This MICCAI paper is the Open Access version, provided by the MICCAI Society. It is identical to the accepted version, except for the format and this watermark; the final published version is available on SpringerLink.

FissionFusion: Fast Geometric Generation and Hierarchical Souping for Medical Image Analysis

Santosh Sanjeev[0000−0003−3664−3844], Nuren Zhaksylyk[0009−0003−6571−8902], Ibrahim Almakky✉[0009−0008−8802−7107], Anees Ur Rehman Hashmi[0009−0002−6232−6826], Mohammad Areeb Qazi[0000−0002−1458−7565], and Mohammad Yaqub[0000−0001−6896−1105]

Mohamed bin Zayed University of Artificial Intelligence, Abu Dhabi, UAE
{santosh.sanjeev, nuren.zhaksylyk, ibrahim.almakky, anees.hashmi, mohammad.qazi, mohammad.yaqub}@mbzuai.ac.ae

Abstract. The scarcity of well-annotated medical datasets requires leveraging transfer learning from broader datasets like ImageNet or pre-trained models like CLIP. Model soups averages multiple fine-tuned models aiming to improve performance on In-Domain (ID) tasks and enhance robustness on Out-of-Distribution (OOD) datasets. However, applying these methods to the medical imaging domain faces challenges and results in suboptimal performance. This is primarily due to differences in error surface characteristics that stem from data complexities such as heterogeneity, domain shift, class imbalance, and distributional shifts between training and testing phases. To address this issue, we propose a hierarchical merging approach that involves local and global aggregation of models at various levels based on models' hyperparameter configurations. Furthermore, to alleviate the need for training a large number of models in the hyperparameter search, we introduce a computationally efficient method using a cyclical learning rate scheduler to produce multiple models for aggregation in the weight space. Our method demonstrates significant improvements over the model souping approach across multiple datasets (around 6% gain in HAM10000 and CheXpert datasets) while maintaining low computational costs for model generation and selection. Moreover, we achieve better results on OOD datasets compared to model soups. Code is available at https://github.com/BioMedIA-MBZUAI/FissionFusion.

Keywords: Model Soups · Medical Image Analysis · Model Merging · Transfer Learning

1 Introduction

Deep learning (DL) has emerged as the de facto standard for various computer vision tasks, consistently achieving state-of-the-art performance. A pivotal contributor to this success lies in the availability of pre-trained models. In recent years, a well-established paradigm has emerged: pre-training models on large-scale data, such as ImageNet [5] followed by fine-tuning on target tasks with

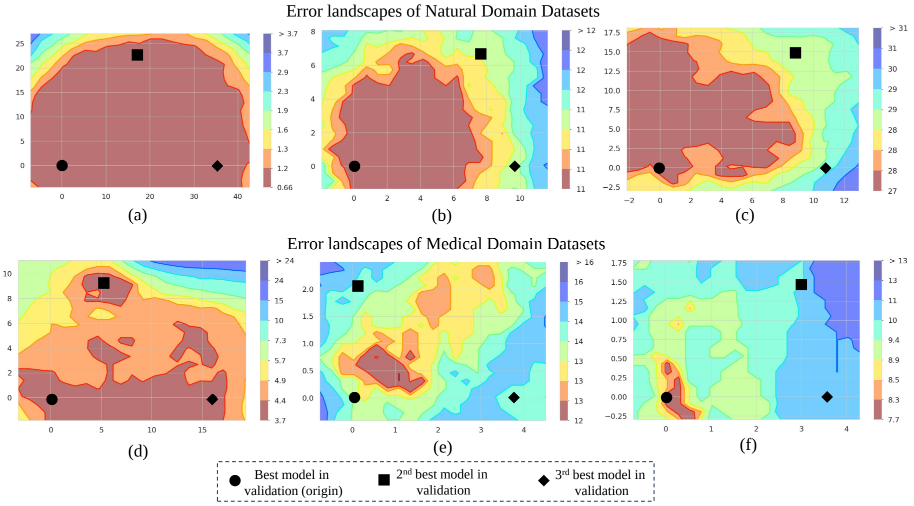

Fig. 1: The validation error on a two-dimensional slice of the error landscapes for various natural and medical domain datasets following [7]. (a) CIFAR-10 [14] (b) CIFAR-100 [14] (c) FGVC-Aircrafts [17] (d) RSNA Pneumonia [28] (e) APTOS [13] (f) HAM10000 [32]. We employ the 3 best-performing models from the validation set, with the best model serving as the reference (origin). (a-c) are characterized by smoothness and convexity, whereas (d-f) exhibit pronounced roughness. The roughness and presence of multiple local minima in loss surfaces of medical datasets are attributed to various intricacies inherent in medical data, including data heterogeneity, domain shift from pre-trained (ImageNet) networks, class imbalance, and distribution shift between the training and testing phases.

