A Scaling Law for Syn2real Transfer: How Much Is Your Pre-training Effective?

Hiroaki Mikami*[1], Kenji Fukumizu*[1,2], Shogo Murai[1], Shuji Suzuki[1], Yuta Kikuchi[1], Taiji Suzuki[3,4], Shin-ichi Maeda[1], and Kohei Hayashi*✉[1]

[1] Preferred Networks, Inc.
{mhiroaki,murai,ssuzuki,kikuchi,ichi,hayasick}@preferred.jp
[2] The Institute of Statistical Mathematics
fukumizu@ism.ac.jp
[3] The University of Tokyo
taiji@mist.i.u-tokyo.ac.jp
[4] AIP-RIKEN

Abstract. Synthetic-to-real transfer learning is a framework in which a synthetically generated dataset is used to pre-train a model to improve its performance on real vision tasks. The most significant advantage of using synthetic images is that the ground-truth labels are automatically available, enabling unlimited expansion of the data size without human cost. However, synthetic data may have a huge domain gap, in which case increasing the data size does not improve the performance. How can we know that? In this study, we derive a simple scaling law that predicts the performance from the amount of pre-training data. By estimating the parameters of the law, we can judge whether we should increase the data or change the setting of image synthesis. Further, we analyze the theory of transfer learning by considering learning dynamics and confirm that the derived generalization bound is consistent with our empirical findings. We empirically validated our scaling law on various experimental settings of benchmark tasks, model sizes, and complexities of synthetic images.

1 Introduction

The success of deep learning relies on the availability of large data. If the target task provides limited data, the framework of transfer learning is preferably employed. A typical scenario of transfer learning is to pre-train a model for a similar or even different task and fine-tune the model for the target task. However, the limitation of labeled data has been the main bottleneck of supervised pre-training. While there have been significant advances in the representation capability of the models and computational capabilities of the hardware, the size and the diversity of the baseline dataset have not been growing as fast [57]. This is partially because of the sheer physical difficulty of collecting large datasets from real environments (e.g., the cost of human annotation).

* Equal contribution

In computer vision, *synthetic-to-real (syn2real) transfer* is a promising strategy that has been attracting attention [9, 12, 22, 29, 44, 56, 61]. In syn2real, images used for pre-training are synthesized to improve the performance on real vision tasks. By combining various conditions, such as 3D models, textures, light conditions, and camera poses, we can synthesize an infinite number of images with ground-truth annotations. Syn2real transfer has already been applied in some real-world applications. Teed and Deng [59] proposed a simultaneous localization and mapping (SLAM) system that was trained only with synthetic data and demonstrated state-of-the-art performance. The object detection networks for autonomous driving developed by Tesla was trained with 370 million images generated by simulation [36].

The performance of syn2real transfer depends on the similarity between synthetic and real data. In general, the more similar they are, the stronger the effect of pre-training will be. On the contrary, if there is a significant gap, increasing the number of synthetic data may be completely useless, in which case we waste time and computational resources. A distinctive feature of syn2real is that we can control the process of generating data by ourselves. If a considerable gap exists, we can try to regenerate the data with a different setting. But how do we know that? More specifically, in a standard learning setting without transfer, a "power law"-like relationship called a *scaling law* often holds between data size and generalization errors [35, 53]. Is there such a rule for pre-training?

In this study, we find that the generalization error on fine-tuning is explained by a simple scaling law,

$$\text{test error} \simeq Dn^{-\alpha} + C, \tag{1}$$

where coefficient $D > 0$ and exponent $\alpha > 0$ describe the convergence speed of pre-training, and constant $C \geq 0$ determines the lower limit of the error. We refer to α as *pre-training rate* and C as *transfer gap*. We can predict how large the pre-training data should be to achieve the desired accuracy by estimating the parameters α, C from the empirical results. Additionally, we analyze the dynamics of transfer learning using the recent theoretical results based on the neural tangent kernel [50] and confirm that the above law agrees with the theoretical analysis. We empirically validated our scaling law on various experimental settings of benchmark tasks, model sizes, and complexities of synthetic images.

Our contributions are summarized as follows.

- From empirical results and theoretical analysis, we elicit a law that describes how generalization scales in terms of data sizes on pre-training and fine-tuning.
- We confirm that the derived law explains the empirical results for various settings in terms of pre-training/fine-tuning tasks, model size, and data complexity (e.g., Figure 1). Furthermore, we demonstrate that we can use the estimated parameters in our scaling law to assess how much improvement we can expect from the pre-training procedure based on synthetic data.
- We theoretically derive a generalization bound for a general transfer learning setting and confirm its agreement with our empirical findings.

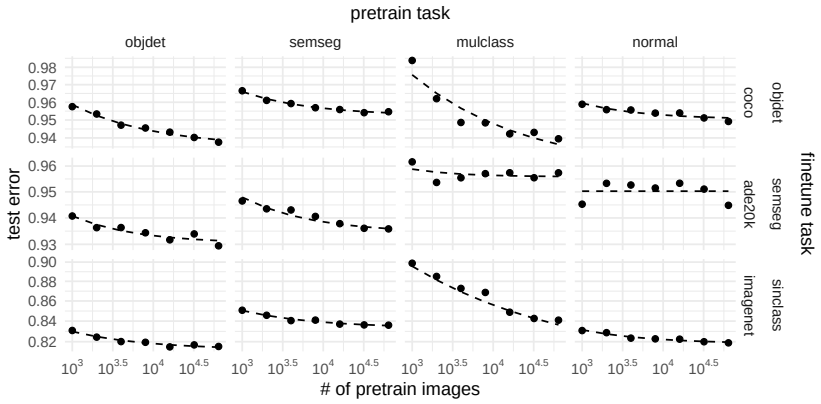


Fig. 1: Empirical results of syn2real transfer for different tasks. We conducted four pre-training tasks: object detection (`objdet`), semantic segmentation (`semseg`), multi-label classification (`mulclass`), surface normal estimation (`normal`), and three fine-tuning tasks for benchmark datasets: object detection for MS-COCO, semantic segmentation for ADE20K, and single-label classification (`sinclass`) for ImageNet. The y-axis indicates the test error for each fine-tuning task. Dots indicate empirical results and dashed lines indicate the fitted curves of scaling law (1). For more details, see Section 4.2.

