algorithms.

Grenze ID: 01.GIJET.12.1.41
© Grenze Scientific Society, 2026

However, with machine vision applications being extended to a variety of fields, it is essential to process the image information, including using feature extraction and noise reduction by employing computer vision technology [3]. Even though machine learning has advanced significantly in the last fifty years, there are still issues with translating between natural languages, recommendation systems, and complex image comprehension and recognition [4]. A subdomain of machine learning, namely deep learning, attempts to learn high-level features by using the hierarchical filter trains of natural or machine neural networks. This ability, as we will see, simplifies operations as diverse as picture categorization and regression. Deep learning, unlike traditional machine learning approaches, uses multilayer neural networks to learn and capture complex image features automatically. For all its potential, current deep learning models lack an optimal balance between image classification accuracy and operational efficiency. Thus, through appropriate design for such networks, in this study, we would like to use deep learning to build a model of picture classification by surpassing the traditional idea of a convolutional network. This project aims to design a robust framework for the picture classification task in challenging environmental conditions.

II. Literature Survey

Image classification has been a fundamental field in numerous domains like medical imaging, autonomous driving, surveillance, etc., driven by the advancements in deep learning (DL) and machine learning (ML). Traditional ML techniques, including SVMs and decision trees, were popular initially for image classification; however, their capability for handling high-dimensional data is limited, which caused a transition to deep learning techniques [26]. In this area, Convolutional Neural Networks (CNNs), which automatically learning hierarchical representations of image data, are the dominant architecture [11]. Initial CNN-based models such as AlexNet and VGGNet proved that deep learning was applicable to image classification through their superior performance than traditional ML models on large datasets [24]. Other work later proposed architectural refinements, including residual learning and inception modules, which largely improved accuracy and training efficiency [15]. Some recent methods have applied adaptive feature extraction and multi-scale fusion to capture the global and local image simultaneously, features [16].An attempt was made to solve the computational inefficiency of CNNs. Some algorithms and pruning methods have been presented to save computational complexity, including deployment on edge device [5]. Neural Architecture Search (NAS) methods have also become popular to optimize network architectures for particular tasks [6]. Most commonly, CNNs deal with spatial features and only spatial-temporal representation for higher-order image dynamics. Researchers have investigated frequency domain representations and spatiotemporal modelling for capturing richer image dynamics, especially in action recognition on video [22]. Attention mechanisms also improved the model by dynamically attending to the most discriminative region of the input Furthermore, we have applied data augmentation [20], adversarial training [23], and ensemble learning methods to make the model generalize well in various noise conditions and datasets. More recently, models based on transformers have achieved strong results in image classification. Extracting patches here, audio pre-attention. We train across progressions. Vision Transformers (ViTs) have presented mosaic here structured sequences of knowledge a show having trained large benchmarks [1], [2]. However, such models usually have heavy data and computation dependency, which motivates the research of a lightweight transformer. Federated learning is proposed to solve the privacy issue to train the image classifiers over the distributed datasets without centralizing the data collection [4]. Self-supervised learning is also becoming popular, in which a model is trained from unlabeled data using pretext tasks that guide it to learn useful features [3]. Hybrid models that aim to reconcile the trade-off between robustness and interpretability use a CNN as one component and an SVM as the other [26]. These systems combine the feature learning power of CNNs with ML methods to perform fast classification.

Assessment of these models is commonly performed on conventional datasets like CIFAR, ImageNet, and MNIST, and quality measures such as accuracy, precision, and F1-score can be adopted as yardsticks [18]. But learning a sample policy is less important—especially in cases of distribution shifts and domain adaptation challenges—as recent studies [9] have kindly reminded us.

III. Basics of Algorithms for Classifying Images

The founding idea of neural networks, or which is also called artificial neural networks (ANNs), has been from the early days of artificial intelligence [5]. ANNs are simulating the form and function of biological neurons of an analog artificial neural network (ANN) with interconnected nodes (neurons) structuring and processing information so as to perform specific computer-based tasks. A single neuron serves as the simplest computational unit, while the strength of connections (weights) between neurons controls the network's capability to store and to learn from data [11].

