Calibrated Transfer Learning for Multi-Class Medical Image Classification

Yu Hlaing Win and JingTao Yao
Department of Computer Science
University of Regina, Regina, Canada
{ywn784, jingtao.yao}@uregina.ca

***Abstract*—While deep transfer learning models for medical image processing often achieve high accuracy, their lack of reliable calibration poses a potential safety risk in clinical applications. Current calibration techniques rely on manual post-hoc processing steps that are challenging to tune for reliable confidence estimations. This study proposes ACR-LLS-MNetCal, which adds Adaptive Confidence Regularizer and Learnable Logit Scaler to the classification head of MobileNetV2 for better calibration performance. The proposed model achieves 94.68% accuracy with excellent calibration (ECE 0.0016-0.0083), representing a calibration improvement of up to 61% compared to baseline model. This study contributes to building a foundation for trustworthy AI models in clinical practice. In addition, the proposed calibration method automatically learns optimal calibration during training that reduces the need for manual post-hoc adjustments.**

***Index Terms*—Transfer Learning, Model Calibration, MobileNetV2, Logit Scaling, Adaptive Confidence Regularization, Medical Imaging**

I. Introduction

Deep learning has improved medical image analysis by learning useful patterns from high-dimensional images [1]. However, their effectiveness is often limited by the availability of annotated medical datasets [2]. Transfer learning addresses this challenge by adapting pretrained models to target medical tasks, offering stable feature representations and efficient training [3]. In this setting, pretrained CNN backbones act as feature extractors and provide different feature-extraction strengths [4]. These backbones can be adapted to the target domain through staged or multi-step training and fine-tuning [5], [6], thereby helping the learned representations align more closely with medical image data [7], [8].

Although these adaptations help improve classification performance, they do not address how reliable the model's confidence is. Many transfer-learning studies focus mainly on accuracy, whereas confidence reliability is less explored, especially when models encounter data from different hospitals or imaging conditions [7], [9]. This makes confidence reliability important in medical imaging. Calibration is commonly used to improve confidence reliability, but it is often applied as a separate post-processing step rather than being integrated into the training or transfer-learning procedure.

Post-hoc calibration methods such as temperature scaling [10] and adaptive temperature scaling [11] adjust output probabilities after the model is trained. They are effective, yet they add an extra calibration stage and do not influence how features are learned, which is important in transfer learning. Mixup-based training can offer some implicit calibration benefits [12]; however, it is not designed to improve reliability directly. Training-time calibration methods, including confidence-aware losses [13] and focal-loss-based strategies [14], influence the learning process and have shown improvements in reliability. These approaches, however, have been explored only to a limited extent in transfer-learning-based medical imaging, and they are rarely combined into a single approach that supports both accuracy and confidence reliability. Integrating calibration into transfer learning may enable more consistent confidence shaping during task-specific adaptation, without relying on an additional post-hoc stage.

In this study, we aim to improve the reliability of predicted-class confidence in transfer-learning-based medical image classification by adjusting confidence during training to more accurately reflect the true probability of correctness. To achieve this, we propose ACR-LLS-MNetCal, a framework that integrates an Adaptive Confidence Regularizer (ACR) and a Learnable Logit Scaler (LLS) into a MobileNetV2-based transfer learning model. ACR penalizes both overconfident and underconfident predictions, and LLS learns class-specific scaling to refine confidence outputs. Both components operate during training rather than as a separate post-processing step. This framework is designed to improve confidence reliability while maintaining strong classification accuracy.

II. Proposed Method: ACR-LLS-MNetCal

A. Architecture of ACR-LLS-MNetCal

The ACR-LLS-MNetCal model uses MobileNetV2 [15] as the backbone architecture, selected for its computational efficiency and proven performance in medical image processing. The input images are resized to 224×224×3 to satisfy the ImageNet pretraining standard. Additionally, we exclude the base model's top classification layer to adapt the model for four-class classification task, and add a custom classification head specific to the target task as shown in Fig. 1.

The custom classification head includes a Global Average Pooling layer, two fully connected layers (256 and 128 neurons with ReLU activation), each followed by dropout regularization (0.4 and 0.3 rates, respectively), and a final dense layer generating logits for four classes: COVID, Lung Opacity, Normal, and Viral Pneumonia.

