iJRASET
International Journal For Research in Applied Science and Engineering Technology

INTERNATIONAL JOURNAL FOR RESEARCH

IN APPLIED SCIENCE & ENGINEERING TECHNOLOGY

Volume: 12 **Issue:** IV **Month of publication:** April 2024

DOI: https://doi.org/10.22214/ijraset.2024.60719

www.ijraset.com

Call: 08813907089 | E-mail ID: ijraset@gmail.com

International Journal for Research in Applied Science & Engineering Technology (IJRASET)
ISSN: 2321-9653; IC Value: 45.98; SJ Impact Factor: 7.538
Volume 12 Issue IV Apr 2024- Available at www.ijraset.com

An Analysis of Convolutional Neutral Network for Image Classification

Tainiyat K Hanchinal
Department of Computer Science and Engineering, Jain College of Engineering and Research

Abstract: ***Convolutional Neural Networks (CNNs) have emerged as a powerful tool for image classification tasks, owing to their ability to automatically learn hierarchical representations from raw pixel data. This paper presents a comprehensive analysis of CNNs for image classification, focusing on their architecture, training process, and performance evaluation metrics. The study investigates various CNN architectures, including AlexNet, GoogLeNet, and ResNet, highlighting their respective strengths and weaknesses in different scenarios. Moreover, the paper explores the impact of hyperparameters such as network depth, filter size, and pooling strategies on classification accuracy and computational efficiency. Furthermore, the training process of CNNs, encompassing data preprocessing, augmentation techniques, and optimization algorithms, is scrutinized to elucidate best practices for achieving optimal performance. Additionally, the paper discusses commonly used evaluation metrics such as accuracy, precision, recall, and F1-score, elucidating their interpretation and significance in assessing model performance. Through a systematic review of existing literature and experimental validation, this analysis aims to provide insights into the underlying mechanisms of CNNs for image classification tasks. Finally, the paper outlines future research directions, emphasizing the need for exploring novel architectures, optimizing hyperparameters, and enhancing interpretability and robustness of CNN models. Overall, this study contributes to a deeper understanding of CNNs for image classification and provides valuable guidance for practitioners and researchers in the field.***

Keywords: Convolutional Neural Networks, Image Classification, CNN Architectures, Training Process, CNN Models, Performance Evaluation Metrics.

I. INTRODUCTION

Computer vision image classification is crucial for our everyday lives, careers, and education. A process comprising picture preprocessing, image segmentation, key feature extraction, and matching identification is used to classify images. We can now gather image data more quickly than ever before and use it for a variety of applications, such as face recognition, traffic identification, security, and medical equipment, thanks to the most advanced image classification systems. With the advent of deep learning, feature extraction and classifier have been combined into a learning framework to solve the drawbacks of the traditional feature selection method. Finding several levels of representation is the aim of deep learning, which assumes that high-level features will capture the more ethereal semantics of the data. A key element of deep learning is the classification of images using convolutional structures. The convolutional neural network is inspired by the architecture of the mammalian visual system. In 1962, Hubel and Wiesel proposed a model of visual structure based on the visual cortex of cats. The concept of a receptive field has been introduced for the first time. Fukushima introduced the initial hierarchical framework that Neocognition would use for image analysis in 1980. Neocognition made use of the local connection between neurons to accomplish network translation invariance.

Numerous deep learning architectures are at one's disposal. The model of a classifier system disclosed in this research was created using convolutional neural networks, the most practical and effective deep neural network for this type of data. Thus, by applying these learning representations to tasks that need less training data, CNNs that have been trained on massive picture datasets for recognition tasks may be effectively utilized.

Many methods have been developed since 2006 to overcome the difficulties of training deep neural networks. Krizhevsky offers a conventional CNN architecture called Alexnet and shows a significant improvement over previous methods for the task of photo classification. In view of Alexnet's success, a number of measures have been suggested to improve its performance. It is advised to use VGGNet, GoogleNet, and ZFNet.

