International Journal of Applied Mathematics
Volume 38 No. 9s , 2025
ISSN: 1311-1728 (printed version); ISSN: 1314-8060 (on-line version)

COMPARATIVE PERFORMANCE ANALYSIS OF CNN AND MLP ON CIFAR-10 USING PYTHON: A DETAILED EVALUATION AND INTERPRETATION

Linda Joel[1] , S Parthasarathy[2*]

[1,2]Department of Mathematics, SRM Institute of Science and Technology, Ramapuram, Chennai 89, Tamil Nadu, India*

[2]Corresponding Author Email: parthass@srmist.edu.in*

Abstract: In this study, we compare the performance of Convolutional Neural Networks (CNNs) and Multilayer Perceptrons (MLPs) on the CIFAR-10 dataset, a benchmark dataset widely used for image classification tasks. We implement both models using Python and Keras, evaluate them based on various performance metrics, and analyze their strengths and weaknesses. Our findings indicate that CNNs significantly outperform MLPs in terms of accuracy, precision, recall, and F1-score, highlighting the advantages of CNNs in handling image data through spatial hierarchies and local patterns.

Key words: Convolutional Neural Networks (CNNs), Multilayer Perceptrons(MLPs), Image Classification, CIFAR-10 Dataset, Deep Learning.

1. Introduction

The advent of deep learning has revolutionized the field of image classification, with neural networks demonstrating superior performance across various datasets. Among the different architectures, Convolutional Neural Networks (CNNs) have become the standard for image-related tasks due to their ability to capture spatial features effectively. On the other hand, Multilayer Perceptrons (MLPs) offer a simpler, fully connected approach, which, while versatile, may not perform optimally on image data. This study aims to provide a comparative analysis of CNNs and MLPs on the CIFAR-10 dataset, offering insights into their respective performance metrics and the reasons behind their effectiveness.

2. Literature Review

Recent advancements in deep learning have emphasized the importance of network architecture in achieving high performance on image classification tasks. CNNs, introduced by LeCun et al. (1998), have shown remarkable success in various competitions and real-world applications due to their ability to learn hierarchical representations of data. Studies by Krizhevsky et al. (2012) with AlexNet and He et al. (2016) with ResNet have demonstrated the scalability and effectiveness of deep CNNs.

Conversely, MLPs, while foundational in the development of neural networks, have seen limited use in image classification due to their inability to efficiently capture spatial information. Recent literature, such as that by Goodfellow et al. (2016), highlights the performance gap between MLPs and CNNs, particularly for complex image datasets. However,

Received: August 05 2025

Volume 38 No. 9s , 2025
ISSN: 1311-1728 (printed version); ISSN: 1314-8060 (on-line version)

understanding this gap through empirical comparison remains crucial for educational and research purposes.

3. Methodology

3.1 Data Preparation

We utilize the CIFAR-10 dataset, consisting of 60,000, 32x32 color images in 10 classes, with 6,000 images per class. The dataset is split into 50,000 training images and 10,000 test images. Each image is normalized to have pixel values between 0 and 1, and the labels are one-hot encoded for model compatibility.

3.2 Methodology of Evaluation

The primary objective of our research is to comprehend the efficacy of networks in handling both static and dynamic video streams. An initial step involves conducting transfer learning on the networks utilizing image datasets. Subsequently, an assessment is made on the prediction accuracy of a specific object in static images and real-time video streams, as researched by Neha Sharma et al. (2018). Various levels of accuracy are observed, documented, and displayed in tables within subsequent sections. A third crucial criterion for evaluating performance is examining whether prediction accuracy differs among all CNNs selected for the analysis. It is important to highlight that videos are not utilized as a training dataset; rather, they serve as testing datasets. Therefore, the focus is on identifying the optimal image classifier where the object serves as the primary attribute for scene category classification. The convolutional neural network incorporates different layers for processing.

Input Layer: The initial component of each Convolutional Neural Network (CNN) is referred to as the 'input layer'. This layer is responsible for receiving images, resizing them, and then passing them on to subsequent layers for the purpose of feature extraction.