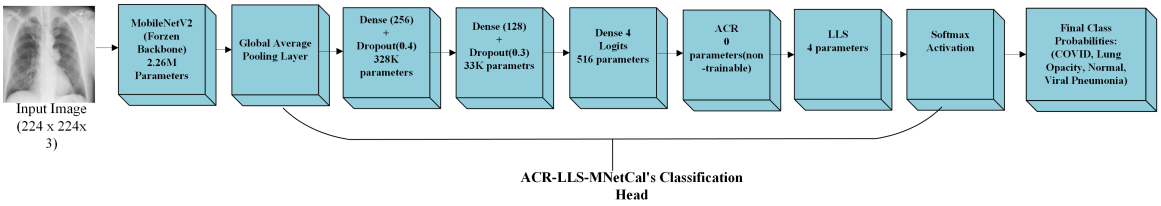

Fig. 1: Architecture of ACR-LLS-MNetCal

We integrate the proposed calibration components sequentially: ACR processes the raw logits for confidence regularization, followed by LLS for class-specific scaling, and then final softmax activation. This design enables efficient learning of chest X-ray features while providing reliable uncertainty quantification for clinical decision-making.

B. Adaptive Confidence Regularizer (ACR)

We implement ACR as a custom regularization layer in Keras, a high-level deep learning library built on TensorFlow. The ACR layer penalizes the difference between the highest and second-highest predicted probabilities. It operates on raw input logits during training without altering the model's forward output. Internally, the layer first computes the softmax probabilities from the input logits, then identifies the two highest predicted class probabilities, p_1 and p_2. Their difference defines the confidence margin, calculated as Margin = p_1 - p_2.

This margin-based method is inspired by margin-sensitive techniques, such as focal loss [14]. The focal loss modulates the standard cross-entropy loss using a confidence-based weighting term. While entropy-based calibration approach [10] evaluates uncertainty across the full probability distribution, the proposed margin-based ACR focuses on the decision boundary between the most likely classes. This margin is used to compute two penalty terms depending on whether it falls below or above a predefined threshold.

The uncertainty penalty, $\mathcal{L}_{\text{u}}$ shown in Eq. 1 is activated when the margin falls below a specified threshold τ. This penalty encourages the model to assign higher confidence when the input predominantly favours one class over the others. The overconfidence penalty, $\mathcal{L}_{\text{o}}$ in Eq. 2 is imposed when the margin exceeds the threshold, thereby penalizing overly confident predictions. This approach is particularly efficient in medical image classification, where small margins may indicate ambiguous diagnostic results.

$$\mathcal{L}_{\text{u}} = \max(0, \tau - \text{margin}) \tag{1}$$

$$\mathcal{L}_{\text{o}} = \max(0, \text{margin} - \tau) \tag{2}$$

Given that both low and overly high confidence margins can lead to miscalibration, we combine the two penalty components into a single loss function. The ACR loss, $\mathcal{L}_{\text{ACR}}$ in Eq. 3 is added to the weighted cross-entropy loss and contributes to the gradient updates. It uses a regularization strength α to scale the overall penalty, penalizing uncertainty more strongly than overconfidence because uncertain predictions are more harmful and cross-entropy already discourages overconfident errors. For this reason, the overconfidence term is downweighted by 0.5.

$$\mathcal{L}_{\text{ACR}} = \alpha \cdot \text{mean}(\mathcal{L}_{\text{u}} + 0.5 \cdot \mathcal{L}_{\text{o}}) \tag{3}$$

We used $\alpha = 0.08$ and $\tau = 0.25$ to optimize calibration and classification accuracy. These parameters were established through hyperparameter tuning, as outlined in Section III-B, and the layer is only active during training to avoid any additional overhead during prediction time.

