Domain-Adaptive PET/CT Tumor Lesion Segmentation Networks through Effective Training Methods

Suhyun Ahn[1,2] and Jinha Park[1,2]

[1] Computer Graphics and Visualization Lab
[2] Korea Advanced Institute of Science and Technology
ahn.ssu@kaist.ac.kr, jinahpark@kaist.ac.kr

Abstract. Fluorodeoxyglucose (FDG) in positron emission tomography (PET) is crucial for tumor detection due to its ability to reflect glucose consumption, especially in tumors like lung cancer, lymphoma, and melanoma. PET combined with computed tomography (CT) is vital for diagnosing various solid tumors. Accurate segmentation of tumor lesions is essential for precise analysis, but manual segmentation is time-consuming. Automating this process is necessary for widespread clinical use. Recent medical imaging advances highlight the widespread adoption of highly accurate deep learning. However, segmenting tumor lesions in whole-body PET/CT remains challenging due to FDG uptake in healthy organs, causing mis-segmentation. Additionally, the scarcity of meaningful whole-body PET/CT datasets and domain shifts affect model generalization. Addressing these challenges, we propose a robust PET/CT model training approach adaptable to domain shifts, integrating domain-specific knowledge and investigating efficient multimodal fusion methods. The models derived from this approach achieve a 72.43% dice score in validation. Our code is available at https://translate.google.com/details?hl=ko&sl=en&tl=ko&op=translate

Keywords: PET/CT · Feature Fusion · Domain Generalization.

1 Introduction

Fluorodeoxyglucose (FDG) is a widely utilized positron emission tomography (PET) tracer in oncology due to its ability to reflect glucose consumption in tissues, particularly increased glucose consumption observed in tumor lesions. PET combined with computed tomography (CT) has become a standard diagnostic tool for a variety of malignant solid tumor types including lung cancer, lymphoma, and melanoma [5]. An essential initial stage in quantitative PET/CT analysis involves segmenting tumor lesions accurately, which facilitates precise feature extraction, tumor characterization, oncological staging, and evaluation of therapy response using image data. However, like a prevailing issue in most medical imaging, manual lesion segmentation demands significant time and resources, making it unfeasible for routine clinical application. Hence, automating

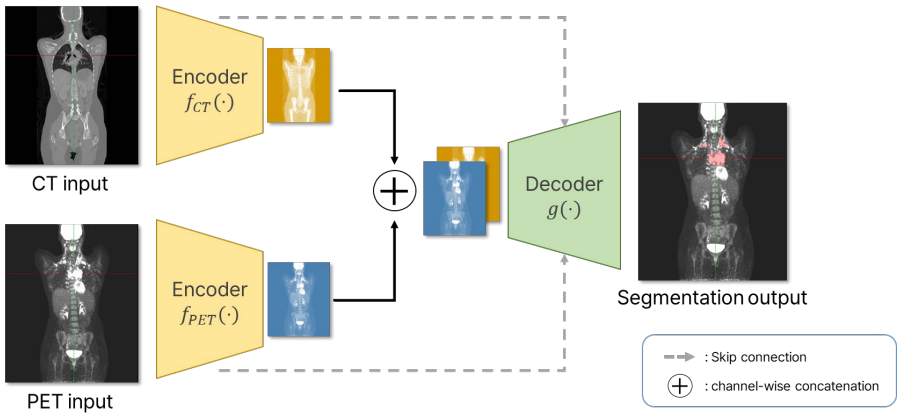

Fig. 1. A figure caption is always placed below the illustration. Please note that short captions are centered, while long ones are justified by the macro package automatically.