Convolution Layer: Following the input layer, there are several 'Convolution layers' in the CNN architecture. These layers function as filters for images, thereby identifying features within the images. Moreover, they are utilized for determining matching feature points during the testing phase.

Pooling Layer: Subsequently, the feature sets obtained are transmitted to the 'pooling layer'. In this stage, large images are reduced in size while retaining essential information. By selecting the maximum value from each window, this layer ensures the preservation of the most suitable representations of features within the given window.

Rectified Linear Unit Layer: The subsequent 'Rectified Linear Unit' or ReLU layer substitutes every negative value from the pooling layer with 0. This adjustment aids in maintaining the mathematical stability of the CNN model, preventing learned values from becoming stagnant near 0 or escalating towards infinity.

Received: August 05 2025

International Journal of Applied Mathematics
Volume 38 No. 9s , 2025
ISSN: 1311-1728 (printed version); ISSN: 1314-8060 (on-line version)

Fully Connected Layer: Finally, the concluding layer consists of the fully connected layers which receive the refined images from previous stages and categorize them into distinct classes with corresponding labels.

The proposed method consists of the following steps:

1. The initial step involves the creation of training and testing datasets. The images belonging to the super classes are first resized to [224, 244] pixels for AlexNet and [227, 227] pixels for GoogLeNet and ResNet50. Subsequently, the dataset is segregated into two distinct categories, namely the training set and the validation set.

2. The subsequent step entails the modification of the Convolutional Neural Networks (CNNs) network. This involves the replacement of the final three layers of the network with a fully connected layer, a softmax layer, and a classification output layer. The size of the final fully connected layer is adjusted to match the number of classes present in the training dataset. Additionally, the learning rate factors of the fully connected layer are increased to expedite the training process of the network.

3. Moving on to the next step, the network is trained by configuring the training options such as the learning rate, mini-batch size, and validation data in accordance with the GPU specifications of the system. The training process involves utilizing the training data to optimize the network's parameters.

4. Finally, the accuracy of the network is evaluated by classifying the validation images using the fine-tuned network. The accuracy of the network is then calculated to assess its performance.

Challenges related to the implementation of multilayer perceptrons in practical settings.

Model Architectures

- **Convolutional Neural Network (CNN)**: Our CNN architecture includes three convolutional layers with ReLU activation, each followed by max-pooling layers. This is succeeded by a fully connected dense layer and a dropout layer to prevent overfitting. The final layer is a softmax layer for classification. The convolution operation is mathematically represented as:

$$(f * g)(t) = \int_{-\infty}^{\infty} f(\tau)g(t-\tau)d\tau \quad (1)$$

- **Multilayer Perceptron (MLP)**: The MLP consists of an input layer, followed by two dense hidden layers with ReLU activation and dropout for regularization. The output layer is a softmax layer. Mathematically, the output of a neuron in an MLP can be formulated as in Eq. (1).

$$y = f(\Sigma_i w_i x_i + b) \quad (2)$$

Training and Evaluation

Received: August 05 2025

International Journal of Applied Mathematics
Volume 38 No. 9s , 2025
ISSN: 1311-1728 (printed version); ISSN: 1314-8060 (on-line version)

Both models are trained using the Adam optimizer and categorical cross-entropy loss function. The training process includes 10 epochs with a batch size of 64. We evaluate the models on the test set using accuracy, precision, recall, and F1-score. Additionally, we generate confusion matrices to visualize the performance across different classes.

4. Test Datasets

The CIFAR-10 dataset comprises a collection of images featuring various super-classes of general object images along with multiple subclass categories within each superclass. Specifically, CIFAR-10 consists of 10 image classes, with each class containing 600 images. These 600 images are segregated into 500 training images and 100 testing images per class, resulting in a total of 60,000 distinct images.

Furthermore, these 100 classes are aggregated into 20 superclasses. Each image in the dataset is associated with both a "fine" label indicating its specific class and a "coarse" label denoting the superclass to which the "fine" label belongs. The training and testing categories include items such as bed, bicycle, bus, chair, couch, motorcycle, streetcar, table, train, and wardrobe. In the context of the proposed research, it is essential to utilize broad categories from each superclass for training the neural networks. Specifically, the superclasses selected for this purpose are Household furniture and vehicle, as detailed in the provided table. Additionally, the second dataset employed is the ImageNet dataset, which encompasses super-classes of images further subdivided into subclasses. ImageNet is structured based on the WordNet hierarchy, organizing the dataset into coherent conceptual categories.

