Precise Lens Status Classification via Projection Tuning for Efficient Adaptation to Data Shifts in Small Cataract Image Datasets

Ji Young Byun
Johns Hopkins University
Baltimore, MD 21218
jbyun13@jhu.edu

Jordan Shuff
Johns Hopkins University
School of Medicine
Baltimore, MD 21231
ksp@jhu.edu

Rengaraj Venkatesh
Aravind Eye Hospital
Pondicherry, India
venkatesh@aravind.org

Nakul S. Shekhawat†
Johns Hopkins University
School of Medicine
Baltimore, Maryland 21287
nshekha1@jhmi.edu

Kunal S. Parikh†
Johns Hopkins University
School of Medicine
Baltimore, MD 21231
jshuff1@jhu.edu

Rama Chellappa†
Johns Hopkins University
Baltimore, MD 21218
rchella4@jhu.edu

Abstract

Cataract is the leading cause of blindness worldwide. Access to cataract screening is important to enable treatment and vision restoration to eliminate avoidable blindness. This paper introduces an artificial intelligence (AI)-driven approach designed to improve access to cataract screening, using external ocular images captured by community health workers utilizing a smartphone-based anterior segment eye imaging modality. The platform integrates segmentation and classification networks by leveraging pretrained foundation models to accurately differentiate between healthy eyes, immature cataracts, and mature cataracts. We evaluated several fine-tuning strategies and proposed *projection tuning* as an efficient and lightweight approach to tackle distribution shift challenges among datasets. In combination with a Vision Transformer model, we demonstrate exceptional lens classification performance using a small cataract image database. Our investigation confirms that our smartphone-based imaging system combined with the proposed framework offers a effective and accurate solution for cataract detection, addressing distribution shift challenges.

1 Introduction

Cataract remains the leading cause of blindness worldwide, and early detection paired with timely medical intervention can greatly enhance visual acuity and overall quality of life [1, 2]. Recently, there has been growing interest in application of artificial intelligence (AI) to automate diagnosis of various ophthalmic conditions through clinical images with foundational models, initially trained on large image datasets, being fine-tuned for downstream medical image tasks [3, 4, 5, 6]. This is particularly important in low resource settings or rural areas where access to an optometrist or ophthalmologist may be limited [7, 8].

The growing trend of utilizing vast amounts of data with foundation models (FMs) has achieved success in general domains. However, in specialized fields like healthcare, the availability of high-

†Corresponding authors

38th Conference on Neural Information Processing Systems (NeurIPS 2024).

quality data is often limited, making it challenging for these models to perform effectively. Moreover, the complexities of nuanced medical tasks demand expert knowledge that general models may struggle to incorporate [9, 10]. Therefore, an effective fine-tuning strategy becomes essential to adapt these models to the unique demands of specialized domains like healthcare, ensuring they can accurately and efficiently leverage the available data.

In situations where data is scarce and complex, common strategies include data purification to improve its quality and relevance, as well as synthesizing additional data to expand the training set [11, 12]. However, these preprocessing methods often fall short in providing a comprehensive representation of the data distribution, and the use of synthetic images can negatively impact model performance [13].

Another approach is fine-tuning the model to better capture domain-specific features [14, 15]. This involves updating all the parameters during training or selectively fine-tuning the last few layers while keeping the earlier layers frozen. However, updating all the weights is often computationally constrained on mobile devices, and fine-tuning only the later layers may lead to vulnerability to data shifts and overfitting, as these topmost layers hold semantic information crucial for decision-making [16, 17]. To address these challenges, we introduced *projection tuning* within the pre-trained Vision Transformer (ViT) model [18]. In the ViT, images are divided into patches, and these flattened patches undergo a linear projection. These projections are then combined with class and position embeddings before being processed by the Transformer encoder block. We hypothesized that projection tuning would enable the ViT model to outperform other fine-tuning strategies, particularly in scenarios of data scarcity and distribution shifts, as it focuses on fine-tuning the linear projection step, which occurs in the earlier layers of the model.

To date, AI-driven, automated diagnostic technologies primarily rely on data acquired by skilled ophthalmologists using specialized imaging modalities in healthcare settings, such as slit lamp or fundus imaging. Although deep learning systems have been developed to accurately identify diseases like diabetic retinopathy [2, 19], glaucoma [20, 21], and cataract[22, 23], the potential to utilize external ocular images from accessible portable, handheld devices has not been widely explored. To overcome these limitations, our dataset was collected by community health workers in Southern India using a inexpensive, smartphone-based anterior segment eye imaging system that we developed. Due to ongoing developments and upgrades to the imaging system during the data collection period, we obtained two datasets that share similarities while exhibiting subtle differences (lighting, hue, magnification, resolution), leading to a domain shift problem that we sought to address in this study.

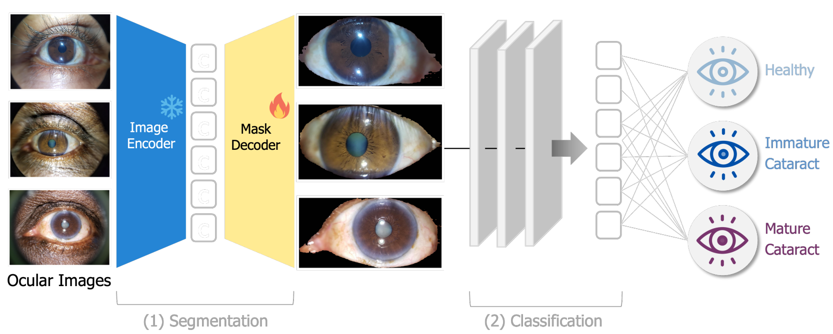


Figure 1: Deep learning framework for cataract diagnosis

In summary, our contribution include:

- Develop an AI platform that utilizes anterior segment images captured by a smartphone-based imaging modality, integrating a foundational model for precise segmentation and classification of lens status.
- Investigate various deep learning architectures and fine-tuning strategies to demonstrate the effectiveness of these approaches in enhancing model performance across datasets.
- Propose a projection tuning method and demonstrate its capability to robustly predict disease status across two cohorts, even with data distribution shifts caused by different hardware designs and issues related to data scarcity.

2 Methods

2.1 Dataset and ethics statement

After obtaining informed consent, community health workers collected smartphone-based eye images of patients attending community eye screenings. Diagnosis labels for each image were obtained using ophthalmologists' clinical diagnoses made via pen light examination at the same screening. The study was approved by the Institutional Review Boards of Aravind Eye Hospital and the Johns Hopkins University School of Medicine.

Smartphone-based anterior segment eye images were gathered in real-world conditions by community health workers at the Aravind Eye Hospital in Tamil Nadu, India using two distinct hardware designs. The first dataset (i.e., $\texttt{CATARACT}_1$) consists of 2,324 images collected using a Samsung M21, including 954 healthy eyes with a clear crystalline lens, 1,054 immature cataracts, and 316 mature cataracts. The second dataset (i.e., $\texttt{CATARACT}_2$) consists of 1,521 images captured with a Samsung S8, including 383 healthy eyes, 985 immature cataracts, and 153 mature cataracts. The primary differences between these datasets are the magnification, type of lighting, and lighting orientation due to differences in design of the hardware imaging systems attached to the Android phones. Representative images from both datasets of each lens status class are shown in Figure 2.

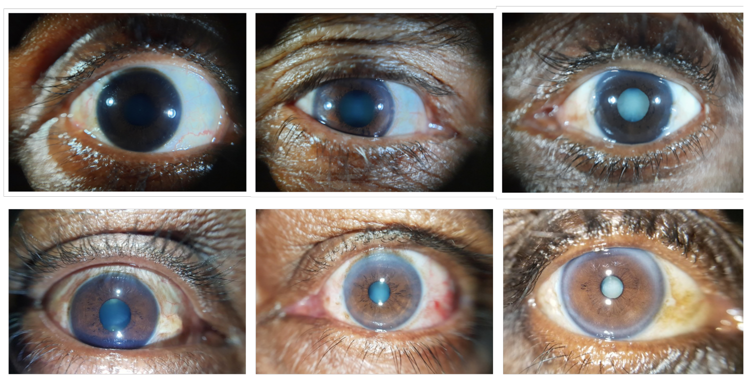


Figure 2: Representative images from the $\texttt{CATARACT}_1$ and $\texttt{CATARACT}_2$ datasets
The top row represents the $\texttt{CATARACT}_1$ dataset, while the bottom row corresponds to the $\texttt{CATARACT}_2$ dataset. The first column displays healthy eyes, the second column shows immature cataracts, and the third column presents mature cataracts with clear differences in lens status.