this process is vital to enable the extensive integration of comprehensive PET image analysis in clinical settings. On the other hand, recent advancements in medical imaging research have demonstrated the widespread adoption of deep learning, exhibiting high levels of accuracy and performance to the extent of establishing it as a standard [7, 12, 15]. These recent advancements in automated lesion segmentation using deep learning methods have demonstrated the fundamental feasibility of PET/CT tasks. However, despite these strides, tumor lesion detection and segmentation in whole-body PET/CT remains a relatively challenging task. The specific challenge in lesion segmentation in FDG-PET lies in the fact that in addition to tumor lesions, healthy organs (e.g., brain, bladder, heart) can exhibit considerable FDG uptake. Therefore, avoiding misegmentation or false positives can be challenging, and in reality, compared to other challenge datasets, relatively lower performance of deep learning models has been observed. Another aspect that adds to the difficulty of this task is the scarcity of publicly available whole-body PET/CT data. While there is a wealth of publicly available datasets for CT images, allowing for utilization in self-supervised learning [15, 12], the situation is quite the opposite for PET images. There is a scarcity of meaningful volumes with significant spatial size in the available datasets for training deep learning models in the context of PET images. Additionally, in medical image analysis, there exists a challenge where, despite the same modality, the model's generalization performance significantly worsens depending on the data acquisition protocol or site. Efforts have been ongoing to address this issue [2, 14]. With the scarcity of publicly available datasets and the necessity for reliable analysis models, modeling considering domain generalization has emerged as a new requirement in this domain.

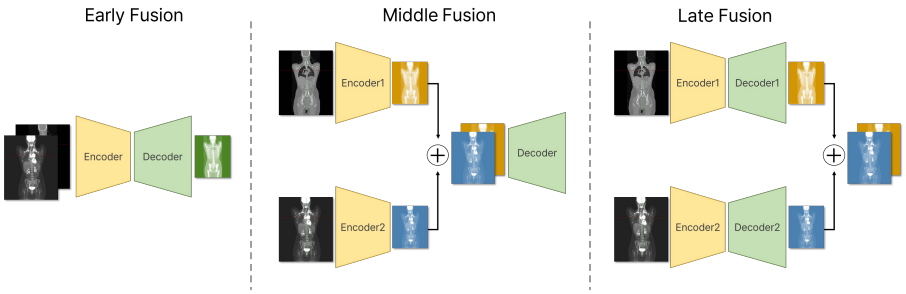

Fig. 2. Overview of the three fusion strategies. Early fusion concatenates the CT and PET volumes in a single input. Middle fusion combines the latent representations from different encoders. Late fusion integrates the outputs of each independent network.

To address these challenges, we formulate and explore a robust PET/CT model training approach that is adaptable to domain shifts. We apply domain knowledge specific to PET to the augmentation methods and adopt training approaches that have proven to be effective for domain generalization based on prior research. Additionally, we investigate effective fusion methods for PET/CT multimodality. The models obtained based on the proposed approach achieve a 72.43% dice score, an average of 8.6254 false positive volumes, and an average of 11.5744 false negative volumes in the validation dataset.

2 Methods

2.1 Network Architectures

The network architecture we adopted for performing whole-body PET/CT segmentation is a modification of the basic UNet structure [7], incorporating middle fusion. Figure 1 provides a summary of the segmentation network 1. The network is designed to have separate encoders for each modality (CT/PET). The encoder for each modality extracts features specific to that modality. The outputs obtained from each encoder are then fed into the middle fusion manner (refer Figure 2), where a channel-wise concatenation is performed, and the combined output is passed to a single decoder. This middle fusion approach enhances the segmentation performance by effectively integrating information from both modalities. The encoder and decoder networks are structured into a total of 5 stages. Each stage consists of 2 convolutional blocks. A single convolution operation is followed by an Instance Normalization layer and SELU activation. The encoder and decoder networks are structured into a total of 5 stages, with each stage comprising 2 convolution blocks. Each block consists of a convolution operation followed by an Instance Normalization layer and SELU activation. In the encoder, downsampling is achieved using a max-pooling layer with a kernel size of 2 and a stride of 2. Conversely, in the decoder, upsampling is performed using transpose convolution layers with the same size and stride. The decoder output

passes through to a convolution with a kernel size of 1, followed by a Softmax layer, resulting in dense predictions.