2 Related Work

Pre-training for visual tasks Many empirical studies show that the performance at a fine-tuning task scales with pre-training data (and model) size. For example, Huh et al. [32] studied the scaling behavior on ImageNet pre-trained models. Beyond ImageNet, Sun et al. [57] studied the effect of pre-training with pseudo-labeled large-scale data and found a logarithmic scaling behavior. Similar results were observed by Kolesnikov et al. [38].

Syn2real transfer The utility of synthetic images as supervised data for computer vision tasks has been continuously studied by many researchers [9, 12, 14, 22, 29, 31, 43–45, 56, 61]. These studies found positive evidence that using synthetic images is helpful to the fine-tuning task. In addition, they demonstrated how data complexity, induced by e.g., light randomization, affects the final performance. For example, Newell and Deng [45] investigated how the recent self-supervised methods perform well as a pre-training task to improve the performance of downstream tasks. In this paper, following this line of research, we quantify the effects under the lens of the scaling law (1).

Neural scaling laws The scaling behavior of generalization error, including some theoretical works [e.g., 3], has been studied extensively. For modern neural networks, Hestness et al. [28] empirically observed the power-law behavior of

generalization for language, image, and speech domains with respect to the training size. Rosenfeld et al. [53] constructed a predictive form for the power-law in terms of data and model sizes. Kaplan et al. [35] pushed forward this direction in the language domain, describing that the generalization of transformers obeys the power law in terms of a compute budget in addition to data and model sizes. Since then, similar scaling laws have been discovered in other data domains [25]. Several authors have also attempted theoretical analysis. Hutter [33] analyzed a simple class of models that exhibits a power-law $n^{-\beta}$ in terms of data size n with arbitrary $\beta > 0$. Bahri et al. [5] addressed power laws under four regimes for model and data size. Note that these theoretical studies, unlike ours, are concerned with scaling laws in a non-transfer setting.

Hernandez et al. [27] studied the scaling laws for general transfer learning, which is the most relevant to this study. A key difference is that they focused on fine-tuning data size as a scaling factor, while we focus on pre-training data size. Further, they found scaling laws in terms of the transferred effective data, which is converted data amount necessary to achieve the same performance gain by pre-training. In contrast, Eq. (1) explains the test error with respect to the pre-training data size directly at a fine-tuning task. Other differences include task domains (language vs. vision) and architectures (transformer vs. CNN).

Theory of transfer learning Theoretical analysis of transfer learning has been dated back to decades ago [7] and has been pursued extensively. Among others, some recent studies [16, 42, 62] derived an error bound of a fine-tuning task in the multi-task scenario based on complexity analysis; the bound takes an additive form $O(An^{-1/2} + Bs^{-1/2})$, where n and s are the data size of pre-training and fine-tuning, respectively, with coefficients A and B. Neural network regression has been also discussed with this bound [62]. In the field of domain adaptation, error bounds have been derived in relation to the mismatch between source and target input distributions [1, 19]. They also proposed algorithms to adopt a new data domain. However, unlike in this study, no specific learning dynamics has been taken into account. In the area of hypothesis transfer learning [18, 64], among many theoretical works, Du et al. [17] has derived a risk bound for kernel ridge regression with transfer realized as the weights on the training samples. The obtained bound takes a similar form to our scaling law. However, the learning dynamics of neural networks initialized with a pre-trained model has never been explored in this context.

3 Scaling Laws for Pre-training and Fine-tuning

The main obstacle in analyzing the test error is that we have to consider interplay between the effects of pre-training and fine-tuning. Let $L(n, s) \geq 0$ be the test error of a fine-tuning task with pre-training data size n and fine-tuning data size s. As the simplest case, consider a fine-tuning task without pre-training ($n = 0$), which boils the transfer learning down to a standard learning setting. In this case, the prior studies of both classical learning theory and neural scaling laws

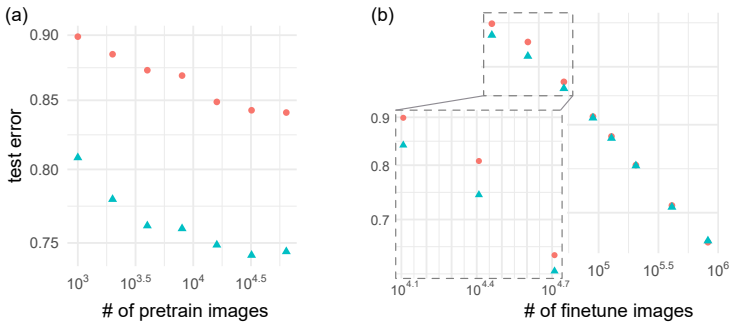


Fig. 2: Scaling curves with different (a) pre-training size and (b) fine-tuning size.

tell us that the test error decreases polynomially[5] with the fine-tuning data size s, that is, $L(0, s) = Bs^{-\beta} + \mathcal{E}$ with decay rate $\beta > 0$ and irreducible loss $\mathcal{E} \geq 0$. The irreducible loss $\mathcal{E}$ is the inevitable error given by the best possible mapping; it is caused by noise in continuous outputs or labels. Hereafter we assume $\mathcal{E} = 0$ for brevity.