2.2 Development of a two-stage deep learning framework

As illustrated in Figure 1, the entire process involves (1) segmentation networks to refine images and (2) classification networks to distinguish between healthy eyes, immature cataracts, and mature cataracts labeled with gold standard ophthalmologist diagnoses.

To fine-tune segmentation models, a dataset of 1,000 randomly selected images was manually annotated with polygonal shapes to accurately delineate the regions of interest. We employed the recently developed FM, Segment Anything (SAM) [24], to explore the effectiveness of large FMs for segmentation in a specialized biomedical use case. These annotated images were then used for model training. The models were trained only on the $\texttt{CATARACT}_1$ dataset to refine their ability to accurately segment relevant areas, including sclera, iris, and pupil. The training process optimized the models using backpropagation with Dice loss. Image augmentation was unnecessary to achieve a 0.95 Dice score, and only the decoder was fine-tuned while the encoder remained frozen.

The segmented images were randomly split at a 7:1.5:1.5 ratio for training, validation, and testing of a fine-tuned deep learning models. No overlap was allowed among training, validation, and testing sets. We also implemented 5-fold cross-validation, a method that helps mitigate biases associated with hyperparameter tuning and algorithm selection.

For classification networks, this study explored the performance of four CNN models (VGG-11 [25], ResNet-18 [26], MobileNetV2 [27], EfficientNet B0 [28]) and five Transformer models (DeiT-Ti, DeiT-distilled [29], MobileViT-S [30], EfficientFormer-L1 [31], ViT-base [18]). Each model was initialized with pre-trained ImageNet [32] weights to enhance performance.

During training, we applied image transformations and augmentations, such as random horizontal flips and Gaussian blur, to improve model generalization. Image standardization and normalization were consistently applied across both training and testing phases to ensure uniformity in data processing.

We utilized the PyTorch deep learning framework [33] (version 2.0.1) for model training with one NVIDIA RTX A5500. The models were trained using backpropagation, with batch sizes of 256 images for CNN models and 8 images for Transformer models. We employed the Adaptive Moment Estimation (ADAM) optimizer [34] with a learning rate set to 1×10^{-4}. The cross-entropy loss function was applied as the objective function for the classification tasks, ensuring the models were optimized effectively for accuracy. The code will be released upon publication.

2.3 Projection tuning

This section outlines a brief overview of how ViT works and presents the projection tuning, designed to optimize the ViT model for efficient fine-tuning and to address the distributional differences between datasets (Figure 4).

Consider an image of resolution H and W with C number of channels. ViT divides each input image into a grid of patches, where each patch is of size $p \times p$ and the total number of patches is $N = \frac{H \times W}{p^2}$ [2]. These patches are then flattened into vectors, resulting in a sequence of patch vectors: $\{\mathbf{x}_i\}_{i=1}^{N}$. Each patch vector $\mathbf{x}_i \in \mathbb{R}^{\times(p^2 \cdot C)}$ is linearly projected into a D-dimensional token embeddings $\mathbf{z}_{0i}$, using a learnable linear projection $\mathbf{E} \in \mathbb{R}^{(p^2 \cdot C) \times D}$. These token embeddings $\{\mathbf{z}_{0i}\}_{i=1}^{N}$ serve as input to the Transformer encoder, which produce the final classification predictions. For more detailed information, please refer to the original paper [18].

Our projection tuning approach solely fine-tunes the linear projection matrix, E, while all other layers are kept frozen. The core idea behind the proposed method is to decouple the module that maps pixel inputs to vector representations (i.e., patcher) from the one that performs actual classification based on these representations. In this framework, the Transformer encoder – which can be regarded as the reasoning and decision-making module – is shared across different datasets, while the fine-tuning stage focuses on effectively aligning input representations.

3 Results

3.1 Segmentation and classification using the $\texttt{CATARACT}_1$ dataset

3.1.1 The effect of segmentation on classification

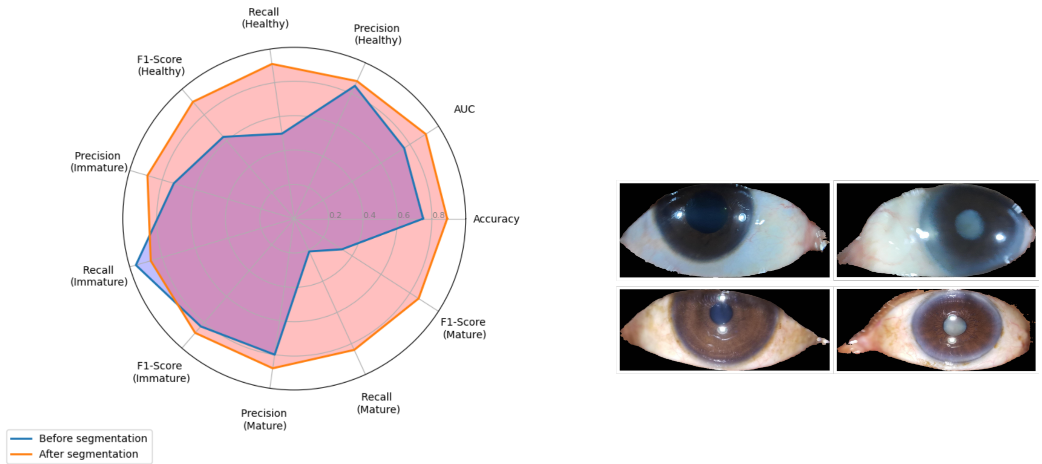
Our two-stage deep learning framework for cataract diagnosis starts with a segmentation module. Segmentation is performed before classification for two key reasons: (1) medical imaging modalities typically exhibit high complexity and dimensionality relative to the dataset size, and (2) our datasets consisted of images captured using a portable imaging system in real-world settings rather than controlled clinical environments, leading to increased variability. Inspired by iris preprocessing procedures used in biometric applications and acknowledging the advantages of minimized skin area and increased focus on the eye in images, we applied segmentation to standardize the images.

We fine-tuned SAM and utilized it for subsequent analyses. As discussed in Section 2.2, the fine-tuning was performed exclusively using the $\texttt{CATARACT}_1$ dataset. Additional fine-tuning with the $\texttt{CATARACT}_2$ dataset was not necessary to achieve comparable segmentation performance.

We employed convolutional neural network models, such as Very Deep Convolutional Networks 11 (VGG-11), Residual Networks 18 (ResNet-18), MobileNet, and EfficientNet, for the classification module (Section 3.1.2). When using the unsegmented whole images, the classification accuracy with ResNet-18 was 75%. After applying segmentation, the classification performance improved by 14%, resulting in an overall performance of 89% (Figure 3). Specifically, for mature cataract classification, which is the minority class among the three, the recall was 21% during testing. However,

[2] H and W are pre-adjusted to be multiples of p

after segmentation, the recall for the mature cataract class increased to 84%. This indicates that segmentation positively impacts overall classification performance by preprocessing the images and effectively guiding the model to focus on relevant anatomical regions, such as the pupil, rather than the features surrounding the eye.



(a) (b)

Figure 3: Comparison of segmentation effects (a) with example images (b).

3.1.2 Quantitative analysis of classification performance

Following the segmentation module, the models generate a binary mask to extract the region of interest. Using this mask, we selected pixel values within the targeted area while assigning zero values to the unwanted regions. Once the image preprocessing was complete, the segmented images were used in the classification module. We fine-tuned all layers to conduct tests using both Convolutional Neural Network (CNN) and Transformer models for this classification task.