2.2 Data Pre-processing

The AutoPET training dataset comprises 1,014 PET/CT scans obtained from 900 patients at the University Hospital Tübingen, Germany. Out of these, 513 scans exhibit no lesions (i.e. there is no foreground mask) [3]. Additionally, there are 188 scans confirmed to have malignant melanoma, 168 with lung cancer, and 145 with lymphoma through histological analysis. We employed a stratified split based on these four patient case categories to construct the training and in-validation sets from the provided 1,014 data samples. This approach aims to achieve an effective dataset composition, not only for segmentation but also to reduce False Negatives and False Positives in negative cases. The ratio between the dataset and validation set is 80:20.

We normalized whole-body PET volumes with dimensions of 400x400 to standardized uptake values (SUV) after normalization. The CT scans obtained from PET/CT scans of the same patients were processed to have the same spatial resolution as the transformed SUV volume. The CT images were normalized to have values within the range of [0, 1] through min-max normalization, starting from the original Hounsfield units range of [-1000, 1000]. For the SUV images, we normalized them to one of two value ranges, [0, 20] or [0, 40], probabilistically. The normalization was performed within the consistent range of [0, 1] using min-max normalization without clipping. We adopted this approach for the following reasons: 1) Even with the same acquisition protocol for PET imaging, there can be variations in the acquired image values among individual patients. For instance, despite using the same SUV acquisition method, there can be a significant difference in the composition of image values between two SUV images. 2) AutoPET's final test set was obtained from a total of 200 images, with only 25% acquired from the same domain as the training data. The remaining 75% were obtained from a different domain, which could lead to changes in data distribution due to variations in acquisition protocol and site. To obtain a model that is agnostic to such distribution changes, we used two different normalization ranges for the SUV data in the training dataset. The ranges for SUV normalization values were determined experimentally.

2.3 Augmentation Methods

To train the model, we utilized input volumes of size $160 \times 160 \times 160$. From the whole-body PET/CT scans, we employed two probabilistic methods for cropping: The first method involves cropping to include a positive mask (i.e., tumor lesions) when the input volume's mask contains the foreground. The second method is a completely spatially random crop. In the first strategy, when the mask contains only the background, it behaves similarly to the second method. This approach was chosen because tumor lesions in the dataset constitute a

Table 1. Detailed Configurations of the data augmentations

Augmentation	**probability**	**parameters**
Random flip	0.5	spatial axis=[0,1,2]
Random Rotate	0.2	range = (-15, +15)
Elastic deformation	0.2	$\alpha = (100, 400), \sigma = (5, 13)$
Gamma Transform	0.3	$\gamma = (0.7, 1.5)$
Intensity Shift	0.2	offsets = (-0.1, 0.1)
Intensity Scale	0.2	scaling factor=0.25
Gaussian Smooth	0.2	$\sigma = (0.25, 1.1)$
Low Resolution	0.3	randomly determined
Gaussian Noise	0.2	randomly determined

very small proportion of the entire volume and are not densely located around a specific organ.

Table 1 depicts the augmentation methods and their configurations utilized during model training. Some of the augmentations were adopted by referring to the nnUNet training strategy [7]. Part of the test set data is acquired through different acquisition protocols and sites compared to the training dataset. This difference can lead to variations in spatial resolution and noise levels of the acquired images. To achieve the generalizability of the model, including these variations, we incorporated Gaussian noise and low-resolution simulation into the augmentation methods. The Gaussian noise augmentation was implemented by stochastically generating multiple pairs of mean and standard deviation. During augmentation, one pair is selected, and noise is added accordingly. To prevent excessive mean shifts on the data normalized to the [0,1] scale, the mean parameter passed to the Gaussian noise was constructed through random sampling between 0 and a value between 0 and 0.1.

