Resolution-Robust Medical Image Registration Method Based on Fourier Neural Operator: Implementation and Reproducibility Aspects*

Nikita A. Drozdov[0009-0002-2785-0294] and Dmitry V. Sorokin[0000-0003-3299-2545]

Laboratory of Mathematical Methods of Image Processing,
Faculty of Computational Mathematics and Cybernetics,
Lomonosov Moscow State University, Moscow, Russia
drozdovna@my.msu.ru, dsorokin@cs.msu.ru

Abstract. This article is a companion paper to our publication for ICPR 2024, "FNOReg: Resolution-Robust Medical Image Registration Method Based on Fourier Neural Operator" [7], in which we proposed an algorithm for medical image registration that offers robustness to input data resolution. This companion paper presents the method's implementation, the methodology for hyperparameter selection, and other details that ensure the reproducibility of our research.

Keywords: Biomedical image registration · Fourier neural operator · Unsupervised learning · Reproducible research

1 Introduction

Image registration is one of the most important tasks in medical imaging. This task can be considered an optimization problem and solved using numerical algorithms. Methods based on this approach include Free-Form Deformation [18], LDDMM [3], Elastix [12], Flash [19], and others. However, all these algorithms require parameter tuning for each pair of images and therefore cannot be effectively used for registering large images in real time. Recently, neural network-based methods have demonstrated state-of-the-art performance in image registration. Their main advantage is fast model inference, requiring only one set of hyperparameters for model training. The most popular models in this field include, but are not limited to, VoxelMorph [2], TransMorph [4], Fourier-Net [10], and Fourier-Net+ [11].

In medical image processing, images often have a large size, requiring significant memory for storage and processing. For deep learning-based methods, this means that these images may not fit in GPU memory during model training or inference. To solve this problem, we propose FNOReg [7], a novel method for

* The work was financially supported by the Russian Science Foundation under the research project No. 22-41-02002.

medical image registration based on the Fourier Neural Operator [13] architecture. The main advantage of this model is its robustness to input data resolution, allowing the model to be trained on images with reduced resolution and used on full-size images without significant degradation in registration quality.

In this companion paper, we discuss the implementation of our method [7] and its features that make it easier to reproduce our results. Section 2 briefly explains the proposed method. Section 3 describes some technical details about the source code of our algorithm. In Section 4, we present the key aspects of the reproducibility of our research. Section 5 summarizes the paper.

2 Proposed method

In our work [7], it was shown that the image registration task can be considered as solving a system of partial differential equations. In recent years, special neural network architectures called neural operators [14] have been used for such tasks. One of them is the Fourier Neural Operator (FNO) [13]. The architecture of the FNO is inspired by the mathematical principles that lie behind the solution of certain types of operator equations. In more detail, the FNO approximates the solution of an operator equation using the following iterative process:

$$u_{t+1}(\boldsymbol{x}) := \sigma\left(W_t u_t(\boldsymbol{x}) + (K u_t)(\boldsymbol{x})\right),\ t = 0, \ldots, N-1, \tag{1}$$

where

$$(K u_t)(\boldsymbol{x}) := \int_{\Omega} \kappa_t(\boldsymbol{x} - \boldsymbol{y}) u_t(\boldsymbol{y})\, d\boldsymbol{y}\,. \tag{2}$$

Here, κ_t is the learnable convolution kernel, which usually has a finite support due to computational constraints. To learn a global convolution kernel, we parametrize κ_t directly in the frequency domain:

$$(K u_t)(\boldsymbol{x}) = \mathcal{F}^{-1}\left(\mathcal{F}(\kappa) \cdot \mathcal{F}(u_t)\right) = \mathcal{F}^{-1}\left(R \cdot \mathcal{F}(u_t)\right)\,, \tag{3}$$

where $R(\boldsymbol{k}) \in \mathbb{C}^{d_{u_{t+1}} \times d_{u_t}}$ is a learnable function in Fourier space, $\mathcal{F}$ is the Fourier transform operator, and $\boldsymbol{k} = (k_1, \ldots, k_n) \in \mathbb{N}^n$ corresponds to non-negative frequencies. $R(\boldsymbol{k})$ has nonzero values only at fixed number of lower frequencies where $k_i \leq k_{max,i}$, which ensures the model's robustness to input data resolution.

