International Journal of Scientific Research in Engineering and Management (IJSREM)
Volume: 08 Issue: 10 | Oct - 2024 SJIF Rating: 8.448 ISSN: 2582-3930

DenseNet: Assessing Limitations in Connectivity and Memory Efficiency

Mohammad Hamza Parvez Alam Siddiqui
hamzasiddiqui7531@gmail.com

***Abstract*—Despite the prevailing trend of introducing skip connections in deep neural networks for enhanced depth and training efficiency, this paper critically examines the limitations of Dense Convolutional Network (DenseNet), a prominent state- of-the-art architecture. The analysis highlights the drawbacks of connecting each layer to every other layer in a feed-forward manner. Additionally, this paper systematically explores the shortcomings and challenges associated with DenseNet, shedding light on its drawbacks. The study culminates in the identification of critical areas for improvement and suggests avenues for optimizing memory utilization during training. Ultimately, this paper aims to provide a comprehensive understanding of the inherent limitations in DenseNet architecture and offers insights into potential advancements in deep neural network design.**

I. Introduction

Convolution Brain Organizations (CNNs) [1] have prompted a change in outlook in the field of PC Vision. CNNs have, as of late, accomplished their best in class by bringing about different picture acknowledgment and item location errands. This has been made conceivable by the new progressions in GPU equipment and the accessibility of enormous, dependable datasets. Upgrades in PC equipment and organized structures have empowered the preparation of exceptionally profound CNNs. It has additionally made the preparation interaction quicker and more productive. LeNet5 [2] presented the prin- cipal CNN engineering comprising only five layers. Followed by this, VGGNet was presented with 19 layers, and of late, Remaining Organizations (ResNets) [3] have outperformed the 100-layer hindrance.

As the number of layers in a profound neural network increases, it is accompanied by the vanishing gradient problem. ResNets [3] and Highway Networks [4] bypass the signal from one layer to the next via identity connections. This counters the issue of vanishing gradient. However, FractalNets (Ultra-Deep Neural Networks without Residuals) [5], without including any pass-through or residual connections, achieve a comparable performance to deep Residual Networks. This demonstrates that residual representations may not be fundamental to the success of intense convolutional neural networks.

Hunag et [6] proposed engineering that tends to the disappearing slope issue with a straightforward availability example to guarantee the most significant data stream between layers in the organization. In this strategy, all layers (with matching element map sizes) are associated straightforwardly with one another. To save the feed-forward nature, each layer acquires extra contributions from every single going before layer and gives its component guides to every single resulting layer.

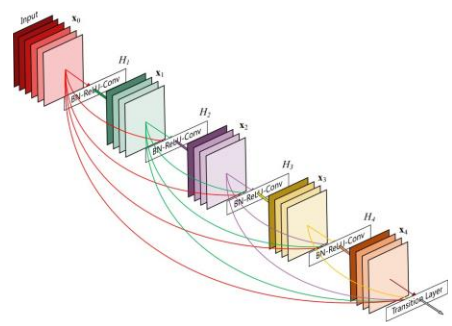

Fig. 1. A 5-layer dense block with a growth rate of k = 4. Each layer takes all preceding feature maps as input.

Figure 1 shows this format schematically. Essentially, as opposed to ResNet, the elements are never joined through summation before they are passed into a layer. The highlights are joined by connection. Subsequently, the `th` ayer has

`sources of info comprising of the component guide`
former convolutional blocks. Its component maps are given to all the resulting layers. This presents $L(L + 1)/2$ associations in a

`- layer organization rather than just \verb` as- sociations, as in customary designs. As a result of its thick availability design, the engineering is known as a Thick Convolutional Organization (DenseNet).

II. DENSENET

The conventional convolutional feed-forward network in- terfaces the result of one layer as the contribution to the following layer. ResNet [3] further adds a skip association with sidestep the layers with a character capability. This assists the angle with streaming straightforwardly through the personality capability.

