

PRESENTATION BY:

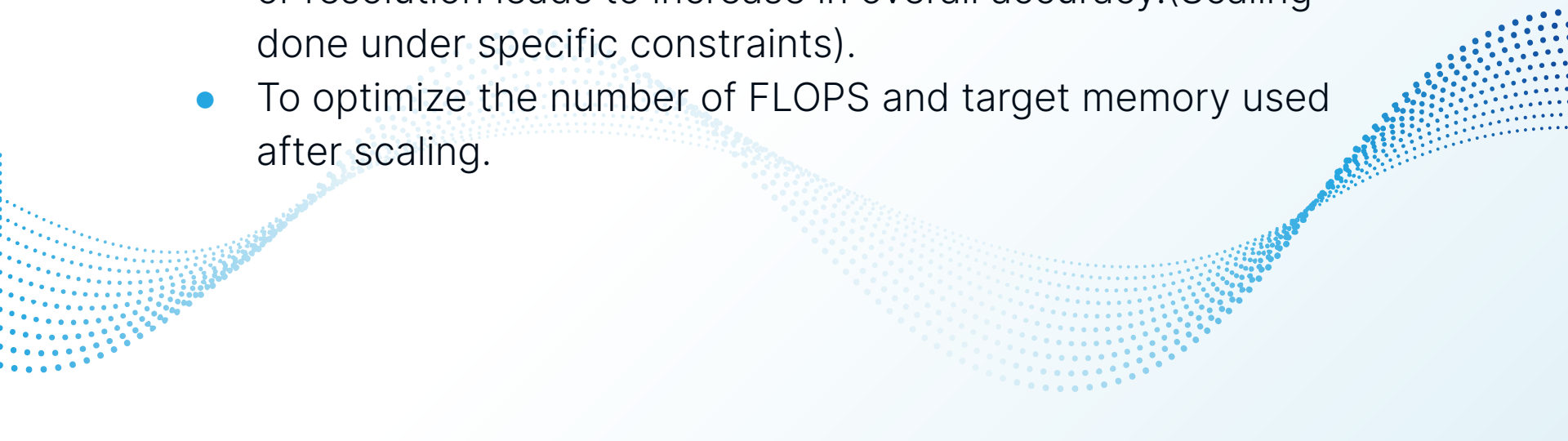
- Sarthak Manoj Ade - 2019A7PS0079P
- Prayaas Karmakar - 2019A3PS0177P
- Abhishek Mishra - 2019A7PS0119P

Efficient Net: Rethinking Model Scaling for Convolution Networks

- MingXing Tan
- Quoc.V.Le



AIM:

- To measure accuracy of our model as a function of scaling and prove the results graphically.
 - To prove that scaling any dimension of network width, depth or resolution leads to increase in overall accuracy. (Scaling done under specific constraints).
 - To optimize the number of FLOPS and target memory used after scaling.
- 
- A decorative graphic consisting of multiple overlapping, wavy lines of blue dots. The dots are arranged in a way that creates a sense of depth and movement, flowing from the bottom left towards the top right, with some lines curving back towards the left.

MOTIVATION:

- In most of the previous models only one dimension of an already existing neural network was scaled up, this paper aims to prove that compound scaling not only archives better accuracy of classification but also optimizes the overall number of FLOPS in the NN architecture.



PREVIOUS IMPLEMENTATIONS:

- https://openaccess.thecvf.com/content_cvpr_2017/papers/Huang_Densely_Connected_Convolutional_CVPR_2017_paper.pdf
- <https://arxiv.org/abs/1605.07146>
- https://openaccess.thecvf.com/content_cvpr_2018/papers/Sandler_MobileNetV2_Inverted_Residuals_CVPR_2018_paper.pdf

ISSUES WITH PREVIOUS MODELS:

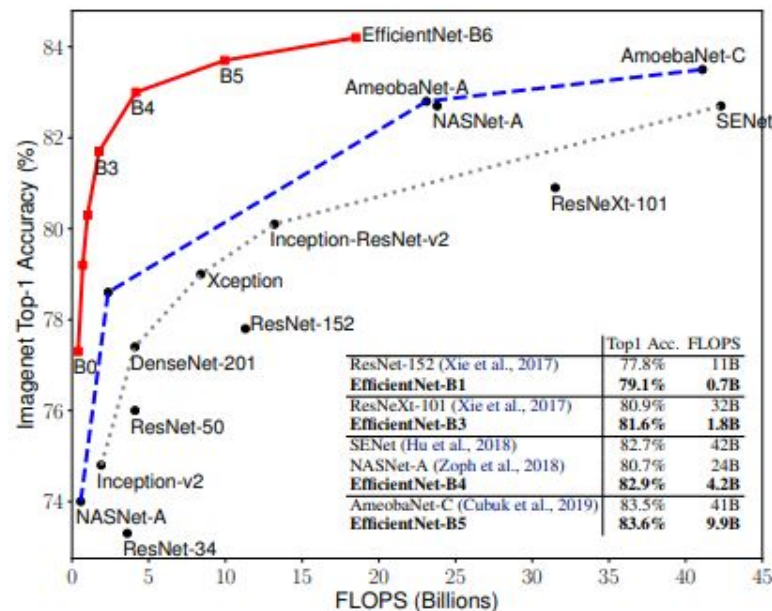
- Heavy depth scaling often resulted in very deep neural networks, this lead to diminishing accuracy due to the problem of vanishing gradients.
- Heavy width scaling helped the network to capture more fine grained features however extremely wide and shallow networks face difficulty to capture high level features.
- Heavy resolution scaling on the other hand leads to stagnation of accuracy which does not increase more on scaling.

METHODOLOGY:

- Compound Scaling using a constant compound coefficient is performed on the Efficient-Net B0 baseline network and the resulting accuracies of the synthesized model are calculated and graphed along with the hyperparameters.
- Φ is initially set to 1 followed by a grid search to find alpha, beta and gamma after which optimal value of Φ is found by iterating over the sample input.

OUTCOME:

- The accuracy of our Efficient nets increases with scaling. Efficient net B7 being the most accurate.



BACKGROUND CONCEPTS:

- In compound scaling instead of scaling all the dimensions separately, all the width, depth and resolution are scaled by constant ratios under some constraints.

$$\text{depth: } d = \alpha^\phi$$

$$\text{width: } w = \beta^\phi$$

$$\text{resolution: } r = \gamma^\phi$$

$$\text{s.t. } \alpha \cdot \beta^2 \cdot \gamma^2 \approx 2$$

$$\alpha \geq 1, \beta \geq 1, \gamma \geq 1$$

$$\max_{d,w,r} \text{Accuracy}(\mathcal{N}(d, w, r))$$

$$\text{s.t. } \mathcal{N}(d, w, r) = \bigodot_{i=1 \dots s} \hat{\mathcal{F}}_i^{d \cdot \hat{L}_i} (X_{\langle r \cdot \hat{H}_i, r \cdot \hat{W}_i, w \cdot \hat{C}_i \rangle})$$

$$\text{Memory}(\mathcal{N}) \leq \text{target_memory}$$

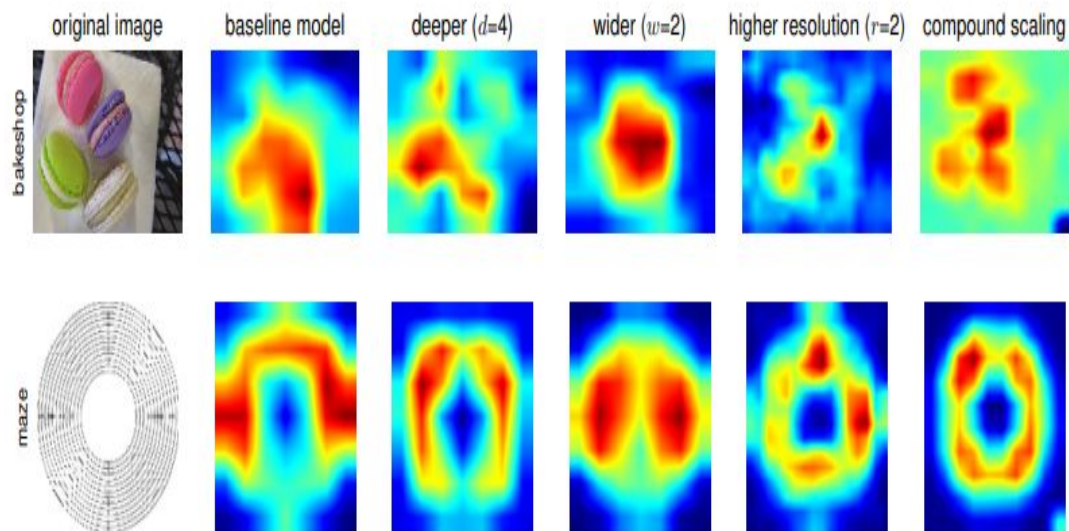
$$\text{FLOPS}(\mathcal{N}) \leq \text{target_flops}$$

