International Journal of Psychosocial Rehabilitation, Vol. 17, Issue 02, 2012
ISSN: 1475-7192

A Novel Deep Learning Based Approach For Image Classification

A. Viswanathan,
Associate Professor in CSE,
KSR College of Engineering,Tiruchengode-637 215.
professorvichu@gmail.com

M. Umamaheswari,
Assistant Professor in CSE,
KSR College of Engineering,Tiruchengode-637 215.
umadeena@gmail.com

S. Sivaprakash
Assistant Professor in CSE
K.S.R College of Engineering, Tiruchengode 637215
shivaiter@gmail.com

M. Sathya,
Assistant Professor in CSE
K.S.R College of Engineering, Tiruchengode 637215
sathimanogaran@gmail.com

Abstract

To address the worsening problem, a deep residual learning technique is provided in this paper. Deeper neural networks are more difficult to train since there is less certainty that each stacked layer corresponds to the requisite underlying mapping. For training networks that are substantially deeper than previously employed networks, we proposed a residual learning technique. Instead of learning unreferenced functions, we deliberately reframe the layers to acquire residual functions with reference to the layer inputs. We offer thorough empirical

evidence that indicates residual networks become more adaptable and precise as network depth rises. Also, investigate residual nets with depths of up to 152 layers on the ImageNet dataset, which is 8 levels deeper than VGG networks which has less complexity.

1. INTRODUCTION

Deep convolutional neural networks had enabled a series of picture classification breakthroughs. Deep networks include low/mid/high-level features as well as classifiers organically in an end-to-end multilayer fashion, with the number of stacked layers increasing the "levels" of features (depth). Recent research shows that network depth is important, and the top results support this. All of them make use of "very deep" models that have a depth in the challenging ImageNet dataset. There are several additional nontrivial sight recognition problems. The relevance of depth begs the question: Is it as easy as stacking more layers to develop better networks? The very well issue of vanishing gradient major issue, which limit convergence from the start, hampered progress on this area.This problem has been mostly resolved by normalized initialization and intermediate normalization layers, enabling networks with tens of layers to start convergent for stochastic gradient descent (SGD) with backpropagation.

The decline in training accuracy indicates that not every system is as easy to enhance. Think about the differences between a deeper design and a shallower one that adds additional layers. By using identity mapping for the new levels and copying the remaining layers from the learned shallower model, the deeper model may be solved via construction. Due to the accessibility of this built-in solution, a more in-depth They all use models that are "very deep" and have a depth in the challenging ImageNet dataset. There are plenty further nontrivial visual recognition problems to be discovered. Think about how a shallower design differs from its deeper, multi-layered sibling.By identity mapping additional levels and duplicating the remaining layers from the learned shorter model, the deeper model may be solved. Models shouldn't produce training errors that are greater than their shallower counterparts.

Deep convolutional neural networks have enabled a flurry of discoveries in image categorization. Deep networks contain low, mid, and high-level features as well as classifiers organically in an end-to-end multilayer way, with the number of stacked layers raising the "levels" of features (depth).

The top findings confirm the recent evidence that network depth is significant. They all employ "extremely deep" models with a depth of sixteen to thirty on the challenging ImageNet dataset. There are several other nontrivial visual recognition issues that exist. Think about the differences between a deeper, more layered design and its shallower equivalent. By replicating the remaining layers from the learnt, shorter model, the deeper model may be constructed and solved. The additional levels are identity mapping. A deeper model shouldn't have a higher training error than its shallower counterpart since there is a built-in solution.

2. RELATED WORKS

Rising the depth, evolving the filter type, growing the width, the number of units in every layer and/or the amount of feature maps (connections), shifting the activation function, and decreasing the number of voxels are some methods used to boost the effectiveness of CNNs in aspects of precision or parameters and computation cost.

In low-level vision and computer graphics, the commonly utilized Multigrid approach revises the system as subtasks at several sizes, with each component being in charge of the residual solution between a coarser and finer scale.