DenseNet, then again, presents direct associations from any layer to every ensuing layer. Thus, the `th` ayer gets the elements from every former layer, `x0, . . . , xl-1`, as information:

$$x_l = H_l([x_0, x_1, \ldots, x_{l-1}])$$

Where `[x0, x1, . . . , xl-1]` refers to the concatenation of the feature-maps

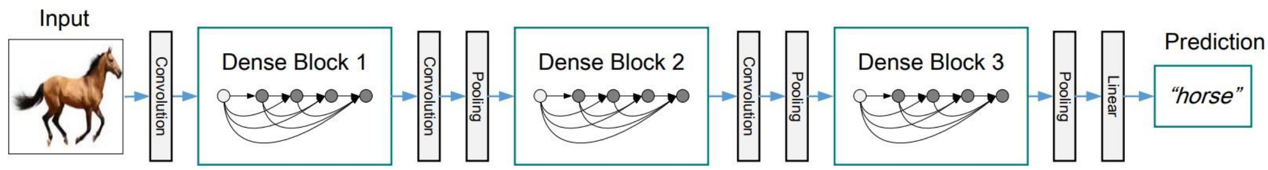

Fig. 2. A profound DenseNet with three thick blocks. The layers between two neighboring blocks are alluded to as progress layers, and the component map sizes are changed by means of convolution and pooling.

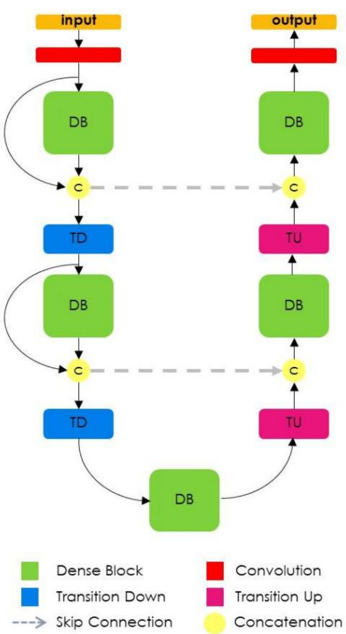

Fig. 3. The architecture has 2 Transition Up (TU) blocks an 2 Transition Down (TD) blocks and gray arrows represent the skip connections from TU blocks to TD blocks.

produced in layers `0. . . . . l-1` and `l is any nonlinearity function.`

III. Experiments

A. Performance Evaluation

DenseNet obtains significant improvements over the state- of-the-art results on four highly competitive object recognition benchmark tasks (CIFAR-10[7], CIFAR-100[8], SVHN[9], and ImageNet [10]). To verify that the DenseNet is as good as mentioned by the authors, it was tested on a Kaggle competition task (Dog-Breed Identification [11]). The results were compared with ResNet [3] and InceptionNet[12] that won the ImageNet [10] Image Recognition Challenge in the past. As seen in Table I, DenseNet performs significantly better (89.9% accuracy) than ResNet and InceptionNet for Dog- Breed Identification [11]. The performance of DenseNet was

Model	Data Augmentation
ResNet50	88.2%
InceptionNet	87.4%
DenseNet	89.9%

TABLE I
Accuracy of different model architectures on Dog-Breed Identification task.

Model	Without Data Augmentation
ResNet50	82.7%
InceptionNet	81.9%
DenseNet	86.4%

TABLE II
Accuracy of different model architectures on Dog-Breed Identification task without data augmentation.

even more significant than other models when no prepro- cessing was performed (as seen in Table II). It shows that connecting every layer's output to each subsequent layers allowed the model to learn the features more effectively.

Furthermore, to evaluate the performance of DenseNet, it was applied to the task of semantic segmentation. CNNs are used in the state-of-the-art methods for semantic segmentation. The architecture for most tasks performs (1) a downsampling of the image to get the features, (2) then an upsampling to get the image back, (3) and post-processing like Conditional Ran- dom Field (CRF)[13], and pre-processing modules like data augmentation is done to get better results. A lot of different CNN architectures were tried and DenseNet achieved the state- of-the-art results as observed by Jegou [14]. Additionally, they also achieved state-of-the-art results on urban scene bench- mark datasets, such as CamVid[15] and Gatech, without any post-processing or pre-training. This shows the effectiveness of DenseNet on a task other than Image Recognition or Object Detection.

