NeuroBridge: Few-Shot Cross-Modal Neuron Re-identification via Dual-Channel Deep Metric Learning

Wenwei Li Mingwei Liao Lingyi Cai Anan Li*
Huazhong University of Science and Technology
liwenwei@hust.edu.cn, liaomingwei@hust.edu.cn, lycai@hust.edu.cn, aali@hust.edu.cn

1. Supplementary Material

1.1. Training and Implementation Details

Frameworks and Libraries The experiments in this study were conducted using the PyTorch deep learning framework. We leveraged the `open-metric-learning` [4] and `pytorch-metric-learning` [3] libraries for implementing key metric learning components, including the loss function and batch sampling strategies.

Hyperparameters For the training configuration, we used the Adam optimizer [1] with an initial learning rate of 1×10^{-4}. The embedding space dimension d was set to 768 to match the output dimension of the ViT-Base model's feature extractor. For the CircleLoss function, the scale factor γ was set to 256 and the margin m was set to 0.25.

Batch Sampling To ensure each training batch contains a rich set of positive and negative pairs, we utilized a `BalancedSampler`. Specifically, each batch was constructed by first selecting 50 distinct neuron identities (classes). For each selected identity, both its *in-vivo* and *ex-vivo* image instances were sampled. This strategy guarantees that every batch contains 50 positive pairs and numerous hard negatives, resulting in an effective batch size of 100 (50 classes $\times$ 2 instances).

Hardware All model training and evaluation tasks were performed on a single NVIDIA A6000 GPU.

1.2. Comparison with Fine-tuning Methods

To provide a more intuitive evaluation of our method against various fine-tuning strategies, we use box plots to visualize the distribution of feature distances. As shown in Figure 1, we compare the feature distance distributions between image pairs from the "Same Neuron" (red box) and "Different Neurons" (blue box). An ideal model should yield small and concentrated distances for same-neuron pairs and large distances for different-neuron pairs, with minimal overlap between the two distributions.

The comparison clearly demonstrates that our proposed method ("Ours") achieves the best performance. The feature distances for "Same Neuron" pairs are not only lower but also more tightly clustered (i.e., a narrower box). Conversely, the distances for "Different Neuron" pairs are significantly higher, creating a clear separation with the least overlap. This contrasts sharply with other methods, such as "Without Fine-Tuning" and "Linear Fine-Tuning", which show considerable overlap. While "Partial Parameter", "LoRA", and "Full Parameter" tuning show improvements, our method exhibits superior discriminative power, confirming its effectiveness in learning highly separable neuron features.

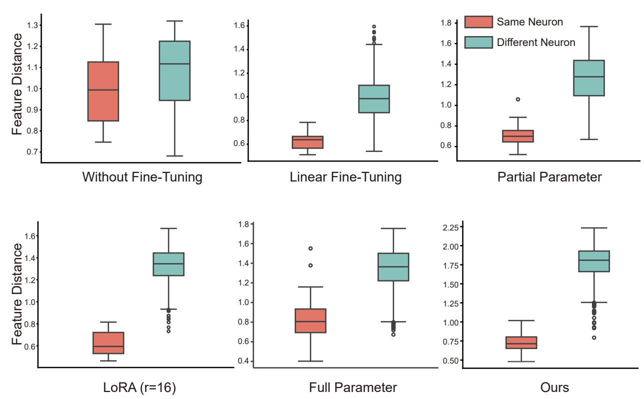

Figure 1. Comparison of feature distance distributions for same-neuron (red) and different-neuron (blue) pairs across various fine-tuning methods. Our method shows the best separation between the two distributions, indicating superior classification performance.

1.3. Comparison with Traditional Similarity Metrics

In addition to quantitative curves, we use similarity matrix heatmaps to visually compare the performance of our

*Corresponding author

method against traditional image similarity metrics. In these heatmaps, each row and column represents a neuron sample. Diagonal elements correspond to the similarity between samples from the same neuron, while off-diagonal elements represent similarity between samples from different neurons. A desirable heatmap should exhibit a strong, bright diagonal and dark off-diagonal regions.

