Intraclass Compactness: A Metric for Evaluating Models Pre-Trained on Various Synthetic Data

Tomoki Suzuki[1], Kazuki Maeno[1], Yasunori Ishii[2], Takayoshi Yamashita[3]
[1]Panasonic Connect Co., Ltd. [2]Panasonic Holdings Corporation [3]Chubu University
{suzuki.tomoki, maeno.kazuki, ishii.yasunori}@jp.panasonic.com, {takayoshi}@isc.chubu.ac.jp

Abstract

Models pre-trained on synthetic data like computer graphics (CG) and formula-driven supervised learning (FDSL) often underperform models pre-trained on real data in downstream tasks. One approach to resolve this accuracy gap involves defining measurable metrics for differences between real and synthetic data or models trained on these data, and then addressing the gaps in these metrics. Conventional metrics often fail to accurately evaluate all synthetic data types, as they are tailored to specific types or designed for real images. Therefore, we propose utilizing the feature compactness measure as an evaluation metric for finding the gap between models. Our experiments show that our metric strongly correlates with downstream task accuracy across a broad range of synthetic data. Additionally, we demonstrate that our metric is useful for designing training methods using synthetic data.

1. Introduction

Pre-trained models are crucial for achieving high accuracy on downstream tasks, but those pre-trained on real data like ImageNet [3] and JFT-300 [23] face privacy, copyright, and bias issues. Synthetic data, generated via methods like computer graphics (CG) [5, 19] and formula-driven supervised learning (FDSL) [10, 11, 17, 24], offer an alternative. However, models pre-trained on synthetic data often exhibit a performance gap than those on real data.

To find the factor of accuracy gaps between models pre-trained on synthetic data and real data, evaluation metrics other than accuracy have been proposed (Table 1). While accuracy shows just a comparison of performance, metrics are values designed to define an ideal state of a data feature or a model's output and measure how closely they approach that ideal. The gap can be reduced by redesigning the data or model based on that behavior. These metrics can be divided into image-based metrics, focusing on the characteristics of specific synthetic data (Table 1(a)), and model-based metrics, which evaluate the feature distributions of models (Table 1(b)). Each type of metric is described below.

	methods \ data types	real	CG	FDSL
(a)	spectrum [25]	✗	✗	✗
	FID [8]	✓	✓	✗
	SIFTer [6]	✓	✗	✓
(b)	transferability [21, 27]	✓	✓	✗
	ours	✓	✓	✓

Table 1. Correlation of evaluation methods with downstream accuracy. ✓/✗ indicates the presence/absence of correlation. Conventional image-based methods (a) and model-based methods (b) fail to correlate with accuracy for certain synthetic data types (e.g., FDSL data). Our proposed metric is robust across all types of data.

image-based methods (Table 1(a)): Image-based metrics are indicators that represent the ideal state of image characteristics. Representative image-based metrics include the Fréchet Inception Distance (FID) [8] and SIFTer [6]. The FID evaluates the similarity of contextual information by comparing the mean and variance of the feature distributions between pre-training data and downstream task data. SIFTer measures the entropy of the distribution of scale-invariant feature transform (SIFT) features [14, 15] in pre-training data. This is based on the finding by Yoshinski et al. [26] that shallow layers, which extract SIFT-like features, are transferable to downstream tasks. On the other hand, there are various methods for generating synthetic data, such as CG data and FDSL data, and each is designed to mimic different characteristics of real data. These image-based metrics cannot correctly evaluate synthetic data generated based on characteristics unfocused by the metric.

model-based methods (Table 1(b)): Model-based metrics, like transferability [21, 27], analyze feature distributions from pre-trained models when given downstream task data. The core idea in these methods is that models creating well-clustered feature distributions are transferable. These methods are not effective for evaluating models trained on pre-training data with labels that are semantically different from the labels of real data. For example, FDSL data uses mathematical parameters as class labels, which do not correspond to real classes (e.g., ”dog”, ”cat”). Therefore, metrics based on the clustering ability of downstream data are unsuitable for FDSL-trained models due to this label mismatch, failing to correlate with downstream accuracy.

