**Journal of Science and Engineering Applications**

Contents are available at https://jsea.iujournals.com

# Optimizing Residual Networks for Image Classification with Extended Training and Standard Augmentation

Hadeel Talib Mangi

*Computer science and information technology department, College of science, University of Hilla, Babylon, Iraq, 51011*

hadeel_talib@hilla-unc.edu.iq

**ARTICLE INFORMATION**

Received date: 26-01-2026
Revised date: 3-03-2026
Accepted date: 25-4-2026

**Keywords**

Image Classification
Deep Learning
CNN
Residual Learning
Data Augmentation

**ABSTRACT**

Image classification is considered an essential task in computer vision and artificial intelligence due to its variety of applications in computer vision and other areas. Despite the fact that deep neural networks have demonstrated impressive performance in image classification and other image-related tasks, their efficiency in image classification and other image-related tasks remains limited by overfitting, poor generalization, and the complexity of using deep neural networks. Despite the fact that using deeper neural networks in image classification and other image-related tasks might improve efficiency, this might not always be the case. This paper proposes an efficient and effective image classification method using a simple neural network. The main aim of this paper is to show that image classification efficiency can be improved using effective training techniques. In order to overcome these challenges, the present research proposes an effective image classification approach that does not require the use of deeper models. Instead, the proposed approach aims to maximize the efficiency of the training process to improve the generalization capability of the model, while maintaining the simplicity of the model. In essence, the proposed approach aims to prove that the training process can be effective enough to produce comparable results without the need for deeper models. The experimental results were able to attain a test accuracy of 90.92%, which is better than the comparative models and proves the idea that the performance of the network can be enhanced without deepening the network.

## 1.Introduction

Image classification is one of the basic computer vision problems that involves mapping an image to a class label based on the visual information present in the image. Image classification acts as a building block for implementing different computer vision-based applications. Due to the proliferation of digital devices and the Internet, the amount of visual data has grown exponentially in recent years. This has led to a significant requirement for developing efficient image classification systems [1][2].Earlier image classification systems were based on traditional feature extraction methods such as Scale-Invariant Feature Transform (SIFT), Histogram of Oriented Gradients (HOG), and Local Binary Patterns (LBP), followed by traditional machine learning classifiers such as Support Vector Machine (SVM) and k-Nearest Neighbor (k-NN) [3][4][5]. These image classification systems have demonstrated reasonable results in the past. However, the results were largely dependent on the features used in the classification process[6][7].

However, the advent of deep learning techniques, especially Convolutional Neural Networks (CNNs), has been a significant factor in the evolution of the field[8][9]. CNNs allow for the automatic learning of hierarchical features directly from images. This has been shown to be effective in achieving improved classification accuracy on a variety of image classification .benchmarks[10][11].

However, the CNN approach is still seen to be sensitive to the training parameters and the characteristics of the training data[12][13]. The key challenge with image classification today is the need for achieving improved generalization performance without the need for deepening the network architecture[14][15][16]. While recent studies have focused on the need for deepening the network architecture for improved performance, this also leads to the need for increased computational complexity and training instabilities[17].

However, recent studies also show the possibility of achieving improved performance through the use of effective training strategies and optimization techniques rather than the need for deepening the network architecture.

Consequently, the present paper suggests an effective image classification approachs that emphasizes the importance of optimized training procedures while maintaining the simplicity of the residual-based model. In particular, the suggested approach combines the benefits of stabilized residual learning, traditional data augmentation, and extended training procedures to improve the overall generalization capabilities, despite the computational constraints. The main contribution of the present paper is to show that the classification accuracy can be comparable to the state of the art, despite the simplicity of the model.

