Receptive Field and Neural Networks Representations

Luis Alfredo Avendaño Muñoz. *
School of Computing
University of Leeds
Leeds, United Kingdom
sclaam@leeds.ac.uk

Abstract

1 Introduction

Neural networks have emerged as the standard approach for a wide range of applications, ranging from image recognition [Deng et al., 2009] to natural language processing [Devlin et al., 2019] and speech synthesis [LeCun et al., 2015; van den Oord et al., 2016]. Neural networks owe their expressiveness to the ability to learn hierarchical features and representations form the data that are particularly suited for the task at hand [Goodfellow et al., 2016]. These features are influenced by the highly non-linear.

The reason that make neural networks so good is the features that they learn from the data ([Zhou et al., 2015]), They learning general features in the first layers while in deeper layer they integrate and combine the simpler features and create more complex and detailed features. One aspect that greatly affects these representations, as well as training dynamics is the receptive field of neural networks, yet, its effect on the representations remains not fully understood.

In this paper we systematically explore the relationship that the receptive field has with the representations of neural networks along with the loss landscape these models. We show that if we manipulate the receptive field by changing the kernel size of the first maxpooling layer leaving every other component static we can demonstrate that the models losses generalization capacity as we increase the receptive field. Surprisingly, for high levels of pruning, it turns out that the negative effect on accuracy have an inverse relationship with the receptive field. We show that the receptive fields greatly affects the loss landscape of the models as shown by the Hessian spectra at initialisation.

We empirically validated our results with two different type of models with similar number of parameters (VGG-like and Resnet50) and two different datasets with different image sizes,(CIFAR10 and Tiny imagenet)

Our contributions are:

- We show that the receptive field is linked to the "ruggedness" of the loss landscape of neural networks affecting its trainability and the robustness of its representations to pruning.
-

2 Related Work

Previous works have been done in exploring the receptive field in deep convolutional neural networks. Luo et al. [2016] investigate the *Effective Receptive Field* (ERF) of a neural network. ERF is smaller

*Use footnote for providing further information about author (webpage, alternative address)—*not* for acknowledging funding agencies.

Preprint. Under review.

than the theoretical receptive field and grows as the training progresses. They also showed how subsampling and dilation affect the ERF. In Kobayashi and Shouno [2020] the authors find that ResNets have orientation-selective neurons and double opponent colour neurons by investigating the preferred stimulus of the receptive field for a particular neuron. Lastly, [Kim et al., 2023] is the closest work to ours, in which they try to link the receptive field to the accuracy of the model. This work uses different models of the ResNet and WideResNet family and compares their performance and receptive fields. The main problem with this approach is that any change in accuracy might not be directly attributable to the receptive field given the differences in the number of parameters, depth, and width between different models. Thus, our work measures the effect of only modifying the receptive. Furthermore, our work shows the specific effect that modifying the receptive field has on a neural network loss landscape, namely the "ruggedness" of the loss landscape, as well as its performance in a better controlled environment.

3 Experimental results

In this section we empirically investigate the effect that the receptive field has in a network's trainability and prunability. We Statistically validate our results on VGG and ReseNet50 architecture and in CIFAR10 and Tiny ImageNet datasets.

Here I never talked of how I calculated the receptive field, which I used a libary that calculates the gradient projection in a dummy input space with a projected gradient of 1 in the middle of all the fueature maps of the last convolutional layer for the two architectures.

3.1 Experimental settings

We used a custom implementation of the ResNet50 and VGG models with a modified maxpooling layer after the first convolutional layer for manipulating the receptive field. Implementation details are in **??**. The dataset are CIFAR10 and Tiny ImageNet. The training hyperparameters are the following:

- Epochs: 200
- Optimizer: SGD
- Learning rate: 0.0001
- Learning rate schedule: Cosine annealing with $T_{max} = 200$
- Momentum: 0.9
- Weight Decay: 5×10^{-5}
- Gradient clipping: 0.1

3.2 Manipulating the Receptive Field

There is plentiful of ways for Manipulating the receptive field. It is known that the presence of skip connections affect the receptive fields along with depth and dilation on convolutions. We wanted to alter the receptive field minimum alterations to the rest of the networks leaving all layers with the exact same number of parameters. We placed a maxpooling layer just after the first convolutional layer on both architectures and we changed the kernel size of that layer. That is the only difference between the different models on each experiment. For each architecture-dataset-receptive field combination we trained 5 models and the result are shown in 3.3.