Thus, conventional metrics fail to compare models pre-trained on various synthetic data. Therefore, we investigated the influence of synthetic data label definitions using feature visualization. The results showed that while label definitions impact the separation of interclass features, the clustering of intraclass features is scarcely affected. Consequently, we propose a metric that eliminates label dependency by focusing on intraclass compactness. Our contribution is to utilize the well-known intraclass compactness metric as a general-purpose evaluation tool for pre-trained models and experimentally demonstrate its effectiveness. The results confirmed that our metric correlates more strongly with downstream task accuracy than conventional metrics and can be applied to various synthetic data. Image generation methods for synthetic data are diversifying. Our metric, independent of generation methods and labels, enables broad evaluation and advances future research.

2. Method

Section 2.1 explains and investigates conventional model-based metrics, highlighting their limitations with synthetic data. Section 2.2 then describes our proposed metric, which improves upon these conventional methods.

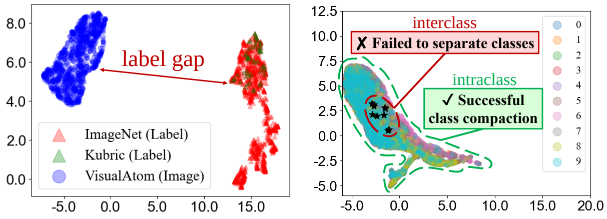
2.1. Conventional model-based metrics and associated issues

2.1.1. Evaluation metrics of pre-trained models based on transferability

Model-based methods aim to rank pre-trained models according to downstream task accuracy when pre-trained models ($m = 1, 2, \ldots, M$) and downstream task data $\mathcal{T}_n = \{(x_n, y_n)\}_{n=1}^{N}$ are given. An easy way to rank the models is to compare the accuracy of the models, but fine-tuning all the models on the downstream task data is computationally expensive. Therefore, model-based methods estimate transferability without fine-tuning. Generally, during transferability evaluation, a downstream task data x_i is input into a feature extractor θ_m of a pre-trained model. From the features $\hat{x_i}$ obtained in this step, the conditional probability $p(y_i \mid x_i; \theta_m)$ of the correct label y_i of the input downstream task data is calculated [21, 27]. Transferability is then calculated as the sum of the logarithms of the conditional probabilities:

$$T_m = \sum_{i=1}^{N} \log p(y_i \mid x_i; \theta_m). \tag{1}$$

In the next step, we calculate the correlation between two model series (i) and (ii): (i) is the model series obtained by arranging the T_m values for multiple pre-trained models in the ascending order, and (ii) is the model series obtained by arranging the accuracy values obtained after fine-tuning the pre-trained models in the ascending order. A transferability metric showing correlation between (i) and (ii) is a good metric reflecting the accuracy of fine-tuned models and is useful for finding the gap between pre-trained models.



(a) Label feature distribution (b) FDSL feature distribution

Figure 1. Effects of labels on conventional transferability evaluation. (a) Distribution of label features in the CLIP model. (b) Feature distribution of CIFAR10 used for the FDSL-trained model. $\star$ represents the mass center of each class feature.

2.1.2. Investigation of issues in conditional probability based on class labels

While real and CG data use names, such as object names and scene names, as class labels, FDSL data assigns class labels based on differences in its generative formula parameters. This labeling scheme is different from the class definitions of real data. Hence, we investigated how differences in class labels between real, CG, and FDSL data affect feature distributions when using transferability as a metric.

To analyze the label difference, we used Contrastive Language-Image Pre-Training (CLIP [20][1]). By measuring the distance in CLIP’s joint feature space between FDSL data features and the text-based label embeddings from real and CG data, we quantify how distant FDSL data is from conventional class definitions. To demonstrate this approach, we extracted features from representative data selected from each class of FDSL data (VisualAtom [24]) and extracted text features from all labels in ImageNet and CG data (Kubric [5]). To visualize the feature distributions, we used Uniform Manifold Approximation and Projection (UMAP [16]).