II. TYPES OF CNN ARCHITECTURE

Convolutional Neural Networks (CNNs) have undergone significant evolution since their inception, resulting in various architectures tailored for different tasks and computational requirements. Below are some notable types of CNN architectures:

A. LeNet

LeNet is a ground-breaking Convolutional Neural Network (CNN) architecture that was first presented by Yann LeCun et al. in 1998. It laid the foundation for contemporary deep learning techniques in computer vision. LeNet is a neural network that was initially created for handwritten digit recognition tasks. It is composed of convolutional layers, subsampling layers, and fully connected layers. Its architecture is based on the principle of feature hierarchy extraction, which allows for efficient pattern recognition by gradually combining low-level characteristics to produce higher-level representations. The success of LeNet encouraged more developments in the field of deep learning research and proved the effectiveness of convolutional operations in capturing spatial hierarchies. LeNet was a pioneer in picture classification, object identification, and other computer vision problems, even though its depth was somewhat shallow in comparison to modern CNN architectures.

B. AlexNet

AlexNet, developed by Alex Krizhevsky, Ilya Sutskever, and Geoffrey Hinton in 2012, transformed the area of computer vision by demonstrating the capabilities of deep convolutional neural networks. Its ground-breaking performance, which greatly outperformed earlier techniques, in the ImageNet Large Scale Visual Recognition Challenge (ILSVRC) represented an important milestone. Multiple convolutional layers make up AlexNet's architecture, which is then followed by max-pooling layers and fully linked layers. It brought forth a number of significant breakthroughs, such as dropout regularization, local response normalization layers, and the widespread application of ReLU activation functions. These components reduced the likelihood of overfitting and allowed the network to efficiently capture complicated hierarchical information. Furthermore, AlexNet demonstrated the value of deep learning architectures and the processing power of GPUs for training massive neural networks. Because of its success, CNN designs became deeper and more complex, which paved the way for later developments in computer vision.

C. VGGNet

VGGNet, introduced by Simonyan and Zisserman in 2014, is a watershed moment in Convolutional Neural Network (CNN) architecture design for image categorization. VGGNet set a new benchmark for deep learning models with its state-of-the-art performance on the ImageNet dataset thanks to its straightforward but efficient structure. VGGNet's uniform architecture, which consists of several convolutional layers with modest 3x3 filters followed by max-pooling layers, is its primary innovation. VGGNet efficiently extracts hierarchical features from input images by stacking convolutional layers deeper, which makes it possible to learn more discriminative representations. VGGNet retains a simple architecture, which makes it easy to comprehend and apply even with its enhanced depth. This homogeneity and simplicity have helped VGGNet become widely used and laid the groundwork for later developments in CNN architecture design.

D. GoogLeNet

GoogLeNet, developed by Szegedy et al. of Google Research in 2014, is a game-changing innovation in Convolutional Neural Network (CNN) architectures. Prominently recognized for its effectiveness and exceptional performance, GoogLeNet pioneered the notion of inception modules, a paradigm shift in CNN information processing. Inception modules have parallel convolutional pathways of varying length, which, in contrast to standard architectures with stacked layers, enable the network to efficiently capture data at different spatial scales. In comparison to earlier models, GoogLeNet showed considerable increases in accuracy and decreased processing costs by allowing the network to extract a variety of features while preserving computational efficiency. Because of its success, network architectures should be optimized for both efficiency and performance. This led to improvements in computer vision research and influenced the design of later CNN structures.

E. ResNet

ResNet, an acronym for Residual Network, is a revolutionary convolutional neural network architecture designed to tackle the difficulty of training exceptionally deep neural networks. The concept of residual learning also referred to as shortcut connections or skip connections was first presented by Kaiming He et al. in 2015. ResNet adds shortcut connections to let gradients flow during training. By allowing information to move directly from older layers to subsequent layers, these shortcuts help to mitigate the vanishing gradient issue that arises during deep network training. ResNet architectures, including variations like ResNet-50, ResNet-101, and ResNet-152, have achieved remarkable performance across various computer vision tasks, such as image classification, object detection, and image segmentation. This is because they allow the training of networks with hundreds or even thousands of layers. ResNet's ground-breaking architecture is still a mainstay in the convolutional neural network community and has greatly advanced deep learning.

