2024 46th Annual International Conference of the IEEE Engineering in Medicine and Biology Society (EMBC) | 979-8-3503-7149-9/24/$31.00 ©2024 IEEE | DOI: 10.1109/EMBC53108.2024.10782322

Multi-Scale Self-Supervised Consistency Training for Trustworthy Medical Imaging Classification

Bonian Han
Department of Statistics
Hangzhou Dianzi University
Hangzhou, China
bonian985@gmail.com

Cristian Moran, Jeong Yang, Young Lee, Zechun Cao, and Gongbo Liang*
Department of Computational, Engineering, and Mathematical Sciences
Texas A&M University-San Antonio
San Antonio, USA
cmora035@jaguar.tamu.edu, {jyang, ylee, zcao, gliang}@tamusa.edu

***Abstract*—Modern neural network models have demonstrated exceptional classification capabilities comparable to human performance in various medical diagnosis tasks. However, their practical application in real-world medical scenarios is hindered by an issue known as *miscalibration*, where these sophisticated tools inaccurately estimate their own prediction confidence, compromising their trustworthiness. To address this challenge, we propose a novel neural network calibration framework that utilizes multi-scale input images and integrates self-supervised consistency enforcement during training. Our experimental results demonstrate the significant enhancement of neural network calibration, concomitant with improvements in model classification performance. Furthermore, the proposed method exhibits the capacity to cultivate more robust feature spaces. Importantly, our approach is a general-purpose solution that is applicable to any imaging modalities. The proposed method can also be combined with other neural network calibration techniques to achieve further performance refinement. This research contributes a valuable tool for augmenting the reliability and trustworthiness of neural network models in diverse medical contexts.**

***Index Terms*—neural network, calibration, robustness, chest x-rays, histopathology images**

I. Introduction

Deep neural networks have been the major driving force for AI development in various domains recently, such as road safety [1], [2], astrophysics [3], [4], adversarial learning [5], [6], and medical imaging [7], [8]. Researchers are actively working on pursuing higher model performance in terms of accuracy [9]–[12]. However, uncertainty quantification is often ignored when evaluating neural network models [13].

In contrast to traditional neural networks, modern deep neural networks frequently exhibit challenges in accurately estimating the confidence of their predictions, which is known as *neural network miscalibration* [14]–[17]. This miscalibration issue is particularly troublesome, especially in healthcare settings, where it is crucial to involve human doctors when the confidence in disease diagnosis is low [18], [19].

This study introduces a general-purpose calibration framework that integrates neural network calibration into the training process by utilizing multi-scale images of the same input and enforces prediction consistency across various representations of the same image. Our method was assessed on two medical imaging datasets with different imaging modalities, demonstrating its capability to enhance model calibration by up to 35.83%, while simultaneously improving classification performance by up to 7.12%. Importantly, our proposed approach can easily be combined with existing calibration methods, such as temperature scaling [20], MMCE [16], or DCA [19], to achieve even more refined calibration results.

II. Background

This paper addresses the challenge of miscalibration in supervised classification tasks using contemporary deep neural networks. Beyond achieving high accuracy, classification networks should also convey their uncertainty and indicate when they are likely to make incorrect predictions. The confidence associated with a prediction, expressed as the probability of belonging to a specific class, should align with its likelihood of being correct [14], [15]. However, modern neural networks often exhibit overconfidence in their predictions [16], [17].

A. Problem Definition

Let $X \in x$ and $Y \in y = \{1, ..., k\}$ are random variables, representing the input and label of a neural network model, that follow a joint distribution $\pi(X,Y) = \pi(Y|X)\pi(X)$. Let h be a modern deep neural network with $h(X) = (\hat{Y}, \hat{P})$, where $\hat{Y}$ is the predicted class label and $\hat{P}$ is the associated probability or confidence. We would like the confidence estimate $\hat{P}$ to be calibrated such that $\hat{P}$ represents a true probability. For instance, given 50 predictions with the average confidence of 0.98, we expect that 49 predictions should be correct (i.e., 98% accuracy). In reality, the average confidence of a modern neural network is often higher than its accuracy [14]–[16]. The perfect calibration can be defined as:

$$\mathbb{P}\left(\hat{Y} = Y | \hat{P} = p\right) = p, \forall p \in [0,1]. \tag{1}$$