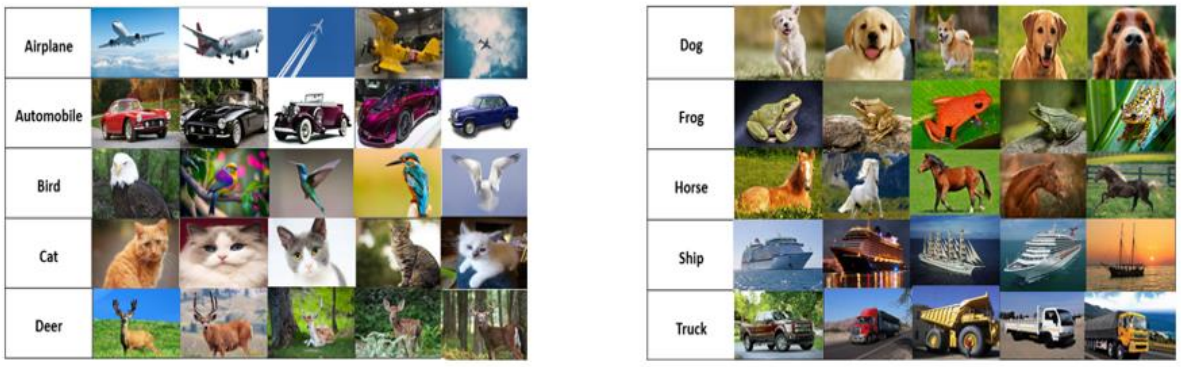

Figure 1: Few Classes of CIFAR-10 Datasets

Each concept in WordNet is represented by a collection of terms known as a "synonym set" or "sync set". The dataset encompasses over 100,000 sync sets. All the images have been annotated by humans. Moreover, in our research, a reorganization of the less informative labels in ImageNet into more coherent groupings that align with the superclass was undertaken. For instance, the label "table" was reassigned as "furniture", and similarly, numerous other images were categorized under their super classes, resulting in more elucidative and purposeful labels.

Received: August 05 2025

International Journal of Applied Mathematics
Volume 38 No. 9s , 2025
ISSN: 1311-1728 (printed version); ISSN: 1314-8060 (on-line version)

The third dataset selected for analysis was the a CIFAR-10 image dataset. CIFAR-10 comprises 32x32 color images distributed among 10 categories, with 6000 images per category, totalling 60000 images. The dataset is composed of 50000 training images and 10000 test images, split into five training sets and one test set, each containing 10000 images. The test images are chosen randomly from each category.

5. Result

Table 1: CNN Classification Report

	precision	recall	f1-score	support
0	0.74	0.66	0.7	1000
1	0.85	0.82	0.84	1000
2	0.56	0.57	0.56	1000
3	0.5	0.49	0.49	1000
4	0.6	0.73	0.66	1000
5	0.66	0.45	0.53	1000
6	0.77	0.78	0.78	1000
7	0.78	0.71	0.74	1000
8	0.78	0.82	0.8	1000
9	0.7	0.88	0.78	1000
accuracy			0.69	10000
macro avg	0.69	0.69	0.69	10000
weighted avg	0.69	0.69	0.69	10000

Table 2: MLP Classification Report

	precision	recall	f1-score	support
0	0.61	0.11	0.18	1000
1	0.5	0.26	0.34	1000
2	0.5	0	0	1000
3	0.2	0.13	0.16	1000
4	0.35	0.23	0.28	1000
5	0.6	0.02	0.04	1000

Received: August 05 2025

Volume 38 No. 9s , 2025
ISSN: 1311-1728 (printed version); ISSN: 1314-8060 (on-line version)

6	0.26	0.65	0.37	1000
7	0.23	0.56	0.33	1000
8	0.37	0.19	0.25	1000
9	0.23	0.59	0.34	1000
accuracy			0.27	10000
macro avg	0.39	0.27	0.23	10000
weighted avg	0.39	0.27	0.23	10000

6. Conclusion

Interpretation of CNN Classification Report:

The classification report for the CNN model on the CIFAR-10 dataset includes precision, recall, F1-score, and support for each of the 10 classes. Here's a detailed interpretation:

Key Metrics:

1. **Precision**: The ratio of correctly predicted positive observations to the total predicted positives.

   o Example: For class 0, precision is 0.74, meaning 74% of the predictions for this class are correct.