F. DenseNet

Proposed by Huang et al. in 2016, DenseNet, also known as Dense Convolutional Network, is a revolutionary architecture in the field of Convolutional Neural Networks (CNNs). DenseNet presents dense connection patterns, where each layer receives direct input from all preceding levels within a dense block, in contrast to standard CNN designs where each layer is connected only to its succeeding layers. This densely connected structure addresses the issues of vanishing gradients and encourages deeper network designs by encouraging feature reuse and facilitating gradient flow during training. DenseNet achieves state-of-the-art performance with substantially fewer parameters than conventional topologies by improving parameter efficiency and feature propagation. Because of its creative architecture, DenseNet is a well-liked option for a variety of computer vision applications, such as segmentation, object identification, and image classification. It also serves as an inspiration for new developments in deep learning research.

III.KEY REASONS FOR THE SIGNIFICANCE OF CNN

A. Hierarchical Feature Learning:

CNNs are good at automatically picking up data representations in a hierarchical format. CNNs are capable of capturing complex patterns and features at many levels of abstraction, ranging from basic edges and textures to sophisticated object shapes and structures, by layering convolutional and pooling procedures. CNNs are able to perform tasks like object detection, segmentation, and picture classification with great efficiency thanks to their hierarchical feature learning capabilities.

B. Translation Invariance:

Translation invariance is achieved by CNNs through the use of convolutional layers with shared weights and spatial pooling procedures. Because of their ability to identify objects regardless of where they are in the input image, CNNs are resistant to changes in position, scale, and orientation.

C. Parameter Sharing and Sparse Connectivity

When compared to fully connected networks, CNNs use sparse connectivity and parameter sharing to drastically lower the number of trainable parameters. CNNs accomplish parameter efficiency while maintaining the capacity to capture spatial dependencies in the data by sharing weights across many spatial locations.

D. Effective Feature Extraction

In CNNs, the convolutional layers perform the function of feature extractors, automatically picking up filters or kernels from the input data that identify significant patterns. To support discriminative representation learning, these learnt filters are refined during training to identify pertinent features including edges, corners, and textures.

E. Scalability and Versatility

CNNs are adaptable for a variety of applications because they can be scaled to handle inputs of different sizes and complexities. CNNs are capable of handling varying input dimensions without requiring major design changes, regardless of the type of input they are processing from high-definition films to low-resolution photos.

F. State-of-the-Art Performance:

On a variety of computer vision tasks, such as object identification, semantic segmentation, image production, and image classification, CNNs have continuously demonstrated state-of-the-art performance. They are now essential tools in industries including healthcare, autonomous cars, robotics, and multimedia analysis due to their unmatched precision and efficiency.

IV.METHODOLOGY

Understanding how networks function with both static and real-time video streams is the main objective of our research. The first step in the subsequent procedure is transfer learning on networks with image datasets. Transfer learning on networks containing image datasets is the next step. The testing of the subsequent stage comes next. Next, the prediction rate of the same item on both live video streams and still photographs is analyzed. The different accuracy rates are noted, noted, and displayed in the tables that are supplied in the sections that follow. The third important criterion for evaluating the performance was determining whether the prediction accuracy of the CNNs employed in the study varied from one another.

It is important to note that videos are used as testing datasets rather than training datasets. Consequently, we are looking for the best image classifier in which the object serves as the main characteristic for classifying scenes into groups.