3.3 Receptive Field, Accuracy and Loss Landscape

Here we show the dense accuracy after training the models and its one-shot pruned accuracy. One interesting observation is that larger receptive field correlates with lower accuracy but simultaneously its one-shot pruning performances is better than those for smaller receptive fields. This behaviour generalises across all the combinations of architecture-dataset. In Section 4.1 we show that the current trend in accuracy (for dense and prune version) only arises for sufficiently large pruning rate (>0.8). We also fine-tuned the models to test their real-world application capabilities as seen in Section 4.1 but here we only show the pruning rate of 0.9 as means to show the one-shot behaviour of these

networks in order to understand their one-shot potential. For each one of the combinations shown here we trained and pruned 5 models, the values presented are the mean with its standard deviation.

The image size for CIFAR10 and Tiny ImageNet is 32x32 and 64x64 respectively. In both cases the receptive fields of each architecture is greater than the size of the image.

VGG			
Receptive Field	**Dense Test Accuracy**	**Pruned Test Accuracy**	**Difference in Accuracy**
181	93.5 ± 0.11	10.9 ± 2.03	82.6 ± 2.02
359	91.1 ± 0.23	32.4 ± 15.7	58.7 ± 15.8
537	87.8 ± 0.19	87.6 ± 0.30	0.18 ± 0.16
715	85.8 ± 0.21	85.8 ± 0.22	0.07 ± 0.04
Resnet 50			
Receptive Field	**Dense Test Accuracy**	**Pruned Test Accuracy**	**Difference in Accuracy**
110	94.8 ± 0.29	56 ± 20.7	38.7 ± 20.5
213	94.0 ± 0.16	91.7 ± 0.98	2.25 ± 0.90
318	92.2 ± 0.18	91.4 ± 0.45	0.85 ± 0.49
423	90.4 ± 0.30	90.2 ± 0.37	0.18 ± 0.09

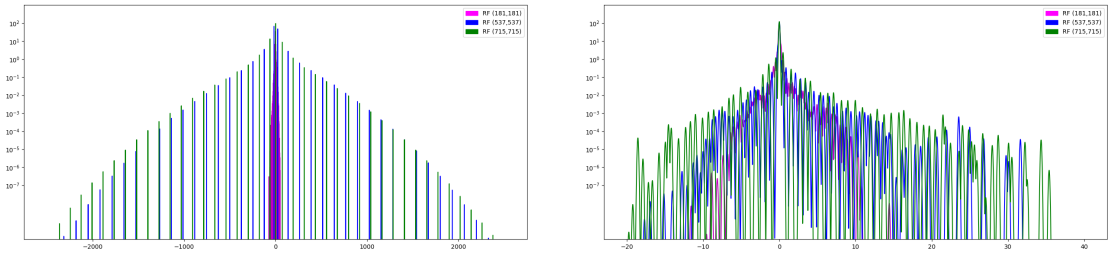
Table 1: **CIFAR10 results:** Here are summarised the results for the experiments performed in CIFAR10. It can be seen that the discrepancy in accuracy between different receptive fields is consistent for these two architectures. Also, as we increase the receptive field we can see that the gap in performance between dense and pruned models diminishes. The pruning rate used is 90%

VGG			
Receptive Field	**Pruned Test Accuracy**	**Dense Test Accuracy**	**Difference in Accuracy**
181	0.75 ± 0.09	61.5 ± 0.33	60.8 ± 0.32
359	0.63 ± 0.17	53.2 ± 0.20	52.6 ± 0.36
537	16.7 ± 7.50	41.0 ± 1.91	24.3 ± 5.81
715	21.8 ± 6.57	38.5 ± 1.69	16.7 ± 7.21
Resnet 50			
Receptive Field	**Pruned Test Accuracy**	**Dense Test Accuracy**	**Difference in Accuracy**
213	5.91 ± 0.89	61.8 ± 0.40	55.9 ± 0.99
318	8.56 ± 2.66	59.1 ± 0.36	50.5 ± 2.55
423	21.4 ± 2.87	56.5 ± 0.27	35.0 ± 2.99