limited training data [21,22]. This strategy has proven particularly effective in domains with constrained data availability, such as medical imaging. The scarcity of annotated medical datasets, coupled with the challenges of data acquisition and ethical considerations, highlights the importance of transfer learning from large-scale datasets. Although public data in medical imaging is increasing, it is usually smaller in size compared to natural image datasets, which has led to the widespread adoption of transfer learning from ImageNet [5], to improve performance on medical tasks [20,23,29].

The common practice in transfer learning is to adapt a pre-trained model to a downstream task by fine-tuning. This involves conducting a grid search to explore various hyperparameter combinations and selecting the model that performs best on the validation set. Another approach is employing ensemble techniques [6], where multiple models are utilized simultaneously, albeit at the expense of increased computational and memory requirements, especially at inference time. Recent research by [25] has noted that fine-tuned models optimized independently from the same pre-trained initialization often converge to similar error basins. A previous work [10] has demonstrated that averaging weights along a single training trajectory can enhance model performance in non-transfer settings. Motivated by these observations, [34] proposed model averaging - Model Soups (MS) as an alternative approach to ensembles, aiming to derive a single model that achieves a good performance in terms of accuracy and inference time on In-Domain (ID) as well as Out-of-Distribution (OOD) datasets. They show that averaging several fine-tuned models trained with different hyperparameter settings results in a better model. This method is particularly effective in scenarios with natural imaging datasets or those without significant domain shifts. Several other works have improved upon the model merging concept by adopting Fisher-based weight averaging [19], Task Arithmetic [8], Pruning [37], Gradient-based Matching [3], Tie-Merging [35], and FedSoups [1] for Federated Learning setting. Unlike ensembles, [34,8,37] as well as our work, require only one model for inference. This is especially important in hospital settings, where compute resources are often limited, such as portable ultrasound devices.

However, complexities arise when dealing with medical datasets. As illustrated in Fig. 1, a significant contrast arises between the error surfaces of natural imaging datasets (a-c) and those of medical datasets (d-f). The presence of multiple local minima in (d-f) result in challenging optimization landscapes, where models are prone to getting stuck in local minima. Consequently, the effectiveness of model averaging is compromised, often leading to subpar performance outcomes. Few works have adopted model souping for medical datasets[30,36,16,18]. Most of these studies have applied uniform or greedy souping to few models, and the majority have not explored or analyzed model souping from the perspective of error surfaces. Additionally, the process of model averaging typically involves training multiple models with different hyperparameter settings, which can be computationally intensive. In contrast to previous research, our work focuses on computationally efficient model generation and averaging in a transfer learning context, especially addressing domains that experience significant shifts.

In Model Soups [34], the process of fine-tuning consists of two main steps: (1) Model generation - fine-tuning models with various hyperparameter configurations, and (2) Model selection - selecting the model with the highest accuracy on the held-out validation set and then performing either uniform or greedy souping. Our study explores the challenges associated with both steps in the medical image analysis domain. Drawing from insights in [25] and [7], we propose a Fast Geometric Generation (FGG) approach to generate models with minimal computational overhead. Additionally, we address the model selection process by introducing Hierarchical Souping (HS), which is better suited for medical data because of the aforementioned complexities. Our key contributions are as follows:

- We propose **Fast Geometric Generation (FGG)** approach, which uses cyclical learning rate scheduler in the weight space for efficient model generation. This approach achieves superior results compared to model soups at a lower computational cost.

