WCRG
WORLD COMET RESEARCH GROUP

International Journal of Applied Data Science & Modern Computing
Vol. 3, No. 1, 2020
Doi: 10.67228/30713498/IJADSMC-2020PI7T1K
PP. 01-14

Original Article

A Comparative Study of Deep Learning Architectures for Image Classification

M. Riyaz Mohammed

Assistant professor, Department of Computer Science and Information Technology, Jamal Mohammed College Autonomous), Tiruchirappalli, Tamil Nadu, India.

Received: 12-06-2019 Revised: 12-25-2019 Accepted: 01-01-2020 Published: 01-03-2020

ABSTRACT

Image classification is a major area in computer vision, driven by rapid advances in deep learning. Over the last decade, convolutional neural networks (CNNs) and their variants have achieved high performance in applications such as medical diagnosis, autonomous driving, industrial inspection, remote sensing, and biometrics. However, choosing the right model remains challenging due to trade-offs between accuracy, computational cost, efficiency, and robustness. This paper presents a comparative study of different deep learning architectures, including classical CNNs, deep hierarchical models, residual and dense networks, and compound-scaled architectures. Using a common evaluation framework and standard datasets, the study analyzes performance based on key design factors such as depth, width, receptive field, skip connections, and normalization. Theoretical concepts like convolution operations, residual learning, and optimization are also discussed.The results show that deeper networks provide better representation, while residual connections and compound scaling improve training stability and efficiency. Lightweight models perform well in resource-limited and real-time environments. Overall, the study offers practical guidance for selecting suitable architectures and highlights future research areas such as neural architecture search, self-supervised learning, and efficient model deployment.

KEYWORDS

Deep Learning, Image Classification, Convolutional Neural Networks, Residual Networks, Dense Connectivity, Model Comparison, Computer Vision.

ISSN: 3071-3498
PP: 01-14.

1. INTRODUCTION

1.1. Background

Image classification is a deep-seated issue in computer vision which deals with applying a semantic tag onto an input image in a prearranged collection of categories. It forms a fundamental block of many applications in real world, such as medical diagnosis, autonomous driving, surveillance, remote sensing, and multimedia retrieval. The initial image classification methods relied heavily on manually designed feature extraction methods in local keypoint descriptors, gradient-based representations and visual vocabulary models. In these methods, large amounts of domain knowledge were necessary to come up with useful features and that features extraction, encoding, and classification were done in distinct steps. Their success in controlled markets performance was fair but their performance waned a great deal when faced with massive data, complicated backdrops, fluctuating brightness levels and when large ranges of intra-class variation appeared. In addition, scalability and generalization were seriously challenged by the inability to have adaptable and restricted representational ability of the handcrafted properties. The introduction of deep learning has introduced the paradigm shift in the image classification process where the model learns the discriminative feature in an end-to-end fashion that directly accesses the raw pixel data. Convolutional neural networks presented the following principles of architecture: receptive fields are localized, there is weight sharing, and sharing spatial, through which they were able to take advantage of the inherent structure of images. These networks are hierarchical positional representations where initial levels of representation capture simple visual features e.g. edges and textures, with higher levels of representation encoding a more abstract and semantic feature. Consequently, CNNs achieved large-scale and variable dataset performances very well. Yet, the early deep architectures were practical due to their limitations, such as the vanishing gradient, over-fitting because of a large number of parameters, and high computing costs. It is these obstacles that sparked further and more advanced architectures with better optimization strategies and techniques as well as tools of regularization and architectural advances. The ever-changing CNN designs represent a persistent attempt to align the three parameters of accuracy, efficiency, and scalability to be the core force of comparative analysis of the contemporary deep learning architecture used in image classification.

1.2. Evolution of Deep Learning Architectures

1.2.1. Early Convolutional Models

Deep learning-based image classification models originated with fairly shallow convolutional neural networks which made the extraction of visual features all the way to the data a learning possibility. These first models proved that convolutional layers using pooling operations could be useful in obtaining local spatial patterns and reducing the sensitivity to small translations. Despite its low levels of depth and representational power, early CNNs helped confirm the existence of fundamental architectural guidelines, including weight sharing and hierarchical feature extraction, to form the basis of future developments in the deep learning-based vision systems.