BACKGROUND CONCEPTS:

Starting from our baseline model Efficient net B0 we follow two major steps to fine tune the hyperparameters for scaling our model:

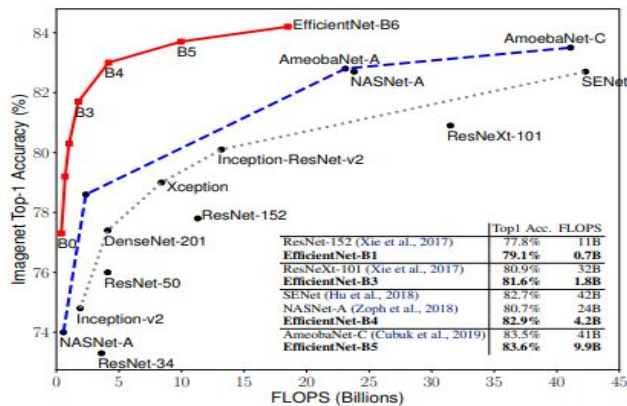
- STEP 1: we first fix $\phi = 1$, assuming twice more resources available, and do a small grid search of α, β, γ based on Equation 2 and 3. In particular, we find the best values for EfficientNet-B0 are $\alpha = 1.2, \beta = 1.1, \gamma = 1.15$, under constraint of $\alpha \cdot \beta^2 \cdot \gamma^2 \approx 2$.
- STEP 2: we then fix α, β, γ as constants and scale up baseline network with different ϕ using Equation 3, to obtain EfficientNet-B1 to B7 (Details in Table 2).

SALIENCY MAP:



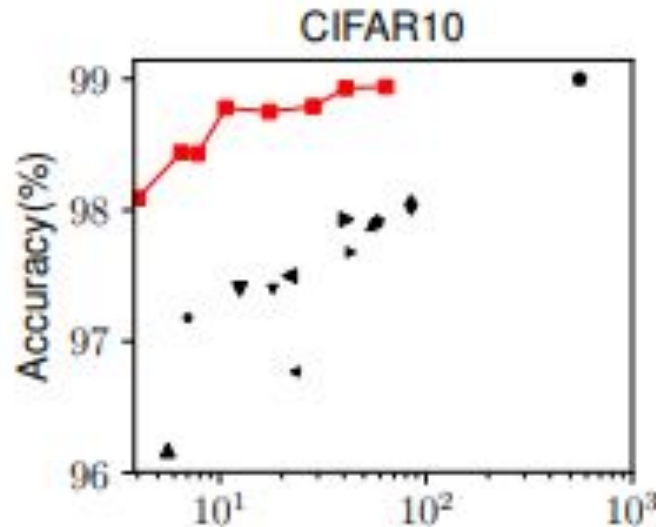
FLOPS USAGE:

- Compound scaling not only optimizes the accuracy of the classification but also optimizes the number of FLOPS in the overall model architecture.