3.1 Induction of scaling law with small empirical results

To speculate a scaling law, we conducted preliminary experiments.[6] We pre-trained ResNet-50 by a synthetic classification task and fine-tuned by ImageNet. Figure 2 (a) presents the log-log plot of error curves with respect to pre-training data size n, where each shape and color indicates a different fine-tuning size s. It shows that the pre-training effect diminishes for large n. In contrast, Figure 2 (b) presents the relations between the error and the fine-tuning size s with different n. It indicates the error drops straight down regardless of n, confirming the power-law scaling with respect to s. The above observations and the fact that $L(0, s)$ decays polynomially are summarized as follows.

Requirement 1 $\lim_{s\to\infty} L(n, s) = 0$.

Requirement 2 $\lim_{n\to\infty} L(n, s) = \text{const}$.

Requirement 3 $L(0, s) = Bs^{-\beta}$.

Requirements 1 and 3 suggest the dependency of n is embedded in the coefficient $B = g(n)$, i.e., the pre-training and fine-tuning effects interact multiplicatively. To satisfy Requirement 2, a reasonable choice for the pre-training effect is

[5] For classification with strong low-noise condition, it is known that the decay rate can be exponential [49]. However, we focus only on the polynomial decay without such strong condition in this paper.

[6] The results are replicated from Appendix C.2; see the subsection for more details.

$g(n) = n^{-\alpha} + \gamma$; the error decays polynomially with respect to n but has a plateau at γ. By combining these, we obtain

$$L(n, s) = \delta(\gamma + n^{-\alpha})s^{-\beta}, \tag{2}$$

where $\alpha, \beta > 0$ are decay rates for pre-training and fine-tuning, respectively, $\gamma \geq 0$ is a constant, and $\delta > 0$ is a coefficient. The exponent β determines the convergence rate with respect to fine-tuning data size. From this viewpoint, $\delta(\gamma + n^{-\alpha})$ is the coefficient factor to the power law. The influence of the pre-training appears in this coefficient, where the constant term $\delta\gamma$ comes from the irreducible loss of the pre-training task and $n^{-\alpha}$ expresses the effect of pre-training data size. The theoretical consideration in Section E.5 suggests that the rates α and β can depend on both the target functions of pre-training and fine-tuning as well as the learning rate.

3.2 Theoretical deduction of scaling law

Next, we analyze the fine-tuning error from a purely theoretical point of view. To incorporate the effect of pre-training that is given as an initialization, we need to analyze the test error during the training with a given learning algorithm such as SGD. We apply the recent development by Nitanda and Suzuki [50] to transfer learning. The study successfully analyzes the generalization of neural networks in the dynamics of learning, showing it achieves minmax optimum rate. The analysis uses the framework of the reproducing kernel Hilbert space given by the neural tangent kernel [34].

For theoretical analysis of transfer, it is important to formulate a task similarity between pre-training and fine-tuning. If the tasks were totally irrelevant (e.g., learning MNIST to forecast tomorrow's weather), pre-training would have no benefit. Following Nitanda and Suzuki [50], for simplicity of analysis, we discuss only a regression problem with square loss. We assume that a vector input x and scalar output y follow $y = \phi_0(x)$ for pre-training and $y = \phi_0(x) + \phi_1(x)$ for fine-tuning, where we omit the output noise for brevity; the task types are identical sharing the same input-output form, and task similarity is controlled by ϕ_1.

We analyze the situation where the effect of pre-training remains in the fine-tuning even for large data size ($s \to \infty$). More specifically, the theoretical analysis assumes a regularization term as the ℓ_2-distance between the weights and the initial values, and a smaller learning rate than constant in the fine-tuning. Hence we control how the pre-training effect is preserved through the regularization and learning rate. Other assumptions made for theoretical analysis concern the model and learning algorithm; a two-layer neural network having M hidden units with continuous nonlinear activation[7] is adopted; for optimization, the averaged SGD [51], an online algorithm, is used for a technical reason.

The following is an informal statement of the theoretical result. See Appendix E for details. We emphasize that our result holds not only for syn2real transfer but also for transfer learning in general.

[7] ReLU is not included in this class, but we can generalize this condition; see [50].

Theorem 1 (Informal) *Let $\hat{f}_{n,s}(x)$ be a model of width M pre-trained by n samples $(x_1, y_1), \ldots, (x_n, y_n)$ and fine-tuned by s samples $(x'_1, y'_1), \ldots, (x'_s, y'_s)$ where inputs $x, x' \sim p(x)$ are i.i.d. with the input distribution $p(x)$ and $y = \phi_0(x)$ and $y' = \varphi(x') = \phi_0(x') + \phi_1(x')$. Then the generalization error of the squared loss $L(n, s) = |\hat{f}_{n,s}(x) - \varphi(x)|^2$ is bounded from above with high probability as*

$$E_x L(n, s) \leq A_1(c_M + A_0 n^{-\alpha})s^{-\beta} + \varepsilon_M. \tag{3}$$

ε_M and c_M can be arbitrary small for large M; A_0 and A_1 are constants; the exponents α and β depend on ϕ_0, ϕ_1, $p(x)$, and the learning rate of fine-tuning.

The above bound (3) shows the correspondence with the empirical derivation of the full scaling law (2). Note that the approximation error ε_M is omitted in (2).

We note that the derived bound takes a multiplicative form in terms of the pre-training and fine-tuning effects, which contrasts with the additive bounds such as $An^{-1/2} + Bs^{-1/2}$ [62]. The existing studies consider the situation where a part of a network (e.g., backbone) is frozen during fine-tuning. Therefore, the error of pre-training is completely preserved after fine-tuning, and both errors appear in an additive way. This means that the effect of pre-training is irreducible by the effect of fine-tuning, and vice versa. In contrast, our analysis deals with the case of re-optimizing the entire network in fine-tuning. In that case, the pre-trained model is used as initial values. As a result, even if the error in pre-training is large, the final error can be reduced to zero by increasing the amount of fine-tuning data.