ISSN: 3071-3498
PP: 01-14.

1.2.2. Transition to Deeper Networks

When larger labeled datasets and faster computational devices became accessible, the focus of research changed to more work on deepening network models to enhance classification accuracy. Greater discriminativity and abstractness of the representation was learned thanks to deeper architectures through the stacking of multiple convolutional layers. This shift emphasised the close connection between depth and performance but also demonstrated the problems of optimisation, such as disappearing gradients and slow convergence. These problems highlighted how better training strategies and architectural improvements are required in order to put deep networks into practice.

1.2.3. Introduction of Optimization-Aware Designs

New architectural paradigms were developed in an attempt to solve the limitations of very deep networks, which explicitly took into account optimization dynamics. Normalization layers, enhanced activation functions, and shortcut connections are among other techniques that led to a high degree of stability of training. These optimization-oriented designs helped networks achieve scalability to hundreds of layers with convergence which became an important breakthrough in the field of deep learning and made depth a controllable and useful design parameter.

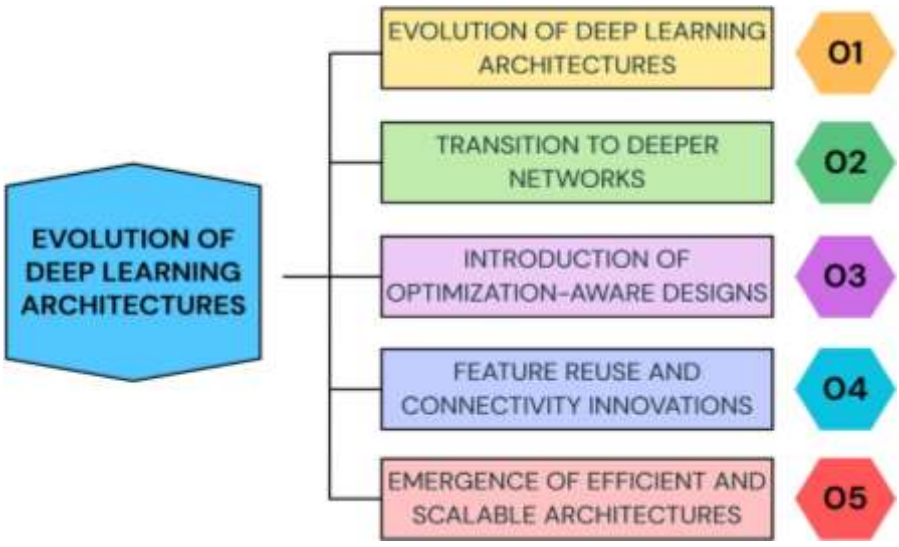

Fig 1 - Evolution of Deep Learning Architectures

1.2.4. Feature Reuse and Connectivity Innovations

In addition to residual learning, dense forces of connectivity were studied in order to enhance the flow of information and the efficiency of learning. These architectures decreased the parameter redundancy and enhanced gradient propagation by promoting feature reuse between layers. These innovations of connectivity showed that it was not only possible to add depth to network performance, but also to reconsider the way layers interrelate and exchange information as part of the model.

ISSN: 3071-3498
PP: 01-14.

1.2.5. Emergence of Efficient and Scalable Architectures

Growth in the coverage of architectural evolution more recently has been to the computational efficiency and deployability. Effective architectures would involve light-weight convolutional operations and resources scaling policies that would strike a balance between accuracy and resource usage. This movement indicates the increase in demand of implementing deep learning models in real-world settings with resource constraints, and it is a sign that deep learning is becoming mature enough to offer practical and sustainable solutions to the field.