B. Memory Consumption

DenseNet implementation, in general, requires higher mem- ory, as seen in Figure 4. If not adequately managed, pre- activation, batch normalization, and contiguous convolution operations can produce feature maps that grow quadratically with the network depth. To counter these problems, the concept of Shared Memory Storage was proposed by Pleiss [16]. During the forward pass, all intermediate outputs were

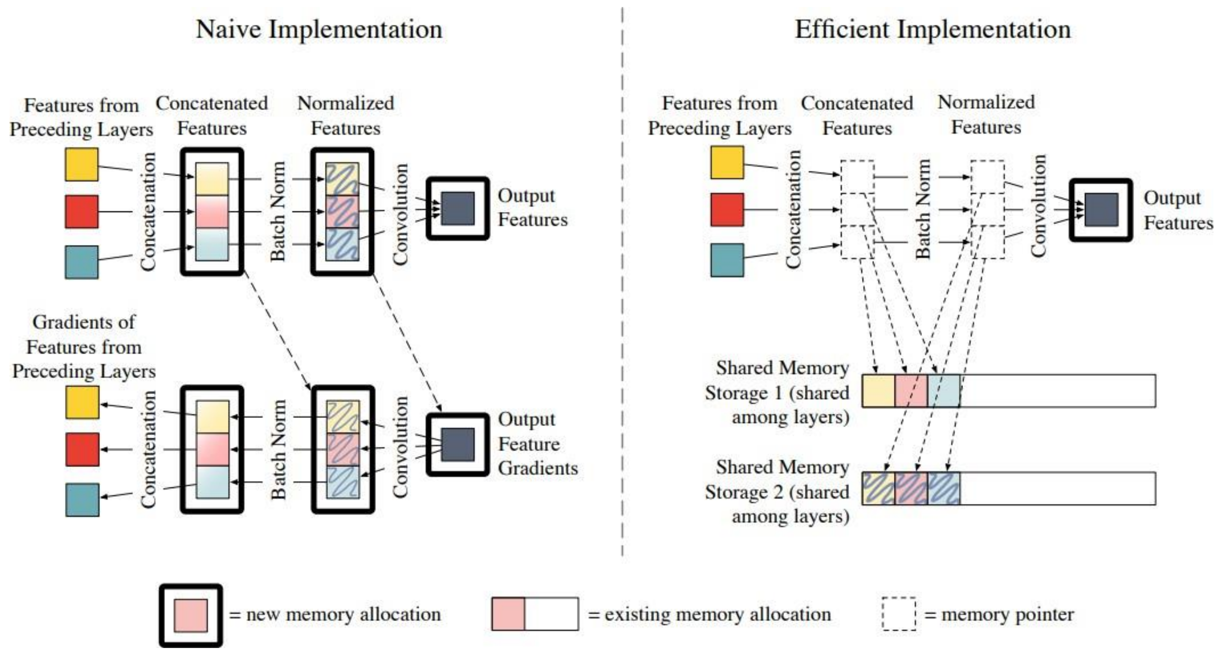

Fig. 4. Unique DenseNet execution is on the left, and its memory-effective execution is on the right. The memory-effective execution stores the result of the link, clump standardization, and ReLU layers in brief capacity cradles, though the first execution designates new memory.

assigned to shared memory blocks. During back-propagation, the concatenated and normalized features were computed dynamically.

C. Training Time

DenseNet's training time per epoch is significantly lower than ResNet and InceptionNet on the same dataset. The experiments showed a decrease of 50 % in training time for DenseNet per epoch. However, to achieve the best possible results, DenseNet needs to be trained over 100 epochs as compared to 20 epochs for ResNets. This is five times more than the other models. So, a tradeoff exists between training time and accuracy for DenseNet.

IV. Conclusion

A large community of researchers believed that the ap- proach of DenseNet was not very novel and that it had been directly/indirectly applied to a lot of tasks earlier. However, when observed closely, the authors put a lot of thought into designing the architecture. This has been proved by the state-of-the-art results achieved by DenseNet in a variety of Image Recognition competitions. The concept has been further tested by a series of experiments performed in this paper and by several other researchers[14]. Overall, DenseNet is not a groundbreaking idea, but the way it is implemented has achieved the desired results and has proven helpful for various tasks.

V. Future Scope

The core idea of DenseNet has opened a lot of research areas. It has already been applied in many domains (Document Analysis, Image Recognition, etc.), and it is believed it will be used in many other areas in the near future. It would be interesting to see if an architecture similar to DenseNet can solve the need for a huge amount of data for training deep learning models. The idea of DenseNet can be combined with the concept of ResNet (introduce skip connections) to avoid learning redundant features over the layers. This is hypothesized to improve the overall training time for DenseNet and to solve the problem of overfitting.