Comparable to our concept, "highway networks" include shortcut linkages with gating capabilities. These gates are data-dependent and have variables, in contrary to our identity shortcuts, that have no parameters.

If a gated shortcut gets "closed," the layers in highway networks show non-residual functions (approaching zero). While all information is continually sent through and our identity shortcuts aren't ever closed, our formulations, on the other hand, continually acquires residual functions, necessitating the acquisition of new residual functions.

The proposed model is grounded on more recent pasts and has a tight relationship to DNIN. One of the key differences between CNNs and conventional neural networks is their depth. AlexNet features eight learned layers without taking into account the pooling layers (five convolutional layers and three fully connected ones). In order to lessen the issue of vanishing gradients and speed up convergence, AlexNet is the first architecture to use the rectified linear unit (ReLU) for the activation function. Highway networks, which are analogous to our work, provide shortcut linkages with gating capabilities.

In contrast to our identity shortcuts, which are parameter-free, these gates are data-dependent and have parameters. The layers in highway networks exhibit non-residual functions when a gated shortcut is "closed" (approaching zero). Our formulation, on the other hand, is always learning residual functions; our identity shortcuts are never closed, and all information is continually pushed through, necessitating the learning of more residual functions.

A few of the methods used to boost the effectiveness of CNNs in aspects of precision as well as parameters and computational complexity include rising the depth, shifting the filter type, rising the width, quantity of items of each layer and/or the number of extracted features (connections), modifying convolution parameters or pooling, changing the activation function, and decreasing the number of voxels.

3. METHODOLOGY

Next assessed are the 18-layer and 34-layer residual nets (EfficientNet s). There is only one difference between the basic topologies and the simple nets above: each pair of 33 filters now has a shortcut link attached to it. In the first comparison, we use zero-padding for increasing dimensions and identity mapping for all shortcuts. As a result, they differ from their basic counterparts in that they lack extra parameters.

The 34-layer EfficientNet performs better than the 18-layer EfficientNet when residual learning is present. Additionally, the 34-layer EfficientNet generalizes to validation data and has a much less training error. This suggests that the degradation issue has been well addressed in this setting, and accuracy rises as a result of higher depth.

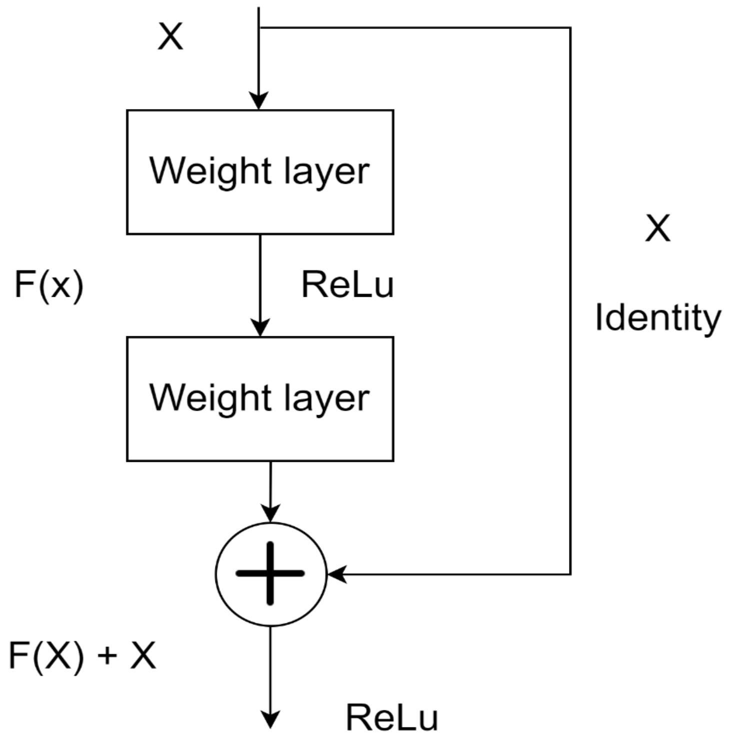

FIGURE 1: LAYER PROCESSING

We describe our deeper networks for ImageNet. Because we are concerned about how much training time we can afford, we alter the construction block to serve as a bottleneck design4. For each residual function F, we use a three-layer stack rather than a two-layer stack. The three layers are 11, 33, and 11 convolutions, with layer 11 recovering dimensions and layer 33 functioning as a bottleneck with smaller input/output dimensions. A situation where the temporal complexity of the two systems is comparable is shown in Figure 5.