C. Learnable Logit Scaler (LLS)

We introduce LLS as a custom Keras layer that applies trainable, class-specific scaling factors to each logit. This LLS layer addresses the class-specific miscalibration issue caused by imbalanced training data. In such scenarios, minority classes tend to be poorly calibrated compared to majority classes. In practice, each scaling weight is initialized at 1.0 and adjusted using backpropagation to enhance class-wise calibration. The LLS layer computes the scaled logits as:

$$\tilde{z} = z \odot s \tag{4}$$

where

- z input logit vector;
- s vector of class-specific scaling weights;
- $\odot$ element-wise multiplication.

The resulting scaled logits, $\tilde{z}$, are passed through the softmax activation function [16] to generate the final class prediction probabilities. This allows the class-specific scaling to directly influence the final probability distribution, enabling the model to automatically adjust class-specific confidence throughout the training without requiring manual tuning.

D. Training Procedure of ACR-LLS-MNetCal

The ACR-LLS-MNetCal employs a two-phase training methodology. It begins with feature extraction by training the model's custom classification head with frozen backbone weights, followed by a comprehensive fine-tuning of the model. Alogrithm 1 presents a complete training procedure of ACR-LLS-MNetCal. The notations used in Algorithm 1 are explained below:

D_{train}, D_{val}: training, validation datasets; w: class weights; M, M^*: base model and fine-tuned model; θ_{head}: classification head parameters; $\theta_{backbone}$: backbone parameters; s_{LLS}: LLS scaling factors; L_{CE}: cross-entropy loss; x: input images;

y: class labels; $\hat{p}$: predicted probabilities; $reduced\_lr$: reduced learning rate; i: class index (runs from 1 to C);

The CONSTRUCTBASEARCHITECTURE() creates a MobileNetV2-based model with a custom classification head. The INTEGRATECALIBRATIONLAYERS(M, α, τ) adds the ACR and LLS layers to the model's classification head using α and τ.

Algorithm 1: Two-Phase Training Algorithm

Input: $D_{\text{train}}, D_{\text{val}}$, α, τ, w
Output: Calibrated model M^*
$M \leftarrow$ CONSTRUCTBASEARCHITECTURE();
$M \leftarrow$ INTEGRATECALIBRATIONLAYERS(M, α, τ);
Pipeline: Logits $\rightarrow$ ACR(α, τ) $\rightarrow$ LLS(s) $\rightarrow$ Softmax;
Phase 1: Feature Extraction;
FREEZEBACKBONE(M);
for *epoch = 1* **to** *30* **do**
  foreach *batch* $(x, y) \in D_{\text{train}}$ **do**
    $z \leftarrow$ FORWARD(M, x);
    $L_{\text{ACR}} \leftarrow \text{ACR}(z, \alpha, \tau)$;
    $\tilde{z} \leftarrow z \odot s$;
    $\hat{p} \leftarrow \text{Softmax}(\tilde{z})$;
    $L_{\text{CE}} \leftarrow -\sum_i w_c(i)\, y_i \log \hat{p}_i$ [16];
    $L_{\text{total}} \leftarrow L_{\text{CE}} + L_{\text{ACR}}$;
    UPDATE$(\theta_{\text{head}}, s_{\text{LLS}})$;
  VALIDATEANDSAVE(M, D_{val});
Phase 2: Fine-tuning;
$M^* \leftarrow$ LOADBESTMODELFROMPHASE1(α, τ);
UNFREEZETOPLAYERS$(M^*, 50)$;
RECOMPILEMODEL$(M^*,$ `reduced_lr`$)$;
for *epoch = 1* **to** *50* **do**
  foreach *batch* $(x, y) \in D_{\text{train}}$ **do**
    $z \leftarrow$ FORWARD(M^*, x);
    $L_{\text{ACR}} \leftarrow \text{ACR}(z, \alpha, \tau)$;
    $\tilde{z} \leftarrow z \odot s$;
    $\hat{p} \leftarrow \text{Softmax}(\tilde{z})$;
    $L_{\text{CE}} \leftarrow -\sum_i w_c(i)\, y_i \log \hat{p}_i$;
    $L_{\text{total}} \leftarrow L_{\text{CE}} + L_{\text{ACR}}$;
    UPDATE$(\theta_{\text{backbone}}, \theta_{\text{head}}, s_{\text{LLS}})$;
  VALIDATEANDSAVE(M^*, D_{val});
return SELECTBESTMODEL(M^*);

Phase 1 (Feature Extraction) freezes the pre-trained backbone for 30 epochs with a learning rate of 0.001. This setup enables the model to retain ImageNet features while allowing the classification head to learn chest X-ray-specific patterns. The training pipeline then sequentially processes the logits through the ACR regularizer and the LLS scaling layer before applying the softmax activation. The total loss updates θ_{head} and s_{LLS}, where L_{CE} ensures accurate classification and L_{ACR} promotes effective confidence calibration.