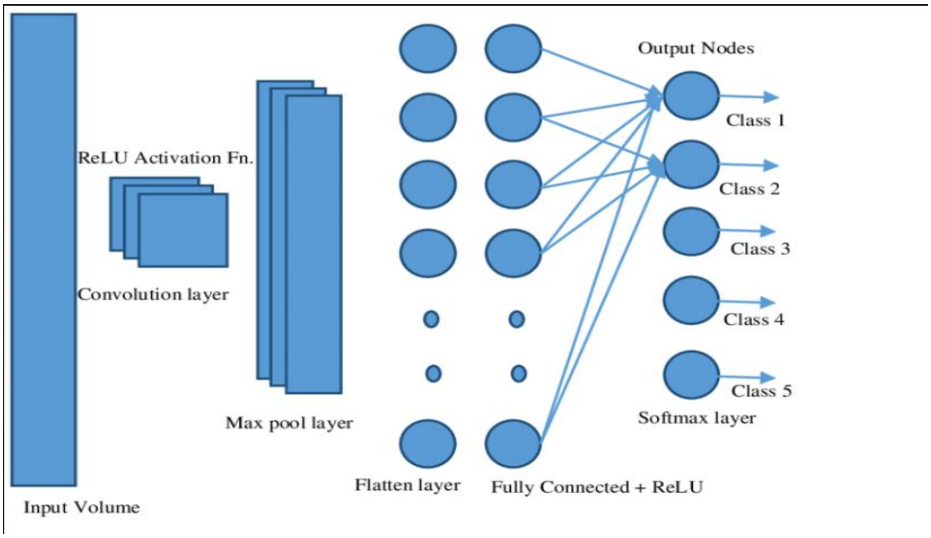


Fig. 1 CNN Architecture Layers.

A. CNN Architecture has different layer as follow:

1) Input Layer

The input layer is responsible for receiving raw image data. Each pixel in the input image is represented by a grid of neurons that makes up the system. The input layer of a CNN maintains the input image's spatial structure, in contrast to fully linked neural networks. The size of the input image and the quantity of color channels (such as RGB) define the input layer's dimensions. As the first input representation for later layers of the CNN, the values of the input layer neurons indicate the intensity or color values of the corresponding pixels in the input image.

2) Convolution Layer

The convolution layer filters the incoming data using a collection of learnable filters. Local patterns and features are extracted from the input data by each filter, also referred to as a kernel. The convolutional layer gains the ability to recognize features like edges, textures, and forms during this process. These features are crucial for tasks like object detection and classification that come after. Hierarchical feature learning in lower layers of the network is made possible by the convolutional layer's output, or "feature maps", which show the existence of learnt features in various spatial positions of the input data.

3) Max Pool Layer

The Max Pooling layer chooses the maximum value within each local region to downsample the input feature maps. This procedure helps with translation invariance and feature extraction by reducing the spatial dimensions of the feature maps while maintaining the most pertinent information. By lowering the computational cost of ensuing layers, max pooling contributes to the network's increased computational efficiency. By concentrating on the most important details in each feature map, it also improves the network's capacity to identify unique patterns and features, which helps the model perform better in tasks like object detection and picture categorization.

4) Flatten Layer

The Flatten layer in a Convolutional Neural Network (CNN) serves as a transition between the convolutional layers and the fully connected layers. It converts the multidimensional feature maps produced by the convolutional layers into a one-dimensional vector, which can be fed into the subsequent fully connected layers for classification or regression tasks. This transformation allows the network to process the spatial information learned by the convolutional layers as a flat input, enabling the network to learn global patterns and relationships across the entire input space, essential for making predictions on the task at hand.

5) Fully Connected Layer

A fully connected layer is one in which every neuron connects to every neuron in the layer before it. Full-connection layers combine data from the whole input volume, in contrast to convolutional layers, which work on small spatial areas. Fully connected layers in CNNs are usually positioned at the end of the network and are used for final classification or regression tasks. They incorporate high-level features that have been learnt by the convolutional and pooling layers that came before them. They facilitate end-to-end learning for tasks like object detection and picture classification by allowing the network to understand intricate nonlinear correlations between features.

6) Output Node

The output nodes show the anticipated class probabilities for each class in the classification task. The number of output nodes in the CNN's last layer usually matches the number of classes in the dataset. The output value of each output node indicates the model's confidence or likelihood that the input image belongs to the class to which it is assigned. Each output node is linked to a distinct class label. The class that has the highest probability at the time of inference is the one that is anticipated for the input image.

A. The following are the Steps in the Suggested Method

1) Creating Training and Testing Dataset

The training and testing dataset is created by resizing the super classes pictures used for training to [224,244] pixels for AlexNet and [227,227] pixels for GoogLeNet and ResNet50. The dataset is then separated into two categories: training and validation data sets.

2) Modifying CNN Network

CNN's network can be modified by adding a fully connected layer, a softmax layer, and a classification output layer in place of the network's last three layers. Assign the last completely linked layer a size equal to the total number of classes in the training set. To train the network more quickly, raise the fully connected layer's learning rate parameters.