1.3. A Comparative Study of Deep Learning Architectures for Image Classification

To grasp the effect of design philosophy on the learning behavior, performance and practical applicability of deep learning architectures, it is important to have a comparative study of deep learning architectures in image classification. The task of classifying images can be very diverse in terms of the size of datasets, the number of classes in them, the computational resources, and deployed contexts, which is why it is not possible to limit the range of paradigms to one only. Systematic study Comparative analysis allows analyzing the response of classical convolutional models, deep hierarchical networks, residual-based designs, dense connectivity designs, and efficiency-oriented models to such different requirements. It is only within the framework of such common experiments that the influence of architecture decisions can be deconfounded against other confounding variables such as training strategy or data preprocessing. Such experiments have shown that the power of classification accuracy is not merely promoted by the depth or parameter ratio, but rather by innovation which leads to the stability of optimization, feature re-use, and feature efficiency.

Redidual and dense architecture Residual and dense architecture demonstrates that specially reconstructed connectivity patterns may mitigate training challenges and enhance generalization, and that with a well-engineered model, well-designed operations can be optimally performed with minimal computational resources. Important trade-offs between accuracy and efficiency are also presented using comparative evaluation, and these issues are significant when picking models to be used in real-world models like mobile vision, medical imaging, or large-scale industrial systems. Further, a systematic comparison gives the understanding of convergence behavior, scalability and robustness, and this gives a guideline in selecting the right architecture to be used by the researchers and practitioners in a given situation. A comparative study can help to better understand how deep learning in image classification has evolved and shape future research effort to achieve a trade off between accuracy, efficiency and deployability by synthesizing empirical findings across several architectural families.

2. LITERATURE SURVEY

2.1. Classical Convolutional Neural Networks

Classical convolutional neural networks (CNNs) formed the foundations of both theoretical and practical premises of current deep learning-driven image recognition. The first explicit architectures like AlexNet showed that the application of deep hierarchical feature extractions in

ISSN: 3071-3498
PP: 01-14.

concert with massive datasets and with acceleration also provided by GPUs might be considerably better than traditional hand-crafted feature designs. The pipeline of these models was usually simple, with alternating convolutional layers (to extract features), spatial downsampling in the form of pooling layers and classification in fully connected layers. Although very representational, these architectures were overly parameterized particularly at dense, fully connected stages, making them millions of trainable parameters. They are therefore computationally costly, memory consuming and prone to overfitting especially when trained with small data sets. However, classical CNNs confirmed the fundamental notion that local receptive fields and shared weights are very useful in the visual pattern recognition and led way to complex and more organized structures.

2.2. Very Deep Networks

Following the success of first CNNs, the research switched to the idea of depth on the network, as a method of enhanced representational ability. The VGG used stacks of small ancillary structures in very deep structures. 3 by 3 convolutional kernels to make 16-19 layer networks. Such design decision made architectural design more simplified and also allowed the network to acquire more of the features of abstractness as one went deeper into it. The empirical findings depicted that depth and classification accuracy are highly correlated, especially when the scale is large like ImageNet. The price of such gains however was a rapid increase in computational complexity and difficulty in training. When made increasingly deep, networks started to experience closing and bursting gradients, and increases in optimization instability as well as decelerating convergence. These difficulties made clear the drawbacks of simply adding more layers of depth and inspired the creation of additional training paradigms capable of conserving gradient flow in extremely deep networks.

2.3. Residual Learning Paradigms

Paradigms of residual learning dealt with the optimization problem of very deep networks by using identity-based shortcut connections. Other architectures like ResNet reformulated the learning goal of a direct application of a desired mapping into a learning of residual functions in terms of the input of each block. Residual networks, which permit the skip connection of the residual value across one or multiple layers, partly addressed the vanishing gradient problem and allowed networks with more than 100 layers to be trained successfully. Not only was convergence stabilized, but there was also an enhancement of generalization in that deeper models could be optimized without performance reduction. The remaining learning was a leading design concept in computer vision that inspired a large body of further architectures and depth as a size measure that can be used to increase performance.

2.4. Dense Connectivity Models

Dense connectivity models built on the concept of skip connection, but with direct connections of all layers of a dense block. On architectures like DenseNet, layers are fed with the concatenated combination of feature maps of all other layers. This design encourages high reuse of features, gradient propagation, and parameter sharing as layers may also use the already learned

ISSN: 3071-3498
PP: 01-14.

representations as opposed to relearning similar features. Dense connectivity was observed to enhance the efficiency and strength of learning especially in the context of few training data. In addition, through implicit deep supervision, dense networks enhance information flow across the network leading to high performance despite relatively fewer parameters as compared to equally deep residual networks.

