How does topology influence gradient propagation and model performance of deep networks with DenseNet-type skip connections?

Kartikeya Bhardwaj[1,*], Guihong Li[2,*], and Radu Marculescu[2]
[1]Arm Inc., San Jose, CA, USA 95134
[2]The University of Texas at Austin, Austin, TX, USA 78712
kartikeya.bhardwaj@arm.com, {lgh,radum}@utexas.edu

Abstract

DenseNets introduce concatenation-type skip connections that achieve state-of-the-art accuracy in several computer vision tasks. In this paper, we reveal that the topology of the concatenation-type skip connections is closely related to the gradient propagation which, in turn, enables a predictable behavior of DNNs' test performance. To this end, we introduce a new metric called NN-Mass to quantify how effectively information flows through DNNs. Moreover, we empirically show that NN-Mass also works for other types of skip connections, e.g., for ResNets, Wide-ResNets (WRNs), and MobileNets, which contain addition-type skip connections (i.e., residuals or inverted residuals). As such, for both DenseNet-like CNNs and ResNets/WRNs/MobileNets, our theoretically grounded NN-Mass can identify models with similar accuracy, despite having significantly different size/compute requirements. Detailed experiments on both synthetic and real datasets (e.g., MNIST, CIFAR-10, CIFAR-100, ImageNet) provide extensive evidence for our insights. Finally, the closed-form equation of our NN-Mass enables us to design significantly compressed DenseNets (for CIFAR-10) and MobileNets (for ImageNet) directly at initialization without time-consuming training and/or searching.[1]

1. Introduction

DenseNets [7] and their variants have been widely adopted by the deep learning community to achieve excellent performance in many computer vision tasks such as image classification, object detection, image segmentation, super resolution, among many others [39, 8, 38]. One of the main contributions of DenseNets is the introduction of *concatenation-type skip connections* where the output channels from all previous layers are concatenated at the input of the current convolutional layer. The concatenation-type skip connections[2] have been particularly valuable to the deep learning literature. For instance, in addition to significant accuracy and efficiency gains in computer vision applications, many state-of-the-art Neural Architecture Search (NAS) techniques have exploited the concatenation-type skip connections into their search space to obtain high-performance models [24, 16, 26, 17, 40]. However, to the best of our knowledge, the properties of concatenation-type skip connections such as their gradient propagation and the resulting effect on model performance has *not* been explored.

Recently, an important method called Dynamical Isometry emerged in order to quantify gradient flow through DNNs [29, 31, 23, 13]. When DNNs achieve "dynamical isometry", the signal flows through such networks without significant amplification or attenuation. This, in turn, helps the learning process and, hence, quantifies the gradient properties of DNNs. The role of Dynamical Isometry in trainability has been demonstrated for networks such as ResNets [5] which have *addition-type skip connections*.

Due to concatenation of channels, DenseNet-type skip connections enforce strong *structural/topological* constraints on the gradient propagation (*i.e.*, the gradients can only follow specific paths during training). In general, the topology (or structure) of graphs/networks directly influences the process taking place over them [21]. For instance, how closely the users of a social network are connected to each other completely determines how fast the information propagates through the network [14, 9]. Consequently, the structural constraints imposed by concatenation of channels must also affect the learning dynamics of DNNs. Motivated by this observation, we study the relationship between topology, gradient flow, and model performance of such deep networks. Note that, to study these topological properties, we do not use the original DenseNets which contain all-to-all connections [7], but rather a generalized version where we can vary the density of skip connections (more details in Section 3).

To this end, we first define our setup of DNNs with concatenation-type skip connections. Then, we propose a new metric called *NN-Mass* to quantify the topological properties of DNNs considered within this setup. Next, we show

*Equal Contribution
[1]Code at https://github.com/SLDGroup/NN_Mass.
[2]Also referred to as DenseNet-type skip connections.

the relationship between NN-Mass (a topological property) and Layerwise Dynamical Isometry (LDI) [13], a property that indicates the faithful gradient propagation through the network [29]. Specifically, we show that irrespective of number of parameters/FLOPS/layers, models with similar NN-Mass and width should have similar LDI, and thus a similar gradient flow that results in comparable accuracy.

To support these theoretical insights, we conduct extensive experiments to show that models with the same width and NN-Mass indeed achieve a similar accuracy irrespective of their depth, number of parameters (#Params), and FLOPS. Moreover, we empirically show that NN-Mass also works for other types of skip connections, e.g., for ResNets, Wide-ResNets (WRNs), and MobileNets which contain addition-type skip connections (ATSC), *i.e.*, residuals or inverted residuals. Finally, we show how the closed-form expression for NN-Mass can be used to *directly* design compressed DNNs, that is, without *any* time-consuming training and (manual or automatic) searching for compressed models.

Overall, we make the following **key contributions**: (*i*) We reveal how topological constraints imposed by DenseNet-type skip connections influence gradient propagation and resulting accuracy; (*ii*) For this setup, we propose a new topological metric called NN-Mass that is theoretically linked to Layerwise Dynamical Isometry and quantifies how efficiently information propagates in neural networks; (*iii*) Our experiments encompass multilayer perceptron as well as CNNs with DenseNet-type skip connections on several datasets (MNIST, CIFAR-10, CIFAR-100, Imagenet). Our results demonstrate that NN-Mass is an excellent indicator of accuracy and support our theory. We further empirically show that NN-Mass also works for ATSC-based networks (ResNets, WRNs, and MobileNets); (*iv*) Finally, NN-Mass allows us to directly design models with up to $3\times$ compression rate (for DenseNets on CIFAR-10), and up to 34%-40% compression rate (for MobileNet-v2 on ImageNet) in #Params/FLOPS while losing minimal accuracy.

The rest of the paper is organized as follows: Section 2 discusses the related work and some preliminaries. Then, Section 3 describes our proposed metric and its theoretical analysis. Section 4 presents detailed experimental results. Finally, Section 5 summarizes our work and contributions.

2. Background and Related Work

Several prior works aim to study the impact of initialization on model convergence and gradients [11, 4, 29, 25, 31, 23]. To this end, Dynamical Isometry has emerged as an important metric for quantifying gradient properties. Moreover, recent model compression literature attempts to connect pruning at initialization to gradient properties [13]. However, none of these studies address the impact of the *topology* of concatenation-type skip connections on gradient propagation. Instead, since our objective is to specifically study the topological properties, we rely on graph theory/network concepts. Hence, our work is orthogonal to prior art that explores the impact of initialization on gradients as those works do not discuss the impact of topology on gradients.