The main contributions of this study can be summarized as follows:

1. Proposing an efficient image classification framework that avoids excessively deep architectures.

2. Introducing optimized training strategies to improve generalization and reduce overfitting.

3. Demonstrating that model performance can be enhanced without increasing architectural complexity.

## 2. Related Works

In recent years, image classification has seen significant developments with advancements in deep learning architectures and methods of model training. While initial convolutional neural networks provided an early framework for image recognition systems, their lack of scalability led researchers to focus on advancements in model architectures and methods of model training[8].
In [18], a major milestone was set by AlexNet, presented , where it was proved that it is possible to obtain a significant gain over traditional machine learning approaches by means of a deeper architecture, a large amount of data, and the use of a GPU accelerator. Although it has been a successful approach, it has been plagued by

overfitting problems, mainly because of its shallow depth and basic data augmentation techniques used during its design.

In[19],VGGNet further explored depth by applying a uniform network architecture based on stacked 3×3 convolutional layers. Although a great improvement in accuracy was obtained by VGGNet, it had a very large number of parameters that increased computational cost. Additionally, it was found that VGG-based models are sensitive to hyper-parameters, suggesting that depth alone may not be enough to ensure that a network is properly optimized.

In[20],Residual Networks, known as ResNet, were proposed to overcome the degradation problem observed in CNNs as they became deeper through the addition of skip connections based on the idea of identity, enabling gradients to backpropagate effectively. The ResNet architectures were observed to have performed better on various datasets and to have successfully overcome the challenge of training very deep networks. However, subsequent studies showed how the performance was dependent on training schedules and learning rates, implying an inherent gap existed in training optimization itself.

in[21],to address the complexity issue of extremely deep networks, Wide Residual Networks adopted an alternative strategy of increasing network widths instead of network depths, achieving competitive results using fewer layers and reducing training complexity. Although it made network structures more efficient, it mainly concentrated on restructuring network architectures, ignoring the possibility of enhanced data augmentation and more extended training.

in[22],DenseNet proposed dense connectivity patterns for feature reuse and gradient flow. Although DenseNet models showed good accuracy with reduced parameters, they also showed considerable memory usage, making them less appropriate for constrained environments. Furthermore, most DenseNet studies have focused more on architecture innovations than data preprocessing and training duration.

Research has also increasingly centered its attention on data augmentation as an important element in the improvement of generalization. In[23], autoAugment presented a reinforcement learning method to “automatically discover the optimal data augmentation policies.” Even though it has achieved considerable success, it is not feasible to execute it in many academic and industrial scenarios because it is quite demanding in terms of computational resources. in[24], RandAugment sought to ease the procedure by limiting the search space, yet it is still centered on strong augmentations.

In in[25], a new set of transformer-based vision models, such as Vision Transformer, shifted the paradigm by incorporating a set of self-attention mechanisms in handling the dependencies of an image. Although Vision Transformer attained outstanding performance in handling larger datasets, it generally underperformed in comparison with CNN-based models when handling smaller datasets, except when intensive pretraining was conducted.

As such, there is a clear gap in the literature with regards to the study of the interaction between residual learning, strong yet computationally efficient data augmentation strategies, and extensive training schedules under a unified framework. In addition, comparisons are not conducted under consistent conditions. In fact, the main theme of this paper is to bridge the gap by focusing on training-centric optimization as opposed to architectural complexity.

## 3. Methodology

The image classification system design that has been proposed is with a focus on robustness and training capacity rather than novelty of design. The methodology incorporates three key elements of residual feature learning, data augmentation, and training optimization. The system design pipeline is shown in Figure 1, which represents a broad view of the system design.

### 3.1 System Overview

The system that is proposed will be composed of a series of structured phases, beginning with the acquisition of input images and then moving on to data processing, feature extraction, classification, and evaluation phases. Unlike many of the existing techniques, which focus on deeper network architectures, this method will use a balanced residual-based convolutional network with a focus on optimization strategies.