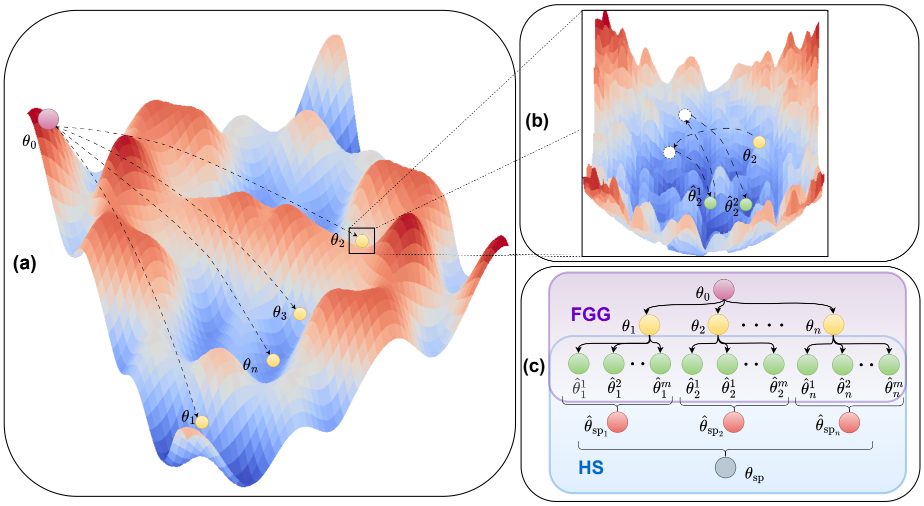
(a)
θ_0
θ_2
θ_3
θ_n
θ_1
(b)
θ_2
$\hat{\theta}_2^1$
$\hat{\theta}_2^2$
(c)
FGG
θ_0
θ_1
θ_2
θ_n
$\hat{\theta}_1^1$ $\hat{\theta}_1^2$ $\hat{\theta}_1^m$
$\hat{\theta}_2^1$ $\hat{\theta}_2^1$ $\hat{\theta}_2^m$
$\hat{\theta}_n^1$ $\hat{\theta}_n^2$ $\hat{\theta}_n^m$
$\hat{\theta}_{\text{sp}_1}$
$\hat{\theta}_{\text{sp}_2}$
$\hat{\theta}_{\text{sp}_n}$
HS
θ_{sp}

Fig. 2: An illustration of (a) Loss landscape of fine-tuned models (b) Fast Geometric Generation(FGG) approach using cyclical learning rate scheduler (c) FGG and the Hierarchical Souping (HS) approach.

- We introduce a novel selection mechanism - **Hierarchical Souping** (HS), tailored specifically for the medical image analysis domain, which performs model averaging at different levels.
- We comprehensively analyze model souping across various datasets in both natural and medical domains. We demonstrate that the combination of FGG and HS for selection significantly enhances results, improves robustness, and increases generalization to out-of-distribution datasets.

2 Methodology

Let $\mathcal{D}_{\text{train}} = \{(x_i, y_i)\}_{i=1}^N$ denote the training dataset where $x_i \in \mathbb{R}^d$ represents the input data and $y_i \in \{1, 2, \ldots, C\}$ denotes the corresponding label from a set of C classes. Similarly, let $\mathcal{D}_{\text{val}} = \{(x_j, y_j)\}_{j=1}^M$ be the validation dataset. We adapt a pre-trained model to our task by performing linear probing for a few epochs as a warm-up step, where we only update the weights of the last layer while freezing the rest of the model. Let θ_0 denote the parameters of the linear probed model. As mentioned above, the souping procedure involves two main steps (1) Generation of the models by fine-tuning (2) Selection of models that perform best on the $\mathcal{D}_{\text{val}}$. The illustration of our proposed approaches (FGG and HS) can be seen in Fig. 2.

Fast Geometric Generation (FGG). We design FGG to generate a range of diverse neural networks while navigating through the weight space taking small steps. The primary objective is to explore variations in network weights without straying into regions likely to lead to low test performance. Instead of employing the conventional hyperparameter grid search, we opt for a more focused strategy. In FGG, we only iterate through a single hyperparameter i.e. (learning rate), and the rest of the hyperparameters are kept constant. This helps to reduce the hyperparameter search space and the computational cost.