Recently, random graph concepts have been used in deep learning. For instance, [35, 34] utilize standard random graphs such as Barabasi-Albert (BA) [3] or Watts-Strogatz (WS) [32] models for NAS. However, like other NAS research, [35, 34] do *not* connect the topology with the gradient flow. In contrast, by considering the concatenation-type skip connections, we aim to quantify the link between topology, gradients, and accuracy. Further, while we do build new models as a proof-of-concept of our theoretically grounded metric, we do not conduct any NAS. Conducting full NAS guided by theoretical metrics is left as a future work.

Preliminaries. We use the following established concepts:

Definition 1 (Average Degree [21])**.** *Average degree ($\hat{k}$) of a network represents the average number of connections across all nodes, $\hat{k}$ = #edges/#nodes.*

Average degree and degree distribution (*i.e.*, distribution of nodes' degrees) are important topological characteristics which directly affect how information flows through a network. How fast a signal can propagate through a network heavily depends on the network topology.

Definition 2 (Layerwise Dynamical Isometry (LDI) [13])**.** *A deep network satisfies LDI if the singular values of Jacobians at initialization are close to 1 for all layers. Specifically, for a multilayer feed-forward network, let $\boldsymbol{s_i}$ ($\boldsymbol{W_i}$) be the output (weights) of layer i such that $\boldsymbol{s_i} = \phi(\boldsymbol{h_i}), \boldsymbol{h_i} = \boldsymbol{W_i s_{i-1}} + \boldsymbol{b_i}$; then, the Jacobian matrix at layer i is defined as: $\boldsymbol{J_{i,i-1}} = \frac{\partial \boldsymbol{s_i}}{\partial \boldsymbol{s_{i-1}}} = \boldsymbol{D_i W_i}$. Here, $\boldsymbol{J_{i,i-1}} \in \mathbb{R}^{w_i, w_{i-1}}$, w_i is the number of neurons in layer i. $\boldsymbol{D_i^{jk}} = \phi'(\boldsymbol{h_i})\delta_{jk}$. ϕ' denotes the derivative of non-linearity ϕ and δ_{jk} is Kronecker delta. Then, if the singular values σ_j for all $\boldsymbol{J_{i,i-1}}$ are close to 1, then the network satisfies the LDI.*

LDI discourages the signal propagating through the DNN from getting attenuated or amplified too much; this ensures faithful propagation of gradients [29].

3. Topological Properties of DNNs

We first describe our setup of DNNs with DenseNet-type skip connections and propose the new topological metrics. We then demonstrate the theoretical relationship between the topology and gradient propagation.

3.1. Modeling DenseNet-type Skip Connections

We start with a generic MLP setup with d_c layers containing w_c neurons each and assume DenseNet-type skip connections superimposed on top of a typical MLP structure (see Fig. 1(a)). Henceforth, unless stated otherwise,

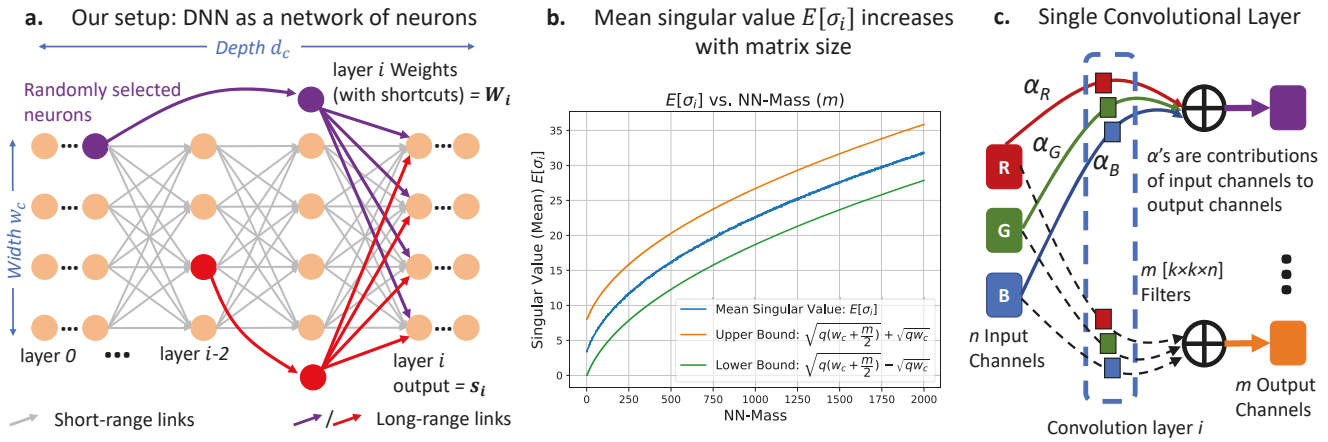

Figure 1: (a) Setup: DNN (depth d_c, width w_c) has layer-by-layer connections due to MLP (gray links) with random concatenation-type skip connections (purple/red links). (b) Simulation of Gaussian matrices sized $(w_c + m/2, w_c)$, where $w_c = 16$: Mean singular values increase as NN-Mass (m) increases and is bounded as shown in Proposition 2 (Appendix D). (c) Convolutional layers form a similar topological structure as MLP: All input channels contribute to all output channels.

DenseNet-type skip connections will be simply referred to as skip connections. Specifically, all neurons at layer i receive skip connections from a maximum of t_c neurons from previous layers. That is, we *randomly* select $\min\{w_c(i-1), t_c\}$ neurons from layers $0, 1, \ldots, (i-2)$, and concatenate them at layer $i-1$ (see Fig. 1(a))[3]; the concatenated neurons then pass through a fully-connected layer to generate the output of layer i ($\boldsymbol{s_i}$). As a result, the weight matrix $\boldsymbol{W_i}$ (which is used to generate $\boldsymbol{s_i}$) gets additional weights to account for the incoming skip connections. Similar to recent NAS research [15], we select links randomly because random architectures are often as competitive as the carefully designed models. Moreover, the random skip connections on top of fixed short-range links make our architectures a small-world network (Fig. 7, Appendix A) [32] which allows us to use graph/network concepts to study their topology.

An important advantage of the above setup is that we can control the density of skip connections (using t_c) to study the topological properties over many DNNs. If the skip connections encompass all-to-all connections, this will result in the original DenseNet architecture. Like standard CNNs (Resnets/DenseNets), we can generalize the setup to contain multiple (N_c) cells of width w_c, depth d_c; skip connections exist only *within* a cell and not across cells.

3.2. Proposed Metrics

Our key objective is to quantify what topological characteristics of DNNs with DenseNet-type skip connections affect their accuracy and gradient flow. We then exploit such properties to *directly* design efficient CNNs by looking at such properties. To this end, we propose two new metrics called *Cell-Density* and *NN-Mass*, as defined below.