In previously published papers focused on diagnosing eye diseases using deep learning approaches, models such as VGG, ResNet, MobileNet, Densely Connected Convolutional Networks (DenseNet), and their variations were commonly used [35, 36, 37, 38, 39]. However, using models with a large number of parameters relative to the dataset size can lead to overfitting and reduced performance. Therefore, we conducted tests using CNN models, including VGG-11, ResNet-18, MobileNetV2, and EfficientNet B0. These models are known for having relatively fewer parameters within their variants, which helps mitigate the risk of overfitting when fine-tuning on smaller datasets. Moreover, their lightweight nature makes them well-suited for future deployment on mobile devices. We also evaluated Transformer models, such as DeiT-Ti, DeiT-distilled, MobileViT-S, EfficientFormer-L1, and Vision Transformer (ViT-base). Transformer models are particularly effective at capturing long-range dependencies in data through self-attention and are efficient in processing inputs due to their ability to parallelize computations. CNN and Transformer model performance are shown in Table 1.

Accuracy results for the CNN models were as follows: VGG-11: 84%; ResNet-18: 89%; MobileNetV2: 86%; EfficientNet B0: 86%. ResNet-18 consistently outperformed the other CNN models across other evaluation metrics, including precision, recall, F1-score, and AUC. For the Transformer models, the accuracy outcomes were as follows: DeiT-Ti: 87%; DeiT-distilled: 88%; MobileViT-S: 88%; EfficientFormer-L1: 87%; ViT-base: 90%. Among these models, ViT-base consistently delivered the highest performance across diverse evaluation metrics.

Transformers generally outperformed CNN models across several evaluation metrics and proved to be more efficient in terms of training time. While CNN models needed at least 20 epochs to reach the desired performance, Transformer models achieved comparable results in 5 epochs, highlighting their efficiency in both training and inference.

Table 1: Comparison of classification performance after segmentation. Light green cells indicate the best performance among CNN models, and light blue cells indicate the best performance among Transformer models. (AUC: Area Under the Curve)

Model	Class	Accuracy	Precision	Recall	F1-score	AUC
VGG-11	Healthy	0.84 ± 0.02	0.84 ± 0.01	0.85 ± 0.02	0.85 ± 0.02	0.86 ± 0.01
	Immature Cataract		0.83 ± 0.02	0.84 ± 0.03	0.84 ± 0.02	
	Mature Cataract		0.82 ± 0.05	0.79 ± 0.07	0.80 ± 0.05	
ResNet-18	Healthy	0.89 ± 0.01	0.88 ± 0.03	0.91 ± 0.02	0.90 ± 0.02	0.91 ± 0.01
	Immature Cataract		0.89 ± 0.02	0.87 ± 0.03	0.88 ± 0.01	
	Mature Cataract		0.88 ± 0.04	0.84 ± 0.05	0.86 ± 0.03	
MobileNetV2	Healthy	0.86 ± 0.01	0.87 ± 0.03	0.88 ± 0.03	0.88 ± 0.02	0.88 ± 0.01
	Immature Cataract		0.85 ± 0.03	0.86 ± 0.03	0.86 ± 0.01	
	Mature Cataract		0.84 ± 0.05	0.79 ± 0.07	0.81 ± 0.05	
EfficientNet B0	Healthy	0.86 ± 0.02	0.87 ± 0.02	0.87 ± 0.02	0.87 ± 0.02	0.88 ± 0.01
	Immature Cataract		0.86 ± 0.02	0.87 ± 0.03	0.86 ± 0.02	
	Mature Cataract		0.82 ± 0.05	0.81 ± 0.03	0.81 ± 0.02	
DeiT-Ti	Healthy	0.87 ± 0.01	0.88 ± 0.03	0.89 ± 0.04	0.88 ± 0.01	0.89 ± 0.01
	Immature Cataract		0.86 ± 0.02	0.87 ± 0.05	0.86 ± 0.02	
	Mature Cataract		0.89 ± 0.07	0.81 ± 0.05	0.84 ± 0.02	
DeiT-distilled	Healthy	0.88 ± 0.01	0.88 ± 0.01	0.90 ± 0.02	0.89 ± 0.00	0.90 ± 0.01
	Immature Cataract		0.87 ± 0.02	0.88 ± 0.01	0.87 ± 0.01	
	Mature Cataract		0.87 ± 0.01	0.81 ± 0.04	0.84 ± 0.03	
MobileViT-S	Healthy	0.88 ± 0.01	0.89 ± 0.02	0.89 ± 0.02	0.89 ± 0.02	0.90 ± 0.01
	Immature Cataract		0.87 ± 0.02	0.88 ± 0.02	0.87 ± 0.01	
	Mature Cataract		0.87 ± 0.02	0.84 ± 0.08	0.85 ± 0.05	
EfficientFormer-L1	Healthy	0.87 ± 0.01	0.86 ± 0.02	0.91 ± 0.02	0.89 ± 0.01	0.89 ± 0.01
	Immature Cataract		0.88 ± 0.02	0.86 ± 0.02	0.87 ± 0.01	
	Mature Cataract		0.87 ± 0.03	0.81 ± 0.04	0.84 ± 0.03	
ViT-base	Healthy	0.90 ± 0.02	0.90 ± 0.02	0.92 ± 0.01	0.91 ± 0.02	0.91 ± 0.02
	Immature Cataract		0.89 ± 0.01	0.90 ± 0.01	0.89 ± 0.01	
	Mature Cataract		0.92 ± 0.03	0.83 ± 0.07	0.87 ± 0.04	

3.2 Impact of data distribution shifts on classification

As previously discussed, our two datasets are largely similar but display subtle differences due to variations in magnification and lighting. To visually capture these differences in the datasets, we plotted pixel value histograms, normalizing them to display the probability density, as shown in Figure 4. This analysis was conducted using segmented images, excluding the masked areas around the eye, which accounted for approximately 30% of all pixels. In the $\texttt{CATARACT}_1$ dataset, a significant pixel value peak was observed near 50, with a large concentration of values at 255. In contrast, the $\texttt{CATARACT}_2$ dataset exhibited a similar peak around 70, with a reduced frequency of pixel values above 160.

To assess the impact of these differences, we first performed zero-shot inference on the images from the $\texttt{CATARACT}_2$ dataset using deep learning models that were fine-tuned on the $\texttt{CATARACT}_1$ dataset. As in Table 2, this resulted in performance decline across both CNN and Transformer models. For CNN models, the performance decreased by up to 35%, with an average drop of 29%. Transformer models experienced a maximum performance drop of 39% and an average decrease of 29.4%.

When the two datasets were combined and models fine-tuned from scratch (mixed-dataset tuning), the performance of CNN models dropped by about 2-5%, while Transformer models exhibited a 1-2% decline. Additionally, when both CNN and Transformer models were fully fine-tuned on the $\texttt{CATARACT}_2$ dataset (full tuning), they maintained performance levels comparable to those achieved with the $\texttt{CATARACT}_1$ dataset.

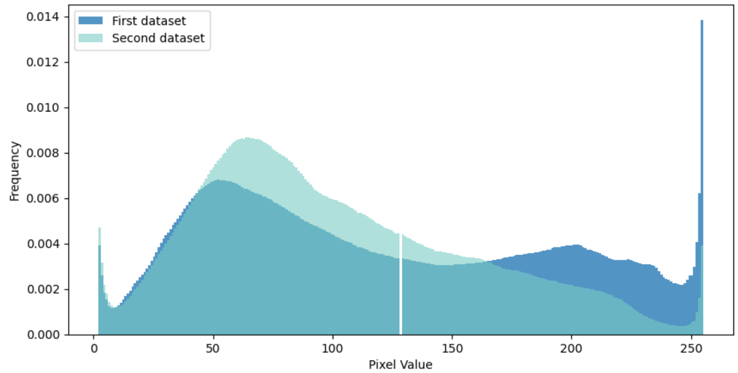


Figure 4: Density distribution plots for the $\texttt{CATARACT}_1$ and $\texttt{CATARACT}_2$ datasets.