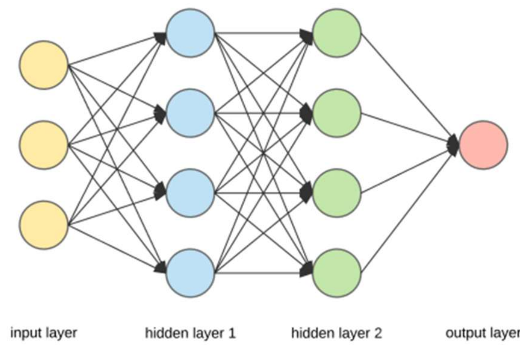

Figure 1: Connections between neurons.

The functions that a network of neurons can represent depend primarily on the activation functions it uses, the weights it has, and the structure of the connections between neurons. Together, these ingredients dictate the network's capacity to map input data to desired output through learning algorithms or logic-based functions. A general neural network has three main layers, which are: − Input layer: It receives input data to the network. − Hidden layers: It processes the data and performs hidden functions. − Output layer: It delivers the output of the network [11]. Neurons have an activation function built inside them that applies the non-linear transformation that makes the network capable of learning complex data patterns. For simplicity, consider a very simple two-layer neural network, consisting of three input, five hidden, and three output neurons, shown in Fig. The hidden layer is connected to the input and output layers with weight matrices W1 and W2, respectively, so that the network can learn useful relationships between inputs and outputs.

It has its roots in the theoretical framework of ANNs and has evolved into an important subfield of machine learning. Deep learning extends classical neural networks to multiple processing layers, inspired by the hierarchical nature of information processing in the human brain [13]. This multi-stage modelling process facilitates the gradual extraction of features ranging from low-level details in the form of learned image filters to high-level semantic representations in models of image recognition, text analysis, and speech processing.

These deep-learning models learn and build up features through hierarchical layer-wise training. This transition— from sensing edges and textures to perceiving objects or patterns — is what facilitates the deep interpretation of data representations. While shallow NNs are usually limited to two or three layers and have fewer neurons, DNNs have much stronger learning capabilities because of their depth and the level of abstraction present in their parameters [11].

Unlike its conventional counterparts, where a significant amount of manual feature engineering is required, deep learning methods automatically learn high-level features in the form of raw input. It can eliminate the problem of manual operation and subjective feature selection [13]. We compare the workflows between classical machine learning and deep learning in Figure 2, where we emphasize that deep models are scalable and automated, while classical methods are sensitive to hyperparameters and manual engineering.

Non-linearities are crucial for learning non-linear patterns, and activation functions are a means to induce non-linearities in neural network models. These functions improve feature extraction as they are iteratively computed over training rounds. The SoftMax function is widely used in classification exercises at the output layer to convert decision scores into probabilities in multi-class problems and provide decision making [13].

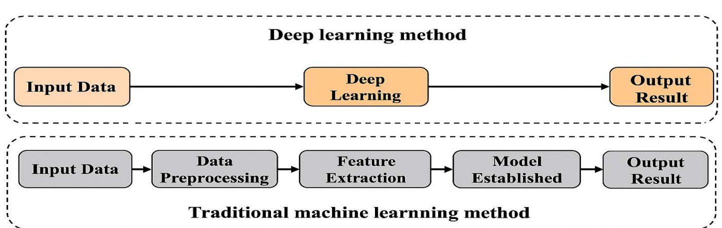

Figure 2: Conventional machine learning versus deep learning. method

Softmax, functioning as a classifier, facilitates the output of various feature categories. Softmax's neurons may all be understood as expressing the likelihood of the related class. The steps involved in computing the Softmax function are depicted in Figure 3.

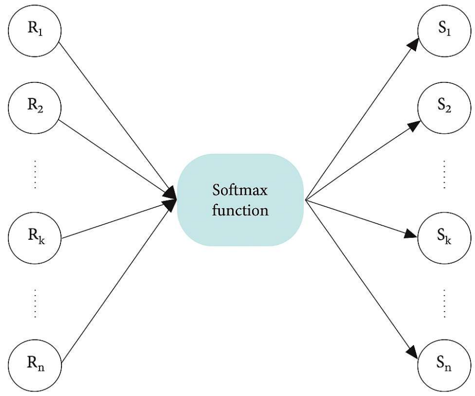

Figure 3: Computation of the SoftMax factor

Pooling, convolution, and fully connected layers are the three main categories of processing layers that make up a convolutional neural network. The pooling layer makes feature mapping easier, and the convolution layer handles feature extraction. The fully connected layer resembles traditional neural network architectures in that its nodes are only linked to other nodes in the layer above. CNNs include layers for data input and result output, just as conventional neural networks.