Definition 3 (Cell-Density)**.** *Density of a cell quantifies how densely its neurons are connected via skip connections. Formally, for a cell c, cell-density ρ_c is given by:*

$$\rho_c = \frac{\textit{Actual \#skip connections within cell } c}{\textit{Total possible \#skip connections within cell } c} = \frac{2\sum_{i=2}^{d_c-1} \min\{w_c(i-1), t_c\}}{w_c(d_c-1)(d_c-2)} \tag{1}$$

For complete derivation, please refer to Appendix B. Informally, density is basically *mass/volume*. Let *volume* be the total number of neurons in a cell ($w_c \times d_c$). Then, we define the NN-Mass (m) as follows:

Definition 4 (Mass of DNNs)**.** *NN-Mass is defined as the sum (over all cells) of product of Cell-Density (ρ_c) and number of neurons per cell.*

$$m = \sum_{c=1}^{N_c} w_c d_c \rho_c = \sum_{c=1}^{N_c} \frac{2 d_c \sum_{i=2}^{d_c-1} \min\{w_c(i-1), t_c\}}{(d_c-1)(d_c-2)} \tag{2}$$

As explained in the next section, NN-Mass quantifies how effectively information can flow through a given DNN topology. For a given width (w_c), models with similar NN-Mass, but different depths (d_c) and #Params, should exhibit a similar gradient flow and, thus, achieve a similar accuracy. Note that, NN-Mass is a function of network width, depth, and skip connections (*i.e.*, the topology of the network). For a fixed number of cells, an architecture can be completely specified by $\{$*depth, width, maximum skip connection candidates*$\}$ per cell = $\{d_c, w_c, t_c\}$. Hence, to create

[3]Here, $w_c(i-1)$ is the total number of candidate neurons from layers $0, 1, \ldots, (i-2)$ that can supply skip connections; if the *maximum* number of neurons t_c that can supply skip connections to the current layer exceeds total number of possible candidates, then all neurons from layers $0, 1, \ldots, (i-2)$ are selected. Neurons are concatenated similar to how channels are concatenated in DenseNets [7].

different architectures with DenseNet-type skip connections, we vary $\{d_c, w_c, t_c\}$ to create architectures with random #Params/FLOPS/layers, and NN-Mass. We then train these architectures and characterize their accuracy, topology, and gradient propagation to understand the relationships among them. But first, we provide our theoretical analysis.

3.3. Relationships among topology, NN-Mass and gradient propagation

Without loss of generality, we assume that the DNN (same setup as above) has only one cell of width w_c and depth d_c.

Proposition 1 (NN-Mass and average degree). *The average degree of a DenseNet-type deep network with NN-Mass m is given by $\hat{k} = w_c + m/2$.*

The proof of the above result is given in Appendix C.

Intuition. Proposition 1 states that the average degree of a deep network is $w_c + m/2$, which, given the NN-Mass m, is independent of depth d_c. The average degree indicates how well-connected the network is. Hence, it controls how effectively the information can flow through a given topology. Therefore, for a given width and NN-Mass, the average amount of information that can flow through various architectures (with different #Params/layers) should be similar (due to the same average degree). Thus, we hypothesize that these topological characteristics might constrain the amount of information being learned by DNNs. Next, we show the impact of topology on gradient propagation.

Proposition 2 (NN-Mass and LDI). *Consider the case of deep linear networks with concatenation-type skip connections, where each layer is initialized using independently and identically distributed values with initialization variance q. For this setup, suppose we are given a small network f_S (depth d_S) and a large network f_L (depth d_L, $d_L >> d_S$), both with same initialization scheme, NN-Mass m, and width w_c. Then, the mean singular value of the initial layerwise Jacobian ($\mathbb{E}[\sigma]$) for both networks is bounded as follows:*

$$\sqrt{q(w_c + m/2)} - \sqrt{qw_c} \leq \mathbb{E}[\sigma] \leq \sqrt{q(w_c + m/2)} + \sqrt{qw_c}$$

That is, the LDI for both models does not depend on the depth if the initialization variance (q) for each layer is depth-independent (which is the case for many initialization schemes). Hence, for such networks, models with similar width and NN-Mass result in similar gradient properties, even if their depths and #Params are different.

Proof: A formal proof of the above result and the bounds under the deep linear network [29, 12, 1, 10] assumption is given in Appendix D. The discussion below is more informal and explains how the above result works for both linear and non-linear DNNs with DenseNet-type skip connections.

To prove this result, it suffices to show that the initial Jacobians $\boldsymbol{J_{i,i-1}}$ have similar properties for both models (and thus their singular values are similar). For our setup, the output of layer i, $\boldsymbol{s_i} = \phi(\boldsymbol{W_i x_{i-1}} + \boldsymbol{b_i})$, where $\boldsymbol{x_{i-1}} = \boldsymbol{s_{i-1}} \cup \boldsymbol{y_{0:i-2}}$ concatenates output of layer $i-1$ ($\boldsymbol{s_{i-1}}$) with the neurons $\boldsymbol{y_{0:i-2}}$ supplying the skip connections (random $\min\{w_c(i-1), t_c\}$ neurons selected uniformly from layers 0 to $i-2$). Hence, $\boldsymbol{J_{i,i-1}} = \partial \boldsymbol{s_i} / \partial \boldsymbol{x_{i-1}} = \boldsymbol{D_i W_i}$. Compared to a typical MLP (see Definition 2), the sizes of $\boldsymbol{D_i}$ and $\boldsymbol{W_i}$ increase to account for incoming skip connections.

For two models f_S and f_L, the layerwise Jacobian ($\boldsymbol{J_{i,i-1}}$) can have two kinds of properties: (*i*) The distribution of values inside Jacobian matrix for f_S and f_L can be different, and/or (*ii*) The sizes of layerwise Jacobian matrices for f_S and f_L can be different. Hence, our objective is to show that when the width (w_c) and NN-Mass (m) are similar, irrespective of the depth of the model (and thus irrespective of #Params/FLOPS), both the distribution and the size of initial layerwise Jacobians are similar.

Let us start by considering a linear network: in this case, $\boldsymbol{J_{i,i-1}} = \boldsymbol{W_i}$. Since the LDI looks at the properties of layerwise Jacobians *at initialization*, and because all models are initialized the same way (*e.g.*, Gaussians with variance scaling[4]), the values inside $\boldsymbol{J_{i,i-1}}$ for both f_S and f_L have same distribution (*i.e.*, point (*i*) above is satisfied). We next show that even the sizes of layerwise Jacobians for both models are similar if the width and NN-Mass are similar.

How is topology related to the layerwise Jacobians? Since the average degree is same for both models (see Proposition 1), on average, the number of incoming skip connections at a typical layer is $w_c \times m/2$. In other words, since the degree distribution for the random skip connections is Poisson [2] with average degree $\bar{k}_{\mathcal{R}|\mathcal{G}} \approx m/2$ (see Eq. 8, Appendix C), an average $m/2$ neurons supply skip connections to each layer[5]. Therefore, the Jacobians will theoretically have the same dimensions $(w_c + m/2, w_c)$ irrespective of the depth of the neural network (*i.e.*, point (*ii*) is also satisfied).