The low-resolution simulation augmentation was also designed to operate probabilistically. We performed this by increasing the spatial resolution size and then reverting it back to the original size. For this approach, we randomly selected a value between 0 and a specified offset as the maximum size for the increase in resolution. By employing these two methods, we aimed to facilitate the training of a model with excellent generalization performance even on PET/CT images from different domains.

2.4 Loss function

The AutoPET challenge evaluation metrics consider not only a high dice score for segmentation results but also low false positives and false negatives. This notes that simply inducing over-segmentation to boost the dice score is not a viable approach for training the model. The evaluation criteria emphasize achieving a balance between segmentation accuracy and minimizing false positives/negatives. Furthermore, in the case of the AutoPET challenge, it has been

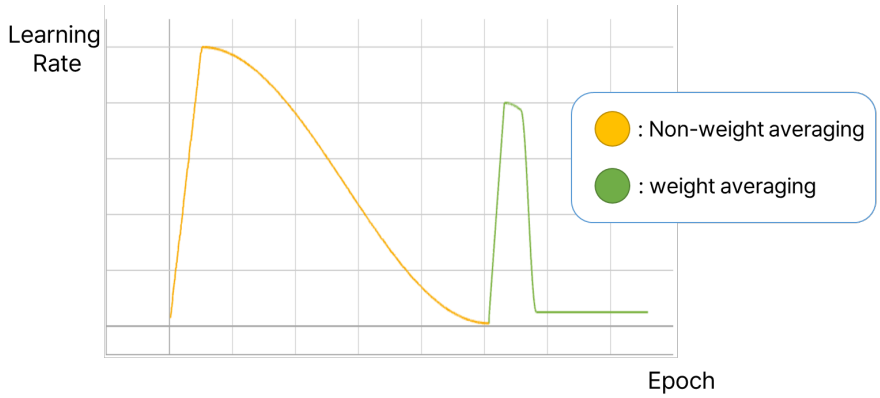

Fig. 3. The learning rate values per iteration for training with Weight Averaging (WA).

proven by last year's participants that obtaining a high dice score was very challenging. Hence, for training the model, we adopted the focal loss [9] along with the dice loss as the loss function. The formula for the loss function is as follows:

$$\mathcal{L} = 1 - \frac{1}{2}\frac{\sum_i^N 2y^i\hat{y}^i + \epsilon}{\sum_i^N y^i + \sum_i^N \hat{y}^i + \epsilon} - \frac{1}{2}\sum_i^N (1-\hat{p}^i)^\gamma \log \hat{p}^i \tag{1}$$

where ϵ is a very small constant used to prevent division errors. y represents the ground truth, and $\hat{y}$ represents the model's prediction. $\hat{p}$ denotes voxel-wise class probability.

2.5 Weight Averaging for Domain Generalization

The AutoPET dataset is composed of both the same domain as the training dataset and different domains [3]. Therefore, to achieve good performance, the model needs to perform well not only on the training dataset's domain but also on various other domains. To address domain generalization, we adopted the Weight Averaging (WA) technique [8]. WA is a method that aims to obtain a model robust to domain shifts in data acquired from different domains, from the perspective of domain generalization. It resolves this by flattening the minima of the trained model.

$$\hat{\theta}_t = \begin{cases} \theta_t, & \text{if } t \leq t_0 \\ \frac{t-t_0}{t-t_0+1} \cdot \hat{\theta}_{t-1} + \frac{1}{t-t_0+1} \cdot \theta_t, & \text{otherwise} \end{cases} \tag{2}$$

where θ_t denotes the model's parameters at iteration t. By periodically updating and averaging the model's parameters during training, the moving average approach helps stabilize the model, making it more robust and enhancing generalization performance across different domains [8]. Specifically, when training

the model, after t iterations, we began weight averaging and utilized annealing to increase the learning rate and then decrease it to a constant learning rate. Figure 3 illustrates the learning rate at the initial state of performing weight averaging. The yellow segment represents the interval during which training is conducted without using weight averaging. On the other hand, the green segment represents the learning rate during the periods when weight averaging is applied. Through annealing, during the initial t_0 iterations, we facilitate substantial movement of the model, which might be oscillating near a specific minima. This prevents the model from getting trapped in a particular minima. Following this, weight averaging (WA) is employed to obtain broader (flatter) minima, contributing to a more stable and generalized model.