Table 2: **Tiny ImageNet Results:** Here are summarised the results for the experiments performed in Tiny ImageNet. Similarly to CIFAR10, the trend of diminishing dense accuracy and gap between dense and pruned accuracy as we increase the receptive field is consistent in the two architectures. The pruning rate is 90%

Why do models with large receptive field behave in this manner? One hypothesis is that the loss landscape changes in such a way that makes more difficult for SGD to found a better solution. In Figures 1 and 2 we show the 90 largest eigen values for both models on CIFAR10, before and after training. As we can see, at the beginning of training the Hessian spectra of models with alter receptive field are wider and encompass larger eigen values. This means that the landscape of that models has much more steeper directions of descent (and ascent) making the landscape more chaotic and difficult to traverse than for smaller receptive fields (maybe search a citation that corroborates this?).

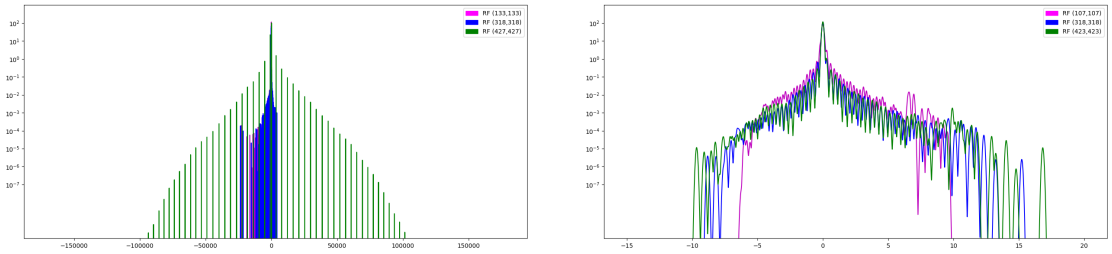
But how does this affect the features or representations that these models learn? Next we show the similarity of the internal representation two different seeds of the ReseNet50 model on CIFAR10., It is observed that the introduction of larger receptive fields results in a divergence of representations within the deepest layers. This phenomenon can be attributed to the heightened "chaotic" nature of the loss landscape, as indicated by the Hessian spectra in Figure 2, particularly in instances where



(a) Before Training (b) After training

Figure 1: Largest 90 eigen values of VGG model on CIFAR10 for different Receptive Fields

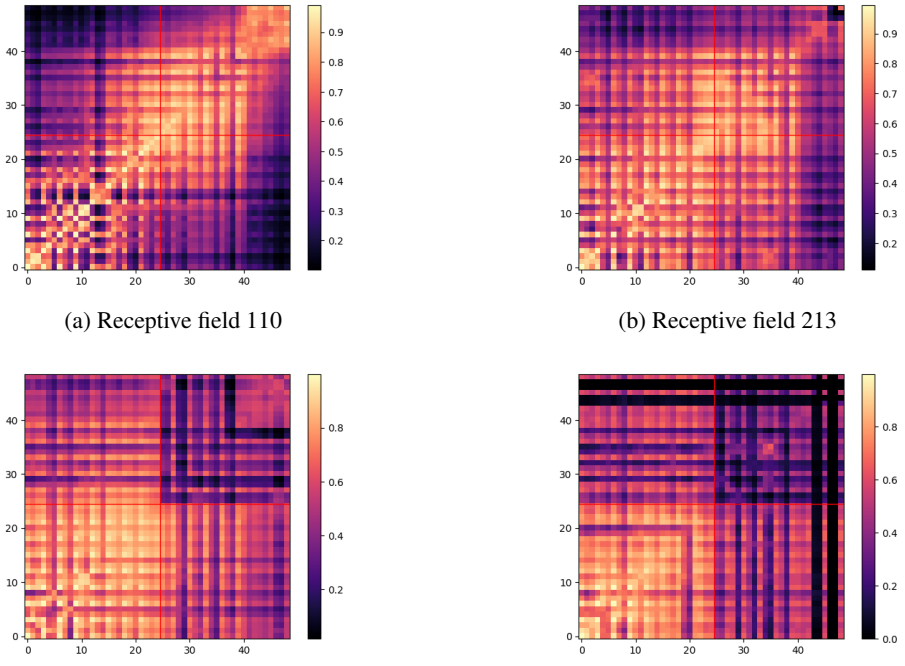
Here I can put labels like λ and $P(\lambda)$. Larger font



(a) Before Training (b) After training

Figure 2: Largest 90 eigen values of ResNet50 model on CIFAR10 for different Receptive Fields

larger receptive fields are employed. Notably, the representations in deeper layers for individual seeds exhibit dissimilarity, as they tend to converge into distinct basins, differing not only from the majority of the network but also from corresponding layers in other seeds. The representation similarly was calculated using the first 1000 images of the test set and we use the Centred linear Kernel Alignment [Kornblith et al., 2019] as similarity measure.