Phase 2 (Fine-tuning) loads the best model from Phase 1 (M^*) and unfreezes the top 50 layers with a reduced learning rate 0.0001 for 50 epochs. The last 50 layers correspond to deeper inverted-residual blocks that learn high-level disease-related patterns and require domain adaptation. Earlier layers capture low-level features that transfer well between domains and are kept frozen to reduce overfitting.

The same pipeline continues processing as in Phase 1; however, in this Phase 2, all trainable parameters θ_{backbone}, θ_{head}, and s_{LLS} are updated jointly via backpropagation. This joint optimization aligns feature representations, classification outputs, and logit scaling within a single training process, resulting in both accurate predictions and well-calibrated confidence estimates.

III. EXPERIMENTAL SETUP

A. Dataset and Preprocessing

We use the COVID-19 Radiography Dataset [17], which includes 21,165 chest X-ray images across four classes: COVID (3,616; 17.1%), Lung Opacity (6,012; 28.4%), Normal (10,192; 48.1%), and Viral Pneumonia (1,345; 6.4%). The dataset is proportionally split into 70% training (14,814 images), 15% validation (3,172 images), and 15% testing (3,179 images) to preserve the class distribution. All images are resized from 299×299 to 224×224 and normalized to [0, 1]. To address class imbalance, we apply data augmentation (rotation ±15°, horizontal flip p=0.5, shift ±10%, shear ±10%, zoom ±15%) and inverse-frequency class weights [18]: 1.46 (COVID), 0.88 (Lung Opacity), 0.52 (Normal), and 3.94 (Viral Pneumonia).

B. ACR Parameter Tuning

We performed a systematic parameter tuning using the `ParameterGrid` function from Scikit-Learn to determine the optimal ACR regularization parameters: α and τ. The parameter tuning involved 16 combinations, formed by pairing four α values $[0.08, 0.1, 0.15, 0.2]$ and four τ values $[0.15, 0.2, 0.25, 0.3]$. Each configuration was trained for 10 epochs with early stopping (patience = 3) to efficiently identify the best parameter combination, with validation loss used as the primary selection criterion.

TABLE I: Top 5 ACR Parameter Configurations

α	τ	Val Accuracy (%)	Val Loss
0.08	**0.25**	**87.04**	**0.3647**
0.15	0.30	87.30	0.3721
0.10	0.15	87.07	0.3744
0.10	0.30	86.60	0.3746
0.10	0.20	86.46	0.3759

TABLE I presents the top five parameter combinations that achieved consistent validation-loss values between 0.3647 and 0.3759, indicating stable parameter behavior. The optimal configuration, α = 0.08 and τ = 0.25, achieved the lowest validation loss of 0.3647, representing the best balance between calibration and classification performance.

Across all 16 combinations, a region with lower validation-loss values appears when α is between 0.08 and 0.15 and τ is between 0.20 and 0.30. Parameter settings beyond these ranges, particularly α = 0.20, resulted in noticeably higher losses, suggesting that overly strong regularization reduces model flexibility. These results highlight the importance of selecting balanced regularization strengths.

IV. RESULTS AND DISCUSSION

A. LLS Behavior

The LLS component applies trainable, class-specific scaling factors to each logit, allowing the model to adjust its confidence for different classes. Values below 1.0 indicate reduced

confidence (down-scaling), while values above 1.0 indicate amplified confidence.