Table 2: Accuracy comparison using the $\texttt{CATARACT}_2$ dataset across different tuning strategies.

Model	Zero shot	Mixed-dataset tuning	Full tuning
VGG-11	50%	82%	84%
ResNet-18	61%	84%	85%
MobileNetV2	67%	83%	84%
EfficientNet B0	51%	83%	83%
DeiT-Ti	58%	86%	83%
DeiT-distilled	63%	86%	84%
MobileViT-S	70%	87%	86%
EfficientFormer-L1	48%	85%	87%
ViT-base	54%	88%	86%

3.3 Projection tuning

As described in Section 2.1, the $\texttt{CATARACT}_1$ dataset consists of 2,324 images, while the $\texttt{CATARACT}_2$ dataset consists of 1,521 images. The differences in the style and distribution of these datasets led to a decline in classification performance (Table 2). The distribution shift and the smaller size of the $\texttt{CATARACT}_2$ dataset likely contributed to the difficulty in capturing the general representations necessary for accurate diagnosis, further impacting performance. These findings highlight the need for fine-tuning with a limited amount of new data and aligning features between the two datasets to improve classification accuracy and overall model robustness.

We focused on the ViT-base model due to its superior performance and resilience, as detailed in Sections 3.1.2 and 3.2. In the ViT-base architecture, image patches are linearly projected and flattened into patch embeddings before being fed into the Transformer encoder (Section 2.3). We explored four approaches to fine-tune this model: (1) fine-tuning all layers with the new dataset, (2) fine-tuning only the last layer, (3) low-rank adaptation (LoRA) [40], and (4) fine-tuning the projection layer.

LoRA is a parameter-efficient transfer learning approach, specifically tailored to enhance the effectiveness of fine-tuning large pre-trained models. By decomposing weight updates into low-rank matrices, it allows for the efficient adaptation of extensive neural networks to specific tasks while significantly reducing the number of trainable parameters without compromising overall performance. In our study, we used the ViT-base model as the backbone and focused solely on fine-tuning the LoRA adapters, targeting the query, key, and value modules of the self-attention mechanism. We set the alpha scaling parameter for LoRA to 16 and evaluated ranks of 2, 4, 8, and 16 to compare the results.

When using the complete set of updated images for training and validation, projection tuning achieved a performance of 81%, training from scratch resulted in 85%, and fine-tuning the last layer also yielded 76% (Figure 5). However, projection tuning demonstrated superior performance compared to the other methods when working with smaller datasets. For example, with only 10% of the dataset, projection tuning achieved 77% performance, whereas training from scratch resulted in 73%, and

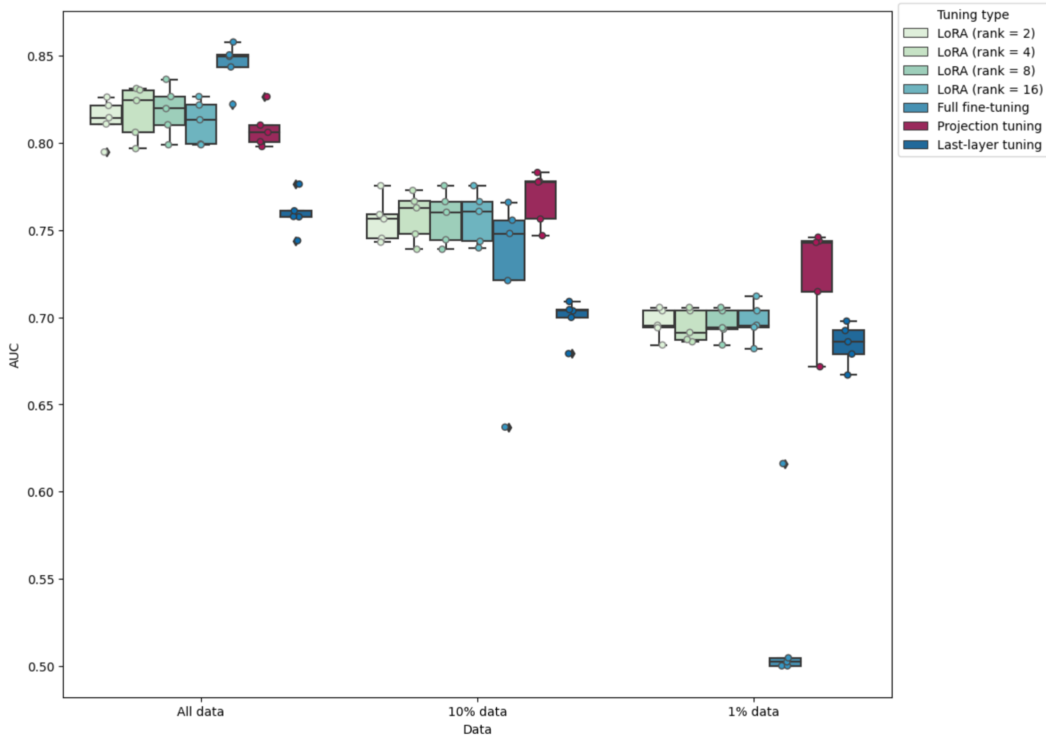


Figure 5: Schematic illustration of projection tuning and its outcomes compared to other fine-tuning methods, including full-layer fine-tuning, last-layer fine-tuning, and LoRA.

fine-tuning the last layer produced 70%. With just 1% of the dataset, projection tuning reached 72% performance, compared to 52% for training from scratch and 69% for fine-tuning the last layer.

In Figure 5, LoRA fine-tuning exhibited a consistent pattern relative to the rank size. As the size of the fine-tuning dataset decreased, LoRA's performance also declined. When using the full dataset, LoRA achieved an AUC between 81% and 82%. With 10% of the dataset, the AUC dropped to 76%, and with just 1% of the dataset, performance ranged from 69% to 70%. In comparison, projection tuning outperformed LoRA by 1% with 10% of the dataset, and by 2% to 3% with 1% of the dataset. Moreover, although LoRA requires fewer parameters to adjust compared to projection tuning, its floating point operations per second (FLOPS) is higher (Table 3). In terms of FLOPS, projection tuning, with its lower computational demands, is more efficient and better suited for resource-constrained environments, such as mobile devices or real-time systems.

To further analyze the differences in feature representation depending on the fine-tuning strategies, we conducted principal component analysis (PCA). Using the ViT-base model, initialized with the pretrained $\texttt{CATARACT}_1$ dataset and then fine-tuned with the $\texttt{CATARACT}_2$ dataset using different strategies, we extracted features from the model's final layer. These features were subsequently analyzed with PCA and visualized in a 2D plot.

As shown in Figure A.1 and A.2 in Appendix, projection tuning demonstrated clear class separation using both 1% and 10% of the $\texttt{CATARACT}_2$ dataset for fine-tuning. While last-layer tuning also produced robust results with both 1% and 10%, the model displayed more dispersion and overlap between classes during testing compared to projection tuning. This likely contributed to the observed decline in performance. Additionally, fine-tuning all layers with 10% of the $\texttt{CATARACT}_2$ dataset failed to achieve clear class separation, and with just 1% of the $\texttt{CATARACT}_2$ dataset, the model exhibited almost no discernible decision boundary. Although the rank order of LoRA had little impact on differentiating the decision boundary, it was evident that 1% of the $\texttt{CATARACT}_2$ dataset was inadequate for fine-tuning the pretrained vit-base model, which differed from the results seen with projection tuning. In contrast, 10% data was necessary to detect variations in the feature space.

These findings highlight several advantages of projection tuning for fine-tuning FMs, particularly when dealing with small datasets and near-out-of-distribution scenarios. First, fully fine-tuning FMs demands significant time and resources. For example, as shown in Table 3, fine-tuning all layers of the ViT-base model requires 85.8M parameters, while projection tuning, which focuses on adjusting the linear layer, only fine-tunes 0.59M parameters. Moreover, while full fine-tuning can risk overfitting due to the broader parameter adjustments, projection tuning's focus on the linear layer reduces this risk. Second, unlike last-layer tuning, which adjusts parameters only after the data has passed through the entire architecture, projection tuning modifies the model before it enters the encoder. This approach enhances the model's adaptability and robustness, contributing to improved performance under challenging conditions.