3.3 Insights and Practical Values

The form of the full scaling law (2) suggests that there are two scenarios depending on whether fine-tuning data is big or small. In "big fine-tune" regime, pre-training contributes relatively little. By taking logarithm, we can separate the full scaling law (2) into the pre-training part $u(n) = \log(n^{-\alpha} + \gamma)$ and the fine-tuning part $v(s) = -\beta \log s$. Consider to increase n by squaring it. Since the pre-training part cannot be reduced below $\log(\gamma)$ as $u(n) > u(n^2) > \log(\gamma)$, the relative improvement $(u(n^2) - u(n))/v(s)$ becomes infinitesimal for large s. Figure 2 (b) confirms this situation. Indeed, prior studies provide the same conclusion that the gain from pre-training can easily vanish [24, 45] or a target task accuracy even degrade [67] if we have large enough fine-tuning data.

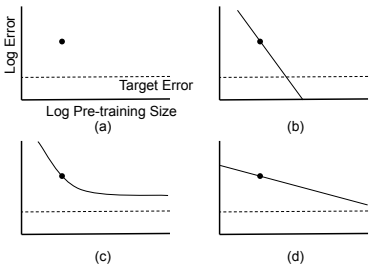


Fig. 3: Pre-training scenarios.

The above observation, however, does not mean pre-training is futile. Dense prediction tasks such as depth estimation require pixel-level annotations, which critically limits the number of labeled data. Pre-training is indispensable in such

"small fine-tune" regime. Based on this, we hereafter analyze the case where the fine-tuning size s is fixed. By eliminating s-dependent terms in (2), we obtain a simplified law (1) by setting $D = \delta s^{-\beta}$ and $C = \delta\gamma s^{-\beta}$. After several evaluations, these parameters including α can be estimated by the nonlinear least squares method (see also Section 4.1).

As a practical benefit, the estimated parameters of the simplified law (1) bring a way to assess syn2real transfer. Suppose we want to solve a classification task that requires at least 90% accuracy with limited labels. We generate some number of synthetic images and pre-train with them, and we obtain 70% accuracy as Figure 3 (a). How can we achieve the required accuracy? It depends on the parameters of the scaling law. The best scenario is (b) — transfer gap C is low and pre-training rate α is high. In this case, increasing synthetic images eventually leads the required accuracy. In contrast, when transfer gap C is larger than the required accuracy (c), increasing synthetic images does not help to solve the problem. Similarly, for low pre-training rate α (d), we may have to generate tremendous amount of synthetic images that are computationally infeasible. In the last two cases, we have to change the rendering settings such as 3D models and light conditions to improve C and/or α, rather than increasing the data size. The estimation of α and C requires to compute multiple fine-tuning processes. However, the estimated parameters tell us whether we should increase data or change the data generation process, which can reduce the total number of trials and errors.

4 Experiments

4.1 Settings

For experiments, we employed the following transfer learning protocol. First, we pre-train a model that consists of backbone and head networks from random initialization until convergence, and we select the best model in terms of the validation error of the pre-training task. Then, we extract the backbone and add a new head to fine-tune all the model parameters. For notations, the task names of object detection, semantic segmentation, multi-label classification, single-label classification, and surface normal estimation are abbreviated as `objdet`, `semseg`, `mulclass`, `sinclass`, and `normal`, respectively. The settings for transfer learning are denoted by arrows. For example, `objdet`→`semseg` indicates that a model is pre-trained by object detection, and fine-tuned by semantic segmentation. All the results including Figure 1 are shown as log-log plots. The details of pre-training, fine-tuning, and curve fitting are described in Appendix A.1.

4.2 Scaling law universally explains downstream performance for various task combinations

Figure 1 shows the test errors of each fine-tuning task and fitted learning curves with Eq. (1), which describes the effect of pre-training data size n for all combinations of pre-training and fine-tuning tasks. The scaling law fits with the empirical fine-tuning test errors with high accuracy in most cases.

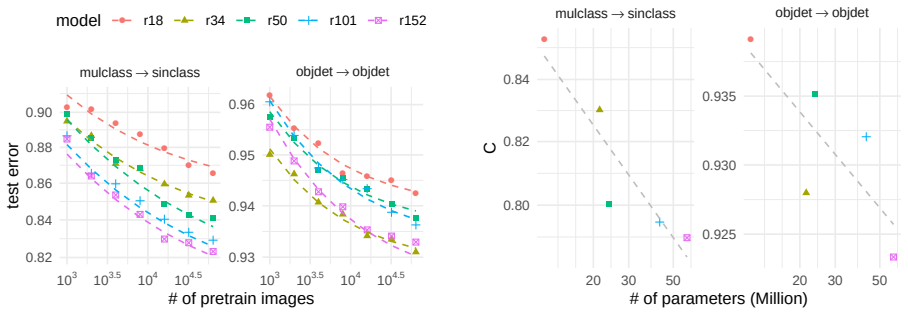


Fig. 4: Effect of model size. Best viewed in color. **Left**: The scaling curves for `mulclass`→`sinclass` and `objdet`→`objdet` cases. The meanings of dots and lines are the same as those in Figure 1. **Right**: The estimated transfer gap C (y-axis) versus the model size (x-axis) in log-log scale. The dots are estimated values, and the lines are linear fittings of them.