In FGG, we start with θ_0 as an initialization and fully train the model using n different learning rates, resulting in a set of parameters $\mathit{\Theta} = \{\theta_1, \ldots, \theta_n\}$. After obtaining $\mathit{\Theta}$, we implement the **fission** process. We initialize the weights $\theta_t \in \mathit{\Theta}$, and carry out a second training process for a set of iterations $I = \{1, \ldots, k\}$ to generate a set of parameters $\hat{\mathit{\Theta}} = \{\hat{\theta}^1, \ldots, \hat{\theta}^m\}$. During this, we employ a learning rate scheduler to cyclically change the learning rate every cycle c, where c is defined as a set of number of iterations and is an even number. This encourages θ_t to diverge from its initialization, while maintaining validation accuracy. We use a function $\alpha(i)$ to control the learning rate at iteration $i \in I$ as follows:

$$\alpha(i) = \begin{cases} \alpha_2.(2t(i)) + \alpha_1(1 - 2t(i)) & \text{if } 0 < t(i) \leq 0.5, \\ \alpha_1.(2t(i) - 1) + \alpha_2(2 - 2t(i)) & \text{if } 0.5 < t(i) \leq 1, \end{cases}$$

where $t(i){=}\frac{1}{c}(\text{mod}(i-1, c) + 1)$, α_2 and α_1 ($\alpha_2 \leq \alpha_1$) are hyperparameters used to control the minimum and maximum learning rates, respectively. Using $\alpha(i)$, we train the network θ_t to form the new set of parameters $\hat{\mathit{\Theta}}$, where each $\hat{\theta} \in \hat{\mathit{\Theta}}$ is the model parameters when the learning rate reaches its minimum value, $\alpha(i) = \alpha_2$. This occurs at the point where $\text{mod}(i-1, c) + 1 = \frac{c}{2}$ and $t(i) = \frac{1}{2}$. We follow this training process for every $\theta_t \in \mathit{\Theta}$, which results in a $\{\hat{\mathit{\Theta}}_1, \ldots, \hat{\mathit{\Theta}}_n\}$.

During high learning rate intervals (close to α_1), the weight θ_t traverses the weight space with larger strides, potentially leading to higher test error. Conversely, in low learning rate episodes, θ_t transitions to smaller steps, reducing test error. This mechanism facilitates incremental movements in weight space, aiding models in evading local minima. Additionally, it gathers diverse models for averaging, reducing the necessity for an extensive grid search. To summarise, **fission** process starts from the base models (each trained with a different learning rate). Then, we cyclically vary the LR, generating multiple models for each base model. This process helps models escape several local minima in rough loss surfaces and generate more generalizable models, facilitating easier model averaging (Fusion using HS).

Hierarchical Souping (HS). In [34], the greedy souping approach outperforms uniform averaging by sequentially adding models to the soup if they improve accuracy on $\mathcal{D}_{\text{val}}$. This approach can yield suboptimal results when the best model is stuck in a local optimum because of the uneven error surface caused due to domain shift. To solve this, we propose Hierarchical Souping, where models are merged at different levels. Starting from the parameter sets $\mathit{\Theta}$ and $\hat{\mathit{\Theta}}$ acquired during the FGG phase, we adopt a local souping approach where the generated models $\{\hat{\Theta}_1, \ldots, \hat{\Theta}_n\}$ are averaged along with the corresponding initialization θ_t (greedy or uniform i.e. $\hat{\theta}_{\mathrm{sp}_t} = \frac{1}{m+1}[\sum_{i=1}^{m} \hat{\theta}_t^i + \theta_t]$) at different levels resulting in $\{\hat{\theta}_{sp_1}, \hat{\theta}_{sp_2}, \ldots, \hat{\theta}_{sp_n}\}$ and a greedy averaging technique is used at the top level giving θ_{sp} (GoG refers to Greedy at all levels whereas GoU refers to Greedy at the top level and Uniform at the lower levels). This approach also known as **fusion** enables networks to escape local minima at the lower levels by local aggregation, giving a good subset of generalizable models that can be averaged in a greedy manner at the top level.

3 Experiments

Implementation Details. Our research investigates two model architectures: DeiT-B (Transformer) and ResNet50 (CNN), both pre-trained on ImageNet. We conduct a 10-epoch warmup with the entire network except the last layer frozen (θ_0). After unfreezing the entire model, we perform full fine-tuning using the LP-FT approach [15]. We use the AdamW optimizer with a cosine annealing scheduler for 50 epochs, a batch size of 128, and an image resolution of 224×224.