Table 3: Comparison of floating point operations per second (FLOPS) alongside the parameter counts for deep learning models.

Model	GFLOPS	# of Parameters
VGG-11	7.61	128.78M
ResNet-18	1.83	11.2M
MobileNetV2	0.33	2.2M
EfficientNet B0	0.42	4.0M
DeiT-Ti	1.08	5.5M
DeiT-distilled	1.08	5.5M
Mobile ViT-S	1.47	4.9M
EfficientFormer-L1	1.32	11.4M
ViT-base	16.87	85.8M
ViT-base + last layer tuning	16.87	2307
ViT-base + projection tuning	16.87	0.59M
ViT-base + LoRA (rank = 8)	17.67	0.44M

4 Discussion

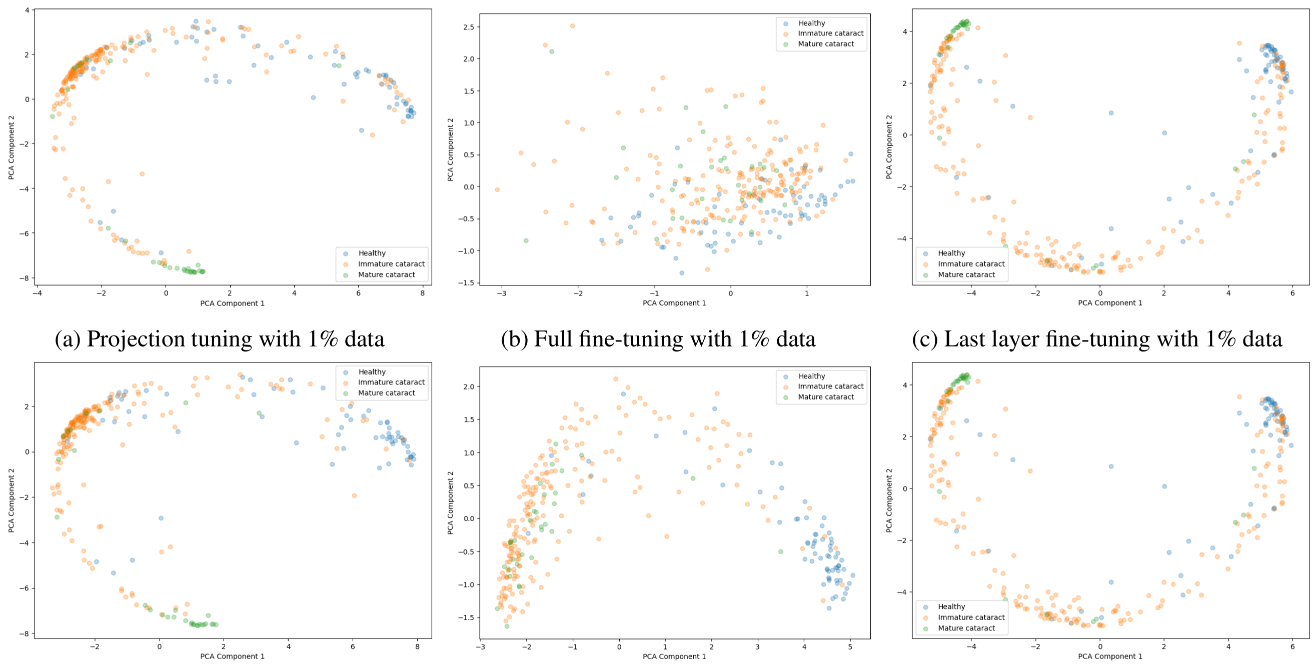
We employed deep learning models with anterior segment eye images captured by a novel smartphone-based diffuse illumination imaging modality to predict lens status in South Indian individuals, beginning with segmentation to eliminate unwanted areas and followed by classification to distinguish between healthy eyes, immature cataracts, and mature cataracts. Our comprehensive evaluation shows that segmentation improves prediction performance, particularly in the context of small datasets. Moreover, we found that projection tuning proved to be an effective fine-tuning strategy, reducing overfitting and improving robustness against data distribution shifts in smaller datasets.

The ViT-base pretrained on $\texttt{CATARACT}_1$ exhibited limited zero-shot performance on $\texttt{CATARACT}_2$, despite having more parameters than other models. This is particularly evident when working with small, real-world datasets like ours, where the benefits of FMs are not fully realized due to the lack of extensive training data. Additionally, FMs in medical imaging face limitations due to incomplete representation of modalities during pretraining, reducing their effectiveness in specialized applications. Our findings suggest that projection tuning is an effective approach for fine-tuning large models on smaller datasets, addressing model capacity with real-world data limitations.

Projection tuning outperformed other fine-tuning methods, such as full-layer and last-layer tuning, especially in near out-of-distribution scenarios with small datasets. Exploring different projection head architectures could further enhance these results. Additionally, comparing projection tuning with other parameter-efficient methods with different vision FMs beyond ViT-base is also necessary to validate its broader applicability.

The ultimate goal of our project is to develop an accurate and efficient cataract detection model for smartphone deployment and broad implementation. Several parameters must be considered to ensure successful deployment on mobile platforms, including model compression to maintain high performance without sacrificing accuracy. Additionally, since our segmentation module required extra annotations for training, it is essential to explore the use of image augmentation as an alternative approach. This could potentially eliminate the need for a separate segmentation step, reduce the associated manual labor, while learning the general representation for inference.

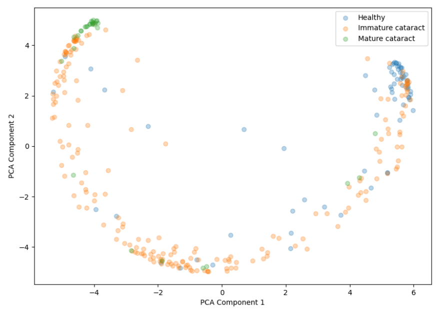
A Appendix



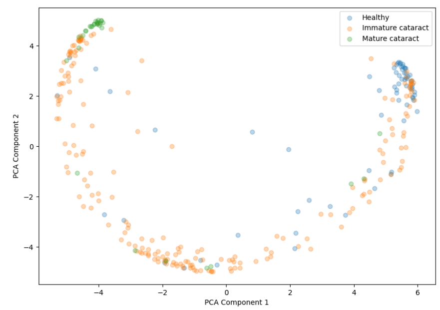
(a) Projection tuning with 1% data (b) Full fine-tuning with 1% data (c) Last layer fine-tuning with 1% data

(d) Projection tuning with 10% data (e) Full fine-tuning with 10% data (f) Last layer fine-tuning with 10% data

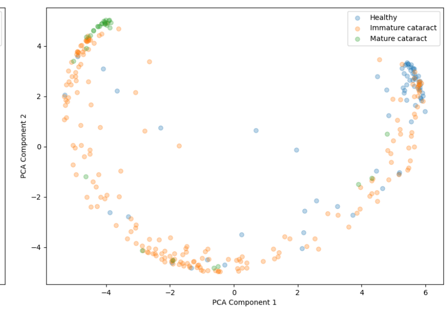
Figure A.1: 2D PCA plots comparing projection tuning, full fine-tuning, and last layer fine-tuning strategies across different dataset sizes.