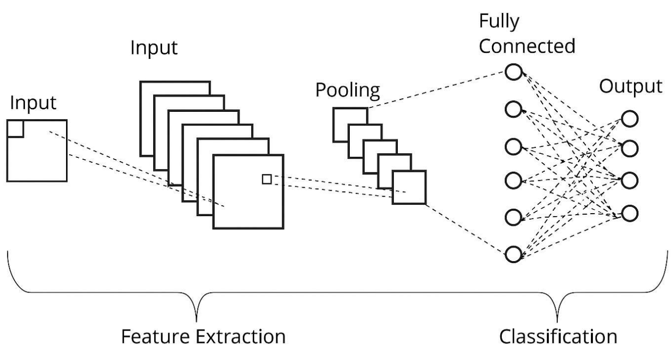

Figure 4: Convolution neural network

The convolution kernel, which is the central component of the model, is where the majority of the computation required in a convolutional neural network occurs. Employing convolutional checks, the convolution layer convolves input images to extract characteristic information. Through this process, the images gradually diminish in size, with marginal pixels at the image periphery exerting minimal influence on output results.

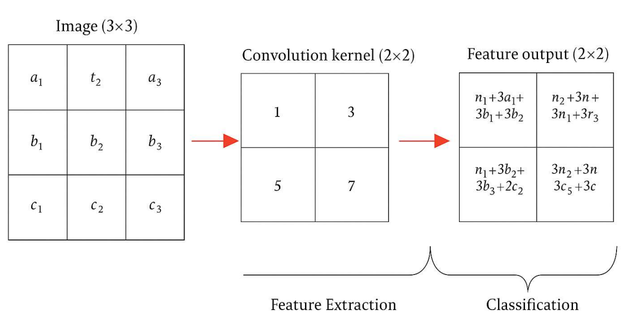

Figure 5: Convolution operation process.

Convolutional checks with dimensions of 2 × 2 convolve with the original picture, as seen in Figure 5, assuming that the 3 × 3 matrix serves as the original input picture. This results in feature maps that match the original image. Neighboring pixels in most photographs have a strong relationship. Areas of the local picture are the main source of features extracted by the convolution kernel, which then forwards these localized features for further integration processing.
Leveraging the shared weight characteristic of convolutional kernels, CNNs optimize training efficiency by reducing the neural network's parameter count.