4.3 Bigger models reduce the transfer gap

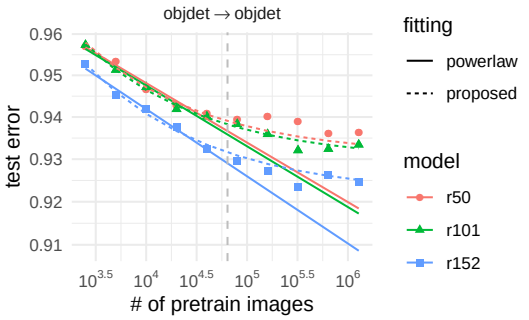
We compared several ResNet models as backbones in `mulclass`→`sinclass` and `objdet`→`objdet` to observe the effects of model size. Figure 4 (left) shows the curves of scaling laws for the pre-training data size n for different sizes of backbone ResNet-x, where $x \in \{18, 34, 50, 101, 152\}$. The bigger models attain smaller test errors. Figure 4 (right) shows the values of the estimated transfer gap C. The results suggest that there is a roughly power-law relationship between the transfer gap and model size. This agrees with the scaling law with respect to the model size shown by Hernandez et al. [27].

4.4 Scaling law can extrapolate for more pre-training images

We also evaluated the extrapolation ability of the scaling law. We increased the number of synthetic images from the original size (n = 64,000) to 1.28 million, and see how the fitted scaling law predicts the unseen test errors where $n >$ 64,000. As a baseline, we compared the power-law model, which is equivalent to the derived scaling law (1) with $C = 0$. Figure 5 (left) shows the extrapolation results in `objdet`→`objdet` setting, which indicates the scaling law follows the saturating trend in regions with large pre-training sizes for all models, while the power-law model fails to capture it. The prediction errors is numerically shown in Figure 5 (right), which again shows our scaling law achieves better prediction performance.

4.5 Data complexity affects both pre-training rate and transfer gap

We examined how the complexity of synthetic images affects fine-tuning performance. We controlled the following four rendering parameters: *Appearance*:



model	powerlaw	proposed
r50	0.0099	0.0024
r101	0.0074	0.0016
r152	0.0091	0.0017

Fig. 5: Ability to extrapolate. **Left:** The solid lines represent the fitted power law and the dashed curves represent the fitted scaling law (1), in which the laws were fitted using the empirical errors where the pre-training size n was less than 64,000 (the first five dots). The vertical dashed line indicates where $n = 64{,}000$. **Right:** The root-mean-square errors between the laws and the actual test errors in the area of extrapolation (the last five dots).

Number of objects in each image; `single` or `multiple` (max 10 objects). *Light*: Either an area and point light is `random`ized or `fix`ed in terms of height, color, and intensity. *Background*: Either the textures of floor/wall are `random`ized or `fix`ed. *Object texture*: Either the 3D objects used for rendering contain texture (`w/`) or not (`w/o`). Indeed, the data complexity satisfies the following ordered relationships: `single` < `multiple` in *appearance*, `fix` < `random` in *light* and *background*, and `w/o` < `w/` in *object texture*[8]. To quantify the complexity, we computed the negative entropy of the Gaussian distribution fitted to the last activation values of the backbone network. For this purpose, we pre-trained ResNet-50 as a backbone with MS-COCO for 48 epochs and computed the empirical covariance of the last activations for all the synthetic data sets.

The estimated parameters are shown in Figure 6, which indicates the following (we discuss the implications of these results further in Section 5.1).

- Data complexity controlled by the rendering settings correlates with the negative entropy, implying the negative entropy expresses the actual complexity of pre-training data.
- Pre-training rate α correlates with data complexity. The larger complexity causes slower rates of convergence with respect to the pre-training data size.
- Transfer gap C mostly correlates negatively with data complexity, but not for *object texture*.

As discussed in Section 4.1, we have fixed the value of D to avoid numerical instability, which might cause some bias to the estimates of α. We postulate, however, the value of D depends mainly on the fine-tuning task and thus has

[8] The object category of `w/o` is a subset of `w/`, and `w/` has a strictly higher complexity than `w/o`.

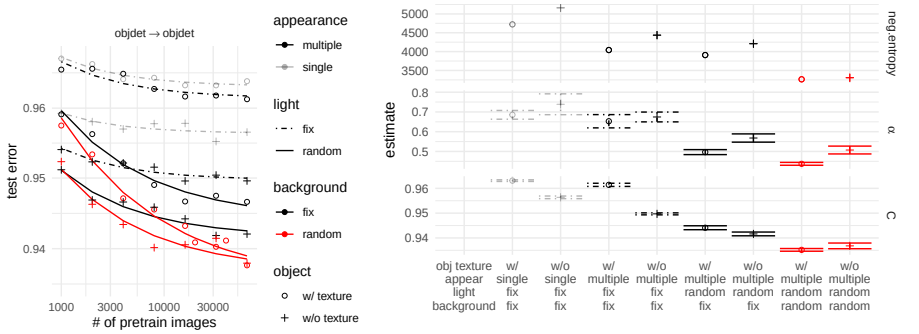


Fig. 6: Effect of synthetic image complexity. Best viewed in color. **Left**: Scaling curves of different data complexities. **Right**: Estimated parameters. The error bars represent the standard error of the estimate in least squares.

a fixed value for different pre-training data complexities. This can be inferred from the theoretical analysis in Appendix E.5: the exponent β in the main factor $s^{-\beta}$ of D does not depend on the pre-training data distribution but only on the fine-tuning task or the pre-training true mapping. Thus, the values of D should be similar over the different complexities, and the correlation of α preserves.

5 Conclusion and Discussion

In this paper, we studied how the performance on syn2real transfer depends on pre-training and fine-tuning data sizes. Based on the experimental results, we found a scaling law (1) and its generalization (2) that explain the scaling behavior in various settings in terms of pre-training/fine-tuning tasks, model sizes, and data complexities. Further, we present the theoretical error bound for transfer learning and found our theoretical bound has a good agreement with the scaling law.