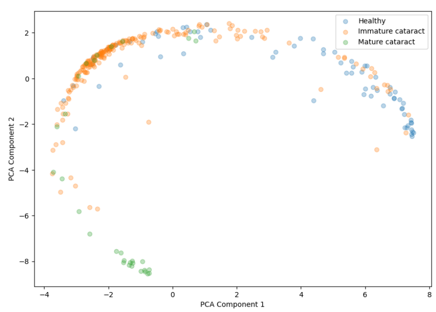
(a) LoRA with 1% data (rank = 2)



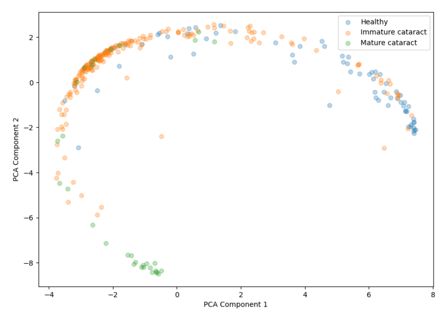
(b) LoRA with 1% data (rank = 4)



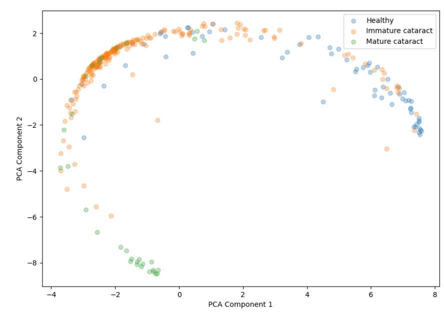
(c) LoRA with 1% data (rank = 8)



(d) LoRA with 10% data (rank = 2)



(e) LoRA with 10% data (rank = 4)



(f) LoRA with 10% data (rank = 8)

Figure A.2: 2D PCA plots showing LoRA fine-tuning across different dataset sizes and ranks.

References

[1] Jaimie D Steinmetz, Rupert RA Bourne, Paul Svitil Briant, Seth R Flaxman, Hugh RB Taylor, Jost B Jonas, Amir Aberhe Abdoli, Woldu Aberhe Abrha, Ahmed Abualhasan, Eman Girum Abu-Gharbieh, et al. Causes of blindness and vision impairment in 2020 and trends over 30 years, and prevalence of avoidable blindness in relation to vision 2020: the right to sight: an analysis for the global burden of disease study. *The Lancet Global Health*, 9(2):e144–e160, 2021.

[2] K Shankar, Abdul Rahaman Wahab Sait, Deepak Gupta, S Kd Lakshmanaprabu, Ashish Khanna, and Hari Mohan Pandey. Automated detection and classification of fundus diabetic retinopathy images using synergic deep learning model. *Pattern Recognition Letters*, 133:210–216, 2020.

[3] Daniel Shu Wei Ting, Louis R Pasquale, Lily Peng, John Peter Campbell, Aaron Y Lee, Rajiv Raman, Gavin Siew Wei Tan, Leopold Schmetterer, Pearse A Keane, and Tien Yin Wong. Artificial intelligence and deep learning in ophthalmology. *British Journal of Ophthalmology*, 103(2):167–175, 2019.

[4] Ryan Poplin, Avinash V Varadarajan, Katy Blumer, Yun Liu, Michael V McConnell, Greg S Corrado, Lily Peng, and Dale R Webster. Prediction of cardiovascular risk factors from retinal fundus photographs via deep learning. *Nature biomedical engineering*, 2(3):158–164, 2018.

[5] Boris Babenko, Akinori Mitani, Ilana Traynis, Naho Kitade, Preeti Singh, April Y Maa, Jorge Cuadros, Greg S Corrado, Lily Peng, Dale R Webster, et al. Detection of signs of disease in external photographs of the eyes via deep learning. *Nature biomedical engineering*, 6(12): 1370–1383, 2022.

[6] Yukun Zhou, Mark A Chia, Siegfried K Wagner, Murat S Ayhan, Dominic J Williamson, Robbert R Struyven, Timing Liu, Moucheng Xu, Mateo G Lozano, Peter Woodward-Court, et al. A foundation model for generalizable disease detection from retinal images. *Nature*, 622 (7981):156–163, 2023.

[7] Matthew J Burton, Jacqueline Ramke, Ana Patricia Marques, Rupert RA Bourne, Nathan Congdon, Iain Jones, Brandon AM Ah Tong, Simon Arunga, Damodar Bachani, Covadonga Bascaran, et al. The lancet global health commission on global eye health: vision beyond 2020. *The Lancet Global Health*, 9(4):e489–e551, 2021.

[8] Diane M Gibson. The geographic distribution of eye care providers in the united states: implications for a national strategy to improve vision health. *Preventive medicine*, 73:30–36, 2015.

[9] Muhammad Imran Razzak, Saeeda Naz, and Ahmad Zaib. Deep learning for medical image processing: Overview, challenges and the future. *Classification in BioApps: Automation of decision making*, pages 323–350, 2018.

[10] Andrés Anaya-Isaza, Leonel Mera-Jiménez, and Martha Zequera-Diaz. An overview of deep learning in medical imaging. *Informatics in medicine unlocked*, 26:100723, 2021.

[11] Devansh Bisla, Anna Choromanska, Russell S Berman, Jennifer A Stein, and David Polsky. Towards automated melanoma detection with deep learning: Data purification and augmentation. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition workshops*, pages 0–0, 2019.

[12] Ms Aayushi Bansal, Dr Rewa Sharma, and Dr Mamta Kathuria. A systematic review on data scarcity problem in deep learning: solution and applications. *ACM Computing Surveys (Csur)*, 54(10s):1–29, 2022.

[13] Sina Alemohammad, Josue Casco-Rodriguez, Lorenzo Luzi, Ahmed Imtiaz Humayun, Hossein Babaei, Daniel LeJeune, Ali Siahkoohi, and Richard G Baraniuk. Self-consuming generative models go mad. *arXiv preprint arXiv:2307.01850*, 2023.

[14] Chuanqi Tan, Fuchun Sun, Tao Kong, Wenchang Zhang, Chao Yang, and Chunfang Liu. A survey on deep transfer learning. In *Artificial Neural Networks and Machine Learning–ICANN 2018: 27th International Conference on Artificial Neural Networks, Rhodes, Greece, October 4-7, 2018, Proceedings, Part III 27*, pages 270–279. Springer, 2018.

[15] Hee E Kim, Alejandro Cosa-Linan, Nandhini Santhanam, Mahboubeh Jannesari, Mate E Maros, and Thomas Ganslandt. Transfer learning for medical image classification: a literature review. *BMC medical imaging*, 22(1):69, 2022.

[16] I Tenney. Bert rediscovers the classical nlp pipeline. *arXiv preprint arXiv:1905.05950*, 2019.

[17] Damai Dai, Li Dong, Yaru Hao, Zhifang Sui, Baobao Chang, and Furu Wei. Knowledge neurons in pretrained transformers. *arXiv preprint arXiv:2104.08696*, 2021.

[18] Alexey DOSOVITSKIY. An image is worth 16x16 words: Transformers for image recognition at scale. *arXiv preprint arXiv:2010.11929*, 2020.

[19] Mohammad Z Atwany, Abdulwahab H Sahyoun, and Mohammad Yaqub. Deep learning techniques for diabetic retinopathy classification: A survey. *IEEE Access*, 10:28642–28655, 2022.

[20] Xiangyu Chen, Yanwu Xu, Damon Wing Kee Wong, Tien Yin Wong, and Jiang Liu. Glaucoma detection based on deep convolutional neural network. In *2015 37th annual international conference of the IEEE engineering in medicine and biology society (EMBC)*, pages 715–718. IEEE, 2015.

[21] Baidaa Al-Bander, Waleed Al-Nuaimy, Majid A Al-Taee, and Yalin Zheng. Automated glaucoma diagnosis using deep learning approach. In *2017 14th International Multi-Conference on Systems, Signals & Devices (SSD)*, pages 207–210. IEEE, 2017.

[22] Yih-Chung Tham, Jocelyn Hui Lin Goh, Ayesha Anees, Xiaofeng Lei, Tyler Hyungtaek Rim, Miao-Li Chee, Ya Xing Wang, Jost B Jonas, Sahil Thakur, Zhen Ling Teo, et al. Detecting visually significant cataract using retinal photograph-based deep learning. *Nature aging*, 2(3): 264–271, 2022.