TABLE II: Evolution of LLS Scaling Factors

Class	Phase 1	Phase 2	Interpretation
COVID	0.8872	0.8759	Moderate logit scaling
Lung Opacity	0.8118	0.7993	Low logit scaling
Normal	1.0365	1.0399	Amplified logit scaling
Viral Pneumonia	1.0040	0.9761	Reduced logit scaling

TABLE II shows that all classes undergo meaningful adjustments from Phase 1 to Phase 2. COVID and Viral Pneumonia show modest reductions, indicating that the model tempers mild overconfidence while keeping the scaling below 1.0. Lung Opacity has the lowest value (0.7993), demonstrating stronger down-scaling for this challenging class. In contrast, Normal maintains values above 1.0, showing consistently amplified confidence. Overall, these adjustments highlight how LLS adapts class-wise confidence automatically without manual tuning.

B. Classification Result and Discriminative Performance

We evaluated the ACR-LLS-MNetCal model on the independent test set (3,179 images). The model achieved 87.57% accuracy in Phase 1 and 94.68% in Phase 2, a 7.11% improvement that demonstrates the effectiveness of the two-phase training strategy for this chest X-ray task.

TABLE III: Classification Report

Class	Precision	Recall	F1-score	Support
COVID	0.9636	0.9742	0.9689	543 (17.1%)
Lung Opacity	0.9571	0.8893	0.9219	903 (28.4%)
Normal	0.9326	0.9680	0.9500	1530 (48.1%)
Viral Pneumonia	0.9704	0.9704	0.9704	203 (6.4%)
Accuracy			0.9468	
Macro Avg	0.9559	0.9505	0.9528	
Weighted Avg	0.9473	0.9468	0.9465	

TABLE III presents the model's classification performance, with an overall test accuracy of 94.68%. The COVID class achieved the highest recall (97.42%) with only 14 misclassifications, resulting in fewer false negatives. Lung Opacity was the most challenging class and had the lowest recall (88.93%) with 100 misclassifications, mostly against the Normal class. Its precision of 95.71% shows that positive predictions remain reliable. Viral Pneumonia achieved balanced precision and recall (97.04% each), resulting in the highest F1-score, and none of these cases were misclassified as COVID. The Normal class exhibited a 96.8% recall, which helps reduce unnecessary clinical treatments. Overall, the model handled class imbalance effectively and achieved AUC scores above 0.98 for all classes (COVID: 0.9987, Lung Opacity: 0.9887, Normal: 0.9882, Viral Pneumonia: 0.9996), indicating strong generalization and reliable performance.

C. Model Calibration Performance

Model reliability describes how well the predicted confidence matches the true correctness. We evaluate this using Brier scores and ECE. Lower Brier scores indicate better probabilistic accuracy, and ECE values below 0.05 are considered well calibrated. Values below 0.01 reflect excellent calibration.

The model achieves Brier scores of COVID (0.0085), Lung Opacity (0.0343), Normal (0.0386), and Viral Pneumonia (0.0031), indicating accurate and well-behaved probability estimates across all classes. The ECE values for COVID (0.0024), Lung Opacity (0.0073), Normal (0.0083), and Viral Pneumonia (0.0016), shown in Table V, fall within the range associated with excellent calibration.

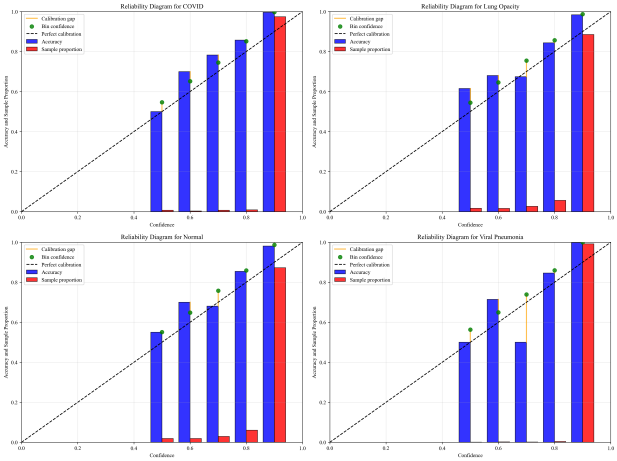

Fig. 2: Reliablility

The reliability diagram in Fig. 2 shows that bin-wise confidence values generally follow the diagonal line of perfect calibration across all four classes. COVID and Viral Pneumonia display the strongest alignment, with only minor deviations. Normal and Lung Opacity show slightly larger mid-confidence deviations, indicating mild miscalibration, but remain within acceptable reliability levels. Overall, the diagram indicates that the model's confidence corresponds well to prediction correctness for chest X-ray classification.