So far, the discussion has considered only a linear network. For a non-linear network, the Jacobian is given as $\boldsymbol{J_{i,i-1}} = \boldsymbol{D_i W_i}$. As explained in [13], $\boldsymbol{D_i}$ depends on pre-activations $\boldsymbol{h_i} = \boldsymbol{W_i x_{i-1}} + \boldsymbol{b_i}$. As established in several deep network mean field theory studies [25, 31], the distribution of pre-activations at layer i ($\boldsymbol{h_i}$) is a Gaussian $\mathcal{N}(0, q_i)$ due to the central limit theorem. Similar to [13, 23], if the input $\boldsymbol{h_0}$ is chosen to satisfy a fixed point $q_i = q^*$, the distribution of D_i becomes independent of the depth ($\mathcal{N}(0, q^*)$). Therefore, the distribution of both $\boldsymbol{D_i}$ and $\boldsymbol{W_i}$ is similar for different models irrespective of the depth, even for non-linear networks. Moreover, the sizes of the matrices will be similar due to similar average degree in f_S and f_L.

Hence, the size and distribution of values in the Jacobian matrix are similar for both the large and the small model

[4] Variance scaling methods also take into account the number of input/output units. Hence, if the width is the same between models of different depths, the distribution at initialization is still similar.

[5] Poisson process assumes a constant rate of arrival of skip connections.

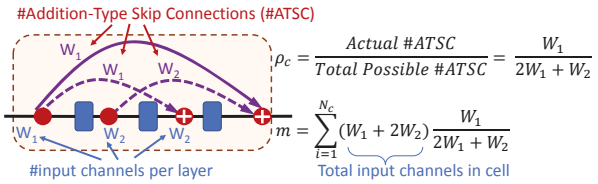

Figure 2: NN-Mass for bottleneck ResNets/MobileNets. Dotted purple lines are possible ATSC (not actually present). Solid purple ATSC are present in MobileNets/ResNets.

(provided the width and NN-Mass are similar); that is, the distribution and mean singular values will also be similar. Thus, LDI is equivalent between different depth DNNs if their width and NN-Mass are similar. As a result, such models have similar gradient flow properties. □

To verify the bounds provided in Proposition 2, we numerically simulate the mean singular values of layerwise Jacobians for deep linear networks using standard Gaussian ($q = 1$) matrices of sizes $(w_c + m/2, w_c)$. Specifically, we vary m for a given width w_c and see the impact of this size variation on mean singular values. Fig. 1(b) shows that as NN-Mass varies, the mean singular values increase and lie within the bounds of Proposition 2. Note that, our results should *not* be interpreted as bigger models yield larger mean singular values. We show in the next section that the relationship between the #Params and mean singular values is significantly worse than that for NN-Mass. Hence, it is the topological properties that enable LDI in different deep networks and *not* the #Params.

Remark 1 (NN-Mass formulation is same for DenseNet-type CNNs). Fig. 1(c) shows a typical convolutional layer. Since all channel-wise convolutions are added together, each output channel is some function of all input channels. This makes the topology of CNNs similar to that of our MLP setup. The key difference is that the nodes in the network (see Fig. 1(a)) are now channels and not individual neurons. Of note, for our CNN setup, we use three cells (similar to DenseNets). More details on CNN setup (including a concrete example for NN-Mass calculations) are given in Appendices E and F.

Remark 2 (NN-Mass generalizes to ResNets and MobileNets). Note that, ResNets/MobileNet-v2 have Addition-Type Skip Connections (ATSC). Then, following Definitions 3 and 4, cell-density (ρ_c) and NN-Mass for ResNets/MobileNets are defined as:

$$\rho_c = \frac{\text{Actual \#ATSC}}{\text{Total possible \#ATSC}}, \quad m = \sum\nolimits_{N_c} i_c \times \rho_c \qquad (3)$$

where, i_c is the total input channels within one cell. For instance, for the bottleneck cells shown in Fig. 2, $i_c = W_1 + 2W_2$, where W_1 and W_2 are number of input channels at various layers within the bottleneck cell. Due to one-to-one, channel-wise additions, the actual #ATSC = W_1 since a maximum of W_1 channels can be added (this cannot exceed the #input channels at the source layer; see solid purple line in Fig. 2). In bottleneck cells, ATSC can be present at two other locations (see dotted purple lines in Fig. 2), each can supply W_1 and W_2 links, respectively. Hence, ρ_c and NN-Mass can be computed as shown in Fig. 2 (right) using Eq. (3). Equations shown in Fig. 2 work for both ResNet and MobileNet bottleneck cells, and a similar process follows for the ResNet-Basicblock cell.

We next provide extensive empirical evidence for our theoretical insights on topology, gradient propagation, LDI, and model accuracy (Proposition 2).

4. Experimental Setup and Results

4.1. Experimental Setup

For experiments on MLPs and CNNs, we generate random architectures (within our setup of DenseNet-type skip connections) with different NN-Mass and number of parameters by varying $\{d_c, w_c, t_c\}$. For random MLPs with different $\{d_c, t_c\}$ and $w_c = 8$ (#cells = 1), we conduct the following experiments on the MNIST dataset: (*i*) We explore the impact of varying #Params and NN-Mass on the test accuracy; (*ii*) We demonstrate how LDI depends on NN-Mass and #Params; (*iii*) We further show that models with similar NN-Mass (and width) result in similar training convergence, despite having different depths and #Params.

After the extensive empirical evidence for our theoretical insights (*i.e.*, the connection between gradient propagation and topology), we next move on to CNN architectures. We conduct the following experiments: (*i*) For three-cell CNNs with random concatenation-type skip connections (*i.e.*, the DenseNet setup), we show that NN-Mass can identify CNNs that achieve similar test accuracy, despite having highly different #Params/FLOPS/layers; (*ii*) We show that NN-Mass is a significantly more effective indicator of model performance than parameter counts; (*iii*) For DenseNet setup, we perform the above experiments for CIFAR-10, CIFAR-100, and ImageNet datasets; (*iv*) We further demonstrate that NN-Mass works for standard ResNets, Wide-ResNets, and MobileNets on ImageNet.

Finally, we exploit NN-Mass to directly design efficient DenseNet-type CNNs (for CIFAR-10) and efficient MobileNet-like networks (for ImageNet) which achieve accuracy comparable to significantly larger models. Overall, we train hundreds of different MLP and CNN architectures with each MLP (CNN) repeated five (three) times with different random seeds, to obtain our results. More setup details (*e.g.*, architecture details, learning rates, *etc.*) are given in Appendix G (see Tables 3, 4, and 5).