Algorithm 1 Ensemble of Averages

Require:
θ_0 pre-trained models;
$\{h_m\}_{m=1}^{H}$ hyperparameter config.
Training:
$\forall m = 1$ to H;
$\theta_m = \text{FineTune}(\Theta_0, h_m)$
Weight selection:
$\text{Rank}\{\theta_m\}_{m=1}^{H}$ by $\text{ValDice}(\theta_m)$
$\mathbb{M} \leftarrow \emptyset$
$\theta_m = \text{FineTune}(\Theta_0, h_m)$
for $m = 1$ to H **do**
    $\text{Dice}_a = \text{ValDice}(\mathbb{M} \cup m)$
    $\text{Dice}_b = \text{ValDice}(\mathbb{M})$
    if $\text{Dice}_a \geq \text{Dice}_b$ **then**
        $\mathbb{M} \leftarrow \mathbb{M} \cup \{m\}$
    end if
end for

Algorithm 2 DiWA (Greedy)

Require:
θ_0 pre-trained models;
$\{h_m\}_{m=1}^{H}$ hyperparameter config.
Training:
$\forall m = 1$ to H;
$\theta_m = \text{FineTune}(\Theta_0, h_m)$
Weight selection:
$\text{Rank}\{\theta_m\}_{m=1}^{H}$ by $\text{ValDice}(\theta_m)$
$\mathbb{M} \leftarrow \emptyset$
$\theta_m = \text{FineTune}(\Theta_0, h_m)$
for $m = 1$ to H **do**
    if $\text{Val}(\theta_{\mathbb{M}\cup m}) \geq \text{Val}(\theta_{\mathbb{M}})$ **then**
        $\mathbb{M} \leftarrow \mathbb{M} \cup \{m\}$
    end if
end for

Multi-runs of Weight Averaging. From the perspective of domain generalization, it has been demonstrated that the WA strategy is effective. However, recent studies suggest that instead of using a single run with one set of hyperparameters, it can be more effective to obtain models using WA across various hyperparameter settings [13, 11, 1]. This involves acquiring a diverse set of models, enabling access to broader flat optima, and potentially achieving higher performance. In this paper, we adopted the methods of Ensemble of Averages (EoA) [1] and Diverse Weight Averaging (DiWA) [11], and the algorithm used is outlined in Algorithm 1 and 2. Both the EoA and DiWA methods utilized a greedy approach and were applied to the Final Test. EoA involves ensemble predictions from each prediction obtained through WA from different runs. On the other hand, DiWA involves performing WA once again on models obtained from WA across different runs.

3 Results

3.1 Implementation Details

In the training step, we maintained the voxel spacing of the original data. An effective batch size of 8 was used, and each model was trained for 1000 epochs. To obtain the pre-trained model θ_t, we used the AdamP [6] optimizer with beta values of (0.9, 0.999) and a weight decay of 0.05. The learning rate started at 1e-4 and was then decreased using a cosine learning rate scheduler. The hidden dimensions of the network were set to [32, 32, 64, 128, 256] for each stage. Additionally, for the CT encoder, we utilized weights trained using the Models Genesis method.

During the prediction phase, we performed tumor lesion segmentation on the whole-body image with background cropping for memory efficiency, based on the PET image. No additional post-processing was applied. All models used were evaluated on the validation dataset using a sliding-window approach with a 0.5 overlap, and a constant kernel. For the final test phase submission, we used the window size of the sliding-window as 192×192×192× size, otherwise 160×160×160 size.