5.1 Implication of complexity results in Section 4.5

The results of Section 4.5 has two implications. First, data complexity (i.e., the diversity of images) largely affects the pre-training rate α. This is reasonable because if we want a network to recognize more diverse images, we need to train it with more examples. Indeed, prior studies [5, 55] observed that α is inversely proportional to the intrinsic dimension of the data (e.g., dimension of the data manifold), which is an equivalent concept of data complexity.

Second, the estimated values of the transfer gap C suggest that increasing the complexity of data is generally beneficial to decrease C, but not always. Figure 6 (right) shows that increasing complexities in terms of *appearance*, *light*, and *background* reduces the transfer gap, which implies that these rendering operations are most effective to cover the fine-tuning task that uses real images.

However, the additional complexity in *object texture* works negatively. We suspect that this occurred because of *shortcut learning* [20]. Namely, adding textures to objects makes the recognition problem falsely easier because we can identify objects by textures rather than shapes. Because CNNs prefer to recognize objects by textures [21, 26], the pre-trained models may overfit to learn the texture features. Without object textures, pre-trained models have to learn the shape features because there is no other clue to distinguish the objects, and the learned features will be useful for real tasks.

5.2 Lessons to transfer learning and synthetic-to-real generalization

Our results suggest the transfer gap C is the most crucial factor for successful transfer learning because C determines the maximum utility of pre-training. Large-scale pre-training data can be useless when C is large. In contrast, if C is negligibly small, the law is reduced essentially to $n^{-\alpha}$, which tells that the volume of pre-training data is directly exchanged to the performance of fine-tuning tasks. Our empirical results suggest two strategies for reducing C: 1) Use bigger models and 2) fill the domain gap in terms of the decision rule and image distribution. For the latter, existing techniques such as domain randomization [60] would be helpful.

5.3 Limitations

We have not covered several directions in this paper. In theory, we assume several conditions that may not fit with the actual setting; the additive fine-tuning model $\phi_0(x) + \phi_1(x)$ in Theorem 1 does not address the transfer to different type of tasks, and the distributional difference of the inputs (synthetic versus real) is not considered. We analyzed only ASGD as the optimization and the effect of the choice is not fully clarified. In spite of these theoretical simplifications, our analysis has revealed the important aspects of the transfer learning as discussed in Section 3. In the experiments, although our theory is justified, we have not investigated the case when a pre-training dataset is not synthetic but real. These topics are left for future work.

Acknowledgments We thank Daisuke Okanohara, Shoichiro Yamaguchi, Takeru Miyato, Katsuhiko Ishiguro for valuable comments and discussions in the early stage of this study. We also thank Masanori Koyama and Kenta Oono for reading the draft and providing detailed feedback. TS was partially supported by JSPS KAKENHI (18H03201, 20H00576), and JST CREST. KF was partially supported by JST CREST JPMJCR2015.

References

1. Acuna, D., Zhang, G., Law, M.T., Fidler, S.: f-domain-adversarial learning: Theory and algorithms. arXiv:2106.11344 (2021)

2. Allen-Zhu, Z., Li, Y., Liang, Y.: Learning and generalization in overparameterized neural networks, going beyond two layers. CoRR abs/1811.04918 (2018)
3. Amari, S.i., Fujita, N., Shinomoto, S.: Four types of learning curves. Neural Computation 4(4), 605–618 (1992)
4. Arora, S., Du, S., Hu, W., Li, Z., Wang, R.: Fine-grained analysis of optimization and generalization for overparameterized two-layer neural networks. In: Proceedings of the 36th International Conference on Machine Learning. vol. 97, pp. 322–332 (2019)
5. Bahri, Y., Dyer, E., Kaplan, J., Lee, J., Sharma, U.: Explaining neural scaling laws. arXiv:2102.06701 (2021)
6. Bartlett, P.L., Foster, D.J., Telgarsky, M.J.: Spectrally-normalized margin bounds for neural networks. In: Advances in Neural Information Processing Systems (2017)
7. Baxter, J.: A model of inductive bias learning. Journal of artificial intelligence research 12, 149–198 (2000)
8. Bolya, D., Zhou, C., Xiao, F., Lee, Y.J.: Yolact: Real-time instance segmentation. In: Proceedings of the IEEE/CVF International Conference on Computer Vision. pp. 9157–9166 (2019)
9. Borrego, J., Dehban, A., Figueiredo, R., Moreno, P., Bernardino, A., Santos-Victor, J.: Applying domain randomization to synthetic data for object category detection. arXiv:1807.09834 (2018)
10. Caponnetto, A., De Vito, E.: Optimal rates for regularized least-squares algorithm. Foundations of Computational Mathematics 7(3), 331–368 (2007)
11. Chen, L.C., Papandreou, G., Schroff, F., Adam, H.: Rethinking atrous convolution for semantic image segmentation. arXiv:1706.05587 (2017)
12. Chen, W., Yu, Z., Mello, S.D., Liu, S., Alvarez, J.M., Wang, Z., Anandkumar, A.: Contrastive syn-to-real generalization. arXiv:2104.02290 (2021)
13. Denninger, M., Sundermeyer, M., Winkelbauer, D., Zidan, Y., Olefir, D., Elbadrawy, M., Lodhi, A., Katam, H.: Blenderproc. arXiv:1911.01911 (2019)
14. Devaranjan, J., Kar, A., Fidler, S.: Meta-sim2: Unsupervised learning of scene structure for synthetic data generation. In: European Conference on Computer Vision. pp. 715–733. Springer (2020)
15. Du, S., Lee, J., Li, H., Wang, L., Zhai, X.: Gradient Descent Finds Global Minima of Deep Neural Networks. In: Proceedings of the 36th International Conference on Machine Learning. vol. 97, pp. 1675–1685 (2019)
16. Du, S.S., Hu, W., Kakade, S.M., Lee, J.D., Lei, Q.: Few-shot learning via learning the representation, provably. arXiv:2002.09434 (2020)
17. Du, S.S., Koushik, J., Singh, A., Poczos, B.: Hypothesis Transfer Learning via Transformation Functions. In: Advances in Neural Information Processing Systems. vol. 30 (2017)
18. Fei-Fei, L., Fergus, R., Perona, P.: One-shot learning of object categories. IEEE Transactions on Pattern Analysis and Machine Intelligence 28(4), 594–611 (2006)
19. Ganin, Y., Ustinova, E., Ajakan, H., Germain, P., Larochelle, H., Laviolette, F., Marchand, M., Lempitsky, V.: Domain-adversarial training of neural networks. The journal of machine learning research 17(1), 2096–2030 (2016)