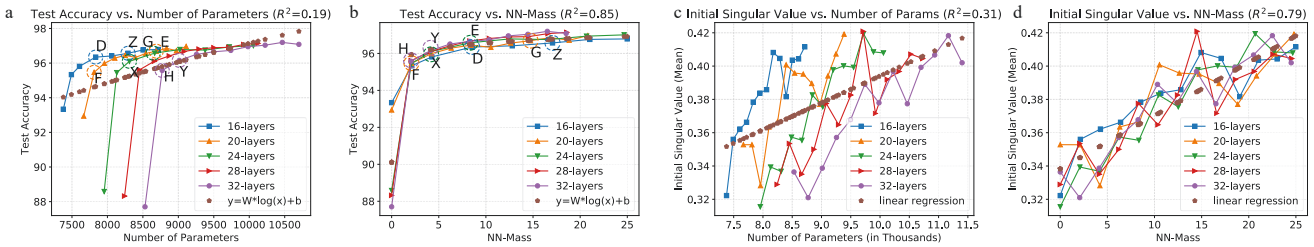

Figure 3: MNIST results: (a) Models with different #Params achieve similar test accuracy. (b) Test accuracy curves of models with different depths/#Params concentrate when plotted against NN-Mass (test accuracy std. dev. $\sim 0.05-0.34\%$). (c,d) Mean singular values of $\boldsymbol{J_{i,i-1}}$ are much better correlated with NN-Mass ($\boldsymbol{R^2 = 0.79}$) than with #Params ($R^2 = 0.31$).

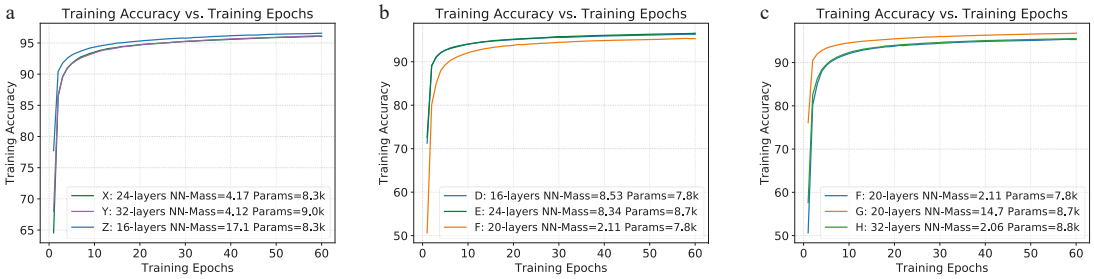

Figure 4: Models X and Y have the same NN-Mass and achieve very similar training convergence, even though they have highly different #Params and depth. Model Z has significantly fewer layers than Y but the same #Params, yet achieves a faster training convergence than Y (Z has higher NN-Mass than Y). The above conclusions hold true for models (D, E, and F) and (F, G and H). Note that, training convergence curves are nearly coinciding for models with same NN-Mass.

4.2. MLP Results (MNIST/Synthetic Data): Topology vs. Gradient Propagation

Test Accuracy. Fig. 3(a) shows test accuracy *vs.* #Params of DNNs with different depths on the MNIST dataset. As evident, even though many models have different #Params, they achieve a similar test accuracy. On the other hand, when the same set of models are plotted against NN-Mass, their test accuracy curves cluster together tightly, as shown in Fig. 3(b). To further quantify the above observation, we generate a linear fit between test accuracy *vs.* log(#Params) and log(NN-Mass) (see brown markers in Fig. 3(a,b)). For NN-Mass, we achieve a significantly higher goodness-of-fit $R^2 = 0.85$ than that for #Params ($R^2 = 0.19$). This demonstrates that NN-Mass can identify DNNs that achieve similar accuracy, even if they have a highly different number of parameters/FLOPS[6]/layers. We next investigate the gradient propagation properties to explain the test accuracy results.

Layerwise Dynamical Isometry (LDI). We calculate the mean singular values of initial layerwise Jacobians, and plot them against #Params (see Fig. 3(c)) and NN-Mass (see Fig. 3(d)). Clearly, NN-Mass ($R^2 = 0.79$) is far better correlated with the mean singular values than #Params ($R^2 = 0.31$). More importantly, just as Proposition 2 predicts, these results show that models with similar NN-Mass and width have equivalent LDI properties, irrespective of the total depth (and, thus #Params) of the network. For example, even though the 32-layer models have more parameters, they have similar mean singular values as the 16-layer DNNs. This clearly suggests that the gradient propagation properties are heavily influenced by the topological characteristics like NN-Mass, and not just by DNN depth and #Params.

Training Convergence. The above results suggest the following hypotheses: (*i*) If the gradient flow between DNNs (with similar NN-Mass and width) is similar, their training convergence should be similar, even if they have highly different #Params and depths; (*ii*) If two models have same #Params (and width), but different depths and NN-Mass, then the DNN with higher NN-Mass should have faster training convergence (since its mean singular value will be higher – see the trend in Fig. 3(d)).

To demonstrate that both hypotheses above hold true, we pick three groups of three models each – (X, Y, and Z), (D, E, and F), and (F, G, and H) from Fig. 3(a,b) and plot their training accuracy *vs.* #epochs in Fig. 4. In Fig. 4(a), Models X and Y have similar NN-Mass but Y has more #Params and depth than X. Model Z has far fewer layers and nearly the same #Params as X, but has higher NN-Mass. Fig. 4(a) shows the training convergence results for X, Y, and Z. As evident, the training convergence of model X (8.3K Params, 24-layers) nearly coincides with that of model Y (9.0K Params, 32-layers). Moreover, even though model Z (8.3K Params, 16-layers) is shallower than the 32-layer model Y (and has far fewer #Params), training convergence of Z is significantly faster than that of Y (due to higher

[6]For our DenseNet-based setup, more parameters lead to more FLOPS. Results for FLOPS are given for CNNs in Appendix H.9.

NN-Mass and, therefore, better LDI). These results clearly show the evidence supporting Proposition 2, and emphasize the concrete links among topology, gradient propagation and model performance for DNNs with DenseNet-type skip connections. Similar observations are found for models (D, E, and F) and (F, G, and H) as shown in Fig. 4(b,c).

4.3. CNN Results on CIFAR-10/100 and ImageNet

Having established a concrete relationship between gradient propagation and topological properties, we now show that NN-Mass can identify efficient CNNs that achieve similar accuracy as models with significantly higher #Params/FLOPS/layers. Unless specified, our CNN models belong to the DenseNet setup. We will explicitly indicate when the results are for ResNets/WRNs/MobileNets.