(a) Receptive field 110

(b) Receptive field 213

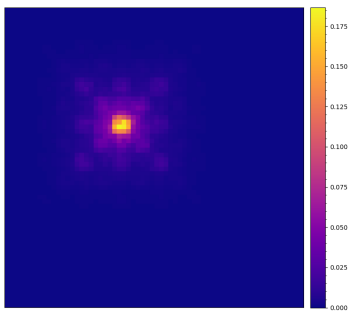
(c) Receptive field 318

(d) Receptive field 423

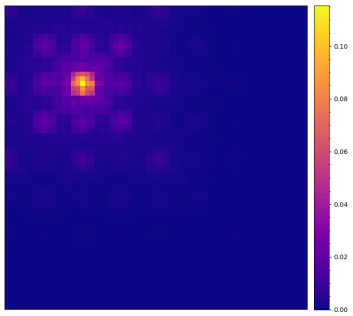
Figure 3: Representation similarly of all layers in ReseNet50 for different receptive fields. The red lines represent the middle layer of the model (25^{th} layer). The largest receptive field (bottom right) has high degree of dissimilarity between its deeper layer representations when compared with the smallest receptive field (top left)

3.4 Receptive Field and Pixel Importance

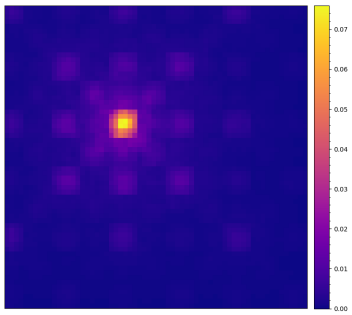
In this section we show the projection of gradient of the last convolutional network into an input space of 64x64. We show that for larger receptive fields the "importance pattern" does not change much but the "importance" decreases from one receptive field to another. This means that even if the model sees similarity for all receptive fields, for larger receptive fields each pixel is less important. We call importance here to the magnitude of the projected gradient on the input space from the last convolutional layer of the randomly initialised model. We place a would be gradient with 1 in the center of all channels of the last convolutional layer and calculate the gradient with respect to a random input with 64x64 dimensions, we take the absolute value of that projected gradient. Then for each receptive field we average 50 of such projections and also take the average of their maximum value. In Figures 4 and 5 we can see the importance pattern for ReseNet50 and VGG. In Figure 5 the check board pattern appears due an odd-sized kernel is applied with an even stride causing what Kim et al. [2023] call *pixel sensitivity imbalance*. In Tables 3 and 4 can be seen the average of the maximum importance value for each receptive field. As we can see the importance for each pixel is decreased as we increase the receptive field. This might explain why having a larger receptive field could hinder performance, since we are paying attention to the same areas of the image but not the same value of attention.



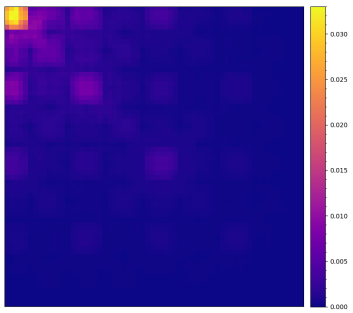
(a) Receptive field 110



(b) Receptive field 213

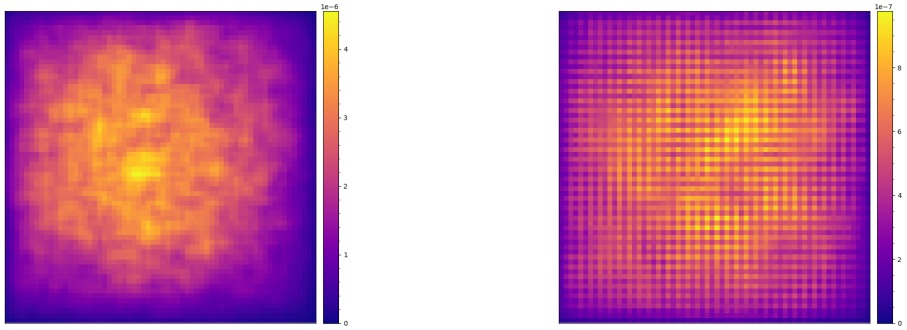


(c) Receptive field 318



(d) Receptive field 423

Figure 4: Projection of the gradient into an input space of 64x64 for different levels of receptive field of ReseNet50