3) Train the Network

Train the network by adjusting the learning rate, mini-batch size, and validation data in accordance with the system's GPU specifications. Utilizing the training data, train the network.

4) Test the Accuracy of the Network

Evaluate the network's accuracy by classifying the validation images with the optimized network and computing the accuracy of the classification. In a similar manner, real-time video feeds are tested to fine-tune the network for accurate outcomes.

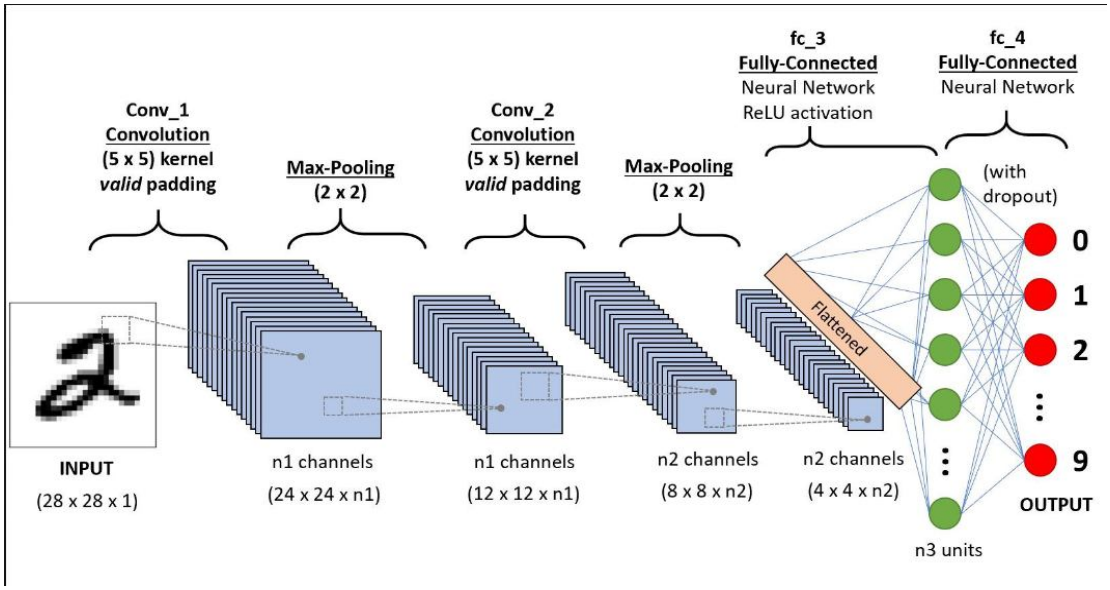


Fig. 2. Steps of CNN Architecture

V. MODEL

Numerous clever pre-trained CNN exist; these CNN have the ability to impart knowledge. It therefore only requires the training and testing datasets at its input layer. The fundamental layers and techniques used in the construction of the networks differ. GoogleNet's Inception Modules mix the filters for the next layer by executing convolutions of different sizes. As opposed to filter concatenation, AlexNet uses the output of the layer before it as its input. Both networks employ the implementation of the Caffe Deep Learning framework and have undergone independent testing. On the other hand, training neural networks becomes more difficult as we move farther away, and accuracy peaks before falling. These two problems are attempted to be addressed via residual learning. A deep convolutional neural network frequently consists of numerous layers that have been trained and layered for the intended use. The network learns a variety of low-, mid-, and high-level features at the end of its layers. Relative to specific attributes, the network aims to learn some residual in residual learning. The feature that is removed from the feature learned from the layer's input is called residual. ResNet does this by creating a shortcut connection that transfers a portion of the layer's input (n+x) directly to another layer. Three neural networks that are now in use AlexNets, Google Nets, and ResNet50 are compared. The transfer learning concepts are then implemented through the training of already existing networks and the construction of new networks for further comparison. The performance of the new models is very different from that of the previous networks, although having the same number of layers as the original models. The various accuracy rates that were determined using the same photographs are shown in the tables in the next section.