Model Performance for CIFAR-10 dataset. Fig. 5(a) shows the test accuracy of various CNNs *vs.* total #Params. As evident, models with highly different #Params (*e.g.*, see models A-E in box W), achieve a similar test accuracy. Note that, there is a large gap in the model size: CNNs in box W range from 5M parameters (model A) to 9M parameters

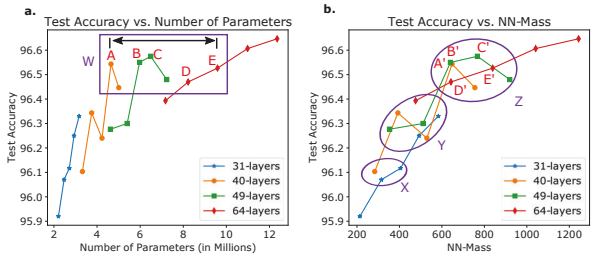

Figure 5: CIFAR-10 Width Multiplier $wm = 2$: (a) Models with very different #Params (box W) achieve similar test accuracies. (b) Models with similar accuracy often have similar NN-Mass: Models in W cluster into Z. Results are reported as the mean of three runs (std. dev. $\sim 0.1\%$).

(models D,E). Again, as shown in Fig. 5(b), when plotted against NN-Mass, the test accuracy curves of CNNs with different depths cluster together (*e.g.*, models A-E in box W cluster into A'-E' within bucket Z). Hence, NN-Mass identifies CNNs with similar accuracy, despite having highly different #Params/layers. The same holds true for models within X and Y boxes. More results with different width multipliers are given in Appendix H.3. For higher width values, the models tend to cluster even more tightly for NN-Mass.

NN-Mass vs. Parameter Count. As shown in Fig. 16 in Appendix H.5, for $wm = 2$, #Params yield an $R^2 = 0.76$ which is lower than that for NN-Mass ($R^2 = 0.84$, see Fig. 16(a, b)). However, for higher widths ($wm = 3$), the parameter count completely fails to predict accuracy ($R^2 = 0.14$ in Fig. 16(c)). For the same width, NN-Mass achieves a significantly higher $R^2 = 0.90$ (see Fig. 16(d)).

Results for CIFAR-100 Dataset. We now corroborate our main findings on CIFAR-100 dataset which is significantly more complex than CIFAR-10. To this end, we train the models in Fig. 5 on CIFAR-100. Fig. 17 (see Appendix H.6) once again shows that several models with highly different number of parameters achieve similar accuracy. Moreover, Fig. 17(b) demonstrates that these models get clustered when plotted against NN-Mass. Further, a high $R^2 = 0.84$ is achieved for a linear fit on the accuracy vs. log(NN-Mass) plot (see Appendix H.6 and Fig. 17).

DenseNet-type CNNs for ImageNet and other results. We provide the ImageNet results in Appendix H.7. More results for the depthwise convolutions (DSConv) with concatenation-type skip links for CIFAR-10 are given in Appendix H.8. Again, NN-Mass identifies CNNs with similar accuracy while having significantly different #Params.

Results for #FLOPS. The results for #FLOPS follow a very similar pattern as #Params (see Fig. 19 in Appendix H.9). In summary, we show that NN-Mass can identify models that yield similar test accuracy, despite having very different #Params/FLOPS/layers.

NN-Mass on Standard DenseNets and VGG. So far, we have used a generalized version of DenseNets which allows us to vary the density of skip connections. For standard DenseNets [7], cell-density (Eq. 1) = 1 (all-to-all connections); thus, NN-Mass for DenseNet = $\sum_{all\ cells}$[(#channels per layer for this cell) $\times$ (#layers per cell)]. For VGG-like models, there are no shortcuts, so NN-Mass = 0. Our theory works for NN-Mass = 0: Fig. 3(b) shows two clusters for [low-depth (16,20) NN-Mass 0] models and [high-depth (24,28,32) NN-Mass 0] models. This is *not* surprising: without shortcuts, the gradient diminishes as depth increases (e.g., see Resnets [5]). Same holds for our NN-Mass=0 CNNs on ImageNet and CIFAR-10.

NN-Mass works for ResNets and MobileNets. For ResNets and MobileNets, we calculate the NN-Mass values using Remark 2. Wide-ResNet (WRN) paper[7] provides #Params and test accuracy for standard ResNets and WRNs for ImageNet (Fig. 6(a)). Fig. 6(b) shows Top1 accuracy vs. NN-Mass. Clearly, NN-Mass outperforms #Params for predicting accuracy ($\boldsymbol{R^2}$**=0.87** vs. R^2=0.64). Note that, RN-50, WRN-34-1.5, and WRN18-3 (26M-101M #Params) achieve similar accuracy (purple box in Fig. 6(a)), and cluster together on NN-Mass plot (purple circle in Fig. 6(b)).

Fig. 6(c,d) shows the Top1 accuracy of MobileNet-v2 (mbn2) vs. #Params (Fig. 6(c)) and NN-Mass (Fig. 6(d)) on ImageNet. The blue line shows the standard MobileNet-v2 models (N_c=17; {0.75, 1.4} are width-multipliers). The red

[7] We directly use the accuracy/#Params from Tables 7,8 in the Wide-ResNet paper [37] (no training is performed).

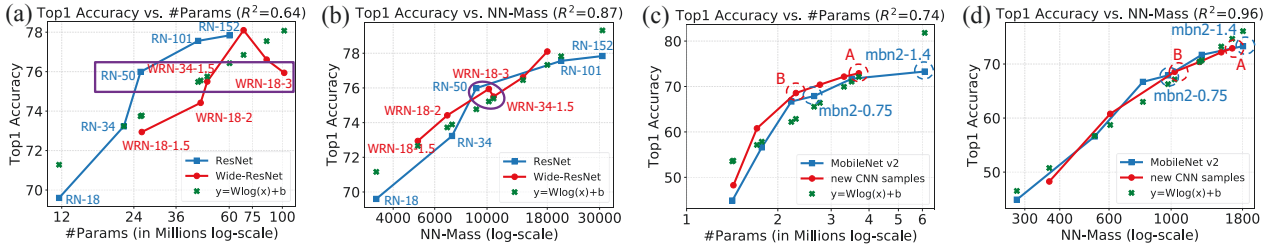

Figure 6: ImageNet results: (a,b) Top1 accuracy vs. (a) #Params (R^2=0.64); (b) NN-Mass ($\mathbf{R^2}$**=0.87**) for standard ResNet and Wide-ResNets. (c,d) Top1 accuracy vs. (c) #Params (R^2=0.74); (d) NN-Mass ($\mathbf{R^2}$**=0.96**) for MobileNet-v2 and some newly sampled CNNs (Better networks are highlighted with *red* letters.).