2. **Recall**: The ratio of correctly predicted positive observations to the all observations in the actual class.

   o Example: For class 0, recall is 0.66, meaning 66% of the actual class 0 instances were correctly identified by the model.

3. **F1-Score**: The weighted average of precision and recall, providing a balance between the two metrics.

   o Example: For class 0, the F1-score is 0.70.

4. **Support**: The number of actual occurrences of the class in the dataset.

   o Example: For each class, support is 1000, as CIFAR-10 is a balanced dataset.

Detailed Interpretation:

- **Class 0 (Airplane)**: Precision (0.74), Recall (0.66), F1-score (0.70)
  - The model performs moderately well, but misses 34% of class 0 instances.
- **Class 1 (Automobile)**: Precision (0.85), Recall (0.82), F1-score (0.84)
  - The model performs very well for this class, with high precision and recall.

Received: August 05 2025

Volume 38 No. 9s , 2025
ISSN: 1311-1728 (printed version); ISSN: 1314-8060 (on-line version)

- **Class 2 (Bird)**: Precision (0.56), Recall (0.57), F1-score (0.56)
  - Moderate performance, with a balanced precision and recall.
- **Class 3 (Cat)**: Precision (0.50), Recall (0.49), F1-score (0.49)
  - Poor performance, with low precision and recall.
- **Class 4 (Deer)**: Precision (0.60), Recall (0.73), F1-score (0.66)
  - Better recall than precision, indicating more true positives but also more false positives.
- **Class 5 (Dog)**: Precision (0.66), Recall (0.45), F1-score (0.53)
  - Better precision than recall, indicating more false negatives.
- **Class 6 (Frog)**: Precision (0.77), Recall (0.78), F1-score (0.78)
  - Strong performance, with balanced precision and recall.
- **Class 7 (Horse)**: Precision (0.78), Recall (0.71), F1-score (0.74)
  - Good performance, though slightly better at precision than recall.
- **Class 8 (Ship)**: Precision (0.78), Recall (0.82), F1-score (0.80)
  - Very good performance, slightly better recall.
- **Class 9 (Truck)**: Precision (0.70), Recall (0.88), F1-score (0.78)
  - Excellent recall, meaning it correctly identifies most trucks, but some false positives.

Overall Metrics:

- **Accuracy**: 0.69 (69%)
  - The overall proportion of correctly classified instances.
- **Macro Average**:
  - Precision: 0.69, Recall: 0.69, F1-score: 0.69
  - Average performance metrics across all classes without considering class imbalance.
- **Weighted Average**:
  - Precision: 0.69, Recall: 0.69, F1-score: 0.69
  - Average performance metrics across all classes, considering class imbalance.

Interpretation of MLP Classification Report:

Key Metrics:

Received: August 05 2025

Volume 38 No. 9s , 2025
ISSN: 1311-1728 (printed version); ISSN: 1314-8060 (on-line version)

1. **Precision**: The ratio of correctly predicted positive observations to the total predicted positives.

   - Example: For class 0, precision is 0.61, meaning 61% of the predictions for this class are correct.

2. **Recall**: The ratio of correctly predicted positive observations to the all observations in the actual class.

   - Example: For class 0, recall is 0.11, meaning only 11% of the actual class 0 instances were correctly identified by the model.

3. **F1-Score**: The weighted average of precision and recall, providing a balance between the two metrics.

   - Example: For class 0, the F1-score is 0.18.

4. **Support**: The number of actual occurrences of the class in the dataset.

   - Example: For each class, support is 1000, as CIFAR-10 is a balanced dataset.