20. Geirhos, R., Jacobsen, J.H., Michaelis, C., Zemel, R., Brendel, W., Bethge, M., Wichmann, F.A.: Shortcut learning in deep neural networks. Nature Machine Intelligence 2(11), 665–673 (2020)
21. Geirhos, R., Rubisch, P., Michaelis, C., Bethge, M., Wichmann, F.A., Brendel, W.: Imagenet-trained cnns are biased towards texture; increasing shape bias improves accuracy and robustness. arXiv:1811.12231 (2018)
22. Georgakis, G., Mousavian, A., Berg, A.C., Kosecka, J.: Synthesizing training data for object detection in indoor scenes. arXiv:1702.07836 (2017)
23. Goyal, P., Dollár, P., Girshick, R., Noordhuis, P., Wesolowski, L., Kyrola, A., Tulloch, A., Jia, Y., He, K.: Accurate, large minibatch SGD: Training ImageNet in 1 hour. arXiv:1706.02677 (2017)
24. He, K., Girshick, R., Dollár, P.: Rethinking imagenet pre-training. arXiv:1811.08883 (2018)
25. Henighan, T., Kaplan, J., Katz, M., Chen, M., Hesse, C., Jackson, J., Jun, H., Brown, T.B., Dhariwal, P., Gray, S., Hallacy, C., Mann, B., Radford, A., Ramesh, A., Ryder, N., Ziegler, D.M., Schulman, J., Amodei, D., McCandlish, S.: Scaling laws for autoregressive generative modeling. arXiv:2010.14701 (2020)
26. Hermann, K.L., Chen, T., Kornblith, S.: The origins and prevalence of texture bias in convolutional neural networks. arXiv:1911.09071 (2019)
27. Hernandez, D., Kaplan, J., Henighan, T., McCandlish, S.: Scaling laws for transfer. arXiv:2102.01293 (2021)
28. Hestness, J., Narang, S., Ardalani, N., Diamos, G., Jun, H., Kianinejad, H., Patwary, M.M.A., Yang, Y., Zhou, Y.: Deep learning scaling is predictable, empirically. arXiv:1712.00409 (2017)
29. Hinterstoisser, S., Pauly, O., Heibel, H., Marek, M., Bokeloh, M.: An annotation saved is an annotation earned: Using fully synthetic training for object instance detection. arXiv:1902.09967 (2019)
30. Hodaň, T., Sundermeyer, M., Drost, B., Labbé, Y., Brachmann, E., Michel, F., Rother, C., Matas, J.: BOP challenge 2020 on 6D object localization. European Conference on Computer Vision Workshops (ECCVW) (2020)
31. Hodaň, T., Vineet, V., Gal, R., Shalev, E., Hanzelka, J., Connell, T., Urbina, P., Sinha, S.N., Guenter, B.: Photorealistic image synthesis for object instance detection. In: 2019 IEEE International Conference on Image Processing (ICIP). pp. 66–70. IEEE (2019)
32. Huh, M., Agrawal, P., Efros, A.A.: What makes imagenet good for transfer learning? arXiv:1608.08614 (2016)
33. Hutter, M.: Learning curve theory. arXiv:2102.04074 (2021)
34. Jacot, A., Gabriel, F., Hongler, C.: Neural tangent kernel: Convergence and generalization in neural networks. In: Advances in Neural Information Processing Systems 31, pp. 8571–8580. Curran Associates, Inc. (2018)
35. Kaplan, J., McCandlish, S., Henighan, T., Brown, T.B., Chess, B., Child, R., Gray, S., Radford, A., Wu, J., Amodei, D.: Scaling laws for neural language models. arXiv:2001.08361 (2020)
36. Karpathy, A.: Tesla ai day. `https://www.youtube.com/watch?v=j0z4FweCy4M` (2021)