Table 1: Exploiting NN-Mass for Model Compression on CIFAR-10 Dataset. All results are reported as mean $\pm$ standard deviation of three runs. DARTS results are from [17].

Model	#Params/#FLOPS	#layers	NN-Mass	Test Accuracy
DARTS$^{\text{I}}$ (NAS)	3.3M/-	-	-	97%
DARTS$^{\text{II}}$ (NAS)	3.3M/-	-	-	97.24%
Manual model	11.89M/3.63G	64	1126	97.02%
Manual model	8.15M/2.54G	64	622	96.99%
NN-Mass based	5.02M/1.59G	40	755	97.00%
NN-Mass based	4.69M/1.51G	37	813	96.93%
NN-Mass based	3.82M/1.2G	31	856	96.82%

I – DARTS First Order, II – DARTS Second Order

line shows *new* CNNs sampled by changing width multipliers and total depth (N_c=22). Again, NN-Mass significantly outperforms #Params ($\mathbf{R^2}$**=0.96** vs. R^2=0.74). Hence, NN-Mass works for standard ResNets/WRNs/MobileNet-v2.

Directly designing compressed CNNs with NN-Mass. We first design regular DenseNet-type CNNs (no DSConv) with skip connections for CIFAR-10 dataset and show how such models can be compressed directly using NN-Mass equation (2) without searching for an efficient network. Detailed procedure is provided in Appendix H.10. To summarize: (*i*) We generate candidate CNNs by varying $\{d_c,w_c,t_c\}$ (for DenseNets) or $\{N_c$, width-multiplier$\}$ (for MobileNets). Note, we do *not* train these CNNs. (*ii*) Next, find the compressed CNN with highest NN-Mass (using Definition 4) given the #Params/MACs constraints. (*iii*) Train this model to verify its accuracy. This is our compressed model.

As shown in Table 1, our models reach a test accuracy of 96.82%-97.00% on CIFAR-10, while reducing the number of parameters and FLOPS by up to 3$\times$ over large CNNs (e.g., 3.82M *vs.* 11.89M parameters). As a reference, DARTS [17], a competitive NAS baseline, achieves a comparable (97%) accuracy with slightly lower 3.3M parameters. Note that, our objective is *not* to beat DARTS or any other baseline, but rather to provide theoretical insights into the behavior of DNNs with DenseNet-type skip connections. The DARTS datapoint is chosen just to show that our efficient, high-accuracy, theoretically grounded CNNs (that do not use specialized search spaces like NAS) are capable of reaching state-of-the-art accuracy and, hence, are practically useful.

Finally, we demonstrate that NN-Mass can be used to compress even the most compact CNNs like MobileNets on ImageNet. Specifically, our model A (see Fig. 6(c,d)) is a

Table 2: Compressed MobileNets via NN-Mass on ImageNet

Network	NN-Mass	Top1	#Params	MACs
MobileNetV2 (1.4)	1807	73.3	6.1M	601M
NN-Mass based A (Fig. 6)	1654	72.9	**3.7M**	**393M**
MobileNetV2 (0.75)	975.0	67.9	2.6M	220M
NN-Mass based B (Fig. 6)	1030	68.56	**2.3M**	**200M**

significantly compressed version of mbn2-1.4 and can be directly identified using its NN-Mass (mbn2-1.4 and model A are very far in Fig. 6(c) but are clustered in Fig. 6(d)). These results are summarized in Table 2. As evident, our NN-Mass-based models allow up to **34**% fewer MACs and **40**% fewer #Params than MobileNet-V2-1.4.

5. Conclusion

We have proposed a new topological metric called *NN-Mass* which quantifies how effectively information flows through DNNs. We have also established concrete theoretical relationships among NN-Mass, topological structure of DenseNet-type networks, and layerwise dynamical isometry that ensures faithful propagation of gradients through DNNs. Our training convergence MLP experiments have demonstrated that models with similar NN-Mass and width but different depths and number of parameters have similar training convergence and gradient flow properties like LDI. Our extensive experiments spanning DenseNets to MobileNets show that NN-Mass identifies models with similar accuracy, despite having a highly different number of parameters/FLOPS/layers. Finally, to show the practical applications of our work, we have exploited the closed-form equation of our NN-Mass metric to directly *design* significantly compressed DenseNet-type CNNs (for CIFAR-10) and MobileNet-like CNNs (for ImageNet).

Since topology is deeply intertwined with the gradient propagation, such topological metrics deserve major attention for future research. Another important venue for further work lies in the intersection of initialization and topology.

Acknowledgement

We thank anonymous reviewers for helpful comments. This research was supported in part by U.S. National Science Foundation (NSF) under Grant CNS-2007284, Semiconductor Research Corporation (SRC) under Grant SRC-2939.001, and Amazon AWS ML Research Program.

References

[1] Sanjeev Arora, Nadav Cohen, Noah Golowich, and Wei Hu. A convergence analysis of gradient descent for deep linear neural networks. *arXiv preprint arXiv:1810.02281*, 2018. 4, 11

[2] Albert-Laszlo Barabasi. *Network Science (Chapter 3: Random Networks)*. Cambridge University Press, 2016. 4

[3] Albert-László Barabási and Réka Albert. Emergence of scaling in random networks. *science*, 286(5439):509–512, 1999. 2

[4] Xavier Glorot and Yoshua Bengio. Understanding the difficulty of training deep feedforward neural networks. In *Proceedings of the thirteenth international conference on artificial intelligence and statistics*, pages 249–256, 2010. 2

[5] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 770–778, 2016. 1, 7, 13

[6] Andrew G Howard, Menglong Zhu, Bo Chen, Dmitry Kalenichenko, Weijun Wang, Tobias Weyand, Marco Andreetto, and Hartwig Adam. Mobilenets: Efficient convolutional neural networks for mobile vision applications. *arXiv:1704.04861*, 2017. 13, 16, 20

[7] Gao Huang, Zhuang Liu, Laurens Van Der Maaten, and Kilian Q Weinberger. Densely connected convolutional networks. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 4700–4708, 2017. 1, 3, 7, 13, 14

[8] Simon Jégou, Michal Drozdzal, David Vazquez, Adriana Romero, and Yoshua Bengio. The one hundred layers tiramisu: Fully convolutional densenets for semantic segmentation. In *Proceedings of the IEEE conference on computer vision and pattern recognition workshops*, pages 11–19, 2017. 1

[9] David Kempe, Jon Kleinberg, and Éva Tardos. Maximizing the spread of influence through a social network. In *Proceedings of the ninth ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 137–146, 2003. 1

[10] Thomas Laurent and James Brecht. Deep linear networks with arbitrary loss: All local minima are global. In *International conference on machine learning*, pages 2902–2907. PMLR, 2018. 4, 11