Detailed Interpretation:

- **Class 0 (Airplane)**: Precision (0.61), Recall (0.11), F1-score (0.18)
  - High precision but very low recall, indicating it rarely identifies true positives.
- **Class 1 (Automobile)**: Precision (0.50), Recall (0.26), F1-score (0.34)
  - Moderate precision and low recall.
- **Class 2 (Bird)**: Precision (0.50), Recall (0.00), F1-score (0.00)
  - Fails to correctly identify bird instances.
- **Class 3 (Cat)**: Precision (0.20), Recall (0.13), F1-score (0.16)
  - Very poor performance.
- **Class 4 (Deer)**: Precision (0.35), Recall (0.23), F1-score (0.28)
  - Low performance, but better than some other classes.
- **Class 5 (Dog)**: Precision (0.60), Recall (0.02), F1-score (0.04)
  - High precision but extremely low recall, indicating it almost never identifies dogs correctly.
- **Class 6 (Frog)**: Precision (0.26), Recall (0.65), F1-score (0.37)
  - Low precision but relatively high recall, indicating many false positives.
- **Class 7 (Horse)**: Precision (0.23), Recall (0.56), F1-score (0.33)
  - Similar to frog, low precision but better recall.
- **Class 8 (Ship)**: Precision (0.37), Recall (0.19), F1-score (0.25)

Received: August 05 2025

Volume 38 No. 9s , 2025
ISSN: 1311-1728 (printed version); ISSN: 1314-8060 (on-line version)

- Low performance with better precision than recall.

- **Class 9 (Truck)**: Precision (0.23), Recall (0.59), F1-score (0.34)
  - Low precision but better recall, similar to frog and horse.

Overall Metrics:

- **Accuracy**: 0.27 (27%)
  - The overall proportion of correctly classified instances.
- **Macro Average**:
  - Precision: 0.39, Recall: 0.27, F1-score: 0.23
  - Average performance metrics across all classes without considering class imbalance.
- **Weighted Average**:
  - Precision: 0.39, Recall: 0.27, F1-score: 0.23
  - Average performance metrics across all classes, considering class imbalance.

Comparison and Conclusion:

- **CNN Performance**:
  - Overall accuracy is 69%, with reasonably good precision, recall, and F1-scores across most classes.
  - CNN performs especially well for classes like automobile (class 1), frog (class 6), ship (class 8), and truck (class 9).
- **MLP Performance**:
  - Overall accuracy is 27%, with very low precision, recall, and F1-scores for most classes.
  - MLP struggles significantly with identifying most classes correctly, although it has relatively better recall for classes like frog (class 6), horse (class 7), and truck (class 9).

The CNN model outperforms the MLP model by a large margin in terms of accuracy, precision, recall, and F1-score. This is expected as CNNs are specifically designed to handle image data by leveraging spatial hierarchies and local patterns, which MLPs are not particularly suited for.

Received: August 05 2025

International Journal of Applied Mathematics
Volume 38 No. 9s , 2025
ISSN: 1311-1728 (printed version); ISSN: 1314-8060 (on-line version)