For model soups experiments, we conduct a grid search over learning rates, seeds, and augmentations similar to [34]. The learning rates (LR) are (1e-3, 5e-4, 1e-4, 5e-5, 1e-5, 5e-6, 1e-6, 1e-7), and augmentations include minimal (random crop covering 90-100% of image), medium (default timm library settings [33]), and heavy (RandAugment with $N = 2, M = 15$) [2]. Each hyperparameter configuration is run with two seeds, resulting in 48 base models [34].

For our method, we vary only the learning rate, fine-tuning eight models initially with heavy augmentation and a fixed seed. In the Supplementary Material, we analyze the learning rate as a top-level hyperparameter using Linear Mode Connectivity (LMC) in Fig. 1. LMC ($\theta = \lambda \cdot \theta_A + (1-\lambda) \cdot \theta_B$) between two models θ_A and θ_B generated during GS is analyzed by varying λ and calculating performance at θ for each model pair. Ideally, LMC should be an inverted parabola or a straight curve if models are in the same basin. We observe that changing seed and augmentation yield smooth curves (models with large differences cause drops in F1), while LR variations result in erratic patterns and significant F1 drops, indicating models lie in different basins. This suggests LR is crucial in guiding models to specific basins, while other hyperparameters aid in converging to global optima. [34,24] also support LR as a critical hyperparameter.

All experiments use the AdamW optimizer with a cyclical learning rate scheduler, where the learning rate ranges between $\alpha_1 = 1e-5$ and $\alpha_2 = 1e-8$. From each base model, we generate five models using FGG, resulting in a total of 48 models. We train the eight initial models for 50 epochs, then perform the second stage training process (FGG) for 17 epochs (4 epochs per cycle) with a cyclical learning rate, collecting a total of five models per base model.

Datasets. For the primary experiments, we consider two natural imaging domain datasets and five medical domain datasets. We use CIFAR10 [14] and CIFAR100 [14], partitioning the training dataset into train/validation sets in

Table 1: Performance comparison of different methods. (GS (best) - best model on the validation set from Grid Search, FGG(best) - best model on the validation set from Fast Geometric Generation, GoU - Greedy of Uniform, GoG - Greedy of Greedy), Bold numbers mean best and underlined are the second best

Model	**Method**	**CIFAR10 (Acc.)**	**CIFAR100 (Acc.)**	**APTOS (F1)**	**HAM10000 (Recall)**	**RSNA (F1)**	**CheXpert (AUC)**	**EyePACs (F1)**
ResNet50	GS(best)	0.9769	0.8380	0.7086	0.6074	0.9444	0.8444	0.4750
	Uniform Soup	0.8703	0.7652	0.5509	0.5698	0.9171	0.5752	0.1738
	Greedy Soup	0.9769	0.8401	**0.7247**	0.6074	0.9444	0.8444	0.4750
	FGG(best)(Ours)	0.9783	0.8464	0.7172	0.6614	0.9518	0.8434	0.4874
	GoU (Ours)	**0.9785**	0.8457	0.6909	**0.6818**	**0.9545**	0.8351	**0.4905**
	GoG (Ours)	**0.9785**	**0.8477**	0.7172	0.6614	0.9518	**0.8488**	0.4900
DeiT-B	GS(best)	0.9871	0.8919	0.6903	0.6487	0.9503	0.8143	0.4807
	Uniform Soup	0.9386	0.8551	0.1637	0.1429	0.4147	0.7177	0.1697
	Greedy Soup	0.9892	0.8968	0.6785	0.6487	0.9503	0.8068	0.4865
	FGG(best)(Ours)	0.9876	0.8963	**0.7011**	0.6393	0.9529	0.8619	**0.5029**
	GoU (Ours)	0.9899	0.8968	0.6976	**0.6495**	**0.9579**	0.7609	0.4903
	GoG (Ours)	**0.9901**	**0.8987**	0.7003	0.6393	0.9529	**0.8644**	0.4940

a 90%:10% ratio. We utilize the official test split provided. For the CheXpert [9] and HAM10000 [32] datasets, we adhere to the official train/validation/test splits. For APTOS [13], EyePACs [12], and RSNA-Pneumonia [28,27] datasets, we split the data into train/validation/test sets in an 80%:10%:10% ratio, given that only the training dataset was publicly available. Notably, all datasets are multiclass, except CheXpert, which is a multilabel dataset. For the OOD experiments, we consider the CIFAR10.1 [26,31] having 2000 test samples from the natural imaging domain. For the medical imaging domain, we use the Messidor and Messidor-2 [4] datasets, sampling 10% of the data for the test set. We also use the MIMIC-CXR [11] dataset and follow its official test split.