37. Kirkpatrick, J., Pascanu, R., Rabinowitz, N., Veness, J., Desjardins, G., Rusu, A.A., Milan, K., Quan, J., Ramalho, T., Grabska-Barwinska, A., Hassabis, D., Clopath, C., Kumaran, D., Hadsell, R.: Overcoming catastrophic forgetting in neural networks. Proceedings of the National Academy of Sciences of the United States of America 114(13), 3521–3526 (mar 2017)
38. Kolesnikov, A., Beyer, L., Zhai, X., Puigcerver, J., Yung, J., Gelly, S., Houlsby, N.: Big transfer (bit): General visual representation learning. arXiv:1912.11370 (2019)
39. Lin, T.Y., Dollár, P., Girshick, R., He, K., Hariharan, B., Belongie, S.: Feature pyramid networks for object detection. In: Proceedings of the IEEE conference on computer vision and pattern recognition. pp. 2117–2125 (2017)
40. Lin, T.Y., Maire, M., Belongie, S., Hays, J., Perona, P., Ramanan, D., Dollár, P., Zitnick, C.L.: Microsoft coco: Common objects in context. In: European conference on computer vision. pp. 740–755. Springer (2014)
41. Long, J., Shelhamer, E., Darrell, T.: Fully convolutional networks for semantic segmentation. In: Proceedings of the IEEE conference on computer vision and pattern recognition. pp. 3431–3440 (2015)
42. Maurer, A., Pontil, M., Romera-Paredes, B.: The benefit of multitask representation learning. Journal of Machine Learning Research 17(81), 1–32 (2016)
43. Mousavi, M., Khanal, A., Estrada, R.: Ai playground: Unreal engine-based data ablation tool for deep learning. In: International Symposium on Visual Computing. pp. 518–532. Springer (2020)
44. Movshovitz-Attias, Y., Kanade, T., Sheikh, Y.: How useful is photo-realistic rendering for visual learning? arXiv:1603.08152 (2016)
45. Newell, A., Deng, J.: How useful is self-supervised pretraining for visual tasks? arXiv:2003.14323 (2020)
46. Neyshabur, B., Bhojanapalli, S., McAllester, D., Srebro, N.: Exploring Generalization in Deep Learning. In: Advances in Neural Information Processing Systems 30. pp. 5947–5956 (2017)
47. Neyshabur, B., Tomioka, R., Srebro, N.: Norm-based capacity control in neural networks. In: Proceedings of The 28th Conference on Learning Theory. pp. 1376–1401 (2015)
48. Nitanda, A., Chinot, G., Suzuki, T.: Gradient descent can learn less overparameterized two-layer neural networks on classification problems (2020)
49. Nitanda, A., Suzuki, T.: Stochastic gradient descent with exponential convergence rates of expected classification errors. In: Proceedings of the Twenty-Second International Conference on Artificial Intelligence and Statistics. Proceedings of Machine Learning Research, vol. 89, pp. 1417–1426 (2019)
50. Nitanda, A., Suzuki, T.: Optimal rates for averaged stochastic gradient descent under neural tangent kernel regime. In: International Conference on Learning Representations (2021)
51. Polyak, B.T., Juditsky, A.B.: Acceleration of stochastic approximation by averaging. SIAM journal on control and optimization 30(4), 838–855 (1992)
52. Ren, S., He, K., Girshick, R., Sun, J.: Faster r-cnn: Towards real-time object detection with region proposal networks. IEEE transactions on pattern analysis and machine intelligence 39(6), 1137–1149 (2016)

53. Rosenfeld, J.S., Rosenfeld, A., Belinkov, Y., Shavit, N.: A constructive prediction of the generalization error across scales. arXiv:1909.12673 (2019)
54. Russakovsky, O., Deng, J., Su, H., Krause, J., Satheesh, S., Ma, S., Huang, Z., Karpathy, A., Khosla, A., Bernstein, M., Berg, A.C., Fei-Fei, L.: ImageNet Large Scale Visual Recognition Challenge. International Journal of Computer Vision (IJCV) 115(3), 211–252 (2015)
55. Sharma, U., Kaplan, J.: A neural scaling law from the dimension of the data manifold. arXiv:2004.10802 (2020)
56. Su, H., Qi, C.R., Li, Y., Guibas, L.J.: Render for cnn: Viewpoint estimation in images using cnns trained with rendered 3d model views. In: Proceedings of the IEEE International Conference on Computer Vision. pp. 2686–2694 (2015)
57. Sun, C., Shrivastava, A., Singh, S., Gupta, A.: Revisiting unreasonable effectiveness of data in deep learning era. In: Proceedings of the IEEE international conference on computer vision. pp. 843–852 (2017)
58. Suzuki, T.: Fast generalization error bound of deep learning from a kernel perspective. In: Proceedings of the Twenty-First International Conference on Artificial Intelligence and Statistics. vol. 84, pp. 1397–1406 (2018)
59. Teed, Z., Deng, J.: Droid-slam: Deep visual slam for monocular, stereo, and rgb-d cameras. arXiv:2108.10869 (2021)
60. Tobin, J., Fong, R., Ray, A., Schneider, J., Zaremba, W., Abbeel, P.: Domain randomization for transferring deep neural networks from simulation to the real world. arXiv:1703.06907 (2017)
61. Tremblay, J., Prakash, A., Acuna, D., Brophy, M., Jampani, V., Anil, C., To, T., Cameracci, E., Boochoon, S., Birchfield, S.: Training deep networks with synthetic data: Bridging the reality gap by domain randomization. arXiv:1804.06516 (2018)
62. Tripuraneni, N., Jordan, M.I., Jin, C.: On the theory of transfer learning: The importance of task diversity. arXiv:2006.11650 (2020)
63. Wei, C., Ma, T.: Improved Sample Complexities for Deep Neural Networks and Robust Classification via an All-Layer Margin. In: International Conference on Learning Representations (2020)
64. Yang, J., Yan, R., Hauptmann, A.G.: Cross-Domain Video Concept Detection Using Adaptive Svms. In: Proceedings of the 15th ACM International Conference on Multimedia. pp. 188–197. MM '07, Association for Computing Machinery, New York, NY, USA (2007)
65. Zhou, B., Zhao, H., Puig, X., Fidler, S., Barriuso, A., Torralba, A.: Scene parsing through ade20k dataset. In: Proceedings of the IEEE conference on computer vision and pattern recognition. pp. 633–641 (2017)
66. Zhou, B., Zhao, H., Puig, X., Xiao, T., Fidler, S., Barriuso, A., Torralba, A.: Semantic understanding of scenes through the ade20k dataset. arXiv:1608.05442 (2016)
67. Zoph, B., Ghiasi, G., Lin, T.Y., Cui, Y., Liu, H., Cubuk, E.D., Le, Q.V.: Rethinking pre-training and self-training. arXiv:2006.06882 (2020)