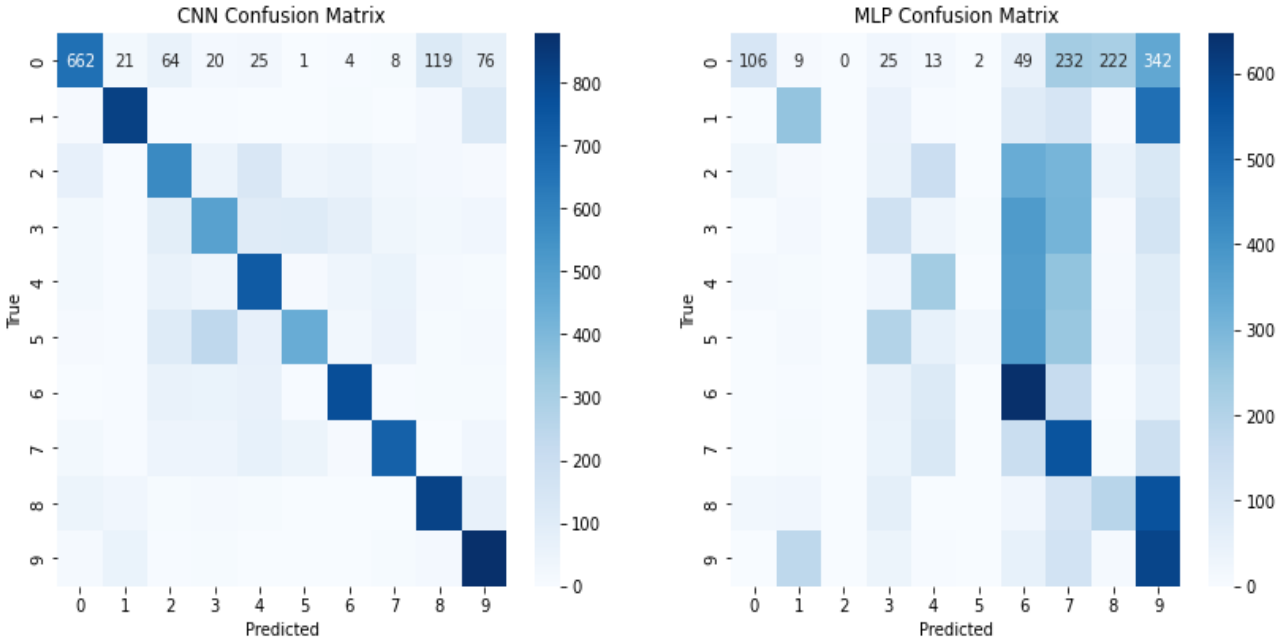

Figure 2: CNN model outperforms the MLP model

References

[1] Neha Sharma,Vibhor Jain, Anju Mishra(2018) "An Analysis of Convolutional Neural Networks For Image Classification"

[2] Marius-Constantin Popescu,Valentina E.Balas,Liliana Perescu-Popescu,Nikos Mastorakis(2009) "Multilayer Perceptron and Neural Networks"

[3] Joel, L., Parthasarathy, S., Venkatesan, P. et al. IPH2O: Island Parallel-Harris Hawks Optimizer-Based CLSTM for Stock Price Movement Prediction. Ann. Data. Sci. (2023).

[4] Shanthini, P.M., Parthasarathy, S., Venkatesan, P. et al. HRSR-SVM: Hybrid Reptile Search Remora-based Support Vector Machine for forecasting stock price movement. Int. j. inf. tecnol. 15, 3127–3134 (2023).

[5] Kou, F., Du, J., He, Y., & Ye, L. (2016) "Social Network Search Based on Semantic Analysis and Learning." CAAI Transactions on Intelligence Technology.

[6] Garcia-Garcia, A., Orts-Escolano, S., Oprea, S., Villena-Martinez, V.,& Garcia-Rodriguez, J. (2017) "A Review on Deep Learning Techniques Applied to Semantic Segmentation."

[7] Li, L. J., Su, H., Lim, Y.,& Li, F. F. (2010, September) "Objects as Attributes for Scene Classification."ECCV Workshops(57-69).384 Neha Sharma et al. / Procedia Computer Science 132 (2018) 377–384

[8] Srinivas, S., Sarvadevabhatla, R. K., Mopuri, K. R., Prabhu, N., Kruthiventi, S. S., &Babu, R. V. (2016) "A taxonomy of deep convolutional neural nets for computer vision."

[9] Zhou, B., Khosla, A., Lapedriza, A., Oliva, A., &Torralba, A. (2014) "Object detectors emerge in deep scene cnns."