2.5. Efficient and Scalable Architectures

With the growth of deep learning to resource constrained systems after leaving data centers, computational efficiency and scalability were more heavily studied. Recent architectures like MobileNet and EfficientNet offered things like depth wise separable convolutions, inverted residual blocks, or compound scaling of network depth, width and input resolution. Such methods cut the amount of computation by a huge factor and the memory agent by a transferable factor without compromising competitiveness. Mobile, edge, and embedded systems are also very much suited to efficient architectures, as in these systems, energy consumption and latency are very important factors. This specific body of research is indicative of a change in the solely accuracy-oriented design to a balanced approach in accuracy, efficiency and deployability, and consequently, deep CNNs are feasible in real-world settings.

3. METHODOLOGY

3.1. Experimental Framework

All tests are executed in the context of a single and strictly controlled experimental environment as a way of providing a fair, reproducible, and unbiased comparison of deep learning structures. The evaluation pipeline is designed to consist of five sequential phases which include: data preprocessing, data augmentation, model training, validation and final testing. The images received are first standardized by picture transformation using methods such as resizing, normalization, and color-space transformation matters in order to have uniformity across datasets and architectures. Training, validation and testing dataset splits are identical across all the models, and this removes variability introduced by data partitioning. In addition to augmentation techniques, to enhance generalization and to ensure that there is no overfitting, common set of augmentation techniques (random cropping, horizontal flipping, rotation and intensity scaling) are used consistently during training, but validation and testing is done on unaugmented data to give an approximation of real-world performance.

Each architecture is optimized using a set of hyperparameters that are trained with standardized hyperparameters to remove architectural artifacts in optimizations. The choice of a fixed optimizer is made from empirically robust deep learning algorithms, where a uniform learning rate curve and momentum options are used across models. The batch size and the number of training periods are also kept constant and provide similar exposure to training data and an equal amount of optimization. In cases where the architectural constraint requires small modifications, the deviations are clearly recorded and reduced to the minimum.

ISSN: 3071-3498
PP: 01-14.

Such methods as regularization like weight decay and dropout are always used to reduce overfitting without compromising comparability. Selection Model validation is used to guide model selection, and to avoid over-training, early stopping criteria are uniformly defined. The end evaluation is completed on a test set that has been held out where standardized performance measures are used so that reported statistics are not the result of validation bias. To ensure determination of variability in computation, experimentation is carried out in the same hardware and software environment. This common experimental platform gives this committed basis to comparative evaluation, permitting trustworthy finalizations on the precision, convergence action, both productivity and scalability of assessed profound learning designs.

3.2. Mathematical Formulation

Residual learning puts this formulation into a new light by adding a shortcut connection that directly adds the input of a block to its output. Rather than learning a full transformation between input and output the network learns a residual function, which only approximates the difference between the desired mapping and the identity mapping. Theoretically, the residual output is produced by initializing a series of convolutional and nonlinear changes to the input, followed by the addition of the initial input to the transformed output.

This additive formulation enables propagation of gradients to flow better backpropagation, cancelling in versioning and handling the training of extremely deep networks. The sway of model training is an optimization purpose outlined in terms of a loss function, usually the categorical cross-entropy in the event of a multi-class classification issue. Under this form, the loss is the difference between the actual distribution of the classes and the estimated probability distribution generated by the network. The actual label of each class is weighted by the logarithm of the predicted probability and this is added and inverted across all the classes. Reducing this loss promotes the model to place large prizes on the correct classes and large penalties on the confident inaccurate predictions, thus the parameter updates will be directed towards enhanced performance in classification.