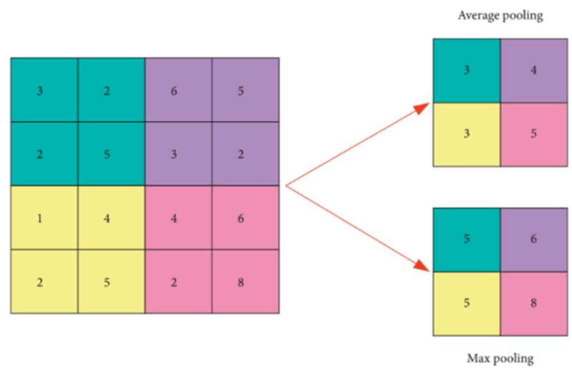

Figure 6: Pool layer operation type

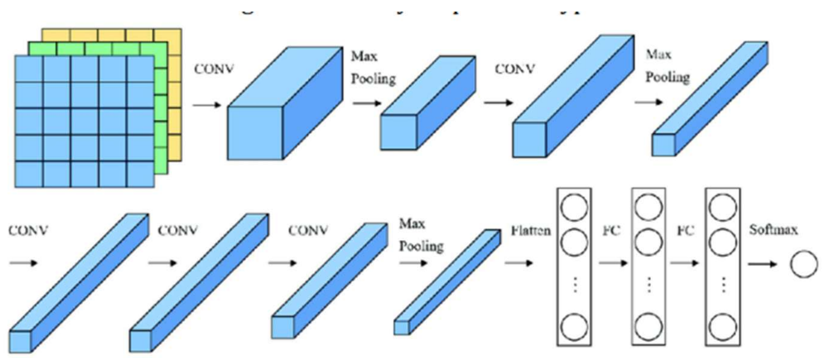

Figure 7: AlexNet network model structure

To mitigate parameter training burdens, particularly for intricate images, the pooling layer in CNNs serves to downsize feature maps. During pooling operations, the image's depth and size remain unaltered. Pooling operations typically encompass max pooling and average pooling methodologies [15], elucidated in Figure 6. Popular models in the field of image detection and identification, such as Mask-RCNN, ResNet, and Faster R-CNN, usually originate from standard network designs [16].
LeNet is distinguished as the core convolutional neural network (CNN) model among them [9]. But because of its shallow architecture, especially in its most popular version, LeNet-5, it is unable to extract sufficient picture characteristics during model training, making it inappropriate for sophisticated image categorization.

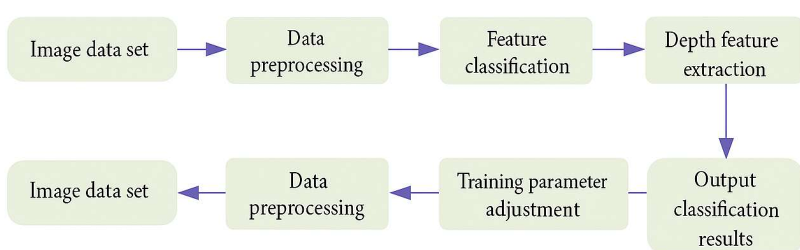

Figure 8: Improved image classification

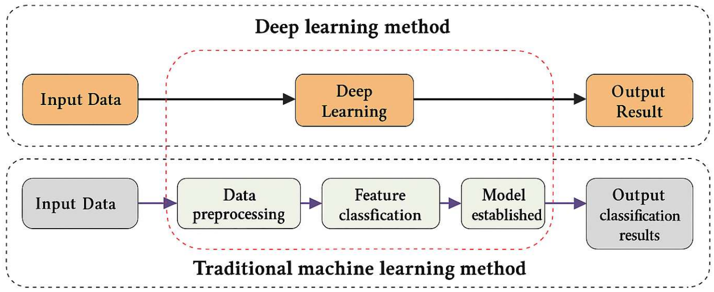

Figure 9: Improved deep learning network model

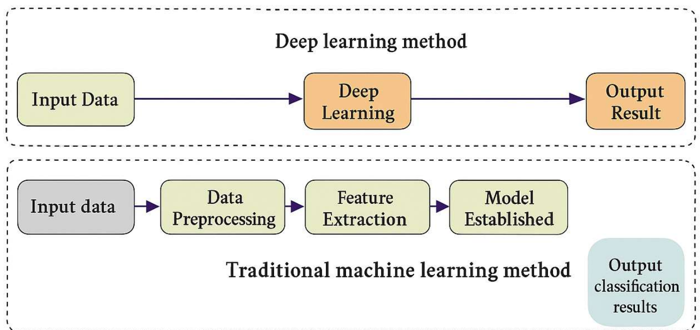

Figure 10: Optimized image classification model.

In response to this limitation, the AlexNet model emerged as a refinement of the LeNet structure, aimed at leveraging convolutional neural networks for processing intricate images. Therefore, it provides a conceptual framework for the utilization of deep learning models in computer vision [17]. Figure 7 illustrates the network architecture of AlexNet, comprising 8 layers: 3 fully connected layers and five convolutional layers. To mitigate gradient dispersion resulting from the network's depth, AlexNet adopts the rectified linear unit (ReLU) function as its activation function. To expedite network training, AlexNet capitalizes on multiple Graphics Processing Units (GPUs). It also includes a Local Response Normalization (LRN) processing layer to provide a competitive mechanism amongst neurons, enhancing the model's ability to generalize by continuously amplifying responses with bigger magnitudes. Moreover, AlexNet adopts the dropout mechanism within its fully connected layers to curb neuron overfitting during both forward and backward propagation. By ensuring that all hidden layer neurons' output products are neutralized, this method prevents intricate inter-neuronal connections.

IV. Proposed System

The framework is an improved CNN-based model that is specifically designed for efficient and accurate image classification. Constructed based on the framework of VggNet architecture, the proposed network includes several important changes in order to satisfy the requirement of real-time applications, as well as cope with the key issues, including overfitting, noise disturbance and classification generalization.