Difference in expectation between confidence and accuracy (i.e., the calibration error) can be defined as:

$$\mathbb{E}_{\hat{p}}\left[\left|\left(\hat{Y} = Y | \hat{P} = p\right) - p\right|\right]. \tag{2}$$

To improve model calibration, we want to reduce the calibration error (Equation 2) as much as possible.

This work is supported by the National Science Foundation under Grant No. 2334243.

Authorized licensed use limited to: Texas A & M University - San Antonio. Downloaded on December 22,2024 at 14:50:03 UTC from IEEE Xplore. Restrictions apply.

B. Existing Calibration Methods

Neural network calibration has been explored in multiple directions, including post-processing techniques, regularization methods, adjustments to the learning process, and the integration of data augmentation strategies.

Temperature scaling [14], [20] is a widely-used post-processing approach for model calibration. Once the model is trained, the temperature parameter T ($T > 0$) is added to the model and needs to be trained on the validation set while all the other parameters are frozen. After that, the temperature parameter will be used for calibration at the testing time. The calibrated confidence, $\hat{q}_i$, using temperature scaling is

$$\hat{q}_i = \max_k \theta_{SM}(\frac{z_i}{T})^{(k)}, \tag{3}$$

where k is the class label ($k = 1, ..., K$), $\theta_{SM}(z_i)$ is the predicted confidence. Though temperature scaling is easy to use and performs well in general, as a post-processing approach, it does not help with feature learning.

Regularization methods use additional regularization terms for calibration, such as Entropy [15], MMCE [16], or DCA [19]. The overall loss may be written as:

$$\text{TotalLoss} = \text{ClassificationLoss} + \beta\text{Regularization}. \tag{4}$$

A weight scalar, β, is needed to adjust the weight of the regularization term that would need to be carefully selected.

Label smoothing [21], [22] is another type of regularization method that focuses on the learning process. In a standard classification learning setup, the model is expected to predict a hard 1 to the correct class and 0 to all others. Label smoothing redistributes some of the probability mass from the correct class to other classes by assigning a value slightly less than 1 to the correct class and distributing the remainder among the other classes. In such a way, a small uncertainty is introduced to prevent the model from becoming overly confident and overfitting to the training data.

Mixup [23], [24] is a data augmentation technique that aims to predict a softer target. Mixup generates a new synthetic example for each training example by taking a weighted average of the pixel values of two randomly chosen examples from the training set. The same weighted average is also applied to the corresponding one-hot encoded labels.

The proposed method incorporates the last three types of strategies that introduce a novel learning process that incorporates the synergy of data augmentation and regularization.

III. Method

The proposed method includes four major components: a multi-scale input generation method, a shared feature extractor, a self-supervised consistency learning module, and the main predicting head with modality encoding. Figure 1 shows the overall architecture of the proposed method.

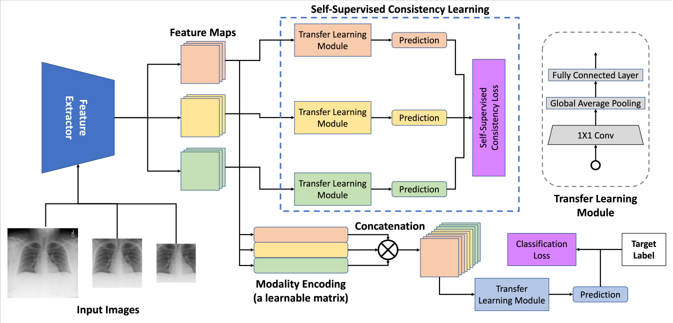


Fig. 1: The overall architecture of the proposed multi-scale self-supervised training framework.