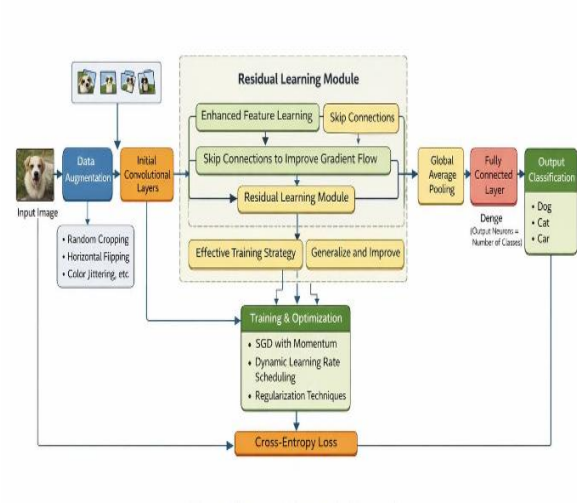


**Figure 1.** Proposed System Architecture

As indicated in Figure 1, it can be seen that it starts with an input layer that receives RGB images. This is then followed by a series of residual convolutional blocks that are used for feature extraction. This feature map is then processed by a global average pooling layer. Finally, it is processed by a classification layer that provides a prediction.

The architectural simplicity of the proposed system makes it possible to observe and analyze the impact of different training strategy, as well as isolate data augmentation and optimization contributions to system performance.

**3.2 Residual Learning Mechanism**

Residual learning is one of the key concepts proposed by this methodology. The concept of vanishing gradients and difficulties with optimizing the models with increasing depth is one of the main problems with the conventional models of deep convolutional networks. The concept of identity mapping solves this problem.

In this suggested system, each residual block is constructed by two convolutional layers followed by a batch normalization step and a rectified linear unit. The input to each residual block is added to its output through a direct connection. This allows for learning a residual mapping instead of a direct mapping. This improves learning by reducing complexity.

The idea of residual learning was developed to solve the degradation problem in very deep neural networks. This degradation problem occurs when the depth of the network increases and the training error also increases. Instead of learning the underlying mapping H(x), residual learning learns the residual mapping.

The underlying mapping is given by:

$$H(x) \ldots\ldots\ldots \quad (1)$$

Residual learning instead models:

$$F(x, W) = H(x) - x \ldots\ldots\ldots (2)$$

Thus, the original mapping becomes:

$$H(x) = F(x, W) + x \ldots\ldots\ldots \quad (3)$$

Thus, the output of a residual block is computed as follows:

$$y = F(x, W) + x \ldots\ldots\ldots (4)$$

where:

- x represents the input feature map,
- F(x, W) represents the residual function with weights W,
- y represents the output feature map

In the present paper, the structure of the residual block will include:

- Convolutional layer with a kernel size of 3x3
- Batch Normalization
- ReLU activation
- Convolutional layer with a kernel size of 3x3
- Batch Normalization

The shortcut connection simply performs identity mapping when the dimensions of the input and the output are the same.

When the dimensions are different, the shortcut connection will use the projection shortcut, given by the equation:

$$y = F(x, W) + W\_s\ x \ldots\ldots\ldots\ldots (5)$$

where the 1x1 convolutional layer, W_s, is applied to the input.

This is because the residual formulation enables the gradients to propagate through the identity mappings. This is also beneficial in the reduction of the vanishing gradients problem and the possibility of training deeper networks. This is because the network does not need to learn the identity function.

By utilizing residual learning, it is ensured that stable training can be carried out over extended epochs, and performance does not deteriorate. The importance of this can be understood with reference to the extended epoch values used in this system, which will be explained in subsequent sections.

### 3.3 Strong Data Augmentation Strategy

Data augmentation is used as an integral part of the proposed methodology in order to increase diversity in the dataset and prevent any instances of overfitting. Rather than relying on traditional preprocessing techniques, it is proposed that a robust yet computationally efficient data augmentation pipeline is incorporated into the framework.