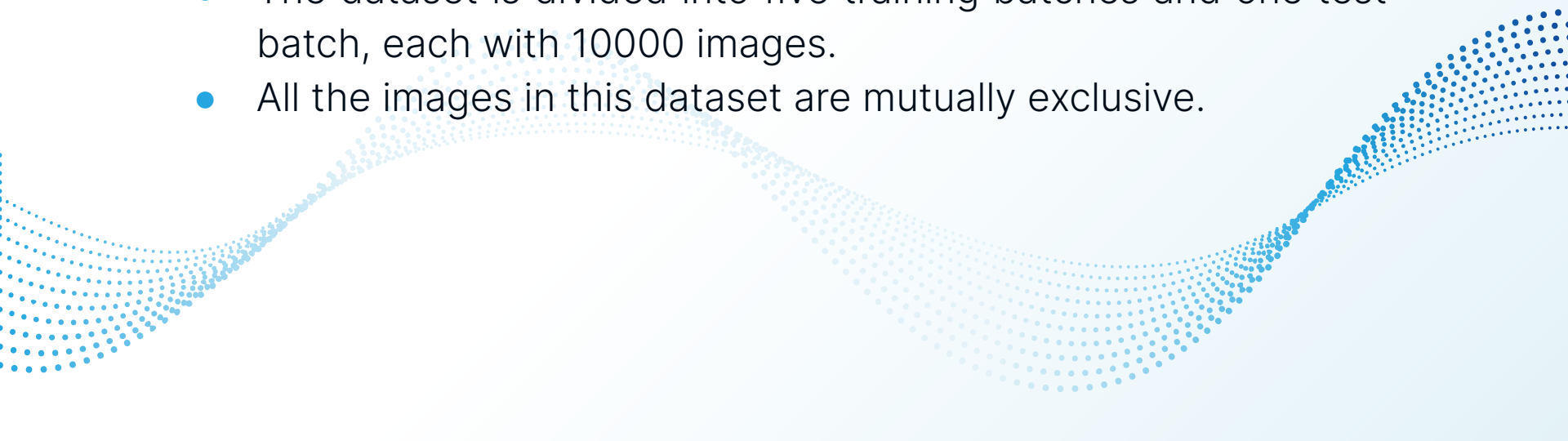


DATASET:

- In our implementation we have scaled resnet-18 model and used CIFAR-10 dataset for testing the accuracy of our model.



DATASET:

- The CIFAR-10 dataset consists of 60000 32×32 colour images in 10 classes, with 6000 images per class. There are 50000 training images and 10000 test images.
 - The dataset is divided into five training batches and one test batch, each with 10000 images.
 - All the images in this dataset are mutually exclusive.
- 
- A decorative graphic consisting of a series of blue dots arranged in a wavy, undulating pattern that spans the bottom half of the slide. The dots are more densely packed in some areas, creating a sense of movement and depth.

HYPOTHESIS:

- Compound Scaling of the Resnet-18 model will result in better accuracy for the classification of the CIFAR-10 dataset.



IMPLEMENTATION:

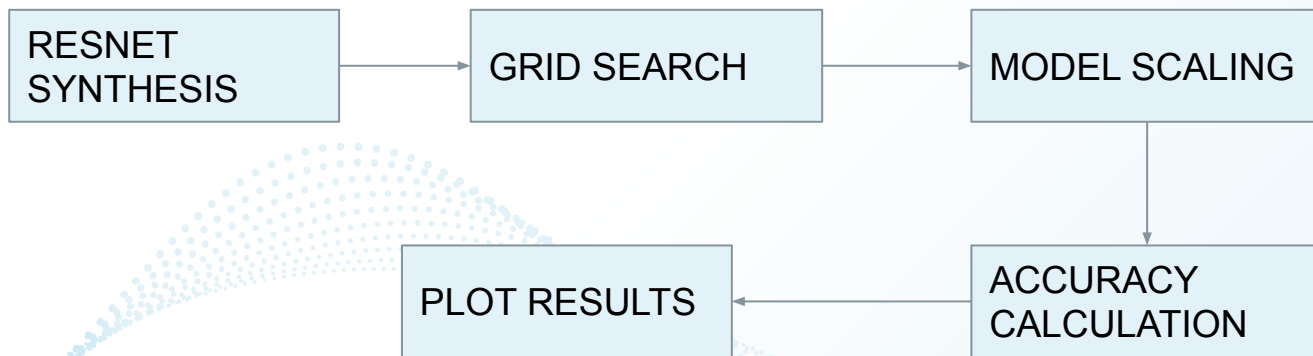
- The Resnet 18 model code was picked up from assignment 2 of this course.
- As grid search was taking a lot of memory and time we decided to pick up the values of the hyperparameters from the research paper directly and scale our model directly on these values.