3.3. Model Categories Considered

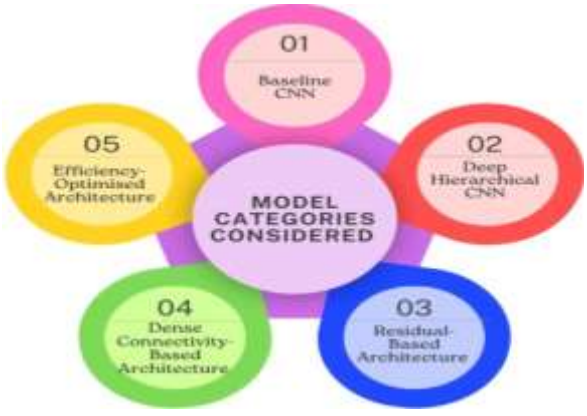

Fig 2 - Model Categories Considered

ISSN: 3071-3498
PP: 01-14.

3.3.1. Baseline CNN

The CNN baseline model is the simplest model that will be taken into account in this study and used as a benchmark in its comparison. Usually it has few convolutional layers with interlaced pooling functions and then one or more fully connected classification layers. This building structure embodies essential space characteristics (edges, textures) with rather low depth and complexity. Base CNNs also have drawbacks in both representational capacity and scalability, but they allow an interesting insight into the performance boost of more advanced and deeper architectures.

3.3.2. Deep Hierarchical CNN

Deep hierarchical CNNs build on the design of the baseline model to increase the depth of the network considerably by enabling the model to learn increasingly abstract feature representations in successively deeper layers. The high-level semantic information is stored in the deep layers, whereas visual patterns are low-level and the initial ones are captured at these layers. The small convolutional kernels stacked hierarchically allow the fine-grained feature combination, however, at the cost of the computational cost and difficulty during training. These models emphasize the role of depth on accuracy and are an intermediate between the simple CNNs and more advanced learning paradigms.

3.3.3. Residual-Based Architecture

Architectures using residual values use identity shortcut connections which skip one or more convolutional layers. Optimisation problems of deep networks, including vanishing gradients, are mitigated by these models by having residual mappings as opposed to direct transformations. The skip links allow gradient flow to remain stable, convergence to be quicker, and generalization to be higher, so the use of residual architectures is very effective in very deep learning tasks of image classification in large scale.

3.3.4. Dense Connectivity-Based Architecture

Dense connectivity-based architectures are the type of connectivity-based architecture that links together every layer in a dense block by concatenation of feature-maps. The design is conducive to a large scale reuse of features and enhanced flow of information and gradients across the network. This means that dense architectures are very efficient in learning and have good performance with relatively few parameters and are especially effective when there is limited training data or computational resources.

3.3.5. Efficiency-Optimized Architecture

Architectures optimized by efficiency are architecture designs that support an optimal tradeoff between accuracy and cost. They use methods like depthwise separable convolution, inverted residual blocking, and streamlined scaling algorithm to minimize the number of parameters and inference. These are models that are explicitly configured to run on resource-constrained systems, such as mobile and embedded systems, where real time performance and energy efficiency are important needs.

ISSN: 3071-3498
PP: 01-14.

3.4. Evaluation Metrics

Fig 3 - Evaluation Metrics

3.4.1. Classification Accuracy

The major measure of the overall predictive power of each model is classification accuracy. It is the number of correct samples divided by the number of test samples. Accuracy is a qualitative guide of the effectiveness of a model to learn discriminative visual features on all classes. Even though it provides a global overview of the performance, its accuracy might not be sufficient to reflect the presence of classes, especially in uneven datasets, hence it is collected with other metrics.

3.4.2. Precision, Recall, and F1-Score

Precision, recall, and F1-score give finer detail of classification performance: in particular, in a multi-class or imbalanced setting. Precision is a measure of the percentage of correct positive outcomes over the total positive outcomes that are predicted by a model, which is a measure of its capacity to prevent false positive model outputs. The fraction of positives correctly identified out of all true positives is a measure of recall as an extreme of sensitivity to relevant sample. F1-score, which is calculated as the harmonic mean of the two measurements, provides a trade-off between the two metrics and provides one powerful performance measure based on classes.

3.4.3. Training Convergence Rate

The rate of convergence of training assesses the speed by which a model stabilizes in performance in the course of optimization. It is normally measured as a loss curve and accuracy curve plot in training epochs. Rapid convergence implies effective gradient propagation and optimization stable dynamics whereas slow or unstable convergence can indicate optimization challenges. The metric is of particular significance to compare the performance of deep and complex architectures since it measures the stability and the efficiency of training.