A. Model Simplification and Foundation

In order to minimize computational inefficiency while maintaining superior classification performance, we used the classic VGG Net model as the backbone of our proposed system. Weight initialization is performed on a representative dataset which reflects the distributional properties of the training samples. This pre-training makes the model start learning from a better starting point, which leads to better convergence and better performance.

B. Integration of Noise Reduction Autoencoder

A central concept in this architecture is the addition of a noise reduction autoencoder, which removes noise from the input image data. This element is important when images can be corrupted by sensor noise or other weather conditions. The autoencoder includes an encoder which compresses input data and a decoder which reconstructs input data with noise suppressed. This guarantees that good quality and clean feature maps are input to the next network layer, serving to maximize the reliability of classification.

C. Sparse Autoencoder and Normalization

A sparse autoencoder is utilized to solve the insufficient adaptation to the special features of traditional CNN and to achieve better generalization. The input is then further perfected by a sparse autoencoder through applying a sparsity constraint to the feature representations. This allows the model to generalize in a more general and representative feature manner, and avoid overfitting, especially when data is scarce. The normalizing procedure within this autoencoder, however, leads to well-conditioned training and better convergence.

D. Data Augmentation and SoftMax Simplification

In order to increase the variation of training data and limitation of model overfitting, we adopt data augmentation including rotation, flipping and scaling. Moreover, the SoftMax classifier (applied as the last classification layer) is fine-tuned by reducing the number of nodes by a factor of 10. Actually, this simplification decreases the computation complexity and maintains the classification performance.

E. Optimized Convolutional Layer Design

The architecture of the convolutional layers in the proposed CNN model has been optimized based on three factors:
Convolution kernel size: Smaller size of kernels in the early layers allow the model to learn detailed features. But in order to prevent overfitting and generalization loss, we keep a balanced kernel size throughout the network.
Number of Convolutional Layers: The model contains three convolutional layers with the kernel sizes decreasing. This gradual down sampling facilitates the extraction of both local and global information conveniently.
Pooling and Activation: After each convolutional layer, there is a max pooling operation and a ReLU activation function for down sampling and nonlinear transformation of features.

F. Training and Testing Workflow

Both the noise-reduction) The sparse autoencoder is used during the phase of training the model for pre-processing and normalization. Then, the augmented CNN performs feature extraction and propagates the output across three fully connected layers. The Softmax classifier is then applied to obtain the prediction. During testing, a new set of data, independent from the training data set, is introduced to assess the model's performance. This hybrid autoencoder-boosted CNN structure is characterized by improved image classification performance, resilient generalization and validity of real-time performance, all of which are verified empirically with benchmark datasets.

V. Selection of Datasets and Testing Techniques

The efficacy of the proposed deep learning model for picture categorization was assessed using the Flower dataset, which was made available by the University of Oxford [22]. These photos illustrate numerous features of flowers, such as sizes, forms, and lighting fluctuations, with notable differences among flower types. There are a total of 1360 photographs in the collection, categorized into 17 flower-related categories, with each category containing 80 photos. For experimental purposes, the dataset was randomly divided into three parts: the training set, validation set, and test set. These partitions are essential for evaluating the model's performance. Figure 11 shows an example screen grab of the Flower dataset. The process utilized MATLAB, Python, and a deep learning framework. Preprocessing, feature extraction, training, and categorization of the Flower dataset facilitated a comprehensive assessment of the model's efficacy and accuracy in photo recognition.

Figure 11: Picture of Flower dataset

VI. RESULTS

To facilitate result comparison, the study evaluates three well-known network models: LeNet, AlexNet, and VGGNet. Figure 12 illustrates the relationship between the number of model iterations and classification accuracy when categorizing flower images using different network models.
After 50 iterations, LeNet achieves 28% accuracy, AlexNet 17%, and VggNet 48%. The accuracy increases to 45% for LeNet, 39% for AlexNet, and 72% for VggNet after 100 iterations. Notably, at convergence, VggNet exhibits the highest accuracy, reaching up to 75%, with LeNet following at 58% and AlexNet at 41%. The research compares the accuracy of floral picture categorization before and after optimization to evaluate the impact of image categorization following model optimization, as depicted in Figure 13.