RESNET ARCHITECTURE

layer name	output size	18-layer	34-layer	50-layer	101-layer	152-layer
conv1	112×112	7×7, 64, stride 2				
conv2_x	56×56	3×3 max pool, stride 2				
		$\begin{bmatrix} 3\times3, 64 \\ 3\times3, 64 \end{bmatrix} \times 2$	$\begin{bmatrix} 3\times3, 64 \\ 3\times3, 64 \end{bmatrix} \times 3$	$\begin{bmatrix} 1\times1, 64 \\ 3\times3, 64 \\ 1\times1, 256 \end{bmatrix} \times 3$	$\begin{bmatrix} 1\times1, 64 \\ 3\times3, 64 \\ 1\times1, 256 \end{bmatrix} \times 3$	$\begin{bmatrix} 1\times1, 64 \\ 3\times3, 64 \\ 1\times1, 256 \end{bmatrix} \times 3$
conv3_x	28×28	$\begin{bmatrix} 3\times3, 128 \\ 3\times3, 128 \end{bmatrix} \times 2$	$\begin{bmatrix} 3\times3, 128 \\ 3\times3, 128 \end{bmatrix} \times 4$	$\begin{bmatrix} 1\times1, 128 \\ 3\times3, 128 \\ 1\times1, 512 \end{bmatrix} \times 4$	$\begin{bmatrix} 1\times1, 128 \\ 3\times3, 128 \\ 1\times1, 512 \end{bmatrix} \times 4$	$\begin{bmatrix} 1\times1, 128 \\ 3\times3, 128 \\ 1\times1, 512 \end{bmatrix} \times 8$
conv4_x	14×14	$\begin{bmatrix} 3\times3, 256 \\ 3\times3, 256 \end{bmatrix} \times 2$	$\begin{bmatrix} 3\times3, 256 \\ 3\times3, 256 \end{bmatrix} \times 6$	$\begin{bmatrix} 1\times1, 256 \\ 3\times3, 256 \\ 1\times1, 1024 \end{bmatrix} \times 6$	$\begin{bmatrix} 1\times1, 256 \\ 3\times3, 256 \\ 1\times1, 1024 \end{bmatrix} \times 23$	$\begin{bmatrix} 1\times1, 256 \\ 3\times3, 256 \\ 1\times1, 1024 \end{bmatrix} \times 36$
conv5_x	7×7	$\begin{bmatrix} 3\times3, 512 \\ 3\times3, 512 \end{bmatrix} \times 2$	$\begin{bmatrix} 3\times3, 512 \\ 3\times3, 512 \end{bmatrix} \times 3$	$\begin{bmatrix} 1\times1, 512 \\ 3\times3, 512 \\ 1\times1, 2048 \end{bmatrix} \times 3$	$\begin{bmatrix} 1\times1, 512 \\ 3\times3, 512 \\ 1\times1, 2048 \end{bmatrix} \times 3$	$\begin{bmatrix} 1\times1, 512 \\ 3\times3, 512 \\ 1\times1, 2048 \end{bmatrix} \times 3$
	1×1	average pool, 1000-d fc, softmax				
FLOPs		1.8×10^9	3.6×10^9	3.8×10^9	7.6×10^9	11.3×10^9