A. Multi-Scale Input Generation

Given an input image, $k(k > 0)$ sub-images are generated using a cropping function $C_{\{\delta,\phi\}}(\cdot)$, with δ indicating the x-/y-coordinate of the center location of the cropped image and ϕ indicates the width and height of the cropped image. For instance, $I_{sub} = C_{\{\delta,\phi\}}(I)$, where $\delta = (100, 100)$ and $\phi = (30, 45)$ generate a sub-image, I_{sub} from I. The center location of I_{sub} aligns at the $(100, 100)$ location of I, and the width and height of I_{sub} is 30 and 45, respectively.

The output of this step is a multi-scale input set containing $k + 1$ images that are derived from the same input image. Each sub-image is the sub-view of the original image. Thus, the sub-image should hold the same image-level classification label. However, a sub-image might contain only part of the full information regarding the corresponding class label.

B. Feature Extraction

After multi-scale input generation, all the $k + 1$ views in the multi-scale input set are passed through a shared feature extractor one after another. The $k + 1$ feature maps are then used by the main predicting head (Section III-C) and the self-supervised consistency learning module (Section III-D).

C. Main Predicting Head with Modality Encoding

The main predicting head contains a modality encoding (ME) matrix and a transfer learning module for classification tasks. The ME matrix is a learnable $(k + 1) \times 8$ matrix that needs to be learned during the training process. The rows in ME provide information about the views in the multi-scale input set and are concatenated to the corresponding feature maps of the views. Then, the feature maps of all the views are concatenated together and used as the input to the transfer learning module for final prediction. The transfer learning module includes a 1×1 convolutional (Conv) layer, a global average pooling (GAP) layer, and a fully connected layer for prediction. Cross-entropy loss is used for evaluating the model classification performance during the training.

D. Self-Supervised Consistency Enforcing

The self-supervised consistency learning (SSCL) module comprises $k + 1$ auxiliary prediction heads. Each head processes the feature maps of a distinct view from the multi-scale

Authorized licensed use limited to: Texas A & M University - San Antonio. Downloaded on December 22,2024 at 14:50:03 UTC from IEEE Xplore. Restrictions apply.

input set and predicts the class label for the corresponding image. Given that all views within the same multi-scale input set originate from a single input image, all auxiliary prediction heads share the same target label. Notably, rather than assessing the accuracy of individual auxiliary head predictions, the primary objective of the SSCL module is to promote consistency in predictions across these auxiliary heads. The Kullback–Leibler (KL) divergence is applied to measure the consistency of the prediction of every two auxiliary heads.

IV. Experiment

We evaluate the performance of the proposed model using publicly available medical imaging datasets for two types of images—histological images and radiology images. The former one is for RGB images, and the latter one is for gray-scale images. Thirty neural network models of five methods for two types of classification tasks—binary classification and multi-class classification—are trained and compared over the two datasets. We denote the five methods as follows:

- the proposed method as MTSL-SCME;
- the model with only the multi-scale input as MTSL;
- the model with the multi-scale input and SSCL module as MTSL-SC;
- the model with the multi-scale input and modality encoding as MTSL-ME;
- and, the model without multi-scale input set, modality encoding, and self-supervised consistency learning as SINGLE since it only uses a single-scale image. The SINGLE model is implemented using ResNet-50 [25] by adding a 1×1 Conv layer before the GAP layer in ResNet-50 and used as the baseline model in this study.

We train each model on each dataset three times. The average performance with a standard deviation of the three training trials is reported in this section.

A. Datasets

The Kather 5000 dataset [26] contains 5000 histological images of 150×150 pixels (Figure 2a). Each image belongs to exactly one of eight tissue categories: tumor epithelium, simple stroma, complex stroma, immune cells, debris, normal mucosal glands, adipose tissue, and background (no tissue). All images are RGB, $0.495 \mu m$ per pixel, digitized with an Aperio ScanScope (Aperio/Leica biosystems), magnification $20\times$. Histological samples are fully anonymized images of formalin-fixed paraffin-embedded human colorectal adenocarcinomas (primary tumors) from the Institute of Pathology, University Medical Center Mannheim, Heidelberg University, Mannheim, Germany). The dataset was randomly partitioned into training and testing datasets with a $4:1$ ratio by us. The images were resized to 160×160.