VI. CONVOLUTIONAL NEURAL NETWORKS (CNNS) FOR IMAGE CLASSIFICATION OF ADVANCEMENTS

A. Architectural Innovations

CNN design advancements have resulted in the development of deeper and more complicated networks, including ResNet, DenseNet, and EfficientNet. These architectures improve computational efficiency and performance by utilizing methods like as dense connectivity, residual connections, and efficient model scaling.

B. Transfer Learning *and Pre-trained Models*

In CNN-based image classification, transfer learning has become a popular technique where pre-trained models developed on massive datasets such as ImageNet are adjusted for particular applications. Especially when labeled data is scarce, this method improves efficiency and allows for faster convergence and greater generalization.

C. Attention Mechanisms

By including attention mechanisms, CNNs are better able to concentrate on pertinent areas of the input image, which improves classification accuracy. The model can selectively attend to prominent portions of the image thanks to attention algorithms that dynamically weight information based on their relevance.

D. Data Augmentation and Regularization Techniques

CNN models are now more robust and broadly applicable thanks to developments in data augmentation and regularization methods such mixup, cutoff, and label smoothing. By introducing changes in the training data and preventing overfitting, these strategies enhance the model's capacity to reliably categorize unknown images.

E. Hardware Acceleration and Model Compression

Developments in hardware acceleration technologies, including as dedicated inference accelerators, GPUs, and TPUs, have made it easier to deploy CNNs for image categorization. CNNs are also more deployable on devices with limited resources thanks to methods like model pruning, quantization, and knowledge distillation, which allow for effective model compression without a noticeable decrease in performance.

VII. FURTHER DISCUSSIONS

In this paragraph, we will delve deeper into the parameters related to weights, the encoding strategies of the structures, and the fitness evaluation of the proposed EvoCNN approach. The experimental results are also discussed, which may provide useful details regarding the possible applications of the proposed EvoCNN methodology. Crossover operators function as the local search, or exploitation search, whereas mutation operators function as the exploration search, or global search. Only fully developing local and global searches might greatly improve performance because they should complement each other.

The approaches that are frequently used for CNN weight optimization rely on gradient data. It is commonly known that gradient based optimizers are sensitive to the initial placements of the parameters that require optimization. Without a good starting point, gradient-based approaches are prone to getting trapped in local minima. Given the abundance of attributes, it appears impossible to find a better starting point for the link weights using GAs. It is evident that a considerable amount of variables are not amenable to effective optimization or efficient chromosomal storage. The suggested EvoCNN technique just encodes the means and standard derivations of the weights in each layer using an indirect encoding strategy. Techniques now in use to find CNN architectures together with an individual's fitness often consider the final classification accuracy. To obtain the final classification accuracy, the training approach typically requires multiple extra epochs, which might take a considerable amount of time.

VIII. SUMMARY

An overview of convolutional neural networks (CNNs) for image categorization is given in this article. It starts off by stressing the value of picture categorization across a range of fields and the shortcomings of conventional feature selection techniques. A remedy for these drawbacks is presented, namely CNNs and deep learning.

The article describes how CNNs use pooling and convolutional layers to extract features at various levels of abstraction, which allows them to become adept at learning hierarchical representations of images. Because CNNs have translation invariance, they can identify patterns in images no matter where they are located. Because of their capacity to recognize pertinent traits and extrapolate to previously unobserved data, they are also data efficient and require fewer training samples.

One important feature of CNNs that is highlighted is transfer learning, which allows them to use pre-trained models that have been taught on big datasets and adjust them for particular image classification tasks. As a result, less training data is needed, and classification performance is enhanced.

Another benefit of CNNs is its scalability, which can be altered by adding or removing layers, altering the quantity of filters, and altering the size of the filters used in the convolutional layers. Because of its adaptability, CNNs may be used for a wide range of image classification tasks, from straightforward classification to more difficult ones like object recognition and segmentation.

The methodology for assessing CNN performance is described in the paper. It entails checking accuracy, transferring learning, and training networks on both static and real-time video streams. The text discusses various CNN layer types, such as fully connected layers, rectified linear unit (ReLU) layers, pooling layers, input layers, and convolution layers.

There is discussion of a number of models and architectures, including AlexNet, GoogLeNet, and ResNet50. The article presents developments in CNNs, such as attention mechanisms, transformer-based designs, meta-learning, AutoML, and neural architecture search, and evaluates how well they function. It also highlights the necessity of robustness against adversarial attacks, explainability, and interpretability in CNNs.