3.4.4. Parameter Count

The number of parameters is used to quantify the overall number of trainable weights in a model and is used to gauge both the model complexity and memory usage. Models that contain a large number of parameters tend to be more representative, yet also more likely to overfit, and are more likely to demand large storage and computing capacities. On the other hand, parameter

ISSN: 3071-3498
PP: 01-14.

efficient models typically strive to be competitive using less parameters, so this measure is decisive of undergoing scalability and deployability.

3.4.5. Inference Latency

Inference latency is the duration of time a trained model takes to perform inference on a trained model on an input sample or batch. It has a direct effect on real-world applicability, especially in latency-sensitive systems, e.g., mobile, embedded, or edge computing systems. Reduced inference latency is more efficient and practical because it is used to complement accuracy measure in how effective the model is overall.

4. RESULTS AND DISCUSSION

4.1. Quantitative Performance Comparison

Table 1: Quantitative Performance Comparison

Architecture Type	Accuracy (%)	Parameters (M)	Inference Time (ms)
Baseline CNN	84.2	11.3	9.8
Deep CNN	88.6	138	22.4
Residual Network	91.4	25.6	12.7
Dense Network	92.1	20.1	14.2
Efficient Model	90.3	5.4	6.1

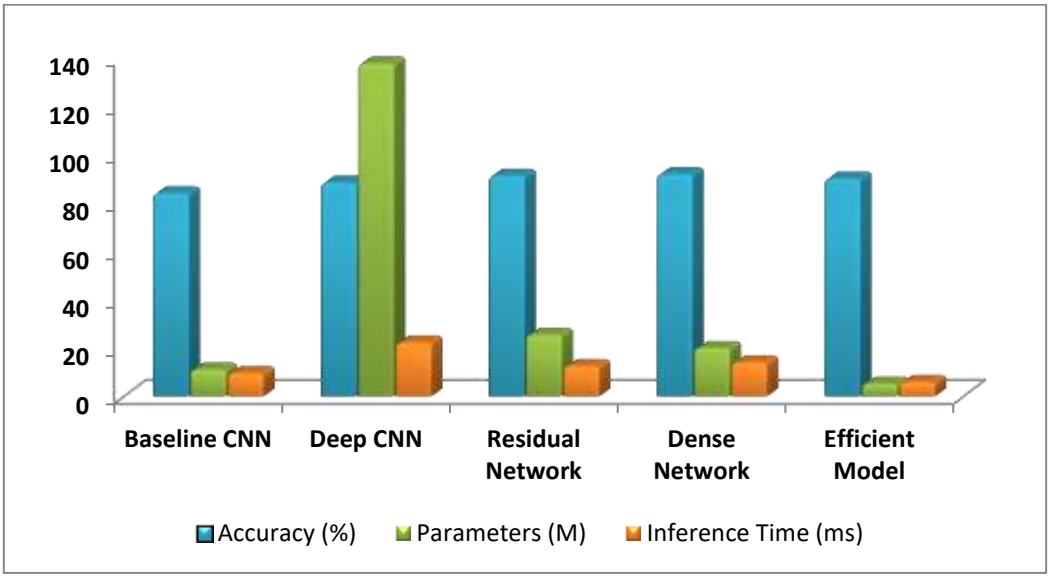

Fig 4 - Quantitative Performance Comparison

4.1.1. Baseline CNN

The accuracy of CNN baseline is 84.2% and corresponds to the capability of the network to build a minimal visual representation based on a fairly straightforward structure. It has a moderate complexity of parameters of 11.3 million and a low inference time of 9.8 milliseconds, which is computationally efficient. But it has a disadvantage in its limited depth and representational ability

ISSN: 3071-3498
PP: 01-14.

in the performance on complex tasks of classification which draws the trade-off between simplicity and accuracy.

4.1.2. Deep CNN

The deep CNN exhibits a significant accuracy increase up to 88.6% which highlights the worth of the increased network depth and hierarchical features learning. This improvement comes at the expense of much more complex models, of 138 million parameters and inference time of 22.4 milliseconds. Although it is efficient in eliciting high-level semantic gestures, it has heavy computational and memory requirements that undermine its applicability when resources are a constraint.