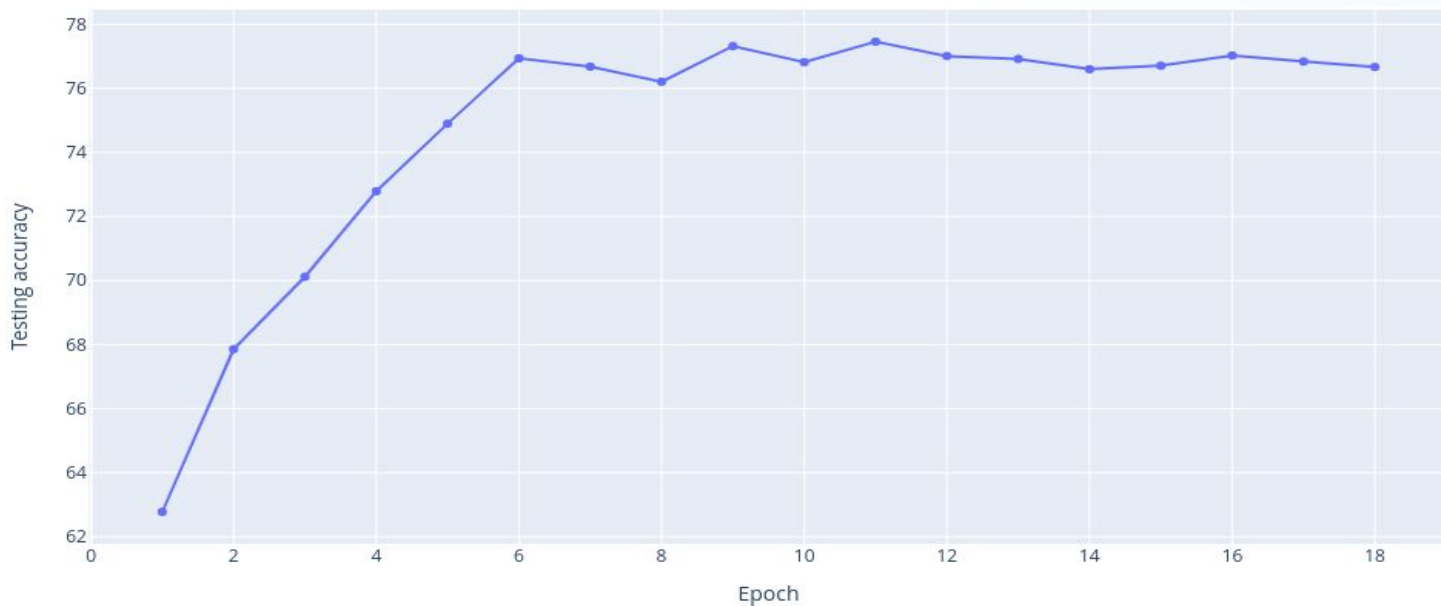
PSEUDO CODE:

- The Basic flowchart of our implementation is as follows:-



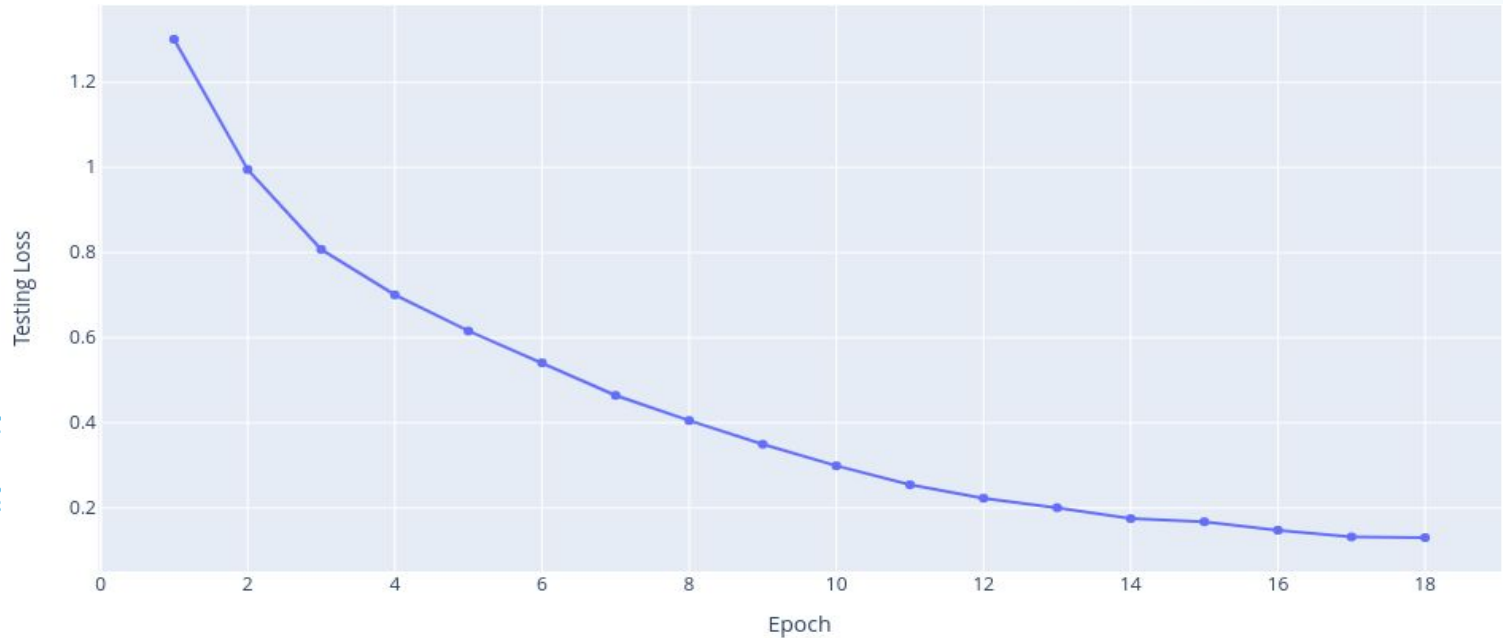
RESULTS:

- The Accuracy of the model increases non linearly with scaling our resnet model.

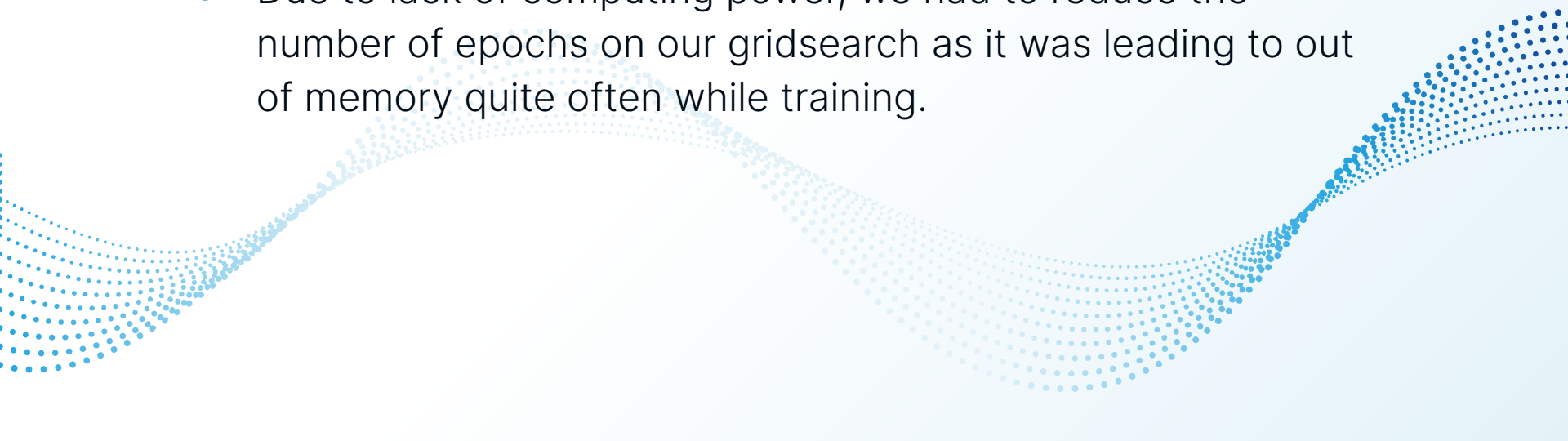


RESULTS:

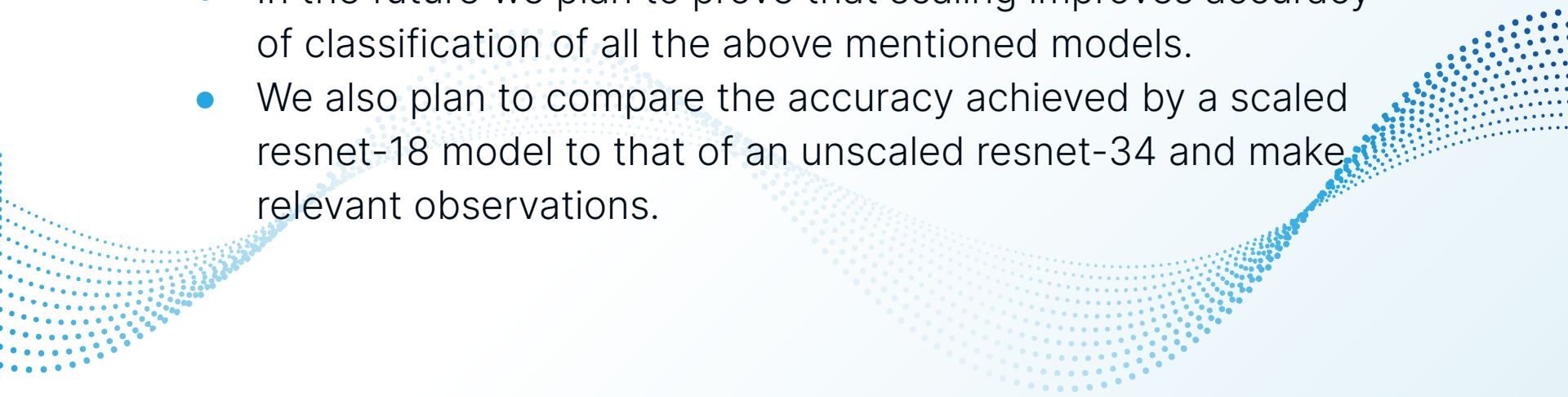
- The graph for the loss function with the number of epochs are shown below.