The Mendeley V2 dataset [27] contains both the optical coherence tomography (OCT) images of the retina and pediatric chest X-ray images. We used the pediatric chest X-ray images (Figure 2b) in this study. The dataset includes 4273 pneumonia images and 1583 normal images. We used the provided training and testing sets in this study. The images were resized to 256×256.

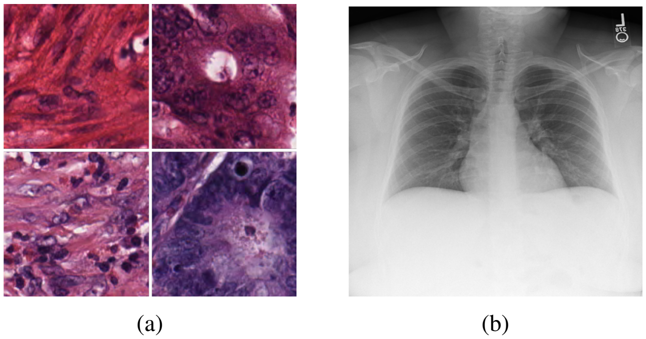

(a) (b)

Fig. 2: Examples from the Kather5000 (left) and Mendeley-V2 (right) datasets.

B. Experiment Setup

The experiments were conducted using an Nvidia A40 GPU card with 48GB of GPU memory. The project was implemented using the `PyTorch` library [28]. The `scikit-learn` library [29] was used for computing the classification evaluation metrics.

We set $k = 2$ to generate the multi-scale input sets. We used δ equal to the center of the image being cropped for the cropping function, $C_{\{\delta,\phi\}}(\cdot)$, and ϕ is 85% of the length and height of the image being cropped. The following procedure generates two crops I_{sub} and I_{subsub} from an input I: $I_{sub} = C_{\{\delta,\phi\}}(I)$ and $I_{subsub} = C_{\{\delta,\phi\}}(I_{sub})$.

The feature extractor of ResNet-50 [25] was used as the shared feature extractor. The batch size was set as 128. The Adam optimizer with a learning rate of $1e-4$ was used to optimize the model parameters.

C. Measurements

Expected Calibration Error (ECE) [30] is the main criteria that are used to measure neural network calibration error that approximates Equation (2) by partitioning predictions into M bins and taking a weighted average of the difference of accuracy and confidence for each bin. All the samples need to be grouped into M interval bins according to the prediction. Let B_m be the set of indices of samples whose prediction confidence falls into the interval $I_m = (\frac{m-1}{M}, \frac{m}{M}]$, $m \in M$. The accuracy of B_m is

$$\mathrm{acc}(B_m) = \frac{1}{|B_m|} \sum_{i \in B_m} 1(\hat{y}_i = y_i), \quad (5)$$

where $\hat{y}_i$ and y_i are the predicted and ground truth label for sample i. The average prediction confidence of bin B_m can be defined as

$$\mathrm{conf}(B_m) = \frac{1}{|B_m|} \sum_{i \in B_m} \hat{p}_i, \quad (6)$$

Authorized licensed use limited to: Texas A & M University - San Antonio. Downloaded on December 22,2024 at 14:50:03 UTC from IEEE Xplore. Restrictions apply.

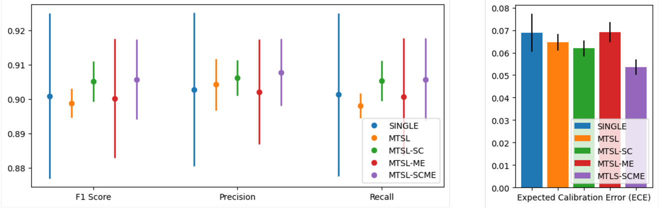


Fig. 3: Classification performance (left) and calibration performance (right) of ResNet-50 on the Kather5000 dataset.