[23] Masum Shah Junayed, Md Baharul Islam, Arezoo Sadeghzadeh, and Saimunur Rahman. Cataractnet: An automated cataract detection system using deep learning for fundus images. *IEEE access*, 9:128799–128808, 2021.

[24] Alexander Kirillov, Eric Mintun, Nikhila Ravi, Hanzi Mao, Chloe Rolland, Laura Gustafson, Tete Xiao, Spencer Whitehead, Alexander C Berg, Wan-Yen Lo, et al. Segment anything. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 4015–4026, 2023.

[25] Karen Simonyan and Andrew Zisserman. Very deep convolutional networks for large-scale image recognition. *arXiv preprint arXiv:1409.1556*, 2014.

[26] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 770–778, 2016.

[27] Mark Sandler, Andrew Howard, Menglong Zhu, Andrey Zhmoginov, and Liang-Chieh Chen. Mobilenetv2: Inverted residuals and linear bottlenecks. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 4510–4520, 2018.

[28] Mingxing Tan. Efficientnet: Rethinking model scaling for convolutional neural networks. *arXiv preprint arXiv:1905.11946*, 2019.

[29] Hugo Touvron, Matthieu Cord, Matthijs Douze, Francisco Massa, Alexandre Sablayrolles, and Hervé Jégou. Training data-efficient image transformers & distillation through attention. In *International conference on machine learning*, pages 10347–10357. PMLR, 2021.

[30] Sachin Mehta and Mohammad Rastegari. Mobilevit: light-weight, general-purpose, and mobile-friendly vision transformer. *arXiv preprint arXiv:2110.02178*, 2021.

[31] Yanyu Li, Geng Yuan, Yang Wen, Ju Hu, Georgios Evangelidis, Sergey Tulyakov, Yanzhi Wang, and Jian Ren. Efficientformer: Vision transformers at mobilenet speed. *Advances in Neural Information Processing Systems*, 35:12934–12949, 2022.

[32] Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, and Li Fei-Fei. Imagenet: A large-scale hierarchical image database. In *2009 IEEE Conference on Computer Vision and Pattern Recognition*, pages 248–255, 2009. doi: 10.1109/CVPR.2009.5206848.

[33] Adam Paszke, Sam Gross, Francisco Massa, Adam Lerer, James Bradbury, Gregory Chanan, Trevor Killeen, Zeming Lin, Natalia Gimelshein, Luca Antiga, et al. Pytorch: An imperative style, high-performance deep learning library. *Advances in neural information processing systems*, 32, 2019.

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

[35] Tiarnan DL Keenan, Qingyu Chen, Elvira Agrón, Yih-Chung Tham, Jocelyn Hui Lin Goh, Xiaofeng Lei, Yi Pin Ng, Yong Liu, Xinxing Xu, Ching-Yu Cheng, et al. Deeplensnet: deep learning automated diagnosis and quantitative classification of cataract type and severity. *Ophthalmology*, 129(5):571–584, 2022.

[36] Hanaa Salem, Kareem R Negm, Mahmoud Y Shams, and Omar M Elzeki. Recognition of ocular disease based optimized vgg-net models. In *Medical Informatics and Bioimaging Using Artificial Intelligence: Challenges, Issues, Innovations and Recent Developments*, pages 93–111. Springer, 2021.

[37] Jay Kant Pratap Singh Yadav and Sunita Yadav. Computer-aided diagnosis of cataract severity using retinal fundus images and deep learning. *Computational Intelligence*, 38(4):1450–1473, 2022.

[38] Yaroub Elloumi. Cataract grading method based on deep convolutional neural networks and stacking ensemble learning. *International Journal of Imaging Systems and Technology*, 32(3): 798–814, 2022.

[39] Zhongwen Li, Lei Wang, Xuefang Wu, Jiewei Jiang, Wei Qiang, He Xie, Hongjian Zhou, Shanjun Wu, Yi Shao, and Wei Chen. Artificial intelligence in ophthalmology: The path to the real-world clinic. *Cell Reports Medicine*, 4(7), 2023.

[40] Edward J Hu, Yelong Shen, Phillip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Shean Wang, Lu Wang, and Weizhu Chen. Lora: Low-rank adaptation of large language models. *arXiv preprint arXiv:2106.09685*, 2021.

NeurIPS Paper Checklist

1. **Claims**

   Question: Do the main claims made in the abstract and introduction accurately reflect the paper's contributions and scope?

   Answer: [Yes]

   Justification: The abstract and introduction clearly outline the paper's claims and contributions with the claims aligning with experimental results.

   Guidelines:

   - The answer NA means that the abstract and introduction do not include the claims made in the paper.
   - The abstract and/or introduction should clearly state the claims made, including the contributions made in the paper and important assumptions and limitations. A No or NA answer to this question will not be perceived well by the reviewers.
   - The claims made should match theoretical and experimental results, and reflect how much the results can be expected to generalize to other settings.
   - It is fine to include aspirational goals as motivation as long as it is clear that these goals are not attained by the paper.

2. **Limitations**

   Question: Does the paper discuss the limitations of the work performed by the authors?

   Answer: [Yes]

   Justification: Throughout the Results and Discussion sections, we examined the factors influencing the approach's performance, clearly stated our assumptions, and addressed the limitations of this work.

   Guidelines:

   - The answer NA means that the paper has no limitation while the answer No means that the paper has limitations, but those are not discussed in the paper.
   - The authors are encouraged to create a separate "Limitations" section in their paper.
   - The paper should point out any strong assumptions and how robust the results are to violations of these assumptions (e.g., independence assumptions, noiseless settings, model well-specification, asymptotic approximations only holding locally). The authors should reflect on how these assumptions might be violated in practice and what the implications would be.
   - The authors should reflect on the scope of the claims made, e.g., if the approach was only tested on a few datasets or with a few runs. In general, empirical results often depend on implicit assumptions, which should be articulated.
   - The authors should reflect on the factors that influence the performance of the approach. For example, a facial recognition algorithm may perform poorly when image resolution is low or images are taken in low lighting. Or a speech-to-text system might not be used reliably to provide closed captions for online lectures because it fails to handle technical jargon.
   - The authors should discuss the computational efficiency of the proposed algorithms and how they scale with dataset size.
   - If applicable, the authors should discuss possible limitations of their approach to address problems of privacy and fairness.
   - While the authors might fear that complete honesty about limitations might be used by reviewers as grounds for rejection, a worse outcome might be that reviewers discover limitations that aren't acknowledged in the paper. The authors should use their best judgment and recognize that individual actions in favor of transparency play an important role in developing norms that preserve the integrity of the community. Reviewers will be specifically instructed to not penalize honesty concerning limitations.

3. **Theory Assumptions and Proofs**

   Question: For each theoretical result, does the paper provide the full set of assumptions and a complete (and correct) proof?

Answer: [NA]

Justification: While this paper does not provide proofs for theoretical assumptions, it thoroughly details all the procedures.

Guidelines:

- The answer NA means that the paper does not include theoretical results.
- All the theorems, formulas, and proofs in the paper should be numbered and cross-referenced.
- All assumptions should be clearly stated or referenced in the statement of any theorems.
- The proofs can either appear in the main paper or the supplemental material, but if they appear in the supplemental material, the authors are encouraged to provide a short proof sketch to provide intuition.
- Inversely, any informal proof provided in the core of the paper should be complemented by formal proofs provided in appendix or supplemental material.
- Theorems and Lemmas that the proof relies upon should be properly referenced.

4. **Experimental Result Reproducibility**

   Question: Does the paper fully disclose all the information needed to reproduce the main experimental results of the paper to the extent that it affects the main claims and/or conclusions of the paper (regardless of whether the code and data are provided or not)?

   Answer: [Yes]

   Justification: In the Method section, we detailed the experimental steps to ensure the reproducibility of our results.