CHALLENGES FACED:

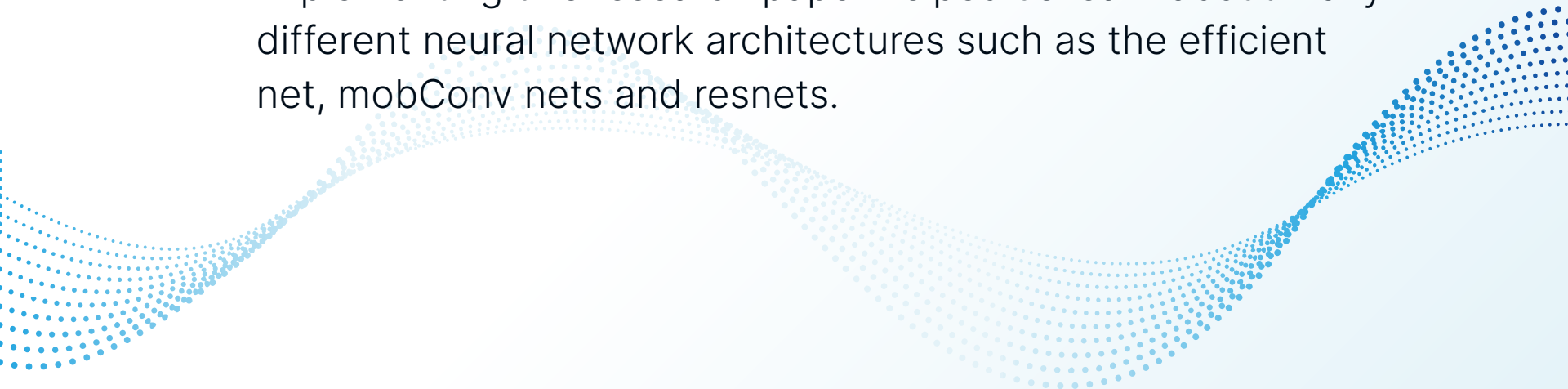
- Lack of proper material related to scaling of resnet 18 lead to little ambiguity on the proper technique to use for the scaling process.
 - Due to lack of computing power, we had to reduce the number of epochs on our gridsearch as it was leading to out of memory quite often while training.
- 

FUTURE SCOPE:

- We Have scaled the Resnet-18 model in this implementation and this can be extended to resnet-34 , resnet-50 and the resnet-101 model.
 - In the future we plan to prove that scaling improves accuracy of classification of all the above mentioned models.
 - We also plan to compare the accuracy achieved by a scaled resnet-18 model to that of an unscaled resnet-34 and make relevant observations.
- 
- A decorative graphic consisting of multiple overlapping, wavy lines of blue dots. The dots are arranged in a way that creates a sense of motion and depth, flowing from the bottom left towards the top right, with some lines curving back towards the left.

LEARNING OUTCOMES:

- Working on this project really helped us improve our coding skills in python and gave us a much deeper insight into the field of neural network based machine learning.
- Implementing this research paper helped us learn about many different neural network architectures such as the efficient net, mobConv nets and resnets.



THANK YOU

A decorative graphic consisting of multiple parallel, wavy lines of small blue dots. These lines flow from the bottom left towards the top right, creating a sense of movement and depth against the solid blue background.