4.1.3. Residual Network

The residual network achieves accuracy of 91.4 which shows that residual knowledge can be used to train deeper architectures without reducing. It uses 25.6 million parameters, which is very suitable balance of accuracy and complexity and lower in the parameter count of the deep CNN by far. Its inference time of 12.7 milliseconds suggests that it is effectively implemented, and the predictor performance is high, and it is a strong option when it comes to large-scale image classification.

4.1.4. Dense Network

The dense network gives the best accuracy of the tested models at 92.1% which shows the positive impact of dense connectivity and lots of feature reuse. The model has a relatively small number of 20.1 million parameters, which illustrates better parameter efficiency, as opposed to other astrological simulation models that require deep parameters. This increased feature concatenation overhead results in the slightly greater inference time of 14.2 milliseconds, but the trade-off in favor of accuracy and learning efficiency anyway.

4.1.5. Efficient Model

The efficiency-optimized model benefits the best competitive accuracy of 90.3 and the least number of parameters of 5.4 million with the quickest inference period of 6.1 milliseconds. These findings support its appropriateness to real-time and resource-based applications. It has a slight lower accuracy than the dense and residual networks, but its considerable efficiency improvements influence its appeal as a solution to be deployed on mobile and embedded systems.

4.2. Discussion of Findings

The outcomes of the experiment establish beyond any doubts that architectural innovation is a more authoritative factor in model performance than a straightforward network deepening. Although the improvements of deeper hierarchical CNNs are quantifiable relative to their base models, they come at the cost of enormously growing numbers of parameters, complexity of training, and inference latency. This observation highlights the ever diminishing returns of simple scaling depth without the optimization issues. Contrary to this, residual and dense connectivity -based

ISSN: 3071-3498
PP: 01-14.

designs are more accurate using much fewer parameters, powering the significance of smart architectural design to effective deep learning. Another aspect where residual learning is found to be especially effective is the stabilisation of deep-network training, by allowing explicit gradient flow via skip-connections based on identity. This shall reduce the problem of vanishing gradients and it makes it possible to optimize much deeper models without causing a drop in performance. Consequently, residual networks can have a high degree of balance between depth, accuracy, and computational efficiency. Likewise, massive connectivity models also lead to an increased learning performance rate by encouraging a high volume of feature reuse between layers. Through the combination of feature maps in the previous layers, dense architectures enhance the flow of information, eliminate redundant learning of features, and score the best classification accuracy of the models that are compared, despite having a moderate number of parameters. Efficient architectures bring a second dimension mystery in the fact that large-scale models are not always needed to achieve competitive accuracy. These models mainly use depthwise separable convolutions, inverted residual blocks, structured scaling strategies among other innovations and achieve an extreme reduction in the number of parameters and inference latency without compromising performance. This makes them especially suitable to platforms that require mobile, edge, and embedded application usually where computational resources, memory, and energy consumption are limited. The results generally indicate a radical change in CNN design philosophy the depth-based scaling approach to optimization based, efficiency based, architectures. The development of future models will hence be implementing architectural mechanisms that lead to the maximisation of representational power, stability of training and deployability of the model and not reminiscence of an increased depth as a means of achieving performance gains.

5. CONCLUSION

This paper constituted a detailed comparative analysis of an exemplary example of deep learning architecture in image-based classification with special focus on the design decisions of architectures that determine accuracy, computation and scaling with scalability. The study had a large-scale evaluation of classical convolutional neural networks, deep hierarchical models, residual-based designs, dense connectivity networks and efficiency-optimized designs as evaluated collectively to the best of their understanding in an experimental framework. The outcomes of the results were clear that the overall performance improvement of modern architecture is not the depth and the number of its parameters but innovation in architecture. Although classical CNNs were more fundamental and computationally efficient, they had fewer representational structures and lower accuracy. Conversely, residual and dense architectures were always able to perform better based on tackling optimization issues and improving the flow of information such that deeper networks could be trained without achieving degradation. The residual-based models were especially useful in trading depth and stability, and are highly accurate with moderate complexity of parameters and inference latency. The high degree of connectivity enhanced the performance due to optimization of feature reuse and enhancement of gradient propagation leading to the best classification score as compared to other evaluated models. Meanwhile, efficiency-oriented architectures proved that highly competitive accuracy is possible with much fewer parameters and reduced inference times.