Figure 1(a) shows the distribution of these features, showing that the FDSL data features are far from real data and CG data class label (text) features. In the CLIP feature space, image and text features are similar, indicating that FDSL data and real or CG data are distant in terms of image features and class label features. General domain shift refers to cases where class labels are the same but image features are different. In this case, image features of data

[1] https://github.com/openai/CLIP is used. The image encoder architecture uses Vision Transformer [4] (ViT-B／32).

with the same label in the CLIP feature space should be similar if the domain shifts. FDSL data are far from real data even in the CLIP feature space, which indicates that FDSL data are out of distribution with real data and CG data not only in terms of image features but also in terms of class labels. Hence, FDSL-trained models cannot correctly represent the feature distributions of real and CG data in relation to the label distributions. In such cases, FDSL-trained models are likely influenced by their dependency on labels y during the calculation of conditional probability $p(y_i \mid x_i; \theta_m)$ in model-based metrics.

Next, we investigated how label mismatch affects transferability by visualizing feature distributions extracted from an FDSL-trained model. Conventional model-based methods define conditional probability $p(y_i \mid x_i; \theta_m)$ on a downstream task as transferability. In other words, transferability is high when the distribution of $\hat{x}$ is separated between classes and concentrated within classes. Therefore, we visualized the distribution of features $\hat{x}$ obtained by inputting real data into a model pre-trained on FDSL (VisualAtom) data. The input downstream task data was CIFAR10 [13].

As shown in Figure 1(b), the FDSL-trained model successfully generates a dense intraclass feature distribution. On the other hand, the model fails to achieve interclass separation, as the feature distributions of different classes overlap. In this way, the effect of the label mismatch between FDSL and real data appears as a failure of interclass separation. However, FDSL data served as a high-performance pre-trained model. Therefore, model-based methods that use the conventional transferability estimation methods cannot accurately evaluate synthetic data with labels that do not match those of real data. Our analysis of FDSL data revealed this label dependency problem. This problem is a potential limitation in conventional transferability evaluation, and the problem does not appear when comparing data with similar class labels, such as real and CG data.

2.2. Proposed metric based on intraclass feature distribution

Despite lower interclass separations on real data because of the label mismatch, an FDSL-trained model shows high performance on downstream tasks. Furthermore, a FDSL-trained model produced sufficiently dense intraclass distributions of features on real data. Thus, the lower intraclass feature variance is an important factor to achieve a high-performance pre-trained model.

Therefore, we propose a metric that uses the within-cluster sum of squares (WCSS) as a metric that can be evaluated for real and synthetic data. The metric for given downstream task data $\mathcal{T}_n = \{(x_n, y_n)\}_{n=1}^{N}$ is calculated as

$$W_m = \frac{1}{(\sum_{i=1}^{k} \sum_{x \in C_i} \|\hat{x} - \mu_i\|^2)/(n-k)}, \tag{2}$$

where k is the number of classes in the downstream task data, C_i is the data group belonging to each class, and μ_i is the centroid vector of features in C_i. As smaller WCSS indicates greater compactness, we take its inverse (Eq. 2) so that higher metric values correspond to higher accuracy.

3. Experiment

3.1. Experiment setting

To validate our proposed metric, we compare it with conventional image-based (FID [8], SIFTer [6]) and model-based (LogMe [27], SFDA [21]) metrics. The comparison is based on the correlation between each metric and downstream task accuracy. We calculate the correlation between each metric and the accuracy of the fine-tuned model using the *weighted Kendall's* τ_w, commonly used in conventional model-based metrics [21, 27]. Details of τ_w can be found in the SciPy implementation[2]. The larger the value of τ_w, the higher the correlation, which makes for a good metric.

The experimental settings for pre-training and the downstream tasks are detailed below. These settings are common to all metrics. The training model was ViT-tiny [4], which has been used in a previous FDSL study [24]. We used cross-entropy as the loss function and followed a previous study for configuring learning settings such as the mini-batch size, learning rate, and data augmentation method. In addition, since the model-based metrics use the output of the feature extractor of a pre-trained model, we used the class tokens of the final transformer block of ViT-tiny.

Regarding pre-training data, we used ImageNet [3] as real data, RCDB [10], VisualAtom [24] as FDSL data; Kubric [5] and VisDA [19] as CG data; and Shaders [1] as other synthetic data. The data scale is 100,000 images for VisDA, 1.2 million images for Kubric, and 1 million images for the other datasets. For comparing a model pre-trained on Kubric with a model pretrained on VisualAtom in Section 3.3, we reduced the amount of data in Kubric to be equivalent to the amount of data in VisualAtom. In particular, we reduced the number of images from the classes with the largest number of images within the class.