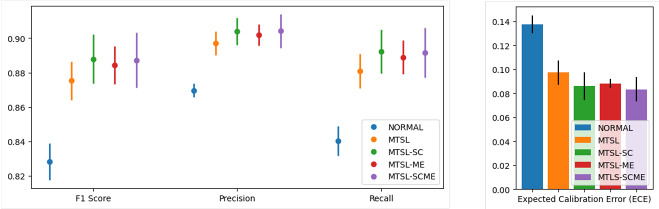
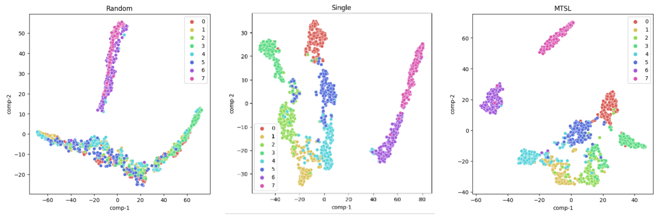


Fig. 4: Classification performance (left) and calibration performance (right) of ResNet-50 on the Mendeley-V2 dataset.



(a)

(b)

(c)

Fig. 5: Feature space visualization of random (left), single scale (middle), and multi-scale (right) models using t-SNE on the Kather5000 dataset.

where $\hat{p}_i$ is the confidence of sample i. ECE can be defined with acc(B_m) and conf(B_m)

$$\text{ECE} = \sum_{m=1}^{M} \frac{|B_m|}{n} \left| \text{acc}(B_m) - \text{conf}(B_m) \right|, \tag{7}$$

where n is the number of samples.

We use ECE to measure the calibration error, and we use F1 score, precision, and recall to measure the classification. The classification metrics are calculated using the weighted average of different classes for multi-class classification tasks. We implemented ECE by ourselves, and we used the `scikit-learn` library for computing F1 score, precision, and recall.

D. Results

1) Model Performance: Figure 3 shows the classification performance and calibration of different models on Kather 5000 with the error bars indicating the standard deviation. The figure reveals that the proposed method not only significantly reduces the calibration error but also improves model classification performance. For instance, the SINGLE model has 0.0688 ECE, and the MTSL-SCME is able to reduce the number by approximately 22% to 0.0536, while the MTSL-SCME also improves the F1 score from 0.9009 to 0.9057. It is also worth noting that the proposed method has a significantly shorter error bar, indicating our method is more stable across multiple training trials.

Figure 4 illustrates the performance of different models on the Mendeley V2 dataset. Similar to the previous dataset, the proposed method significantly reduces the calibration error and improves classification performance. The proposed MTSL-SCME method is able to reduce the calibration error from 0.1376 to 0.0833, approximately 39.46% reduction. The method is also able to improve classification performance by approximately 7.12%, from 0.8282 F1 score to 0.8872.

2) Feature Embedding: Compared to post-process calibration methods, such as temperature scaling, a trainable method is useful in the learning space, helping the model to learn a more representative feature space. Figure 5 uses t-SNE [31] to visualize the feature space learned by the proposed method. We first randomly selected 1024 samples from the Kather 5000 dataset and extracted features using three feature extractors. Then, we use t-SNE to project the feature maps of the samples into a 2D space. Each dot in the figure is one sample with color coding for the label information. Ideally, the samples from the same class should be densely close to each other (i.e., small intra-class distance), while samples from different classes should be further away (i.e., large inter-class distance).

Figure 5a demonstrates the feature embeddings from a random feature extractor with randomly initialized weights. The samples are embedded into two clusters. The samples from different classes are also mixed together. Figure 5b illustrates the feature embeddings from the SINGLE model. Though the samples of the eight classes are roughly separable, the samples from the same class are still widely spread in the feature space, and the clusters of several classes are close to each other. For instance, Class ID 6 (pink) and Class ID 7 (purple) are almost connected to each other. Figure 5c shows the embedding of our MTSL model, which embeds samples from the same class densely close to each other, while the clusters are much more easily separable than the SINGLE model's. The figure reveals that our feature extractor may have a more vital representative ability than the SINGLE model.