IX. CONCLUSIONS

In this study, a unique evolutionary technique is being developed for autonomously evolving CNN architectures and weights for image classification tasks. This objective has been successfully attained by proposing a new representation for the weight initialization strategy, a new encoding scheme for variable length chromosomes, a new genetic operator for chromosomes with varying lengths, a slacked binary tournament selection for selecting promising individuals, and an efficient fitness evaluation method to accelerate evolution. Given the short training duration, it is helpful to understand deep learning. Evolutionary algorithms will be incorporated into future research to handle the classification feature extraction difficulty and minimize the amount of parameters needed for this process, thereby improving our system.

REFERENCES

[1] Narayana Darapaneni; B Krishnamurthy; Anwesh Reddy Paduri, "Convolution Neural Networks: A Comparative Study for Image Classification", IEEE 15th International Conference on Industrial and Information Systems (ICIIS), RUPNAGAR, India, ISSN: 2164-7011, RUPNAGAR, India.

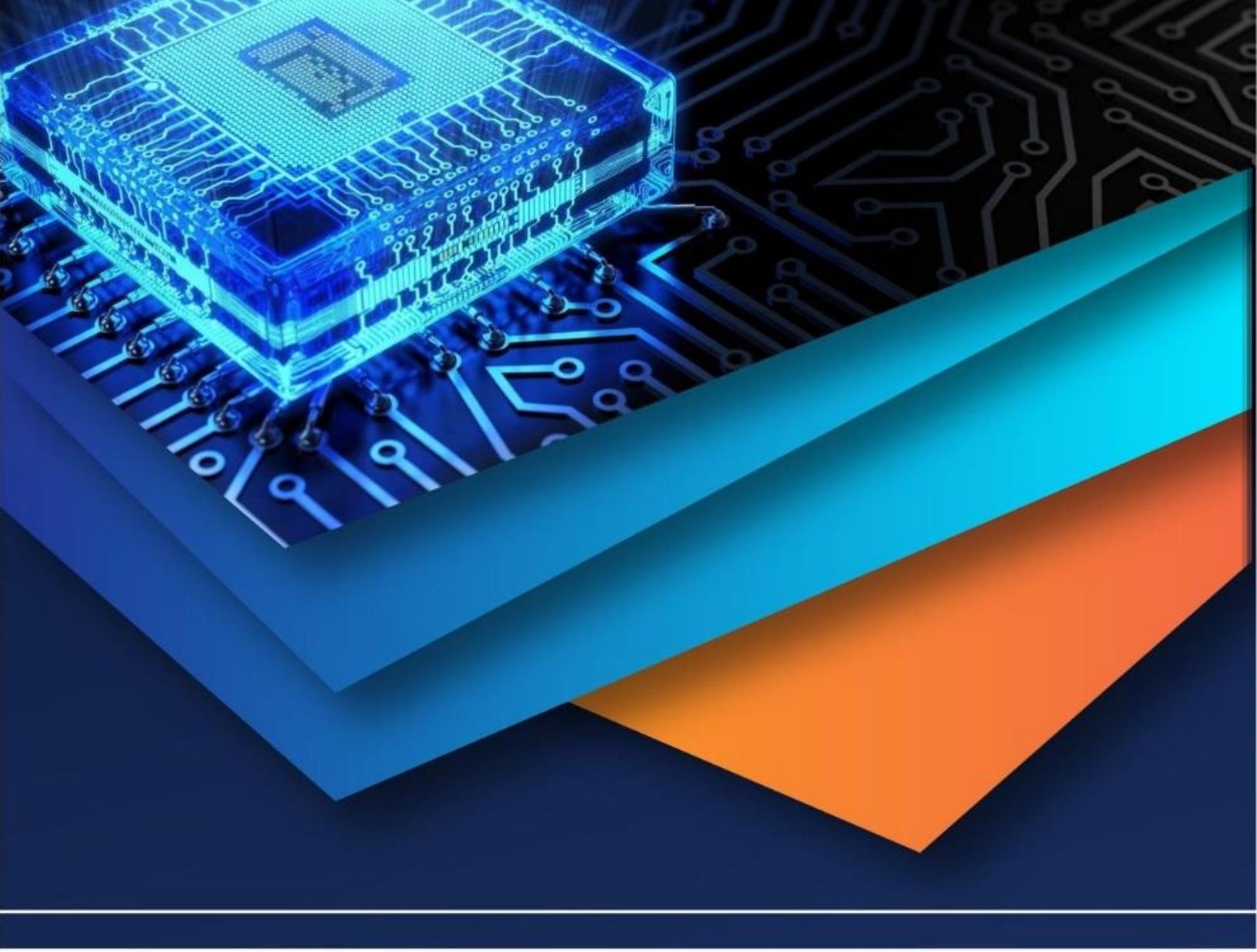
[2] Neha Sharma, Vibhor Jain, Anju Mishra, "An Analysis Of Convolutional Neural Networks For Image Classification", Procedia Computer Science Volume 132, Pages 377-384, 2018.