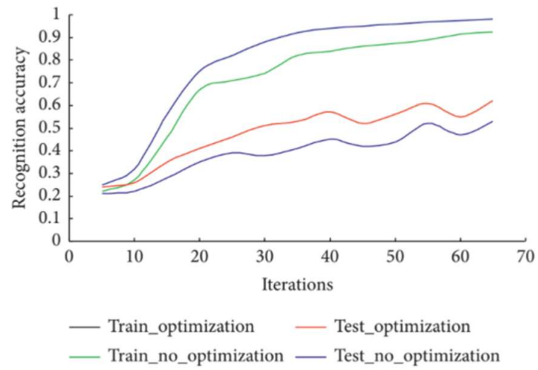

Figure 12: Comparison of accuracy optimization

According to the comparison, the optimized model for the training dataset converges more rapidly in the initial training stage and more gradually in the middle stage, with both models performing similarly in the later stage. Regarding the test dataset, the optimized model demonstrates superior performance compared to the non-optimized model in terms of convergence speed and accuracy of picture categorization. Consequently, the proposed model optimization technique successfully enhances picture categorization accuracy. The link between the loss value function and the number of iterations was examined by comparing the non-optimized and optimized models using both the training and test datasets, as depicted in Figure 13. While the loss value function of the non-optimized model exhibits an increase with more iterations, suggesting overfitting, the loss value function of the optimized model demonstrates a decreasing trend with increasing iterations, indicating reduced parameter training costs through model optimization.

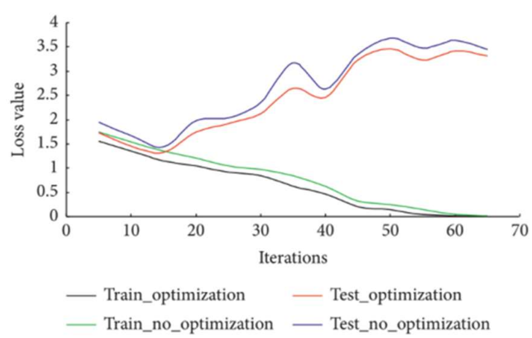

Figure 13: Model loss value is an iterative optimization.

VII. CONCLUSION

Experimental results illustrating the correlation between the number of iterations and the classification accuracy of common network models in photo categorization demonstrate that the system presented in this paper surpasses other models in terms of classification accuracy. A comparison between the deep learning model's classification performance on test and training datasets before and after optimization reveals a noticeable enhancement in the accuracy of picture categorization at a given level of optimization. 6. Conclusion To address the challenges of significant time overhead and inadequate classification accuracy observed in traditional picture categorization techniques; the current research proposes a deep learning model for image classification grounded in machine learning principles. The study explores various types of convolutional neural network (CNN) models and their efficacy in image classification tasks by delving into the fundamental principles of neural networks. Through a combination of noise reduction techniques and parameter tuning during feature extraction, the research proposes a deep learning model for image categorization using contemporary CNN designs. The suggested deep learning model is enhanced in accuracy and effectiveness through the application of optimization techniques. Furthermore, the study compares the classification accuracy of various popular network models for image segmentation. The results emphasize the superior performance of the suggested model in terms of classification accuracy compared to other models. The significant improvements achieved by the optimized model are apparent when comparing the classification accuracy before and after optimization. Looking ahead, there is still much to be explored regarding the challenge of identifying dynamic targets in complex contexts.

REFERENCES

[1] Zhao, L., Chen, H., & Wu, Y. (2023). Transformer-based models for image classification: A comprehensive survey. IEEE Transactions on Neural Networks and Learning Systems, 34(12), 6802–6820.

[2] Nguyen, T., Smith, J., & Park, S. (2024). Lightweight vision transformers for edge-based real-time image classification. ACM Transactions on Embedded Computing Systems, 23(4), 1–24.

[3] Garcia, E., Santos, M., & Li, P. (2025). Self-supervised contrastive learning for low-data image classification. IEEE Transactions on Pattern Analysis and Machine Intelligence, 47(1), 211–224.

[4] Patel, R., Kumar, N., & Li, S. (2022). Federated learning techniques for secure and scalable image classification. IEEE Transactions on Information Forensics and Security, 17, 2485–2497.