When the input and output dimensions are about the same, the identity shortcuts can be utilised immediately (solid line shortcuts). When the dimensions increase (dotted line shortcuts), we have 2 alternatives: (A) The shortcut continues to perform identity mapping, but with more zero entries to account for the extra dimensions. This option adds no additional parameters; (B) To match dimensions, the projection shortcut in Eqn.(2) is utilised (done using 11 convolutions). In all cases, the shortcuts employ a two-stride approach to explore maps of two sizes.

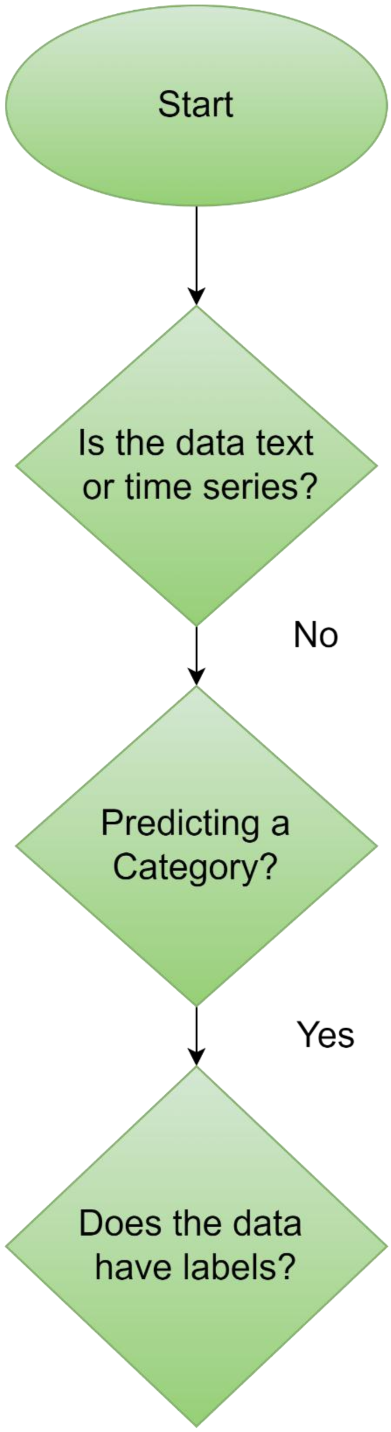

FIGURE 2: WORK FLOW DIAGRAM

Our simple baselines have been influenced by the philosophy of VGG networks (Fig. 2, middle). The convolutional layers usually include 33 filters and follow two fundamental design principles: If the output feature map size remains constant, each layer has the same number of filters, and if it decreases, the number of filters increases to maintain the time complexity per layer.

Direct downsampling is accomplished using convolutional layers with a stride of two. The network is finished with a 1000-way fully-connected layer with softmax and a global average pooling layer. There are a total of 34 weighted layers.

The identity shortcuts can be used immediately when the input and output dimensions are the same (solid line shortcuts). dotted line shortcuts), we have two options as the dimensions increase: Identity mapping is still carried out via the shortcut in (A), but with extra zero entries to take into account the larger dimensions.

This option doesn't introduce any new parameters; (B) the projection short-cut in Eqn. (2) is used to match the dimensions (done using 11 convolutions). The shortcuts take a 2 stride to cross feature maps with two sizes in both options.