In our work [7], we proposed the FNOReg model, whose architecture is depicted in Fig. 1. This model includes several additions and enhancements over the classical FNO, resulting in better performance and robustness during training.

Our first enhancement is the utilization of feature extractors (depicted as red rectangles in Figure 1) designed with the Spectral Transform layer [6]. Unlike a standard convolution layer, the Spectral Transform layer has a global receptive field, allowing us to extract non-local details from the feature maps in our network. This improvement leads to better registration quality while preserving the model's robustness to input resolution.

Our second improvement involves adding residual connections in the Fourier layers after the activation function. These connections bring more stability to the learning process, thus making FNOReg more robust than the standard FNO.

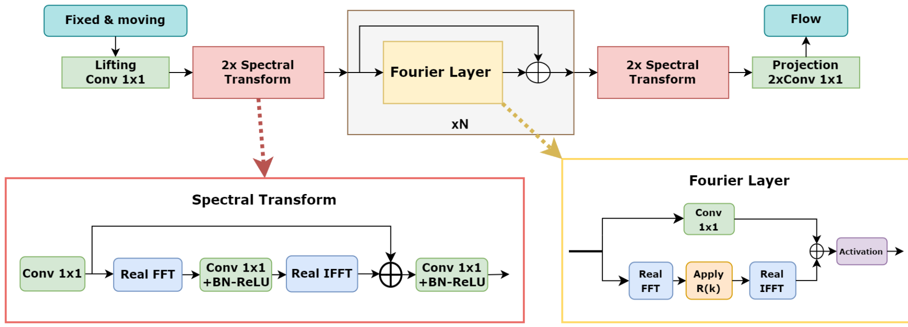

Fig. 1. FNOReg model architecture.

3 Implementation details

We implemented our method using the PyTorch [17] library, which has become a de facto standard among deep learning researchers. For FNO-based models, we used the neuralop [13] library, which contains the original implementation of the Fourier Neural Operator. The source code for our method is available in an open GitHub repository[1], where one can find instructions for accurately reproducing our results. The repository also contains scripts for model training and evaluation, as well as modules for data processing and result plotting.

Following good coding practices, our implementation includes different classes and functions with informative parameter names. For convenience, we have written separate scripts for model training and evaluation (`train_*.py` and `evaluate_*.py`, respectively). These scripts also support versioning of experiments and saving detailed logs of the model training process, thereby enhancing the reproducibility of our results (see Section 4.2 for details).

4 Reproducibility aspects

4.1 Dataset selection

The right choice of dataset is crucial for reproducible research. If one evaluates their models on less well-known data or even on a closed dataset, the results become difficult to compare with those of other models for the same task. Therefore, to obtain reproducible results and enable a representative comparison with other research papers, the dataset should be well-known and openly available for anyone to download. For these reasons, we chose the OASIS-1 dataset [15] for training and evaluating our models. OASIS-1 consists of a cross-sectional collection of 414 MRI scans from subjects aged 18 to 96, with each scan having segmentation masks for 35 important anatomical areas. 2D data can be obtained

[1] https://github.com/anac0der/fnoreg

by extracting a specific slice in the xy-plane from the original 3D data. In our experiments, we used the preprocessed version of the 2D and 3D OASIS-1 dataset from [9], where all scans were bias-corrected, affinely aligned, and cropped to a size of 160×192 and $160 \times 192 \times 224$ for 2D and 3D data, respectively. Images from the raw and preprocessed datasets, along with different segmentation maps, are depicted in Fig. 2.