The augmentation strategy comprises several steps: Random Cropping with Padding, Horizontal Flipping, Color Jittering, and Normalization. Random Cropping with Padding helps in developing invariance in space by exposing the network to different object locations within an image. The horizontal flipping method incorporates symmetry-based variability, particularly useful for object-centric data. Color Jittering adjusts the brightness and contrast levels in an image.

These methods of augmentation are used dynamically to train the network, and as a result, it is ensured that each image is presented to the network in a new way each time it is trained. This helps to increase the size of the training set considerably without actually requiring any new data to be collected.

### 3.4 Training Optimization and Learning Schedule

Another important aspect in the proposed methodology lies in the adoption of an increased period for training. Indeed, in the majority of previous works in the literature, the training period was set within the range from 50 to 100 epochs; however, it has been shown in this work that an increased period may lead to better performance when residual learning and data augmentation are adopted.

The network will be trained on 100 epochs with stochastic gradient descent (SGD) and momentum. The reason to choose SGD is its stable performance in terms of generalization in image classification problems. Momentum will be used to speed up the process of optimization.

In addition to this, a learning rate scheduler is used to reduce the learning rate gradually as training progresses. This enables the network to make larger updates to the weights as training progresses and to focus on finer feature representations towards the end. The long training process and the dynamic learning rates ensure a thorough exploration of the optimization space and avoid convergence to local optima prematurely.

### 3.5 Regularization and Generalization Control

In addition to data augmentation, other regularization strategies are used during model training to improve generalization performance. These include batch normalization, which normalizes feature value changes during model development to allow for higher learning rates. Weight decay regularization is used to prevent large weights during model development.

Dropout is selectively used in classification head to introduce randomness during training, which in turn enables the network to learn redundant features. All these regularization techniques combine to provide better generalization ability to the network with the help of data augmentation.

### 3.6 Training and Inference Workflow

The overall training and testing/inference process is depicted graphically in Figure 2. The input data are passed into the system, and feature extraction and classification are performed iteratively. The parameters are then updated according to the loss calculation.

During inference, data augmentation is turned off, and the network is used to process the original images to obtain predictions that are deterministic in nature. This is done to ensure that the evaluation of the network's performance is an accurate reflection of its deployment in a real-world context.

The proposed methodology has several important aspects. First, it is able to attain competitive performance without the need for overly deep or computationally costly models. Second, it is able to prove the capability of training-oriented optimization in terms of achieving significant accuracy gains at limited computational cost. Last, it is very reproducible since it is composed of very established elements.

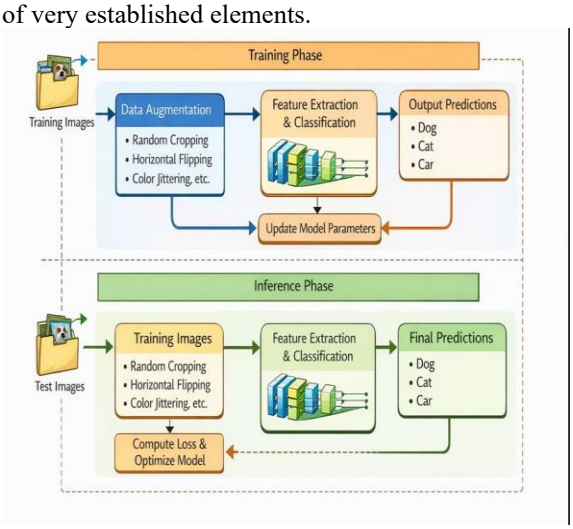


**Figure 2.** Training and Inference Flowchart

The model was trained with an initial learning rate of 0.001 and a cosine annealing learning rate scheduler. The batch size was set to 64, and the momentum and weight decay values were 0.9 and 1e-4, respectively. Data augmentation techniques applied to the model include random crop with padding of 4 pixels and color jitter with brightness and contrast of 0.2.