As illustrated in Figure 2, the heatmap generated by our "Proposed Method" displays a sharp and distinct diagonal line, indicating high intra-neuron similarity and very low inter-neuron similarity. In contrast, methods like Mean Squared Error, Pearson Correlation Coefficient, and Cosine Similarity produce noisy heatmaps with many bright off-diagonal blocks, signifying a high rate of false positives. This visualization strongly validates that the deep metric features learned by our model are significantly more discriminative and robust for neuron identification than traditional similarity measures.

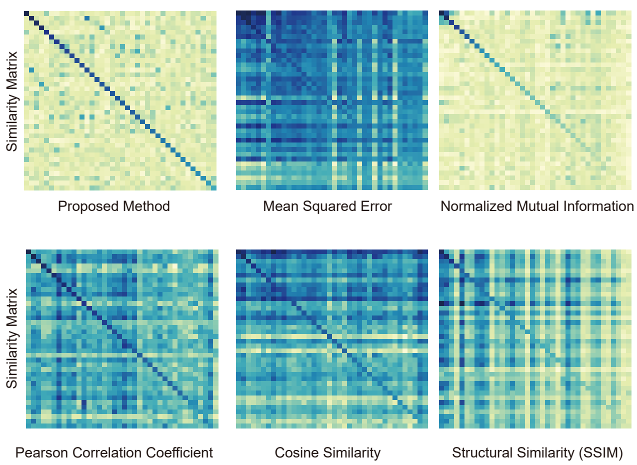

Figure 2. Similarity matrix heatmaps comparing our proposed method with traditional metrics. Our method yields a clean heatmap with a strong diagonal, demonstrating its superior ability to distinguish between different neurons.

1.4. Visualization of Latent Space

To further validate the effectiveness of our proposed method for cross-modality neuron feature alignment, we employed the t-distributed Stochastic Neighbor Embedding (t-SNE) [2] algorithm to perform non-linear dimensionality reduction on the high-dimensional features of neurons extracted from Two-Photon and fMOST imaging. By compressing the original 768-dimensional features into a 2D space, we can intuitively observe the spatial relationships between paired samples from different modalities in the learned embedding space.

As shown in Figure 3, neurons from Two-Photon imaging are represented by circles with a black outline, while those from fMOST imaging are represented by solid colored circles. A cyan line connects a "Matched Pair" from the two modalities. The visualization clearly shows that successfully matched pairs (e.g., those labeled 1.00, 1.03) are positioned in close proximity to each other in the 2D space. This visually confirms that our model has successfully learned modality-invariant features, mapping samples of the same neuron from different imaging techniques to nearby locations in the embedding space.

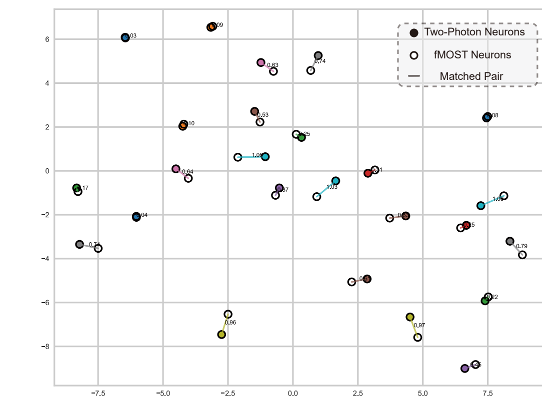

Figure 3. t-SNE visualization of the learned feature embeddings for cross-modality neurons. Two-Photon neurons are shown with black outlines, and fMOST neurons are solid. Cyan lines connect matched pairs, whose proximity in the 2D space demonstrates the effectiveness of our feature alignment method.

References

[1] Diederik P Kingma. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*, 2014.

[2] Laurens van der Maaten and Geoffrey Hinton. Visualizing data using t-sne. *Journal of machine learning research*, 9 (Nov):2579–2605, 2008.

[3] Kevin Musgrave, Serge Belongie, and Ser-Nam Lim. Pytorch metric learning. *arXiv preprint arXiv:2008.09164*, 2020.

[4] Aleksei Shabanov, Aleksei Tarasov, and Sergey Nikolenko. Stir: Siamese transformer for image retrieval postprocessing. In *Proceedings of the 33rd ACM International Conference on Information and Knowledge Management*, pages 4852–4857, 2024.