(a) Receptive Field 181

(b) Receptive Field 359

(c) Receptive Field 537

(d) Receptive Field 715

Figure 5: Projection of the gradient into an input space of 64x64 for different levels of receptive field of ReseNet50

Receptive Field	**Maximum Projected Gradient Value**
181	1.24e-05 $\pm$ 2.14e-06
359	2.48e-06 $\pm$ 4.88e-07
537	1.32e-07 $\pm$ 2.48e-08
715	7.26e-08 $\pm$ 1.45e-08

Table 3: VGG gradient projection on a 64X64 input image

Receptive Field	**Maximum Projected Gradient Value**
110	0.322 $\pm$ 0.122
213	0.179 $\pm$ 0.079
318	0.111 $\pm$ 0.039
423	0.064 $\pm$ 0.022

Table 4: Maximum projected gradient for ReseNet50

4 Conclusions and Future work

In this work, we showed how the receptive field affects the loss landscape of neural networks along with its representations and its behaviour on the input image. It is important to note that there are other ways to manipulate the receptive field such as changing the kernel size of the convolutional networks,stride, and dilation of the convolutional networks. In this work, we wanted to have the minimum intervention that would not modify the number of weights of the models as we manipulated the receptive field of the models. Additionally, in our experiments, as we changed the receptive field,

also the size of the feature maps was altered. Future work will concentrate on testing other ways to manipulate the receptive field to test the robustness of the findings in this work.

References

J. Deng, W. Dong, R. Socher, L.-J. Li, K. Li, and L. Fei-Fei. ImageNet: A large-scale hierarchical image database. In *2009 IEEE Conference on Computer Vision and Pattern Recognition*, pages 248–255, June 2009. doi: 10.1109/CVPR.2009.5206848.

J. Devlin, M.-W. Chang, K. Lee, and K. Toutanova. BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 4171–4186, Minneapolis, Minnesota, June 2019. Association for Computational Linguistics. doi: 10.18653/v1/N19-1423.

I. Goodfellow, Y. Bengio, and A. Courville. *Deep Learning*. MIT Press, 2016.

B. J. Kim, H. Choi, H. Jang, D. G. Lee, W. Jeong, and S. W. Kim. Dead pixel test using effective receptive field. *Pattern Recognition Letters*, 167:149–156, Mar. 2023. ISSN 0167-8655. doi: 10.1016/j.patrec.2023.02.018.

G. Kobayashi and H. Shouno. Interpretation of ResNet by Visualization of Preferred Stimulus in Receptive Fields, July 2020.

S. Kornblith, M. Norouzi, H. Lee, and G. Hinton. Similarity of Neural Network Representations Revisited, July 2019.

Y. LeCun, Y. Bengio, and G. Hinton. Deep learning. *nature*, 521(7553):436–444, 2015.

W. Luo, Y. Li, R. Urtasun, and R. Zemel. Understanding the Effective Receptive Field in Deep Convolutional Neural Networks. In *Advances in Neural Information Processing Systems*, volume 29. Curran Associates, Inc., 2016.

A. van den Oord, S. Dieleman, H. Zen, K. Simonyan, O. Vinyals, A. Graves, N. Kalchbrenner, A. Senior, and K. Kavukcuoglu. WaveNet: A Generative Model for Raw Audio. *arXiv:1609.03499 [cs]*, Sept. 2016.

B. Zhou, A. Khosla, A. Lapedriza, A. Oliva, and A. Torralba. Object Detectors Emerge in Deep Scene CNNs, Apr. 2015.

Supplementary Material

4.1 Model Implementation details

Here are the implementation details.

One Shot solutions with multiple pruning rates

For every combination 5 models were trained, pruned and fine-tuned. The error bars correspond to the standard deviation.