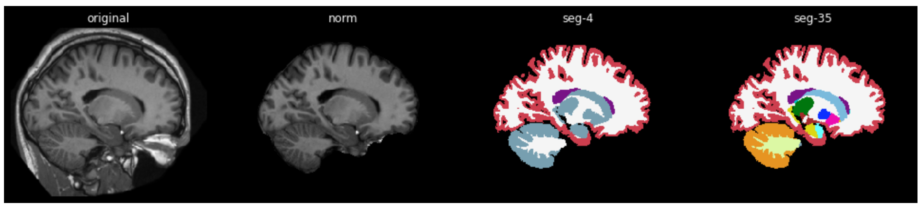

Fig. 2. Vertical slices of images from the OASIS-1 dataset. From left to right: image from the raw version of the dataset, image from the preprocessed version [9], 4-label tissue-type segmentation, and 35-label segmentation of major anatomical regions.

From the point of view of popularity, the OASIS-1 dataset has been used for training and evaluation in most classical and state-of-the-art papers in the field of neural network-based medical image registration, such as VoxelMorph [2], TransMorph [4], Fourier-Net [10], LapIRN [16], ViT-V-Net [5], and H-ViT [8]. This makes it easier to compare the results from our work with those of others and to reproduce the values presented in our paper.

4.2 Versioning of experiments

Versioning experiments is an important feature of reproducible research in the field of deep learning. It is easier to compare different models if you have the ability to evaluate any model from your past experiments. Our implementation offers the user such features. When starting the training of the model with the training scripts, a new directory is created. This directory contains model checkpoints, final model weights, TensorBoard logs [1], and plots of validation metric progress. Each experiment is assigned a unique number, which can later be used to evaluate the model from that experiment. The user can also set parameters for the model and training process in a configuration file.

More specifically, each training script should be launched with the following command-line parameters:

1. `--gpu_num` – the number of the graphics card in your system on which the model will be trained.
2. `--config_file` – the name of the configuration file. This file is in JSON format and contains the setup of hyperparameters for the current experiment (model name, number of layers, number of convolutional filters in each layer, learning rate, batch size, etc.).

3. `--exp_num` – an optional argument for experiment restarting. By default, experiments are numbered in the order in which they are launched. If you want to restart experiment number N with different parameters, you should change these parameters in the configuration file located in the experiment folder. Then you can rerun the training script with the `exp_num` argument equal to N.
4. `--ckpt_epoch` – an optional argument to specify the epoch from which you want to restart the learning process.
5. `--size` – the size of the smaller dimension in the input image shape (useful for experiments at reduced resolution).

For model evaluation, the corresponding script should be launched with the following parameters:

1. `--gpu_num` – the number of the graphics card in your system on which the model will be evaluated.
2. `--exp_num` – the model will be downloaded from the folder of the experiment with this number.
3. `--ckpt_epoch` – an optional argument that defines the epoch of the training process from which the model will be downloaded.

The TensorBoard web interface, which displays plots of training and validation losses, can be launched using the following terminal command:

```
tensorboard --logdir path_to_exp_folder/logs
```

Here, `path_to_exp_folder` is the path to the folder that was created for saving the experiment.

4.3 Tuning of hyperparameters

In deep learning, the choice of hyperparameters is one of the most important parts of the entire training pipeline. It is difficult to find the optimal combination of hyperparameters because the search space is often not finite. There are several methods for searching for suboptimal combinations of parameters (manual search, grid search, random search, etc.), all of them have their own pros and cons.

Here, we will focus on the search for the optimal combination of parameters λ and γ, which are part of the loss function for our models using grid search. The parameter λ controls the regularization strength, while the parameter γ controls the contribution of the Dice loss in the overall loss function. Our search strategy for the optimal combination of these parameters is similar to coordinate descent. First, we performed a grid search for the λ parameter on the OASIS 2D dataset, where we did not use auxiliary information (and therefore set $\gamma = 0$ for the 2D data). Next, we repeated the process for γ on the 3D data, using a fixed value ofλ that corresponds to the value with the best performance from the previous step.