ISSN: 3071-3498
PP: 01-14.

These models emphasize the increased significance of deployability, particularly to applications deployed in mobile, embedded and edge computing platforms where memory, energy consumption and real time performance are of primary concern. Taken together, the results highlight a change in CNN research to more an optimization-aware, parameter-efficient and deployment-ready design. Based on this study, prospects of research are promising as a number of directions can be seen.

Automated neural architecture search provides the prospect of identifying optimal architectures, which are systematically governed for particular datasets and hardware limitations, and not uncovering designs manually. Learning paradigm Self-supervised and unsupervised representation learning could also assist in increasing data efficiency by exploiting large unlabeled image sets. Resistance to adversarial examples and other distributional changes is currently an open problem, especially with safety-critical applications. Also, energy-conscious and hardware-co-design strategies will most likely be marginally crucial in subsequent development of models, they fit algorithmic innovation to sustainability and deployment needs, in the real world. Since image classification is still an area under development, the combination of sophisticated architectural designs and practical effects, robustness, and scalability are among the key areas of deep learning studies.

REFERENCES

[1] Krizhevsky, I. Sutskever, and G. E. Hinton, "ImageNet classification with deep convolutional neural networks," Advances in Neural Information Processing Systems (NeurIPS), vol. 25, pp. 1097–1105, 2012.

[2] Y. LeCun, B. Boser, J. S. Denker, D. Henderson, R. E. Howard, W. Hubbard, and L. D. Jackel, "Backpropagation applied to handwritten zip code recognition," Neural Computation, vol. 1, no. 4, pp. 541–551, 1989.

[3] K. Simonyan and A. Zisserman, "Very deep convolutional networks for large-scale image recognition," International Conference on Learning Representations (ICLR), 2015.

[4] Szegedy et al., "Going deeper with convolutions," Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), pp. 1–9, 2015.

[5] R. He, J. Sun, and X. Tang, "Delving deep into rectifiers: Surpassing human-level performance on ImageNet classification," IEEE International Conference on Computer Vision (ICCV), pp. 1026–1034, 2015.

[6] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), pp. 770–778, 2016.

[7] K. He, X. Zhang, S. Ren, and J. Sun, "Identity mappings in deep residual networks," European Conference on Computer Vision (ECCV), pp. 630–645, 2016.

[8] G. Huang, Z. Liu, L. van der Maaten, and K. Q. Weinberger, "Densely connected convolutional networks," Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), pp. 4700–4708, 2017.

[9] G. Huang, S. Liu, L. van der Maaten, and K. Q. Weinberger, "CondenseNet: An efficient DenseNet using learned group convolutions," Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), pp. 2752–2761, 2018.

[10] G. Howard et al., "MobileNets: Efficient convolutional neural networks for mobile vision applications," arXiv preprint arXiv:1704.04861, 2017.

[11] M. Sandler, A. Howard, M. Zhu, A. Zhmoginov, and L. Chen, "MobileNetV2: Inverted residuals and linear bottlenecks," Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), pp. 4510–4520, 2018.

[12] F. Chollet, "Xception: Deep learning with depthwise separable convolutions," Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), pp. 1251–1258, 2017.

ISSN: 3071-3498
PP: 01-14.

[13] M. Tan and Q. V. Le, "EfficientNet: Rethinking model scaling for convolutional neural networks," Proceedings of the 36th International Conference on Machine Learning (ICML), pp. 6105–6114, 2019.

[14] S. Han, H. Mao, and W. J. Dally, "Deep compression: Compressing deep neural networks with pruning, trained quantization and Huffman coding," International Conference on Learning Representations (ICLR), 2016.

[15] J. Deng et al., "ImageNet: A large-scale hierarchical image database," Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), pp. 248–255, 2009.