3.2 Feature Fusion

In this section, we compare the performance of models and feature fusion strategies. The models used for comparison are Swin UNETR [4] and U-Net [7]. Due to VRAM limitations, the comparison of feature fusion was conducted with U-Net as the reference. Table 2 presents the evaluation results of the trained models on the validation dataset.

When utilizing the same early fusion, U-Net achieved a dice score of 0.6723. In contrast, Swin-UNETR demonstrated better performance with a dice score of 0.6817, outperforming U-Net. However, with middle fusion, U-Net showed the highest performance, achieving a dice score of 0.6934. Furthermore, using late fusion also resulted in an improvement of over 0.012 compared to early fusion. When conducting experiments to address the imbalance in model capacity and ensure similar parameters, the results did not significantly differ from the previous findings. Consequently, we concluded that middle fusion is the most effective

Table 2. Fusion methods

Network	Fusion methods	params	Dice Score (↑)
SwinUNETR	early fusion	15.7 M	0.6817
U-Net	early fusion	5.7 M	0.6723
U-Net	middle fusion	15.8 M	**0.6934**
U-Net	late fusion	11.5 M	0.6844
U-Net	early fusion	15.4 M	0.6691
U-Net	late fusion	15.9 M	0.6853

Table 3. The prediction results on the validation dataset for models using different WA methods. * denotes the model submitted to the AutoPET final phase.

Model	Dice Score (↑)	False Positive (↓)	False Negative (↓)
Vanilla model	0.6948	**8.5077**	13.4579
WA (Best)	0.7097	9.7973	**10.8431**
EoA (uniform)	0.7160	9.9053	**11.3314**
DiWA (uniform)	0.6597	9.9179	13.9808
*EoA (greedy)	**0.7242**	8.8002	11.3881
*DiWA (greedy)	0.7200	8.6254	11.5744

for learning multi-modal information of PET/CT to perform segmentation, and hence adopted middle fusion as the baseline.

3.3 Weight Averaging

In this section, we conduct an in-depth experiment on Weight Averaging (WA). Table 4 shows the prediction results of the models on the validation dataset for each WA method. The 'Vanilla model' represents the model trained using the conventional training approach without WA. 'WA (BEST)' indicates the best-performing model among models trained with a single run, representing the WA method.

Models trained using the Ensemble of Averages (EoA) [1] and Diverse Weight Averaging (DiWA) [11] methods, were acquired under various hyperparameter settings. The augmentation methods were modified to be stronger or weaker based on the degree specified in Table 1. During the WA step, optimizers such as AdamP [6], AdamW [10], and Ada-belief [16] were used. The learning rate utilized during WA varied accordingly.

Among all the models, the EoA model acquired in a greedy manner exhibited the highest dice score (0.7242) and demonstrated overall satisfactory performance, including low False Positives and False Negatives. The 'vanilla model,' which did not have WA applied, showed the lowest False Positives but conversely had the highest False Negatives. When applying WA and obtaining the 'WA (Best)' model from a single run, we observed a significant reduction in False Negatives, resulting in a dice score increase of 0.7097 compared to when WA was not applied. Both EoA and DiWA demonstrated competitive performance. However, it's worth noting that uniformly applying WA in DiWA led to an overall performance decrease.

Overall, models with a high Dice score, low False Positives, and low False Negatives are suitable for the autoPET evaluation metrics. Furthermore, active consideration should be given to the domain generalization of the model. Therefore, we decided to submit the EoA and DiWA models acquired through a greedy approach. Since the results of the final test submission for this autoPET challenge are not disclosed, the actual performance remains undisclosed.

4 Conclusion

We introduced various approaches to train a domain-agnostic model for whole-body PET/CT scans. We explored tailored augmentation methods to enhance the robustness to acquisition protocols and sites and investigated the utilization of the weight averaging method to effectively achieve domain generalizability at a low cost. Additionally, we evaluated feature fusion strategies for leveraging multi-modality information. Through these efforts, we expect to obtain models that are both high-performing and possess strong generalizability.