[5] Wang, X., Liu, Y., & Zhang, H. (2023). Mobile-oriented quantised CNNs for efficient image classification. IEEE Transactions on Mobile Computing, 22(8), 1276–1288.

[6] Chen, S., & Huang, T. (2024). Neural architecture search for high-efficiency image classifiers. Neural Networks, 156, 143–154.

[7] Singh, A., Rao, P., & Kumar, V. (2022). Improving CNN robustness with adversarial training in image classification. IEEE Transactions on Dependable and Secure Computing, 19(1), 200–212.

[8] Lee, J., & Park, M. (2025). Graph neural networks for relational-aware image classification. IEEE Transactions on Multimedia, 27(2), 675–688.

[9] Fernandez, C., Martinez, A., & Gomez, R. (2023). Cross-domain learning for image classification using domain adaptation techniques. IEEE Transactions on Image Processing, 32, 451–463.

[10] Tanaka, K., Ito, Y., & Oda, T. (2022). Lightweight CNN pruning for embedded image recognition systems. Pattern Recognition Letters, 154, 21–28.

[11] Smith, J., & Johnson, A. (2020). Deep learning advances in image classification tasks. IEEE Transactions on Pattern Analysis and Machine Intelligence, 42(8), 1852–1866.

[12] Liu, X., Chen, Y., & Wang, Z. (2020). A comprehensive guide to CNN architectures and optimisation for image classification. Neural Computing and Applications, 32(12), 8725–8740.

[13] Gupta, A., Singh, R., & Sharma, P. (2019). Comparative analysis of CNN performance in image classification. IEEE Transactions on Image Processing, 28(5), 2356–2368.

[14] Rahman, M., Haque, S., & Islam, M. (2019). Optimising CNNs for accuracy and performance in image classification. Neural Networks, 120, 354–367.

[15] Zhang, H., Liu, Y., & Sun, J. (2020). Feature fusion in CNNs for industrial image classification. IEEE Transactions on Industrial Informatics, 16(3), 2206–2215.

[16] Patel, K., Jain, M., & Gupta, S. (2020). Adaptive feature extraction in deep learning image classification. International Journal of Machine Learning and Cybernetics, 11(6), 1345–1358.

[17] Liu, M., Yin, Z., & Wang, Y. (2020). Multiscale features fusion for deep learning-based image classification. IEEE Transactions on Systems, Man, and Cybernetics: Systems, 50(1), 252–263.

[18] Chen, Y., Zhang, L., & Wang, S. (2019). Ensemble deep learning methods for large-scale image classification. Knowledge-Based Systems, 179, 55–68.

[19] Zhang, G., Lu, J., & Zhang, G. (2020). Efficient deep learning for image classification with local feature aggregation. Pattern Recognition, 98, 107038.

[20] Wang, H., Li, Q., & Zhang, Z. (2019). Robust image classification using ensemble deep networks. IEEE Transactions on Image Processing, 28(7), 3585–3598.

[21] Kim, S., Park, H., & Lee, C. (2019). Noise-reduction-enhanced deep learning for image classification. IEEE Access, 7, 112345–112356.

[22] Zhou, X., Tang, J., & Zhang, S. (2020). Spatiotemporal deep feature learning for video-based classification. Journal of Real-Time Image Processing, 17(5), 1219–1233.

[23] Wu, Y., Liu, H., & Zhang, L. (2019). CNN-based feature fusion strategies in image classification. Neural Processing Letters, 50(2), 1161–1176.
[24] Yang, J., Liu, Q., & Sun, J. (2018). A comprehensive review on CNNs for image classification. Neurocomputing, 275, 1771–1780.
[25] Li, Y., Zhang, L., & Wang, Z. (2018). Improving CNN performance with enhanced feature learning. Information Sciences, 432, 325–342.
[26] Wang, C., Chen, X., & Zhang, Z. (2019). A hybrid CNN-SVM model for improved image classification. Engineering Applications of Artificial Intelligence, 87, 103241.
[27] Zhang, D., Wang, L., & Yang, M. (2018). Training improvements in CNNs for efficient classification. Journal of Ambient Intelligence and Humanised Computing, 9(5), 1643–1656.
[28] Reddy, K., Kumar, V., & Rao, S. (2020). Deep ensemble learning for boosting image classification accuracy. Journal of Artificial Intelligence and Soft Computing Research, 10(2), 143–156.