4. RESULT & DISCUSSION

The proposed method is evaluated using the 1000-class ImageNet 2012 classification dataset. 1.28 million training pictures and 50,000 validation images are used to train and assess the models. On the 100k test images, we also receive a final result from the test server. The top-1 and top-5 error rates are evaluated.

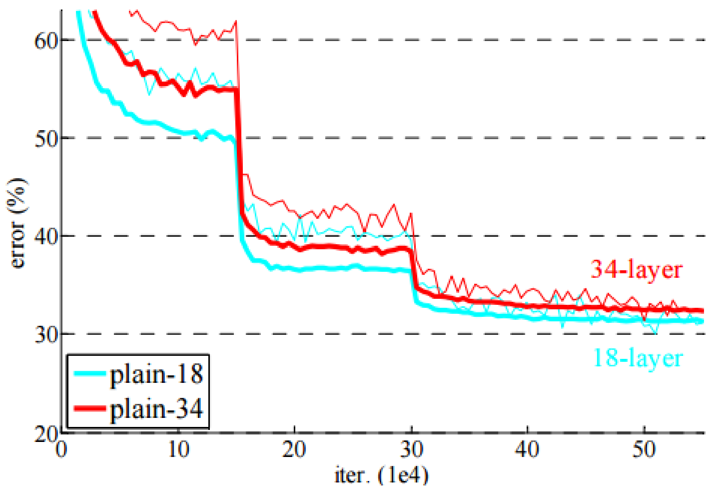

FIGURE 3: ANALYSING THE TRAINING ERROR RAT E

We have done multi-scale testing in our current implementation;however, we have not done multi-scale training owing to time restrictions. Furthermore, only the XceptionNetphase has been tested on a large scale (but not yet for the RPN step). The models are trained and evaluated using 1.28 million training photos and 50k validation images. We also obtained a final result from the test server for the 100k test photographs. Both the top-1 and top-5 error rates are assessed.