D. Baseline Comparison and Ablation Study

To analyze the impact of ACR and LLS, we compared three configurations: the baseline without ACR/LLS, a single-dense version that removes the Dense(128) and Dropout(0.3) layers, and the proposed ACR-LLS-MNetCal model. This design isolates the calibration components by modifying only the calibration layers or head capacity while keeping the backbone fixed.

TABLE IV: Model Performance Comparison

Model	Accuracy	F1-Score
Baseline	94.02%	0.9461
Single Dense	94.02%	0.9450
Proposed	**94.68%**	**0.9528**

As shown in TABLE IV, the proposed model improves accuracy from 94.02% to 94.68% (+0.66%), while the single-dense model remains similar to the baseline. Although the accuracy gain is moderate, the main improvement lies in calibration. TABLE V shows that the proposed model achieves

the lowest ECE across all classes, reducing ECE by 50% (COVID), 61.2% (Lung Opacity), 61.8% (Normal), and 57.9% (Viral Pneumonia) compared to the baseline. The single-dense model yields higher ECE across all classes, suggesting that reduced head capacity alone is insufficient for improving calibration. These results confirm that ACR and LLS contribute directly to the calibration gains while maintaining strong overall performance.

TABLE V: Model Calibration Comparison (ECE Values)

Model	COVID	Lung Opacity	Normal	Viral Pneumonia
Baseline	0.0048	0.0188	0.0217	0.0038
Single Dense	0.0096	0.0189	0.0248	0.0033
Proposed	**0.0024**	**0.0073**	**0.0083**	**0.0016**

E. Interpretability via Grad-CAM

Grad-CAM visualizations in Fig. 3 show where the model focuses for each class.

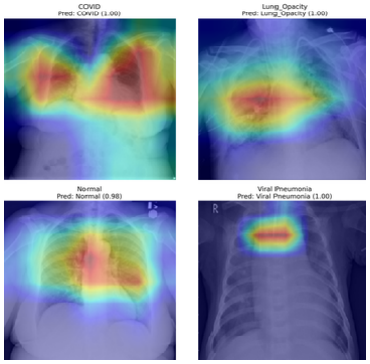

Fig. 3: Grad-CAM visualizations

High-activation regions (red/yellow) indicate strong attention, while low-activation regions (blue) show weaker attention. Across all four classes, the ACR-LLS-MNetCal model highlights meaningful lung areas: diffuse activation for COVID, a localized region for Lung Opacity, a moderate pattern for Normal, and a small distinct region for Viral Pneumonia. These visualizations show that the model focuses on relevant lung features rather than background artifacts.

V. Conclusion

We propose the ACR-LLS-MNetCal to address the challenge of achieving reliable confidence in medical image classification while maintaining strong accuracy. ACR regulates prediction confidence during training, while LLS learns class-specific scaling factors, removing the need for post-hoc calibration or manual threshold tuning. The proposed model achieves 94.68% accuracy and consistently low ECE values (0.0016–0.0083), reflecting a close alignment between predicted confidence and actual correctness. The evaluation on a diverse and imbalanced chest X-ray dataset further shows that the model remains well-calibrated across all classes, including the smallest class, Viral Pneumonia.

This study demonstrates that integrating calibration-aware components into the training process can deliver reliable confidence estimates and competitive accuracy. The proposed ACR-LLS-MNetCal offers a promising and distinct approach to reliability-focused medical AI and provides a solid foundation for future extensions to larger datasets and real-world clinical settings.

References

[1] H. Zhang and Y. Qie. "Applying deep learning to medical imaging: A Review," *Applied Sciences*, vol. 13, no. 18, Art. no. 10521, 2023.

[2] G. Tummalapalli, O. Gurrapu, K. N. Kumar, J. V. Suman, A. V. Rao, and M. Prabhu. "Deep learning approaches for enhancing image classification accuracy in medical imaging," in *2025 Devices for Integrated Circuit*, pp. 16–21, 2025.