References

[1] Devansh Arpit et al. *Ensemble of Averages: Improving Model Selection and Boosting Performance in Domain Generalization*. 2022. arXiv: 2110.10832 [cs.LG].

[2] Mohammad Atwany and Mohammad Yaqub. "DRGen: Domain Generalization in Diabetic Retinopathy Classification". In: *Medical Image Computing and Computer Assisted Intervention – MICCAI 2022*. Ed. by Linwei Wang et al. Cham: Springer Nature Switzerland, 2022, pp. 635–644. ISBN: 978-3-031-16434-7.

[3] Sergios Gatidis et al. "A whole-body FDG-PET/CT Dataset with manually annotated Tumor Lesions". In: *Scientific Data* 9.1 (Oct. 2022), p. 601. ISSN: 2052-4463. DOI: 10.1038/s41597-022-01718-3. URL: https://doi.org/10.1038/s41597-022-01718-3.

[4] Ali Hatamizadeh et al. *Swin UNETR: Swin Transformers for Semantic Segmentation of Brain Tumors in MRI Images*. 2022. arXiv: 2201.01266 [eess.IV].

[5] Mathieu Hatt et al. "Characterization of PET/CT images using texture analysis: the past, the present... any future?" In: *European Journal of Nuclear Medicine and Molecular Imaging* 44.1 (June 2016), pp. 151–165. DOI: 10.1007/s00259-016-3427-0. URL: https://doi.org/10.1007/s00259-016-3427-0.

[6] Byeongho Heo et al. *AdamP: Slowing Down the Slowdown for Momentum Optimizers on Scale-invariant Weights*. 2021. arXiv: 2006.08217 [cs.LG].

[7] Fabian Isensee et al. *nnU-Net: Self-adapting Framework for U-Net-Based Medical Image Segmentation*. 2018. arXiv: 1809.10486 [cs.CV].

[8] Pavel Izmailov et al. *Averaging Weights Leads to Wider Optima and Better Generalization*. 2019. arXiv: 1803.05407 [cs.LG].

[9] Tsung-Yi Lin et al. *Focal Loss for Dense Object Detection*. 2018. arXiv: 1708.02002 [cs.CV].

[10] Ilya Loshchilov and Frank Hutter. *Decoupled Weight Decay Regularization*. 2019. arXiv: 1711.05101 [cs.LG].

[11] Alexandre Ramé et al. *Diverse Weight Averaging for Out-of-Distribution Generalization*. 2023. arXiv: 2205.09739 [cs.CV].

[12] Yucheng Tang et al. *Self-Supervised Pre-Training of Swin Transformers for 3D Medical Image Analysis*. 2022. arXiv: 2111.14791 [cs.CV].

[13] Mitchell Wortsman et al. *Model soups: averaging weights of multiple fine-tuned models improves accuracy without increasing inference time*. 2022. arXiv: 2203.05482 [cs.LG].

[14] Chundan Xu et al. "Improved Domain Generalization for Cell Detection in Histopathology Images via Test-Time Stain Augmentation". In: *Medical Image Computing and Computer Assisted Intervention – MICCAI 2022*. Ed. by Linwei Wang et al. Cham: Springer Nature Switzerland, 2022, pp. 150–159. ISBN: 978-3-031-16434-7.

[15] Zongwei Zhou et al. "Models Genesis". In: *Medical Image Analysis* 67 (Jan. 2021), p. 101840. DOI: 10.1016/j.media.2020.101840. URL: https://doi.org/10.1016%2Fj.media.2020.101840.

[16] Juntang Zhuang et al. *AdaBelief Optimizer: Adapting Stepsizes by the Belief in Observed Gradients*. 2020. arXiv: 2010.07468 [cs.LG].