VGG			
Receptive Field	**Dense Test Accuracy**	**Pruned TestAccuracy**	**Difference In Accuracy**
181	93.52±0.115	72.22±27.11	21.30±27.18
359	91.15±0.232	90.23±1.32	0.916±1.50
537	87.87±0.193	87.87±0.193	0.0\pm$0.0
715	85.88±0.217	85.880±0.217	0.0±0.0
ResNet50			
Receptive Field	**Dense Test Accuracy**	**Pruned TestAccuracy**	**Difference In Accuracy**
110	94.69±0.213	92.34±1.084	2.350±0.921
213	94.03±0.236	93.81±0.218	0.220±0.234
318	92.22±0.244	92.14±0.227	0.080±0.036
423	90.23±0.169	90.23±0.145	-0.003±0.035

Table 5: vgg and resent50 cifar10 pruning rate 0.8

VGG			
Receptive Field	**Dense Test Accuracy**	**Pruned TestAccuracy**	**Difference In Accuracy**
181	61.58±0.333	19.14±4.610	42.44±4.669
359	53.25±0.207	10.09±1.931	43.16±2.085
537	41.05±1.917	40.09±2.129	0.958±0.360
715	38.57±1.691	37.87±1.566	0.700±0.524
ResNet50			
Receptive Field	**Dense Test Accuracy**	**Pruned TestAccuracy**	**Difference In Accuracy**
213	61.83±0.401	40.56±3.039	21.27±3.170
318	59.10±0.368	43.09±1.880	16.01±1.824
423	56.53±0.279	49.27±0.580	7.260±0.412

Table 6: vgg and resent50 tiny Image net pruning rate 0.8

Pruning Rate 0.7

VGG			
Receptive Field	**Dense Test Accuracy**	**Pruned TestAccuracy**	**Difference In Accuracy**
181	93.52±0.115	93.32±0.104	0.206±0.155
359	91.15±0.232	91.10±0.230	0.058±0.050
537	87.88±0.193	87.88±0.193	0.0±0.0
715	85.88±0.217	85.88±0.217	0.0±0.0
ResNet50			
Receptive Field	**Dense Test Accuracy**	**Pruned TestAccuracy**	**Difference In Accuracy**
110	94.69±0.213	94.24±0.204	0.453±0.012
213	94.03±0.236	93.93±0.203	0.097±0.093
318	92.22±0.244	92.24±0.254	-0.020±0.010
423	90.23±0.169	90.23±0.175	-0.007±0.006

Table 7: VGG and Resnet50 on CIFAR10 and pruning rate 0.7

VGG			
Receptive Field	**Dense Test Accuracy**	**Pruned TestAccuracy**	**Difference In Accuracy**
181	61.58±0.333	48.29±1.390	13.29±1.462
359	53.25±0.207	40.02±2.558	13.23±2.646
537	41.05±1.917	40.96±1.934	0.092±0.070
715	38.57±1.691	38.51±1.658	0.068±0.087
ResNet50			
Receptive Field	**Dense Test Accuracy**	**Pruned TestAccuracy**	**Difference In Accuracy**
213	61.83±0.401	55.33±0.987	6.504±1.194
318	59.10±0.368	54.28±0.398	4.816±0.319
423	56.53±0.279	54.41±0.449	2.128±0.216

Table 8: VGG and Resnet50 on Tiny ImageNet and pruning rate 0.7

Pruning Rate 0.6

VGG			
Receptive Field	**Dense Test Accuracy**	**Pruned TestAccuracy**	**Difference In Accuracy**
181	93.52±0.115	93.48±0.109	0.048±0.050
359	91.15±0.232	91.15±0.240	0.004±0.015
537	87.88±0.193	87.88±0.193	0.000±0.000
715	85.88±0.217	85.88±0.217	0.000±0.000
ResNet50			
Receptive Field	**Dense Test Accuracy**	**Pruned TestAccuracy**	**Difference In Accuracy**
110	94.69±0.213	94.61±0.190	0.080±0.026
213	94.03±0.236	94.01±0.234	0.020±0.020
318	92.22±0.244	92.23±0.225	-0.010±0.020
423	90.23±0.169	90.23±0.169	0.000±0.000