4 Results and Discussion

Table 1 shows that our method achieves better results in both natural and medical imaging domains. Different classification metrics were chosen for different datasets, as they are the metrics of interest used for those datasets [21,**?**]. In CIFAR datasets, GoU and GoG achieve better results than model soups, though it is insignificant due to the smooth convex error surface. However, in medical datasets, we observe around **6%** improvement in Recall (GoG) for the HAM10000 dataset (ResNet50) and around **6%** gain (GoG) in AUC (DeiT-B) for the CheXpert dataset. Both GoU and GoG approaches achieve similar results except for CheXpert (DeiT-B), where GoU lags behind GoG. GoG consistently outperforms Greedy Soup in almost all cases and at a significantly lower computational cost, as we do not perform a full grid search. Model soups GS requires 2400 epochs to generate 48 models (50 epochs each), whereas our FGG requires 536 epochs (8×50 + 8×17), four times less than GS. The time per epoch is the same in both settings. Unlike ensembles, we hierarchically average model weights to get one model without incurring additional inference or memory costs.

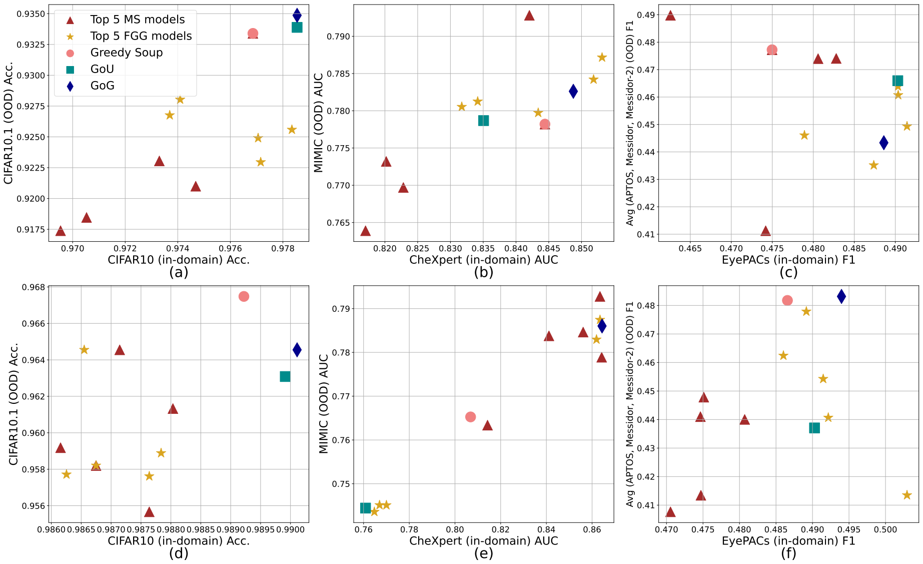

Fig. 3: OOD analysis for different architectures on various datasets (a) CIFAR10 v/s CIFAR10.1 - ResNet50 (b) CheXpert v/s MIMIC - ResNet50 (c) APTOS v/s (EyePacs, Messidor, Messidorv2) - ResNet50 (d) CIFAR10 v/s CIFAR10.1 - DeiT-B (e) CheXpert v/s MIMIC - DeiT-B (f) APTOS v/s (EyePacs, Messidor, Messidorv2) - DeiT-B. We do not plot the results of Uniform Soups as it performs poorly.

Greedy soups do not perform as expected when fine-tuning on medical datasets due to uneven error surfaces. For example, for DeiT-B on the CheXpert dataset, the best model has a high validation score but a low test score, indicating poor generalization. The FGG process overcomes this issue by escaping numerous local minima, while acquired models cluster around the same area in the error surface, facilitating smoother averaging. Local averaging in the HS approach allows smoother averaging at local surfaces and greater diversity at the top level. We conduct an ablation study in Table 1 in the Supplementary Material exploring greedy souping on FGG models and HS on grid-search generated models.