References

[1] 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 Comput.*, vol. 1, no. 4, pp. 541–551, Dec. 1989. [Online]. Available: http://dx.doi.org/10.1162/neco.1989.1.4.541

[2] Y. Lecun, L. Bottou, Y. Bengio, and P. Haffner, "Gradient-based learning applied to document recognition," in *Proceedings of the IEEE*, 1998, pp. 2278–2324.

[3] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," *CoRR*, vol. abs/1512.03385, 2015. [Online]. Available: http://arxiv.org/abs/1512.03385

[4] R. K. Srivastava, K. Greff, and J. Schmidhuber, "Training very deep networks," in *Advances in Neural Information Processing Systems 28*, C. Cortes, N. D. Lawrence, D. D. Lee, M. Sugiyama, and R. Garnett, Eds. Curran Associates, Inc., 2015, pp. 2377–2385. [Online]. Available: http://papers.nips.cc/paper/5850-training-very-deep-networks.pdf

[5] G. Larsson, M. Maire, and G. Shakhnarovich, "Fractalnet: Ultra-deep neural networks without residuals," *CoRR*, vol. abs/1605.07648, 2016. [Online]. Available: http://arxiv.org/abs/1605.07648

[6] G. Huang, Z. Liu, and K. Q. Weinberger, "Densely connected convolutional networks," *CoRR*, vol. abs/1608.06993, 2016. [Online]. Available: http://arxiv.org/abs/1608.06993

[7] A. Krizhevsky, V. Nair, and G. Hinton, "Cifar-10 (canadian institute for advanced research)." [Online]. Available: http://www.cs.toronto.edu/~kriz/cifar.html

[8] ——, "Cifar-100 (canadian institute for advanced research)." [Online]. Available: http://www.cs.toronto.edu/~kriz/cifar.html

[9] Y. Netzer, T. Wang, A. Coates, A. Bissacco, B. Wu, and A. Y. Ng, "Reading digits in natural images with unsupervised feature learning," in *NIPS Workshop on Deep Learning and Unsupervised Feature Learning 2011*, 2011. [Online]. Available: http://ufldl.stanford.edu/housenumbers/nips2011_housenumbers.pdf

[10] O. Russakovsky, J. Deng, H. Su, J. Krause, S. Satheesh, S. Ma, Z. Huang, A. Karpathy, A. Khosla, M. S. Bernstein, A. C. Berg, and F. Li, "Imagenet large scale visual recognition challenge," *CoRR*, vol. abs/1409.0575, 2014. [Online]. Available: http://arxiv.org/abs/1409.0575

[11] A. Khosla, N. Jayadevaprakash, B. Yao, and L. Fei-Fei, "Novel dataset for fine-grained image categorization," in *First Workshop on Fine-Grained Visual Categorization, IEEE Conference on Computer Vision and Pattern Recognition*, Colorado Springs, CO, June 2011.

[12] C. Szegedy, W. Liu, Y. Jia, P. Sermanet, S. E. Reed, D. Anguelov, D. Erhan, V. Vanhoucke, and A. Rabinovich, "Going deeper with convolutions," *CoRR*, vol. abs/1409.4842, 2014. [Online]. Available: http://arxiv.org/abs/1409.4842

[13] J. D. Lafferty, A. McCallum, and F. C. N. Pereira, "Conditional random fields: Probabilistic models for segmenting and labeling sequence data," in *Proceedings of the Eighteenth International Conference on Machine Learning*, ser. ICML '01. San Francisco, CA, USA: Morgan Kaufmann Publishers Inc., 2001, pp. 282–289. [Online]. Available: http://dl.acm.org/citation.cfm?id=645530.655813

[14] S. Je´gou, M. Drozdzal, D. Va´zquez, A. Romero, and Y. Bengio, "The one hundred layers tiramisu: Fully convolutional densenets for semantic segmentation," *CoRR*, vol. abs/1611.09326, 2016. [Online]. Available: http://arxiv.org/abs/1611.09326

[15] G. J. Brostow, J. Shotton, J. Fauqueur, and R. Cipolla, "Segmentation and recognition using structure from motion point clouds," in *ECCV (1)*, 2008, pp. 44–57.

[16] G. Pleiss, D. Chen, G. Huang, T. Li, L. van der Maaten, and K. Q. Weinberger, "Memory-efficient implementation of densenets," *CoRR*, vol. abs/1707.06990, 2017. [Online]. Available: http://arxiv.org/abs/1707.06990