Table 9: vgg and resent50 cifar10 pruning rate 0.6

VGG			
Receptive Field	**Dense Test Accuracy**	**Pruned TestAccuracy**	**Difference In Accuracy**
181	61.58±0.333	56.93±0.512	4.656±0.663
359	53.25±0.207	47.94±1.896	5.304±1.963
537	41.05±1.917	41.05±1.913	-0.000±0.071
715	38.57±1.691	38.58±1.659	-0.008±0.039
ResNet50			
Receptive Field	**Dense Test Accuracy**	**Pruned TestAccuracy**	**Difference In Accuracy**
213	61.83±0.401	59.28±0.468	2.552±0.489
318	59.10±0.368	57.26±0.467	1.840±0.356
423	56.53±0.279	55.97±0.459	0.568±0.212

Table 10: vgg and resent50 tiny Image net pruning rate 0.6

Pruning Rate 0.5

VGG			
Receptive Field	**Dense Test Accuracy**	**Pruned TestAccuracy**	**Difference In Accuracy**
181	93.52±0.115	93.51±0.128	0.010±0.029
359	91.15±0.232	91.16±0.237	-0.004±0.005
537	87.88±0.193	87.88±0.193	0.000±0.000
715	85.88±0.217	85.88±0.217	0.000±0.000
ResNet50			
Receptive Field	**Dense Test Accuracy**	**Pruned TestAccuracy**	**Difference In Accuracy**
110	94.69±0.213	94.71±0.202	-0.023±0.012
213	94.03±0.236	94.02±0.243	0.010±0.026
318	92.22±0.244	92.22±0.229	-0.003±0.015
423	90.23±0.169	90.23±0.169	0.000±0.000

Table 11: VGG and ResNet50 on CIFAR10 with pruning rate 0.5

VGG			
Receptive Field	**Dense Test Accuracy**	**Pruned TestAccuracy**	**Difference In Accuracy**
181	61.58±0.333	59.85±0.241	1.738±0.240
359	53.25±0.207	51.13±0.665	2.116±0.735
537	41.05±1.917	41.05±1.915	-0.004±0.018
715	38.57±1.691	38.57±1.686	0.008±0.013
ResNet50			
Receptive Field	**Dense Test Accuracy**	**Pruned TestAccuracy**	**Difference In Accuracy**
213	61.83±0.401	60.86±0.458	0.966±0.400
318	59.10±0.368	58.46±0.510	0.640±0.401
423	56.53±0.279	56.29±0.408	0.248±0.180

Table 12: VGG and ResNet50 Tiny ImageNet with pruning rate 0.5

Fine-tuning pruned solutions

Here we fine-tuned the pruned solutions while preserving the mask for 10 epochs with the following hyper-parameters

- Initial Learning Rate: 0.0001,
- Weight Decay:5e-4
- Momentum:0,9
- Gradient clip: 0.1

VGG			
Receptive Field	**Dense Test Accuracy**	**Pruned TestAccuracy**	**Difference In Accuracy**
181	93.52±0.115	91.42±2.026	2.106±2.111
359	91.15±0.232	90.81±0.293	0.348±0.366
537	87.88±0.193	87.85±0.223	0.032±0.070
715	85.88±0.217	85.78±0.286	0.096±0.140
ResNet50			
Receptive Field	**Dense Test Accuracy**	**Pruned TestAccuracy**	**Difference In Accuracy**
110	94.69±0.213	90.86±0.701	3.830±0.795
213	94.03±0.236	93.55±0.211	0.477±0.197
318	92.22±0.244	92.03±0.200	0.190±0.053
423	90.23±0.169	90.23±0.147	-0.007±0.087

Table 13: VGG and ResNet50 on CIFAR10 with fine tuning and pruning rate 0.9

VGG			
Receptive Field	**Dense Test Accuracy**	**Pruned TestAccuracy**	**Difference In Accuracy**
181	61.58±0.333	30.21±2.595	31.37±2.534
359	53.25±0.207	3.558±2.027	49.69±2.188
537	41.05±1.917	38.21±2.699	2.842±0.884
715	38.57±1.691	36.45±1.549	2.126±0.762
ResNet50			
Receptive Field	**Dense Test Accuracy**	**Pruned TestAccuracy**	**Difference In Accuracy**
213	61.83±0.401	44.56±1.205	17.27±1.346
318	59.10±0.368	44.28±0.648	14.82±0.633
423	56.53±0.279	46.16±0.716	10.37±0.577

Table 14: VGG and ReseNet50 on Tiny ImageNet with fine tuning and pruning rate of 0.9