We use downstream task data from CIFAR10 [13], CIFAR100 [13], Stanford Cars [12], Flowers [18], IN100 [9], Describable Textures Dataset (DTD) [2], and UCF101 [22]. For fair comparison, we adopted the experimental settings from previous studies, as the amount of data used can differ depending on the metric. We calculated the model-based and proposed metrics using only the test data from the downstream task.

3.2. Experiment results

Table 2 shows the correlation between each metric and the accuracy of fine-tuned models. The conventional image-

[2] https://docs.scipy.org/doc/scipy/reference/generated/scipy.stats.weightedtau.html

Table 2. Comparison of the correlation between evaluation metrics and the accuracy of downstream tasks. The highest values are displayed in **bold**.

Method	C10	C100	Cars	Flowers	IN100	DTD	UCF101	Average
FID	–0.14	–0.13	0.13	–0.4	0.06	–0.65	–0.26	–0.20
SIFTer	0.41	0.37	**0.53**	0.69	**0.53**	0.68	0.32	0.50
LogME	0.32	0.32	0.13	0.72	0.13	**0.76**	0.37	0.39
SFDA	0.32	0.32	0.23	0.69	0.13	0.62	0.37	0.38
WCSS (Eq. (2))	**0.54**	**0.41**	0.47	**0.79**	0.47	**0.76**	**0.74**	**0.59**

based metrics showed low average correlations, as did the model-based metrics. The results support our analysis by showing that model-based metrics (LogME and SFDA) are label dependent and therefore not suitable for evaluating FDSL data across a range of downstream tasks (Figure 1(b)). In contrast, our proposed metric achieved a significantly higher average correlation. Thus, intraclass compactness was less affected by the labels of downstream task data, as revealed in the analysis results in Figure 1(b), and it was observed for various downstream task data. These results demonstrate that during model pre-training, intraclass compactness strongly correlates with downstream task accuracy regardless of the data type (real or synthetic data).

3.3. Combined model training via FDSL and CG data based on the proposed metric

This section shows how the proposed metrics improve the pre-training model and offers insights into future directions for advancing synthetic-image research.

Intraclass compactness measures feature closeness within each class. The correlation between feature compactness and accuracy suggests that better pre-trained models are more robust to intraclass fluctuations. To improve this robustness, we can use data that are robust against different intraclass fluctuations. In this paepr, we attempt to improve robustness by learning a combination of FDSL and CG data. Figure 2 (left) plots WCSS against IN100 downstream accuracy. Despite their differences, FDSL and CG data exhibit similar intraclass compactness. This suggests they differ in robustness. Figure 2 (right) qualitatively confirms this, showing the top 4 feature-similar images retrieved from IN100 queries by models pre-trained on FDSL and CG data. These results imply the FDSL-trained model resists color shifts by selecting images with fine corner shapes, while the CG-trained model resists shape shifts by choosing images with similar colors.

We then tested if dataset combinations based on the metric, each with distinct robustness, enhance performance. We chose VisualAtom [24] (FDSL) and Kubric [5] (CG) for their strong standalone performance. We evaluated three combination methods: multitask learning, continual learning, and image mixing. In multitask learning, we merged FDSL and CG as one dataset, using their combined classes during training. This doubles the data size (2 million) versus using only FDSL (1 million) or CG (1 million). Thus, we randomly sampled half of the data from each class in each dataset. For continuous learning, after pre-training on FDSL data, an additional pre-training step was performed on CG data. The data quantity was handled as in multitask learning. For image mixing, we overlaid FDSL onto CG using PIXMIX [7] and trained models. Table 3 shows that multitask and continual learning boost intraclass compactness (WCSS) over single datasets. All combination methods also improved downstream accuracy. Thus, optimizing intraclass compactness guides improved synthetic-data pre-training and suggests new research directions.

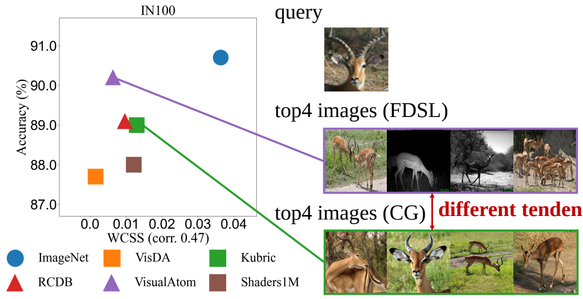


Figure 2. Investigation of downstream task accuracy improvement methods guided by WCSS. (left) WCSS vs. accuracy scatter plot. (right) Randomly selected query images from IN100.