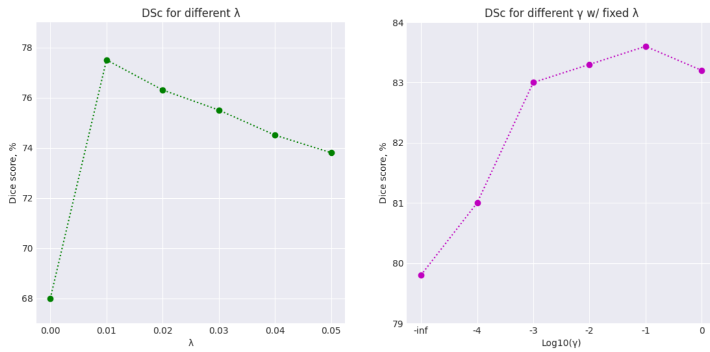

Fig. 3. Dice scores of different models with varying combinations of λ and γ parameters of the loss function. Left: performance of the FNOReg (large) model on the OASIS 2D dataset with different λ values. Right: performance of the FNOReg model on the OASIS 3D dataset with different γ and $\lambda = 0.01$ (corresponding to the best Dice score in the left plot).

Results of the parameters search are presented in Fig. 3. It can be seen from the left plot that $\lambda = 0.01$ yields the best performance, confidently surpassing the model scores with other values of λ. On the right plot, model Dice scores with $\gamma = 0.01$, $\gamma = 0.1$, and $\gamma = 1$ are almost the same (0.833, 0.836, and 0.832, respectively). However, the model with $\gamma = 0.01$ obtains a smoother deformation field than the models with $\gamma = 0.1$ and $\gamma = 1$, because regularization gets weaker as γ increases. This can also be seen by counting the mean percentage of folded voxels (voxels in which the determinant of the deformation field Jacobian is negative) for each model: 0.329 for $\gamma = 0.01$, 1.349 for $\gamma = 0.1$, and 4.389 for $\gamma = 1$. Taking this into account, we chose the value of $\gamma = 0.01$ as optimal, obtaining $\lambda = 0.01$ and $\lambda = \gamma = 0.01$ as the final combinations of hyperparameters in the loss function for 2D and 3D data, respectively.

5 Conclusion

This paper provides a detailed description of the important technical aspects of our main research work [7]. We introduce various details about the implementation of the resolution-robust medical image registration method proposed in our main paper. Our implementation is open to everyone and consists of high-quality code, which will greatly assist researchers in the fields of medical imaging and deep learning. The methodology for hyperparameter selection is also presented, including a detailed description of our strategy and rationale for choosing the final combination of parameters. Additionally, we address several aspects such as versioning of experiments and dataset selection, which enhance the reproducibility of our research.

Acknowledgements For computational experiments, neural network training, and fine-tuning, we utilized the MSU-270 supercomputer of Lomonosov Moscow State University with Nvidia Tesla A100 80GB GPUs.