[11] Yann A LeCun, Léon Bottou, Genevieve B Orr, and Klaus-Robert Müller. Efficient backprop. In *Neural networks: Tricks of the trade*, pages 9–48. Springer, 2012. 2

[12] Jaehoon Lee, Lechao Xiao, Samuel Schoenholz, Yasaman Bahri, Roman Novak, Jascha Sohl-Dickstein, and Jeffrey Pennington. Wide neural networks of any depth evolve as linear models under gradient descent. In H. Wallach, H. Larochelle, A. Beygelzimer, F. dÁlché-Buc, E. Fox, and R. Garnett, editors, *Advances in Neural Information Processing Systems*, volume 32, pages 8572–8583. Curran Associates, Inc., 2019. 4, 11

[13] Namhoon Lee, Thalaiyasingam Ajanthan, Stephen Gould, and Philip H. S. Torr. A signal propagation perspective for pruning neural networks at initialization. In *International Conference on Learning Representations*, 2020. 1, 2, 4, 12

[14] Jure Leskovec, Mary McGlohon, Christos Faloutsos, Natalie Glance, and Matthew Hurst. Patterns of cascading behavior in large blog graphs. In *Proceedings of the 2007 SIAM international conference on data mining*, pages 551–556. SIAM, 2007. 1

[15] Liam Li and Ameet Talwalkar. Random search and reproducibility for neural architecture search. *arXiv preprint arXiv:1902.07638*, 2019. 3

[16] Chenxi Liu, Barret Zoph, Maxim Neumann, Jonathon Shlens, Wei Hua, Li-Jia Li, Li Fei-Fei, Alan Yuille, Jonathan Huang, and Kevin Murphy. Progressive neural architecture search. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pages 19–34, 2018. 1

[17] Hanxiao Liu, Karen Simonyan, and Yiming Yang. Darts: Differentiable architecture search. *arXiv preprint arXiv:1806.09055*, 2018. 1, 8, 19, 20

[18] Giacomo Livan, Marcel Novaes, and Pierpaolo Vivo. *Introduction to random matrices: theory and practice*, volume 26. Springer, 2018. 12

[19] Vladimir A Marčenko and Leonid Andreevich Pastur. Distribution of eigenvalues for some sets of random matrices. *Mathematics of the USSR-Sbornik*, 1(4):457, 1967. 13

[20] Remi Monasson. Diffusion, localization and dispersion relations on "small-world" lattices. *The European Physical Journal B-Condensed Matter and Complex Systems*, 12(4):555–567, 1999. 11, 12

[21] Mark Newman, Albert-Laszlo Barabasi, and Duncan J Watts. *The structure and dynamics of networks*, volume 19. Princeton University Press, 2011. 1, 2

[22] Mark EJ Newman and Duncan J Watts. Renormalization group analysis of the small-world network model. *Physics Letters A*, 263(4-6):341–346, 1999. 11, 12

[23] Jeffrey Pennington, Samuel Schoenholz, and Surya Ganguli. Resurrecting the sigmoid in deep learning through dynamical isometry: theory and practice. In *Advances in neural information processing systems*, pages 4785–4795, 2017. 1, 2, 4

[24] Hieu Pham, Melody Y Guan, Barret Zoph, Quoc V Le, and Jeff Dean. Efficient neural architecture search via parameter sharing. *arXiv preprint arXiv:1802.03268*, 2018. 1

[25] Ben Poole, Subhaneil Lahiri, Maithra Raghu, Jascha Sohl-Dickstein, and Surya Ganguli. Exponential expressivity in deep neural networks through transient chaos. In *Advances in neural information processing systems*, pages 3360–3368, 2016. 2, 4

[26] Esteban Real, Alok Aggarwal, Yanping Huang, and Quoc V Le. Regularized evolution for image classifier architecture search. In *Proceedings of the aaai conference on artificial intelligence*, volume 33, pages 4780–4789, 2019. 1

[27] Mark Rudelson and Roman Vershynin. Non-asymptotic theory of random matrices: extreme singular values. In *Proceedings of the International Congress of Mathematicians 2010 (ICM 2010) (In 4 Volumes) Vol. I: Plenary Lectures and Ceremonies Vols. II–IV: Invited Lectures*, pages 1576–1602. World Scientific, 2010. 13

[28] Mark Sandler and et al. Inverted residuals and linear bottlenecks: Mobile networks for classification, detection and segmentation. *arXiv:1801.04381*, 2018. 18

[29] Andrew M Saxe, James L McClelland, and Surya Ganguli. Exact solutions to the nonlinear dynamics of learning in deep linear neural networks. *arXiv preprint arXiv:1312.6120*, 2013. 1, 2, 4, 11

[30] Karen Simonyan and Andrew Zisserman. Very deep convolutional networks for large-scale image recognition. *arXiv preprint arXiv:1409.1556*, 2014. 13

[31] Wojciech Tarnowski, Piotr Warchoł, Stanisław Jastrzębski, Jacek Tabor, and Maciej A Nowak. Dynamical isometry is achieved in residual networks in a universal way for any activation function. *arXiv preprint arXiv:1809.08848*, 2018. 1, 2, 4

[32] Duncan J Watts and Steven H Strogatz. Collective dynamics of 'small-world'networks. *nature*, 393(6684):440, 1998. 2, 3, 12

[33] John Wishart. The generalised product moment distribution in samples from a normal multivariate population. *Biometrika*, pages 32–52, 1928. 12

[34] Mitchell Wortsman, Ali Farhadi, and Mohammad Rastegari. Discovering neural wirings. *arXiv preprint arXiv:1906.00586*, 2019. 2

[35] Saining Xie, Alexander Kirillov, Ross Girshick, and Kaiming He. Exploring randomly wired neural networks for image recognition. *arXiv preprint arXiv:1904.01569*, 2019. 2

[36] Fisher Yu and Vladlen Koltun. Multi-scale context aggregation by dilated convolutions. *arXiv preprint arXiv:1511.07122*, 2015. 20

[37] Sergey Zagoruyko and Nikos Komodakis. Wide residual networks. *arXiv preprint arXiv:1605.07146*, 2016. 7, 13, 16

[38] Yulun Zhang, Yapeng Tian, Yu Kong, Bineng Zhong, and Yun Fu. Residual dense network for image super-resolution. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 2472–2481, 2018. 1

[39] Peng Zhou, Bingbing Ni, Cong Geng, Jianguo Hu, and Yi Xu. Scale-transferrable object detection. In *proceedings of the IEEE conference on computer vision and pattern recognition*, pages 528–537, 2018. 1

[40] Barret Zoph, Vijay Vasudevan, Jonathon Shlens, and Quoc V Le. Learning transferable architectures for scalable image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 8697–8710, 2018. 1, 19