[3] F. Zhuang, Z. Qi, K. Duan, D. Xi, Y. Zhu, H. Zhu, H. Xiong, and Q. He. "A comprehensive survey on transfer learning," *Proceedings of the IEEE*, vol. 109, no. 1, pp. 43–76, 2021.

[4] M. Rahimzadeh, A. Attar, and S. M. Sakhaei. "A fully automated deep learning-based network for detecting COVID-19 from a new and large lung CT Scan dataset," *Biomedical Signal Processing and Control*, vol. 68, Art. no. 102588, 2021.

[5] G. Ayana, K. Dese, A. Abagaro, K. C. Jeong, S.-D. Yoon, and S.-W. Choe. "Multistage transfer learning for medical images," *Artificial Intelligence Review*, vol. 57, Art. no. 232, 2024.

[6] R. Zhang, Z. Guo, Y. Sun, Q. Lu, Z. Xu, Z. Yao, M. Duan, S. Liu, Y. Ren, L. Huang, and F. Zhou. "COVID19XrayNet: A two-step transfer learning model for the COVID-19 detecting problem based on a limited number of chest X-Ray images," *Interdisciplinary Sciences: Computational Life Sciences*, vol. 12, no. 4, pp. 555–565, 2020.

[7] H. E. Kim, A. Cosa-Linan, N. Santhanam, M. Jannesari, M. E. Maros, and T. Ganslandt. "Transfer learning for medical image classification: a literature review," *BMC Medical Imaging*, vol. 22, no. 1, Art. no. 69, 2022.

[8] S. Atasever, N. Azginoglu, D. S. Terzi, and R. Terzi. "A comprehensive survey of deep learning research on medical image analysis with focus on transfer learning," *Clinical Imaging*, vol. 94, pp. 18–41, 2023.

[9] R. Wang, P. Chaudhari, and C. Davatzikos. "Embracing the disharmony in medical imaging: A Simple and effective framework for domain adaptation," *Medical Image Analysis*, vol. 76, Art. no. 102309, 2022.

[10] C. Guo, G. Pleiss, Y. Sun, and K. Q. Weinberger. "On calibration of modern neural networks," in *Proceedings of the 34th International Conference on Machine Learning*, vol. 70, pp. 1321–1330, 2017.

[11] S. A. Balanya, J. Maroñas, and D. Ramos. "Adaptive temperature scaling for robust calibration of deep neural networks," *Neural Computing and Applications*, vol. 36, pp. 8073–8095, 2024.

[12] 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 *Advances in Neural Information Processing Systems*, vol. 32, 1246, 2019.

[13] J. Moon, J. Kim, Y. Shin, and S. Hwang. "Confidence-aware learning for deep neural networks," in *Proceedings of the 37th International Conference on Machine Learning*, vol. 119, pp. 7034–7044, 2020.

[14] J. Mukhoti, V. Kulharia, A. Sanyal, S. Golodetz, P. H. S. Torr, and P. K. Dokania. "Calibrating deep neural networks using focal loss," in *Proceedings of the 34th International Conference on Neural Information Processing Systems*, vol. 33, pp. 15288–15299, 2020.

[15] M. Sandler, A. Howard, M. Zhu, A. Zhmoginov, and L.-C. Chen. "MobileNetV2: Inverted residuals and linear bottlenecks," in *2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 4510–4520, 2018.

[16] I. Goodfellow, Y. Bengio, and A. Courville. *Deep Learning*. Cambridge, MA, USA: MIT Press, 2016.

[17] M. E. H. Chowdhury, T. Rahman, A. Khandakar, R. Mazhar, M. A. Kadir, Z. B. Mahbub, K. R. Islam, M. S. Khan, A. Iqbal, N. Al-Emadi, M. B. I. Reaz, and M. T. Islam. "Can AI help in screening viral and COVID-19 pneumonia?" *IEEE Access*, vol. 8, pp. 132665–132676, 2020.

[18] H. He and E. A. Garcia. "Learning from imbalanced data." *IEEE Transactions on Knowledge and Data Engineering*, vol. 21, no. 9, pp. 1263–1284, 2009.