V. Discussion

Deep neural networks have been the major technique for building the next general computer-aided diagnosis tools. The society is excited about neural networks' near-human classification performance. Researchers are pushing the metrics higher and higher. However, to be able to use such an advantageous technique in real-world medical practice, the reliability of the neural networks is the key. Unfortunately, neural network models often incorrectly capture their own confidence, which is problematic in the medical field since it dramatically affects medical experts' decisions about how much we should trust the decision.

This work proposes a neural network calibration framework that jointly utilizes multi-scale input images and self-

Authorized licensed use limited to: Texas A & M University - San Antonio. Downloaded on December 22,2024 at 14:50:03 UTC from IEEE Xplore. Restrictions apply.

TABLE I: Detailed performance (±std) of different models on the Kather 5000 and Medeley V2 datasets.

Dataset	Model	ECE (↓)	F1 (↑)	Precision (↑)	Recall (↑)
Kather 5000	**SINGLE**	0.0688 ± 0.0085	0.9009 ± 0.0241	0.9028 ± 0.0223	0.9013 ± 0.0237
	MTSL	0.0647 ± 0.0037	0.8988 ± 0.0043	0.9042 ± 0.0057	0.8980 ± 0.0036
	MTSL-SC	0.0619 ± 0.0035	0.9051 ± 0.0058	0.9062 ± 0.0052	0.9053 ± 0.0059
	MTSL-ME	0.0691 ± 0.0046	0.9002 ± 0.0174	0.9021 ± 0.0153	0.9017 ± 0.0170
	MTSL-SCME	$\mathbf{0.0536 \pm 0.0035}$	$\mathbf{0.9057 \pm 0.0117}$	$\mathbf{0.9078 \pm 0.0098}$	$\mathbf{0.9057 \pm 0.0012}$
Mendeley V2	**SINGLE**	0.1376 ± 0.0073	0.8282 ± 0.0107	0.8697 ± 0.0041	0.8403 ± 0.0087
	MTSL	0.0973 ± 0.0102	0.8753 ± 0.0112	0.8969 ± 0.0069	0.8809 ± 0.0099
	MTSL-SC	0.0859 ± 0.0115	$\mathbf{0.8879 \pm 0.0143}$	0.9039 ± 0.0080	$\mathbf{0.8921 \pm 0.0127}$
	MTSL-ME	0.0883 ± 0.0036	0.8843 ± 0.0111	0.9019 ± 0.0061	0.8889 ± 0.0098
	MTSL-SCME	$\mathbf{0.0833 \pm 0.0100}$	0.8872 ± 0.0161	$\mathbf{0.9041 \pm 0.0097}$	0.8916 ± 0.0144

supervised consistency enforcement to calibrate the neural network models during the training. We tested the proposed method on two medical imaging datasets for binary classification and multi-class classification tasks. The datasets also include both RGB and gray-scale imaging modalities. The experimental result shows the proposed method is able to improve not only the model calibration but also classification performance.

One limitation of this work is we only focus on convolutional neural networks, specifically ResNet, in this paper. We chose ResNet for our experiment because it is the most widely used foundational model for imaging analysis in the medical field. For instance, PubMed has indexed 1775 papers with a "ResNet" keyword between 2022 and 2023. However, there are only 820 papers mentioned "Vision Transformer," 416 for "AlexNet," 529 for "DenseNet," and 369 for "EfficientNet."[1] However, given the general-purpose nature of the proposed method, I believe the method is also applicable to other popular foundational models.