References

1. Abadi, M., Agarwal, A., Barham, P., Brevdo, E., Chen, Z., Citro, C., Corrado, G.S., Davis, A., Dean, J., Devin, M., et al.: Tensorflow: Large-scale machine learning on heterogeneous distributed systems. arXiv preprint arXiv:1603.04467 (2016)
2. Balakrishnan, G., Zhao, A., Sabuncu, M.R., Guttag, J., Dalca, A.V.: Voxelmorph: a learning framework for deformable medical image registration. IEEE transactions on medical imaging **38**(8), 1788–1800 (2019)
3. Beg, M.F., Miller, M.I., Trouvé, A., Younes, L.: Computing large deformation metric mappings via geodesic flows of diffeomorphisms. International journal of computer vision **61**, 139–157 (2005)
4. Chen, J., Frey, E.C., He, Y., Segars, W.P., Li, Y., Du, Y.: Transmorph: Transformer for unsupervised medical image registration. Medical image analysis **82**, 102615 (2022)
5. Chen, J., He, Y., Frey, E.C., Li, Y., Du, Y.: Vit-v-net: Vision transformer for unsupervised volumetric medical image registration. arXiv preprint arXiv:2104.06468 (2021)
6. Chi, L., Jiang, B., Mu, Y.: Fast fourier convolution. Advances in Neural Information Processing Systems **33**, 4479–4488 (2020)
7. Drozdov, N., Sorokin, D.: FNOReg: Resolution-Robust Medical Image Registration Method Based on Fourier Neural Operator. In: 27th International Conference on Pattern Recognition (2024)
8. Ghahremani, M., Khateri, M., Jian, B., Wiestler, B., Adeli, E., Wachinger, C.: H-vit: A hierarchical vision transformer for deformable image registration. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 11513–11523 (2024)
9. Hoopes, A., Hoffmann, M., Greve, D.N., Fischl, B., Guttag, J., Dalca, A.V.: Learning the effect of registration hyperparameters with hypermorph. The journal of machine learning for biomedical imaging **1** (2022)
10. Jia, X., Bartlett, J., Chen, W., Song, S., Zhang, T., Cheng, X., Lu, W., Qiu, Z., Duan, J.: Fourier-net: Fast image registration with band-limited deformation. In: Proceedings of the AAAI Conference on Artificial Intelligence. vol. 37, pp. 1015–1023 (2023)
11. Jia, X., Thorley, A., Gomez, A., Lu, W., Kotecha, D., Duan, J.: Fourier-net+: Leveraging band-limited representation for efficient 3d medical image registration. arXiv preprint arXiv:2307.02997 (2023)
12. Klein, S., Staring, M., Murphy, K., Viergever, M.A., Pluim, J.P.: Elastix: a toolbox for intensity-based medical image registration. IEEE transactions on medical imaging **29**(1), 196–205 (2009)
13. Li, Z., Kovachki, N., Azizzadenesheli, K., Liu, B., Bhattacharya, K., Stuart, A., Anandkumar, A.: Fourier neural operator for parametric partial differential equations. arXiv preprint arXiv:2010.08895 (2020)
14. Li, Z., Zheng, H., Kovachki, N., Jin, D., Chen, H., Liu, B., Azizzadenesheli, K., Anandkumar, A.: Physics-informed neural operator for learning partial differential equations. ACM/JMS Journal of Data Science (2021)

15. Marcus, D.S., Wang, T.H., Parker, J., Csernansky, J.G., Morris, J.C., Buckner, R.L.: Open access series of imaging studies (oasis): cross-sectional mri data in young, middle aged, nondemented, and demented older adults. Journal of cognitive neuroscience **19**(9), 1498–1507 (2007)
16. Mok, T.C., Chung, A.C.: Large deformation diffeomorphic image registration with laplacian pyramid networks. In: Medical Image Computing and Computer Assisted Intervention–MICCAI 2020: 23rd International Conference, Lima, Peru, October 4–8, 2020, Proceedings, Part III 23. pp. 211–221. Springer (2020)
17. Paszke, A., Gross, S., Massa, F., Lerer, A., Bradbury, J., Chanan, G., Killeen, T., Lin, Z., Gimelshein, N., Antiga, L., et al.: Pytorch: An imperative style, high-performance deep learning library. Advances in neural information processing systems **32** (2019)
18. Rueckert, D., Sonoda, L.I., Hayes, C., Hill, D.L., Leach, M.O., Hawkes, D.J.: Non-rigid registration using free-form deformations: application to breast mr images. IEEE transactions on medical imaging **18**(8), 712–721 (1999)
19. Zhang, M., Fletcher, P.T.: Fast diffeomorphic image registration via fourier-approximated lie algebras. International Journal of Computer Vision **127**, 61–73 (2019)