[3] Redmon J, and Angelova A, "Real-time grasp detection using convolutional neural networks", IEEE International Conference on Robotics and Automation, pp. 1316–1322, 2015.

[4] Lei M., Yu L., Xueliang Z., Yuanxin Y., Gaofei Y and Brian Alan J, "Deep learning in remote sensing applications: A meta-analysis and review", ISPRS Journal of Photogrammetry and Remote Sensing 2018, pp. 166–177.

[5] Deepan P. and Sudha L.R., "Object Classification of Remote Sensing Image Using Deep Convolutional Neural Network", The Cognitive Approach in Cloud Computing and Internet of Things Technologies for Surveillance Tracking Systems, 2020, pp. 107-120.

[6] Yu H., Yang W., Xia G.S. and Liu G, "A color-texture-structure descriptor for high resolution satellite image classification", Jo. of Remote Sensing, 2016, pp. 259-269.
[7] Cheng G., Han J. and Lu X, "Remote Sensing Image Scene Classification", Benchmark and State of the Art, Proceedings of the IEEE, 2017, pp. 1-19.
[8] Hang Chang, Cheng Zhong, Ju Han, Jian-Hua Mao, "Unsupervised Transfer Learning via Multi-Scale Convolutional Sparse Coding for Biomedical Application.", IEEE Transactions on Pattern Analysis and Machine Intelligence, 23 January 2017.
[9] Ahonen, T., Hadid, A., and Pietikinen, "M. Face description with local binary patterns: Application to face recognition.", Pattern Analysis and Machine Intelligence, 2037–2041. 2016.
[10] Maxwell A., Warner T.A. and Fang F. Implementation of machine-learning classification in remote sensing: an applied review, International Journal of Remote Sensing,2018, pp. 2784-2817.
[11] Wong Y.C., Lai J.A., Ranjit S.S., Syafeeza A.R. and Hamid N, "A. Convolutional Neural Network for Object Detection System for Blind People", Journal of Telecommunication, Electronic and Computer Engineering, 2019, pp. 1-6.
[12] Koh C., Chang J., Tai C., Huang D., Hsieh H. and Liu Y.Bird, "Sound Classification using Convolutional Neural Networks", International Journal of computer vision, 2019,pp. 1-10.
[13] Wen Y., Zhou T., Liu L. and Xia C. "Automatic Convolutional Neural Architecture Search for Image Classification under Different Scenes", IEEE Transaction on Innovation and Application in Edge Computing, 2019,pp. 38495- 38506.
[14] Zhang W., Tang P. and Zhao L, "Remote Sensing Image Scene Classification Using CNN-CapsNet, Remote Sens".,2019, pp. 1-22.
[15] Deepan P. and Sudha L.R, "Remote sensing image scene classification using dilated convolutional neural networks", International Journal of Emerging Trends in Engineering Research, Vol. 8, No.7, 2020, pp.3622-3630.



10.22214/IJRASET

45.98

IMPACT FACTOR: 7.129

IMPACT FACTOR: 7.429

INTERNATIONAL JOURNAL FOR RESEARCH

IN APPLIED SCIENCE & ENGINEERING TECHNOLOGY