In conclusion, this work proposes a general-purpose neural network calibration framework that may be used for any type of imaging modality with any foundational models for any classification tasks. Our evaluation result demonstrates its capability to enhance model calibration by up to 35.83% while simultaneously improving classification performance by up to 7.12%. It is important to note that the proposed method may be combined with other popular calibration techniques, such as temperature scaling, to achieve an even better calibration performance. The proposed method offers a promising solution for neural network calibration that assists the reliable integration of advanced neural network models into real-world medical practices across diverse imaging modalities and classification tasks.

References

[1] G. Liang, J. Zulu, X. Xing, and N. Jacobs, "Unveiling roadway hazards: Enhancing fatal crash risk estimation through multiscale satellite imagery and self-supervised cross-matching," *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing*, vol. 17, pp. 535–546, 2024.

[2] W. Song, T. Salem, H. Blanton, and N. Jacobs, "Remote estimation of free-flow speeds," in *2019 IEEE International Geoscience and Remote Sensing Symposium*, 2019, pp. 791–794.

[3] Y. Zhang, G. Liang, Y. Su, and N. Jacobs, "Multi-branch attention networks for classifying galaxy clusters," in *2020 25th International Conference on Pattern Recognition (ICPR)*. IEEE, 2021, pp. 9643–9649.

[4] S.-C. Lin, Y. Su, G. Liang, Y. Zhang, N. Jacobs, and Y. Zhang, "Estimating cluster masses from sdss multiband images with transfer learning," *Monthly Notices of the Royal Astronomical Society*, vol. 512, no. 3, pp. 3885–3894, 2022.

[5] Y. Zhang, G. Liang, T. Salem, and N. Jacobs, "Defense-pointnet: Protecting pointnet against adversarial attacks," in *2019 IEEE International Conference on Big Data (Big Data)*. IEEE, 2019, pp. 5654–5660.

[6] G. Liang, J. Guerrero, F. Zheng, and I. Alsmadi, "Enhancing neural text detector robustness with μ attacking and rr-training," *Electronics*, vol. 12, no. 8, p. 1948, 2023.

[7] L. Liu, J. Chang, G. Liang, and S. Xiong, "Simulated quantum mechanics-based joint learning network for stroke lesion segmentation and tici grading," *IEEE Journal of Biomedical and Health Informatics*, 2023.

[8] L. Liu, Y. Wang, J. Chang, P. Zhang, G. Liang, and H. Zhang, "Llrhnet: multiple lesions segmentation using local-long range features," *Frontiers in Neuroinformatics*, vol. 16, p. 859973, 2022.

[9] X. Xing, M. U. Rafique, G. Liang, H. Blanton, Y. Zhang, C. Wang, N. Jacobs, and A.-L. Lin, "Efficient training on alzheimer's disease diagnosis with learnable weighted pooling for 3d pet brain image classification," *Electronics*, vol. 12, no. 2, p. 467, 2023.

[10] Y. Zhang, G. Liang, and N. Jacobs, "Dynamic feature alignment for semi-supervised domain adaptation," *arXiv preprint arXiv:2110.09641*, 2021.

[11] G. Liang, C. Greenwell, Y. Zhang, X. Xing, X. Wang, R. Kavuluru, and N. Jacobs, "Contrastive cross-modal pre-training: A general strategy for small sample medical imaging," *IEEE Journal of Biomedical and Health Informatics*, vol. 26, no. 4, pp. 1640–1649, 2021.

[12] L. Liu, J. Chang, Y. Wang, G. Liang, Y.-P. Wang, and H. Zhang, "Decomposition-based correlation learning for multi-modal mri-based classification of neuropsychiatric disorders," *Frontiers in Neuroscience*, vol. 16, p. 832276, 2022.

[13] E. Xing, L. Liu, X. Xing, Y. Qu, N. Jacobs, and G. Liang, "Neural network decision-making criteria consistency analysis via inputs sensitivity," in *2022 26th International Conference on Pattern Recognition (ICPR)*. IEEE, 2022, pp. 2328–2334.