Table 3. Mean WCSS and top-1 accuracy across downstream tasks for different pre-training methods. The highest values are displayed in **bold**.

pre-train	WCSS	accuracy
VisualAtom	0.0067	83.4
Kubric	0.0141	82.9
FDSL (VisualAtom) + CG (Kubric)		
multitask learning	0.0176	84.0
continuous learning	**0.3710**	**85.0**
PIXMIX	0.0112	83.6

4. Conclusion

We proposed a method for assessing pre-trained models on arbitrary synthetic images by utilizing the intraclass compactness (WCSS). By analyzing CLIP features, we found that WCSS was effective because the difference in label definitions between real and synthetic images affects the evaluation of some models using conventional metrics, but has little effect on intraclass compactness. We demonstrated the effectiveness of WCSS through its correlation with downstream tasks. We also confirmed WCSS can guide performance improvement strategies, like combining synthetic data. Thus, WCSS is valuable for designing future synthetic data and training methods.

References

[1] Manel Baradad, Chun-Fu Chen, Jonas Wulff, Tongzhou Wang, Rogerio Feris, Antonio Torralba, and Phillip Isola. Procedural image programs for representation learning. In *NeurIPS*, 2022. 3

[2] M. Cimpoi, S. Maji, I. Kokkinos, S. Mohamed, , and A. Vedaldi. Describing textures in the wild. In *CVPR*, 2014. 3

[3] Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, and Li Fei-Fei. Imagenet: A large-scale hierarchical image database. In *CVPR*, pages 248–255, 2009. 1, 3

[4] Alexey Dosovitskiy, Lucas Beyer, Alexander Kolesnikov, Dirk Weissenborn, Xiaohua Zhai, Thomas Unterthiner, Mostafa Dehghani, Matthias Minderer, Georg Heigold, Sylvain Gelly, Jakob Uszkoreit, and Neil Houlsby. An image is worth 16x16 words: Transformers for image recognition at scale. In *ICLR*, 2021. 2, 3

[5] Klaus Greff, Francois Belletti, Lucas Beyer, Carl Doersch, Yilun Du, Daniel Duckworth, David J. Fleet, Dan Gnanapragasam, Florian Golemo, Charles Herrmann, Thomas Kipf, Abhijit Kundu, Dmitry Lagun, Issam Laradji, Hsueh-Ti (Derek) Liu, Henning Meyer, Yishu Miao, Derek Nowrouzezahrai, Cengiz Oztireli, Etienne Pot, Noha Radwan, Daniel Rebain, Sara Sabour, Mehdi S. M. Sajjadi, Matan Sela, Vincent Sitzmann, Austin Stone, Deqing Sun, Suhani Vora, Ziyu Wang, Tianhao Wu, Kwang Moo Yi, Fangcheng Zhong, and Andrea Tagliasacchi. Kubric: A scalable dataset generator. In *CVPR*, pages 3749–3761, 2022. 1, 2, 3, 4

[6] Ryo Hayamizu, Shota Nakamura, Sora Takashima, Hirokatsu Kataoka, Ikuro Sato, Nakamasa Inoue, and Rio Yokota. SIFTer: Self-improving synthetic datasets for pre-training classification models. In *CVPRW*, 2024. 1, 3

[7] Dan Hendrycks, Andy Zou, Mantas Mazeika, Leonard Tang, Bo Li, Dawn Song, and Jacob Steinhardt. Pixmix: Dreamlike pictures comprehensively improve safety measures. In *CVPR*, pages 16783–16792, 2022. 4

[8] Martin Heusel, Hubert Ramsauer, Thomas Unterthiner, Bernhard Nessler, and Sepp Hochreiter. Gans trained by a two time-scale update rule converge to a local nash equilibrium. In *NeurIPS*. Curran Associates, Inc., 2017. 1, 3