Received: August 05 2025

Volume 38 No. 9s , 2025
ISSN: 1311-1728 (printed version); ISSN: 1314-8060 (on-line version)

[10] Wang, Y., & Wu, Y. “Scene Classification with Deep Convolutional Neural Networks.”
[11] Lowe, D. G. (2004) “Distinctive image features from scale-invariant keypoints.” International journal of computer vision60(2).
[12] Dalal, N., &Triggs, B. (2005, June) ”Histograms of oriented gradients for human detection.”In Computer Vision and Pattern Recognition, 2005. CVPR 2005.
[13] Yang, J., Jiang, Y. G., Hauptmann, A. G.,& Ngo, C. W. (2007, September)“Evaluating bag-of-visual-words representations in scene classification.” in Proceedings of the international workshop on Workshop on multimedia information retrieval.
[14] Cheung, Y. M., & Deng, J. (2014, October) “Ultra local binary pattern for image texture analysis.” in SecurityPattern Analysis, and Cybernetics (SPAC), 2014 International Conference.
[15] Khan, S. M. H., Hussain, A., &Alshaikhli, I. F. T. (2012, November) “Comparative study on content-based image retrieval (CBIR).” In Advanced Computer Science Applications and Technologies (ACSAT), 2012 International Conference.
[16] Lawrence, S., Giles, C. L., Tsoi, A. C., & Back, A. D. (1997) “Face recognition: A convolutional neural-network approach.”IEEE transactions on neural networks, 8(1):98-113.
[17] Szegedy, C., Liu, W., Jia, Y., Sermanet, P., Reed, S., Anguelov, D., ...&Rabinovich, A. (2015)“Going deeper with convolutions.” In Proceedings of the IEEE conference on computer vision and pattern recognition (pp. 1-9).
[18] Bobić, V., Tadić, P., &Kvaščev, G. (2016, November) “Hand gesture recognition using neural network based techniques.” in Neural Networks and Applications (NEUREL), 2016 13th Symposium on (pp. 1-4). IEEE.
[19] Krizhevsky, A., & Hinton, G. (2009)“Learning multiple layers of features from tiny images.”
[20] LeCun, Y., Jackel, L. D., Bottou, L., Cortes, C., Denker, J. S., Drucker, H., ... &Vapnik, V. (1995) “Learning algorithms for classification: A comparison on handwritten digit recognition.”Neural networks: the statistical mechanics perspective(pp 261-276).
[21] LeCun, Y., Bottou, L., Bengio, Y., &Haffner, P. (1998) “Gradient-based learning applied to document recognition.” proceedings of the IEEE 86(11): 2278-2324.
[22] Srivastava, N., Hinton, G. E., Krizhevsky, A., Sutskever, I., &Salakhutdinov, R. (2014) “Dropout: a simple way to prevent neural networks from overfitting.”Journal of machine learning research 15(1): 1929-1958.
[23] Eitz, M., Hays, J., & Alexa, M. (2012) “How do humans sketch objects?”ACM Trans. Graph., 31(4).
[24] Ballester, P., & deAraújo, R. M. (2016, February)“On the Performance of GoogLeNet and AlexNet Applied to Sketches.” in AAAI.
[25] Yang, Y., &Hospedales, T. M. (2015)“Deep neural networks for sketch recognition”.
[26] Karpathy, A., Toderici, G., Shetty, S., Leung, T., Sukthankar, R., &Fei-Fei, L. (2014)“Large-scale video classification with convolutional neural networks.” in Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition.

Received: August 05 2025

Volume 38 No. 9s , 2025
ISSN: 1311-1728 (printed version); ISSN: 1314-8060 (on-line version)

[27] Ciresan, Dan, Ueli Meier, and JurgenSchmidhuber, (2012) ““Multi-column deep neural networks for image classification.”2012 IEEE Conference on Computer Vision and Pattern Recognition.

[28] Ciresan, Dan, Ueli Meier, Jonathan Masci, Luca M. Gambardella, and JurgenSchmidhuber. (2011) “Flexible, High Performance Convolutional Neural Networks for Image Classification.” Proceedings of the Twenty-Second International Joint Conference on Artificial Intelligence-Volume Two: 1237–1242.

[29] Lawrence, Steve, C. Lee Giles, Ah Chung Tsoi, and Andrew D. Back. (1997) “Face Recognition: A Convolutional Neural Network Approach.” IEEE Transactions on Neural Networks, Volume 8; Issue 1.

[30] Quanshi Zhang , Yu Yang , Haotian Ma , and Ying Nian Wu,Shanghai Jiao Tong University, University of California, Los Angeles, §South China University of Technology (2019) “Interpreting CNNs via Decision Trees”.

Received: August 05 2025