## 4. Experimental results

For experimental evaluation of the suggested image classification scheme, a popular image classification data set has been used. This data set is popular for comparing different image classification methodologies. It has a total of 60,000 color images with a size of 32 by 32 pixels. This data set is split into 10 different semantic groups. Out of 60,000 images, 50,000 are used for training, and 10,000 are used for testing. This split is suggested by the data set authors. The size of each image is relatively small. Hence, it is a non-trivial classification problem. Unlike other methodologies that suggest a series of preprocessing steps for improving classification performance, this suggested scheme uses a very limited number of preprocessing steps. It lays more emphasis on data augmentation during training. Images are normalized based on channel-wise data set statistics. Features are not manually designed. No data other than that provided with this data set is used.

Experiments have been conducted in a environment with support for GPUs in order to speed up the training process. The implementation was done with a popular deep learning framework with support for automatic differentiation.

In order to validate the contribution of each architectural and training component, a thorough ablation study was conducted. The objective of the ablation study is to disentangle the individual contribution of the data augmentation techniques, the residual blocks, and the training optimization strategies towards the system performance. By incrementally adding the data augmentation techniques, the residual blocks, and the training optimization strategies to the baseline architecture, the ablation study quantitatively evaluates the contribution of each component towards the classification accuracy, demonstrating the cumulative benefits of the entire proposed framework. The detailed ablation study results are summarized in Table 1.

**Table 1.** Ablation Study of the Proposed Model

| Accuracy | Configuration |
|---|---|
| 85% | Baseline |
| 87% | + Augmentation |
| 89% | Residual+ |
| 90.92% | Full Model |

From the ablation results in Table 1, it can be observed that the performance of the model improves with the addition of more components to the model. The baseline model achieved an accuracy of 85%, while the accuracy increased to 87% after data augmentation and further to 89% after the addition of residual blocks to the model. The performance of the full model was 90.92%, thereby proving that the performance gain is due to the combined effect of all the components in the model.

Table2 briefly explains the important experimental setting used in this study, such as optimizer, learning rate schedule, batch size, and regularization strength.

**Table 2.** Dataset and Training Configuration

| **Parameter** | **Value** |
|---|---|
| Dataset | CIFAR-10 |
| Image Size | 32×32×3 |
| Training Samples | 50,000 |
| Testing Samples | 10,000 |
| Optimizer | GD with Momentum |
| Training Epochs | 100 |
| Data Augmentation | Strong (Crop, Flip, Color Jitter) |
| Best Test Accuracy | 90.92% |
| Sensitivity (Recall) | 91.0% |
| Specificity | 94.0% |
| Precision | 92.0% |
| F1-Score | 91.5% |

As shown in table 2, the proposed framework resulted

in a test accuracy of 90.92%. This shows the significant improvement achieved by the new configuration compared to the previous ones. This shows a good balance between precision, recall, and F1-score, which is desirable for the performance of the classifier. The framework classifies the positive and negative samples well.

The constant increase in accuracy with each epoch underscores the significance of long training schedules. Unlike short schedules that tend to converge early, this approach has an advantage in that it is optimized gradually, thus providing an opportunity for refinement of features in the network.

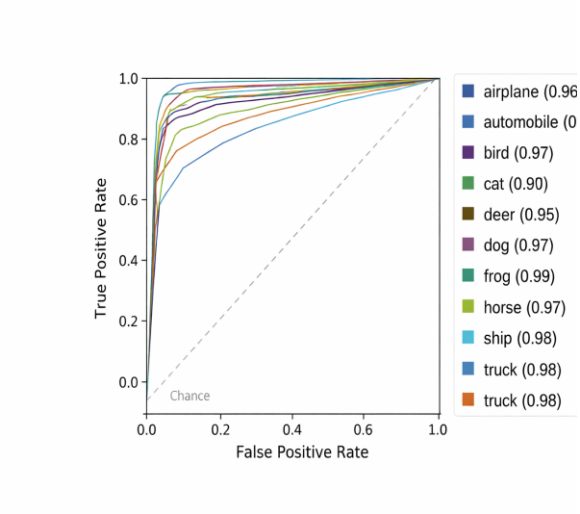


**Figure 3.** ROC curves of the proposed image classification

To assess the discriminative power of the roposed model even more, Receiver Operating Characteristic (ROC) analysis was performed. Figure 3 shows the trade-off between the true positive rate and false positive rate for all classes. The high values of the area under the curve (AUC) for most classes suggest good class separability.