[9] Hirokatsu Kataoka, Kazushige Okayasu, Asato Matsumoto, Eisuke Yamagata, Ryosuke Yamada, Nakamasa Inoue, Akio Nakamura, and Yutaka Satoh. Pretraining without natural images. In *ACCV*, 2020. 3

[10] Hirokatsu Kataoka, Ryo Hayamizu, Ryosuke Yamada, Kodai Nakashima, Sora Takashima, Xinyu Zhang, Edgar Josafat Martinez-Noriega, Nakamasa Inoue, and Rio Yokota. Replacing labeled real-image datasets with auto-generated contours. In *CVPR*, pages 21232–21241, 2022. 1, 3

[11] Hirokatsu Kataoka, Kazushige Okayasu, Asato Matsumoto, Eisuke Yamagata, Ryosuke Yamada, Nakamasa Inoue, Akio Nakamura, and Yutaka Satoh. Pretraining without natural images. *IJCV*, 2022. 1

[12] Jonathan Krause, Michael Stark, Jia Deng, and Li Fei-Fei. 3d object representations for fine-grained categorization. pages 554–561, 2013. 3

[13] Alex Krizhevsky, Geoffrey Hinton, et al. Learning multiple layers of features from tiny images. *Master's thesis, Department of Computer Science, University of Toronto*, 2009. 3

[14] David G Lowe. Object recognition from local scale-invariant features. In *ICCV*, pages 1150–1157. Ieee, 1999. 1

[15] David G. Lowe. Distinctive image features from scale-invariant keypoints. *Int. J. Comput. Vision*, 60(2): 91–110, 2004. 1

[16] Leland McInnes, John Healy, and James Melville. Umap: Uniform manifold approximation and projection for dimension reduction. *arXiv preprint arXiv:1802.03426*, 2018. 2

[17] Kodai Nakashima, Hirokatsu Kataoka, Asato Matsumoto, Kenji Iwata, Nakamasa Inoue, and Yutaka Satoh. Can vision transformers learn without natural images? In *AAAI*, pages 1990–1998, 2022. 1

[18] Maria-Elena Nilsback and Andrew Zisserman. Automated flower classification over a large number of classes. In *ICVGIP*, pages 722–729, 2008. 3

[19] Xingchao Peng, Ben Usman, Neela Kaushik, Dequan Wang, Judy Hoffman, and Kate Saenko. Visda: A synthetic-to-real benchmark for visual domain adaptation. In *CVPRW*, 2018. 1, 3

[20] Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, Gretchen Krueger, and Ilya Sutskever. Learning transferable visual models from natural language supervision. In *ICML*, pages 8748–8763. PMLR, 2021. 2

[21] Wenqi Shao, Xun Zhao, Yixiao Ge, Zhaoyang Zhang, Lei Yang, Xiaogang Wang, Ying Shan, and Ping Luo. Not all models are equal: Predicting model transferability in a self-challenging fisher space. In *ECCV*, pages 286–302. Springer, 2022. 1, 2, 3

[22] Khurram Soomro, Amir Roshan Zamir, and Mubarak Shah. Ucf101: A dataset of 101 human actions classes from videos in the wild. *arXiv preprint arXiv:1212.0402*, 2012. 3

[23] Chen Sun, Abhinav Shrivastava, Saurabh Singh, and Abhinav Gupta. Revisiting unreasonable effectiveness

of data in deep learning era. In *ICCV*, pages 843–852, 2017. 1

[24] Sora Takashima, Ryo Hayamizu, Nakamasa Inoue, Hirokatsu Kataoka, and Rio Yokota. Visual atoms: Pre-training vision transformers with sinusoidal waves. In *CVPR*, pages 18579–18588, 2023. 1, 2, 3, 4

[25] Antonio Torralba and Aude Oliva. Statistics of natural image categories. *Network: computation in neural systems*, 14(3):391, 2003. 1

[26] Jason Yosinski, Jeff Clune, Yoshua Bengio, and Hod Lipson. How transferable are features in deep neural networks? In *NeurIPS*. Curran Associates, Inc., 2014. 1

[27] Kaichao You, Yong Liu, Jianmin Wang, and Mingsheng Long. Logme: Practical assessment of pretrained models for transfer learning. In *ICML*, pages 12133–12143. PMLR, 2021. 1, 2, 3