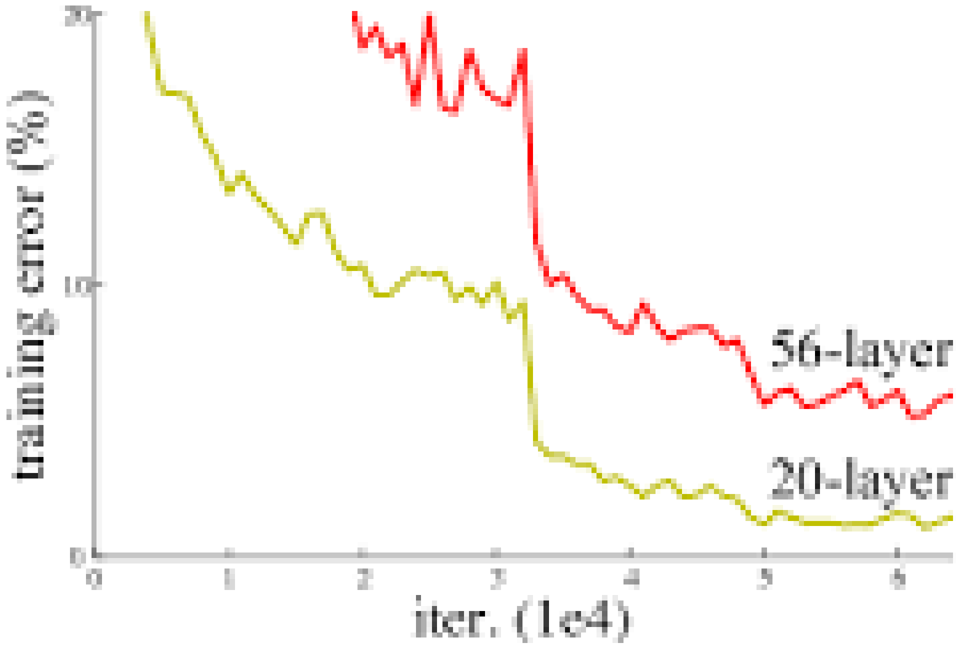

FIGURE 4: ANALYSING THE TRAINING ERROR RATE

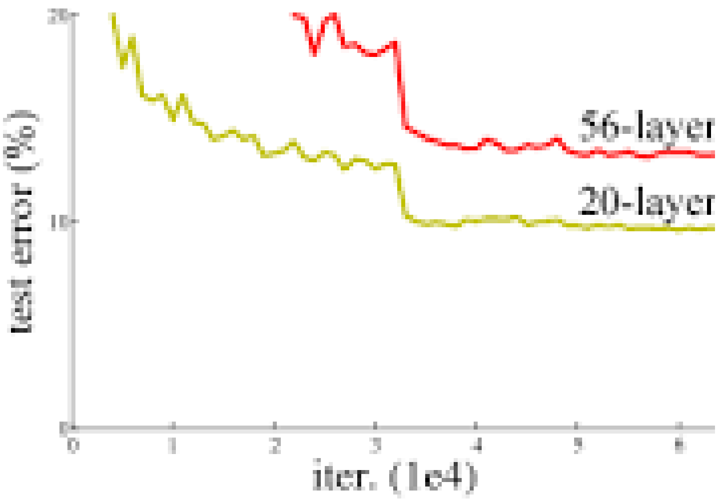

FIGURE 5: ANALYSING THE TEST ERROR RATE

All of the findings in the example above were acquired using single-scale training and testing, with the image's shorter side set at s = 600 pixels. Using maxout layers, multi-scale training and testing has been constructed by choosing a scale from a feature pyramid. Due to time constraints, we have not yet conducted multi-scale training in our present implementation, but we have instead conducted multi-scale testing. Additionally, we have only tested on many scales for the Fast R-CNN phase (but not yet for the RPN step).

CONCLUSION

A novel XceptionNet in network image classification model is proposed in this work. An original nonlinear DrMLPconv filter is employed in this model. To hasten the learning process, this layer is built on a residual block applied to very tiny convolutional filter sizes (3 3) The use of these layers improves categorization accuracy. There is also a suggested, in-depth analytical, and useful DrNIN model that discusses the effects of several layers on improving accuracy in great detail. Future work should focus on creating new CNN models that have accuracy levels comparable to or higher than our proposed model while requiring less training time and parameter consumption.

REFERENCES

1. M. Lin, Q. Chen, and S. Yan, “Network in network,” 2013.
2. L.-C. Chen, G. Papandreou, I. Kokkinos et al., “DeepLab: semantic image segmentation with deep convolutional nets, atrous convolution, and fully connected CRFs,” *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 40, no. 4, p. 834, 2018.
3. S. Zagoruyko and N. Komodakis, “Wide residual networks,” 2016.
4. C. Szegedy, “Going deeper with convolutions,” in *Proceedingsof the 2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 1–9, Boston, MA USA, June 2015.
5. D. Ciresan, U. Meier, and J. Schmidhuber, “Multi-column deep neural networks for image classification,” 2012.
6. K.Gregor and Y. LeCun, “Emergence of complex-like cells in a termporal product network with local receptive fields,” 2010.
7. Krizhevsky, I. Sutskever, and G. Hinton, “Imagenet classification with deep convolutional neural networks,” *Advances in Neural Information Processing Systems*, vol. 25, pp. 1097–1105, 2012.
8. K. Jarrett, K. Kavukcuoglu, M. A. Ranzato, and Y. LeCun, “What is the best multi-stage architecture for object recognition?” in *Proceedings of the IEEE International Conference on Computer Vision*, pp. 2146–2153, Corfu, Greece, September 2009.

9. Y. LeCun, K. Kavukcuoglu, and C. Farabet, “Convolutional networks and applications in vision,” in *Proceedings of the International Symposium on Circuits and Systems (ISCAS 2010)*, pp. 253–256, Paris, France, June 2010.
10. M. D. Zeiler and R. Fergus, “Stochastic pooling for regularization of deep convolutional neural networks,” 2013.