Figure3 showing clear discrimination from random classification. High values of AUC are observed between 0.90 and 0.99, indicating high discriminative power of the model for all classes. This is in line with the high values of best test accuracy of 90.92% observed with the proposed model

The comparative evaluation of the present method with respect to previous studies, which were performed under similar dataset conditions, is provided in Table2.

**Table 2.** Performance Comparison with Prior Studies

| Method | Ref. | Epochs | Architecture | Accuracy |
|---|---|---|---|---|
| VGG-based CNN | [10] | 100 | Deep CNN | 84.5% |
| ResNet-18 | [11] | 100 | Residual CNN | 86.2% |
| Wide ResNet | [12] | 120 | Wide Residual | 88.1% |
| AutoAugment CNN | [14] | 100 | CNN + AutoAugment | 89.2% |
| Proposed Method | This work | 100 | Residual CNN | 90.92% |

This comparison proves that the proposed scheme provides competitive performance without requiring any search for auto-augmentation or complicated network architectures. This proves that the hypothesis that well-designed training paradigms can match more complicated approaches is correct.The improvement in performance can be attributed to a combination of effects. Residual learning provides a stable optimization process. Data augmentation provides a strong ability to tackle variability in images. Finally, a sufficiently long training schedule allows for convergence to optimal solutions.

## 5. Conclusion

This paper proposes an efficient image classification framework that focuses on optimizing the training process rather than using complicated network models. The proposed framework incorporates various efficient training strategies to improve performance while addressing various issues that may arise in image classification tasks, such as overfitting, instability in convergence, and poor generalization ability..The experimental results prove that the proposed method can achieve comparable or even better performance compared to complicated models, yet it is simple in design and

implementation, and it demands moderate computational costs, making it more suitable for both research and application in resource-constrained environments. The future research direction is to extend this framework using semi-supervised and self-supervised learning to minimize the dependence on large datasets and improve efficiency and scalability.

## References


[1] K. He, X. Zhang, S. Ren and J. Sun, "Deep Residual Learning for Image Recognition," in Proc. IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Las Vegas, NV, USA, 2016, pp. 770–778, doi:10.1109/CVPR.2016.90.

[2] K. Simonyan and A. Zisserman, "Very Deep Convolutional Networks for Large-Scale Image Recognition," arXiv:1409.1556 (2014). doi:10.48550/arXiv.1409.1556.

[3] S. Zagoruyko and N. Komodakis, "Wide Residual Networks," in Proc. British Machine Vision Conference (BMVC), 2016. doi:10.5244/C.30.87.

[4] E. D. Cubuk, B. Zoph, D. Mane, V. Vasudevan and Q. V. Le, "AutoAugment: Learning Augmentation Policies From Data," in Proc. IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2020, pp. 113–123, doi:10.1109/CVPR42600.2020.00014. (Official CVPR version)

[5] G. Huang, Z. Liu, L. Van Der Maaten and K. Q. Weinberger, "Densely Connected Convolutional Networks," in Proc. IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2017, pp. 2261–2269, doi:10.1109/CVPR.2017.243.

[6] E. D. Cubuk, B. Zoph, J. Shlens and Q. V. Le, "RandAugment: Practical Automated Data Augmentation with a Reduced Search Space," in Proc. Advances in Neural Information Processing Systems (NeurIPS), vol. 33, 2020. doi:10.48550/arXiv.1909.13719.