[14] C. Guo, G. Pleiss, Y. Sun, and K. Q. Weinberger, "On calibration of modern neural networks," in *International conference on machine learning*. PMLR, 2017, pp. 1321–1330.

[15] G. Pereyra, G. Tucker, J. Chorowski, Ł. Kaiser, and G. Hinton, "Regularizing neural networks by penalizing confident output distributions," *arXiv:1701.06548*, 2017.

[16] A. Kumar, S. Sarawagi, and U. Jain, "Trainable calibration measures for neural networks from kernel mean embeddings," in *Prco. ICML*, 2018, pp. 2810–2819.

[17] T. Popordanoska, R. Sayer, and M. Blaschko, "A consistent and differ-

[1] https://pubmed.ncbi.nlm.nih.gov accessed on Dec 24, 2023.

Authorized licensed use limited to: Texas A & M University - San Antonio. Downloaded on December 22,2024 at 14:50:03 UTC from IEEE Xplore. Restrictions apply.

entiable lp canonical calibration error estimator," *Advances in Neural Information Processing Systems*, vol. 35, pp. 7933–7946, 2022.

[18] X. Jiang, M. Osl, J. Kim, and L. Ohno-Machado, "Calibrating predictive model estimates to support personalized medicine," *Journal of the American Medical Informatics Association*, vol. 19, no. 2, pp. 263–274, 2012.

[19] G. Liang, Y. Zhang, X. Wang, and N. Jacobs, "Improved trainable calibration method for neural networks on medical imaging classification," in *British Machine Vision Conference (BMVC)*, 2020.

[20] G. Hinton, O. Vinyals, and J. Dean, "Distilling the knowledge in a neural network," *arXiv:1503.02531*, 2015.

[21] C. Szegedy, V. Vanhoucke, S. Ioffe, J. Shlens, and Z. Wojna, "Rethinking the inception architecture for computer vision," in *Proc. CVPR*, June 2016.

[22] R. Müller, S. Kornblith, and G. E. Hinton, "When does label smoothing help?" in *Proc. NeurIPS*, 2019, pp. 4694–4703.

[23] H. Zhang, M. Cissé, Y. N. Dauphin, and D. Lopez-Paz, "mixup: Beyond empirical risk minimization," in *Proc. ICLR*, 2018.

[24] S. Thulasidasan, G. Chennupati, J. A. Bilmes, T. Bhattacharya, and S. Michalak, "On mixup training: Improved calibration and predictive uncertainty for deep neural networks," in *Proc. NeurIPS*, 2019, pp. 13 888–13 899.

[25] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2016, pp. 770–778.

[26] J. N. Kather *et al.*, "Multi-class texture analysis in colorectal cancer histology," *Scientific reports*, vol. 6, p. 27988, 2016.

[27] D. Kermany and M. Goldbaum, "Labeled optical coherence tomography (oct) and chest x-ray images for classification," *Mendeley Data*, vol. 2, 2018.

[28] A. Paszke, S. Gross, F. Massa, A. Lerer, J. Bradbury, G. Chanan *et al.*, "Pytorch: An imperative style, high-performance deep learning library," in *Advances in Neural Information Processing Systems 32*. Curran Associates, Inc., 2019, pp. 8024–8035.

[29] L. Buitinck, G. Louppe, M. Blondel, F. Pedregosa, A. Mueller, O. Grisel, V. Niculae *et al.*, "API design for machine learning software: experiences from the scikit-learn project," in *ECML PKDD Workshop: Languages for Data Mining and Machine Learning*, 2013, pp. 108–122.

[30] M. P. Naeini, G. Cooper, and M. Hauskrecht, "Obtaining well calibrated probabilities using bayesian binning," in *Proc. AAAI*, 2015.

[31] L. Van der Maaten and G. Hinton, "Visualizing data using t-sne." *Journal of machine learning research*, vol. 9, no. 11, 2008.

Authorized licensed use limited to: Texas A & M University - San Antonio. Downloaded on December 22,2024 at 14:50:03 UTC from IEEE Xplore. Restrictions apply.