OOD Analysis. We conduct an analysis on OOD scenarios for both natural and medical domain datasets. Performance comparison of souped models on OOD data is important as averaged models should demonstrate robustness to distribution shifts [34]. In Fig. 3, we compare the performance of various approaches in both ID and OOD datasets, where the top five models are selected based on validation set results. In almost all cases, GoU or GoG yields similar or better results than greedy soups in ID and OOD tasks at a much lower computational cost. Our approach results in higher ID and OOD performance gains, particularly for medical imaging datasets, attributed to the FGG approach aiding models in escaping local minima. HS also contributes by facilitating easier averaging between models in the error surface.

5 Conclusion

This work investigates challenges associated with model averaging in transfer learning settings, particularly in medical imaging domain where significant domain shifts can occur. To address this, we introduce Fast Geometric Generation (FGG) leveraging hyperparameter significance and a cyclical learning rate scheduler. Moreover, we propose a Hierarchical Souping mechanism, which involves averaging models at different levels based on the smoothness of the error surface and hyperparameter significance. The proposed generation and selection methodologies yield notable performance enhancements compared to the traditional model souping approach. While our work achieves improved results, we observe instances where models from grid search occasionally outperform averaged models due to very rough error landscapes. This suggests a potential improvement for enhancing generalizability by smoothing the error surface, an aspect we plan to focus on in future endeavors.

Disclosure of Interests. The authors have no competing interests to declare that are relevant to the content of this article.

References

1. Chen, M., Jiang, M., Dou, et al.: Fedsoup: Improving generalization and personalization in federated learning via selective model interpolation. In: International Conference on MICCAI. Springer (2023)
2. Cubuk, E.D., Zoph, B., Shlens, J., Le, Q.: Randaugment: Practical automated data augmentation with a reduced search space. In: Advances in NeurIPS. vol. 33, pp. 18613–18624. Curran Associates, Inc. (2020), https://proceedings.neurips.cc/paper_files/paper/2020/file/d85b63ef0ccb114d0a3bb7b7d808028f-Paper.pdf
3. Daheim, N., Möllenhoff, T., Ponti, E., Gurevych, I., Khan, M.E.: Model merging by uncertainty-based gradient matching. In: The Twelfth ICLR (2024), https://openreview.net/forum?id=D7KJmfEDQP
4. Decencière, E., Zhang, et al.: Feedback on a publicly distributed image database: The messidor database. Image Analysis and Stereology **0** (07 2014). https://doi.org/10.5566/ias.1155
5. Deng, J., Dong, W., Socher, et al.: Imagenet: A large-scale hierarchical image database. In: 2009 IEEE Conference on Computer Vision and Pattern Recognition (2009)
6. Dietterich, T.G.: Ensemble methods in machine learning. In: Multiple Classifier Systems. pp. 1–15. Springer Berlin Heidelberg, Berlin, Heidelberg (2000)
7. Garipov, T., Izmailov, et al.: Loss surfaces, mode connectivity, and fast ensembling of dnns. In: Advances in Neural Information Processing Systems (2018)
8. Ilharco, G., Ribeiro, M.T., Wortsman, M., Schmidt, L., Hajishirzi, H., Farhadi, A.: Editing models with task arithmetic. In: The Eleventh ICLR (2023), https://openreview.net/forum?id=6t0Kwf8-jrj