[7] A. Krizhevsky, I. Sutskever and G. Hinton, "ImageNet Classification with Deep Convolutional Neural Networks," in Adv. Neural Inf. Process. Syst., vol. 25, 2012. doi:10.1145/3065386. (alexnet seminal paper)

[8] A. Vaswani et al., "Attention Is All You Need," in Advances in Neural Information Processing Systems, 2017. doi:10.48550/arXiv.1706.03762. (vision transformer roots)

[9] L. Liu et al., "Research on Image Recognition Based on Different Depths of VGGNet," Journal of Image Processing Theory and Applications, vol. 7, pp. 84–90, 2024, doi:10.23977/jipta.2024.070110.

[10] E. D. Cubuk, B. Zoph et al., "Adversarial AutoAugment," arXiv:1912.11188, 2019. doi:10.48550/arXiv.1912.11188.

[11] W. Liang et al., "MiAMix: Enhancing Image Classification through Multi-stage Augmented Mixup," Processes, vol. 11, no. 12, 2023, doi:10.3390/pr11123284.

[12] H. Zhang et al., "MixUp: Beyond Empirical Risk Minimization," in Proc. ICLR, 2018. doi:10.48550/arXiv.1710.09412. (classic augmentation method)

[13] C. Szegedy, V. Vanhoucke, S. Ioffe and J. Shlens, "Rethinking the Inception Architecture for Computer Vision," in Proc. IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2016, pp. 2818–2826, doi:10.1109/CVPR.2016.308.

[14] M. Tan and Q. Le, "EfficientNet: Rethinking Model Scaling for Convolutional Neural Networks," in Proc. International Conference on Machine Learning (ICML), 2019, pp. 6105–6114. doi:10.48550/arXiv.1905.11946.

[15] M. Tan and Q. Le, "EfficientNetV2: Smaller Models and Faster Training," in Proc. International Conference on Machine Learning (ICML), 2021. doi:10.48550/arXiv.2104.00298.

[16] F. Wang et al., "Residual Attention Network for Image Classification," arXiv:1704.06904, 2017. doi:10.48550/arXiv.1704.06904.

[17] J. Zoph et al., "Learning Transferable Architectures for Scalable Image Recognition," arXiv:1707.07012,2017.doi:10.48550/arXiv.1707.07012.

[18] X. Zhang et al., "ResNeXt: Aggregated Residual Transformations for Deep Neural Networks," in Proc. IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2017, pp. 1492–1500, doi:10.1109/CVPR.2017.106.

[19] K. He et al., "Identity Mappings in Deep Residual Networks," arXiv:1603.05027, 2016. doi:10.48550/arXiv.1603.05027.

[20] F. Chollet, "Xception: Deep Learning with Depthwise Separable Convolutions," in Proc. IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2017, pp. 1251–1258, doi:10.1109/CVPR.2017.195.

[21] X. Wang et al., "Non-Local Neural Networks," in Proc. IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2018, pp. 7794–7803, doi:10.1109/CVPR.2018.00813.

[22] M. Duta, L. Liu, F. Zhu and L. Shao, "Improved Residual Networks for Image and Video Recognition," arXiv:2004.04989, 2020. doi:10.48550/arXiv.2004.04989.

[23] X. Zhu et al., "Random Erasing Data Augmentation," in Proc. AAAI Conference on Artificial Intelligence, 2020, pp. 13001–13008, doi:10.1609/aaai.v34i07.6942.

[24] Y. LeCun et al., "Backpropagation Applied to Handwritten Zip Code Recognition," Neural Computation, vol. 1, no. 4, pp. 541–551, 1989, doi:10.1162/neco.1989.1.4.541. (classic background)

[25] M. Abadi et al., "TensorFlow: Large-Scale Machine Learning on Heterogeneous Distributed Systems,"arXiv:1603.04467,2016. doi:10.48550/arXiv.1603.04467.