9. Irvin, J., Rajpurkar, P., Ko, et al.: Chexpert: A large chest radiograph dataset with uncertainty labels and expert comparison. In: Proceedings of the AAAI conference on artificial intelligence (2019)
10. Izmailov, P., Podoprikhin, D., Garipov, T., Vetrov, D., Wilson, A.G.: Averaging weights leads to wider optima and better generalization. arXiv preprint arXiv:1803.05407 (2018)
11. Johnson, A.E., Pollard, T.J., Berkowitz, S.J., Greenbaum, N.R., Lungren, M.P., Deng, C.y., Mark, R.G., Horng, S.: Mimic-cxr, a de-identified publicly available database of chest radiographs with free-text reports. Scientific data **6**(1), 317 (2019)
12. Kaggle: Diabetic Retinopathy Detection. https://www.kaggle.com/competitions/diabetic-retinopathy-detection/data (2015)
13. Karthik, Maggie, S.D.: Aptos 2019 blindness detection (2019), https://kaggle.com/competitions/aptos2019-blindness-detection
14. Krizhevsky, A., Hinton, G.: Learning multiple layers of features from tiny images. Tech. Rep. 0, University of Toronto, Toronto, Ontario (2009), https://www.cs.toronto.edu/~kriz/learning-features-2009-TR.pdf
15. Kumar, A., Raghunathan, A., et al.: Fine-tuning can distort pretrained features and underperform out-of-distribution. In: ICLR (2022)
16. Kvak, D., Chromcová, A., Biroš, et al.: Chest x-ray abnormality detection by using artificial intelligence: A single-site retrospective study of deep learning model performance. BioMedInformatics (2023)
17. Maji, S., Kannala, J., Rahtu, E., Blaschko, M., Vedaldi, A.: Fine-grained visual classification of aircraft. Tech. rep. (2013)
18. Maron, R.C., Hekler, A., Haggenmüller, et al.: Model soups improve performance of dermoscopic skin cancer classifiers. European Journal of Cancer (2022)
19. Matena, M.S., Raffel, C.A.: Merging models with fisher-weighted averaging. Advances in Neural Information Processing Systems (2022)
20. Matsoukas, C., Haslum, J.F., Söderberg, M., Smith, K.: Is it time to replace cnns with transformers for medical images? arXiv preprint arXiv:2108.09038 (2021)
21. Matsoukas, C., Haslum, J.F., Sorkhei, M., Söderberg, M., Smith, K.: What makes transfer learning work for medical images: Feature reuse & other factors. In: Proceedings of the IEEE/CVF Conference on CVPR (2022)
22. Morid, M.A., Borjali, et al.: A scoping review of transfer learning research on medical image analysis using imagenet. Computers in biology and medicine (2021)
23. Morid, M.A., Borjali, A., Del Fiol, G.: A scoping review of transfer learning research on medical image analysis using imagenet. Computers in Biology and Medicine **128** (2021). https://doi.org/10.1016/j.compbiomed.2020.104115
24. Moussa, C., van Rijn, et al.: Hyperparameter importance of quantum neural networks across small datasets. In: International Conference on Discovery Science. Springer (2022)
25. Neyshabur, B., Sedghi, et al.: What is being transferred in transfer learning? Advances in neural information processing systems (2020)
26. Recht, B., Roelofs, R., Schmidt, L., Shankar, V.: Do cifar-10 classifiers generalize to cifar-10? arXiv preprint arXiv:1806.00451 (2018)
27. Shih, G., Wu, C.C., Halabi, et al.: Augmenting the national institutes of health chest radiograph dataset with expert annotations of possible pneumonia. Radiology: Artificial Intelligence **1**(1), e180041 (2019)
28. Stein, A., Wu, C., Carr, C., et al.: Rsna pneumonia detection challenge (2018), https://kaggle.com/competitions/rsna-pneumonia-detection-challenge

29. Tajbakhsh, N., Shin, J.Y., Gurudu, et al.: Convolutional neural networks for medical image analysis: Full training or fine tuning? IEEE TMI (2016)
30. Tenescu, A., Bercean, B.A., et al.: Averaging model weights boosts automated lung nodule detection on computed tomography. In: Proceedings of the 2023 13th International Conference on Bioscience, Biochemistry and Bioinformatics (2023)
31. Torralba, A., Fergus, R., Freeman, W.T.: 80 million tiny images: A large data set for nonparametric object and scene recognition. IEEE TPAMI (2008)
32. Tschandl, P., Rosendahl, C., Kittler, H.: The ham10000 dataset, a large collection of multi-source dermatoscopic images of common pigmented skin lesions. Scientific data **5**(1), 1–9 (2018)
33. Wightman, R.: Pytorch image models. https://github.com/rwightman/pytorch-image-models (2019)
34. Wortsman, M., Ilharco, et al.: Model soups: averaging weights of multiple fine-tuned models improves accuracy without increasing inference time. In: Proceedings of the 39th ICML. PMLR (2022), https://proceedings.mlr.press/v162/wortsman22a.html
35. Yadav, P., Tam, et al.: Resolving interference when merging models. arXiv preprint arXiv:2306.01708 (2023)
36. Zhang, G., Lai, Z.F., et al.: A histopathological image classification method based on model fusion in the weight space. Applied Sciences (2023)
37. Zimmer, M., Spiegel, C., et al.: Sparse model soups: A recipe for improved pruning via model averaging. arXiv preprint arXiv:2306.16788 (2023)