   Guidelines:

   - The answer NA means that the paper does not include experiments.
   - If the paper includes experiments, a No answer to this question will not be perceived well by the reviewers: Making the paper reproducible is important, regardless of whether the code and data are provided or not.
   - If the contribution is a dataset and/or model, the authors should describe the steps taken to make their results reproducible or verifiable.
   - Depending on the contribution, reproducibility can be accomplished in various ways. For example, if the contribution is a novel architecture, describing the architecture fully might suffice, or if the contribution is a specific model and empirical evaluation, it may be necessary to either make it possible for others to replicate the model with the same dataset, or provide access to the model. In general. releasing code and data is often one good way to accomplish this, but reproducibility can also be provided via detailed instructions for how to replicate the results, access to a hosted model (e.g., in the case of a large language model), releasing of a model checkpoint, or other means that are appropriate to the research performed.
   - While NeurIPS does not require releasing code, the conference does require all submissions to provide some reasonable avenue for reproducibility, which may depend on the nature of the contribution. For example
     (a) If the contribution is primarily a new algorithm, the paper should make it clear how to reproduce that algorithm.
     (b) If the contribution is primarily a new model architecture, the paper should describe the architecture clearly and fully.
     (c) If the contribution is a new model (e.g., a large language model), then there should either be a way to access this model for reproducing the results or a way to reproduce the model (e.g., with an open-source dataset or instructions for how to construct the dataset).
     (d) We recognize that reproducibility may be tricky in some cases, in which case authors are welcome to describe the particular way they provide for reproducibility. In the case of closed-source models, it may be that access to the model is limited in some way (e.g., to registered users), but it should be possible for other researchers to have some path to reproducing or verifying the results.

5. **Open access to data and code**

Question: Does the paper provide open access to the data and code, with sufficient instructions to faithfully reproduce the main experimental results, as described in supplemental material?

Answer: [No]

Justification: Our data is not publicly available, but we plan to release the code.

Guidelines:

- The answer NA means that paper does not include experiments requiring code.
- Please see the NeurIPS code and data submission guidelines (`https://nips.cc/public/guides/CodeSubmissionPolicy`) for more details.
- While we encourage the release of code and data, we understand that this might not be possible, so "No" is an acceptable answer. Papers cannot be rejected simply for not including code, unless this is central to the contribution (e.g., for a new open-source benchmark).
- The instructions should contain the exact command and environment needed to run to reproduce the results. See the NeurIPS code and data submission guidelines (`https://nips.cc/public/guides/CodeSubmissionPolicy`) for more details.
- The authors should provide instructions on data access and preparation, including how to access the raw data, preprocessed data, intermediate data, and generated data, etc.
- The authors should provide scripts to reproduce all experimental results for the new proposed method and baselines. If only a subset of experiments are reproducible, they should state which ones are omitted from the script and why.
- At submission time, to preserve anonymity, the authors should release anonymized versions (if applicable).
- Providing as much information as possible in supplemental material (appended to the paper) is recommended, but including URLs to data and code is permitted.

6. **Experimental Setting/Details**

   Question: Does the paper specify all the training and test details (e.g., data splits, hyperparameters, how they were chosen, type of optimizer, etc.) necessary to understand the results?

   Answer: [Yes]

   Justification: The Method section provides a detailed description of the experimental setup, ensuring the results are understandable and meaningful.

   Guidelines:

   - The answer NA means that the paper does not include experiments.
   - The experimental setting should be presented in the core of the paper to a level of detail that is necessary to appreciate the results and make sense of them.
   - The full details can be provided either with the code, in appendix, or as supplemental material.

7. **Experiment Statistical Significance**

   Question: Does the paper report error bars suitably and correctly defined or other appropriate information about the statistical significance of the experiments?

   Answer: [Yes]

   Justification: This paper does not include experiments with error bars or statistical significance tests, but the Method section details the train/test split, initialization, and other experimental setups. Additionally, if necessary, description is included in the captions.

   Guidelines:

   - The answer NA means that the paper does not include experiments.
   - The authors should answer "Yes" if the results are accompanied by error bars, confidence intervals, or statistical significance tests, at least for the experiments that support the main claims of the paper.

- The factors of variability that the error bars are capturing should be clearly stated (for example, train/test split, initialization, random drawing of some parameter, or overall run with given experimental conditions).
- The method for calculating the error bars should be explained (closed form formula, call to a library function, bootstrap, etc.)
- The assumptions made should be given (e.g., Normally distributed errors).
- It should be clear whether the error bar is the standard deviation or the standard error of the mean.
- It is OK to report 1-sigma error bars, but one should state it. The authors should preferably report a 2-sigma error bar than state that they have a 96% CI, if the hypothesis of Normality of errors is not verified.
- For asymmetric distributions, the authors should be careful not to show in tables or figures symmetric error bars that would yield results that are out of range (e.g. negative error rates).
- If error bars are reported in tables or plots, The authors should explain in the text how they were calculated and reference the corresponding figures or tables in the text.

8. **Experiments Compute Resources**

   Question: For each experiment, does the paper provide sufficient information on the computer resources (type of compute workers, memory, time of execution) needed to reproduce the experiments?

   Answer: [Yes]

   Justification: The Method section of the paper provides sufficient details about the computer resources used.

   Guidelines:

   - The answer NA means that the paper does not include experiments.
   - The paper should indicate the type of compute workers CPU or GPU, internal cluster, or cloud provider, including relevant memory and storage.
   - The paper should provide the amount of compute required for each of the individual experimental runs as well as estimate the total compute.
   - The paper should disclose whether the full research project required more compute than the experiments reported in the paper (e.g., preliminary or failed experiments that didn't make it into the paper).

9. **Code Of Ethics**

   Question: Does the research conducted in the paper conform, in every respect, with the NeurIPS Code of Ethics `https://neurips.cc/public/EthicsGuidelines`?

   Answer: [Yes]

   Justification: The authors ensured anonymity and reviewed the NeurIPS Code of Ethics.

   Guidelines:

   - The answer NA means that the authors have not reviewed the NeurIPS Code of Ethics.
   - If the authors answer No, they should explain the special circumstances that require a deviation from the Code of Ethics.
   - The authors should make sure to preserve anonymity (e.g., if there is a special consideration due to laws or regulations in their jurisdiction).

10. **Broader Impacts**

    Question: Does the paper discuss both potential positive societal impacts and negative societal impacts of the work performed?

    Answer: [Yes]

    Justification: The paper addressed the societal impacts of the work throughout the Introduction and Discussion sections.

    Guidelines:

    - The answer NA means that there is no societal impact of the work performed.

- If the authors answer NA or No, they should explain why their work has no societal impact or why the paper does not address societal impact.
- Examples of negative societal impacts include potential malicious or unintended uses (e.g., disinformation, generating fake profiles, surveillance), fairness considerations (e.g., deployment of technologies that could make decisions that unfairly impact specific groups), privacy considerations, and security considerations.
- The conference expects that many papers will be foundational research and not tied to particular applications, let alone deployments. However, if there is a direct path to any negative applications, the authors should point it out. For example, it is legitimate to point out that an improvement in the quality of generative models could be used to generate deepfakes for disinformation. On the other hand, it is not needed to point out that a generic algorithm for optimizing neural networks could enable people to train models that generate Deepfakes faster.
- The authors should consider possible harms that could arise when the technology is being used as intended and functioning correctly, harms that could arise when the technology is being used as intended but gives incorrect results, and harms following from (intentional or unintentional) misuse of the technology.
- If there are negative societal impacts, the authors could also discuss possible mitigation strategies (e.g., gated release of models, providing defenses in addition to attacks, mechanisms for monitoring misuse, mechanisms to monitor how a system learns from feedback over time, improving the efficiency and accessibility of ML).

11. **Safeguards**

    Question: Does the paper describe safeguards that have been put in place for responsible release of data or models that have a high risk for misuse (e.g., pretrained language models, image generators, or scraped datasets)?

    Answer: [NA]

    Justification: This paper does not involve releasing data or models that carry a high risk of misuse.

    Guidelines:

    - The answer NA means that the paper poses no such risks.
    - Released models that have a high risk for misuse or dual-use should be released with necessary safeguards to allow for controlled use of the model, for example by requiring that users adhere to usage guidelines or restrictions to access the model or implementing safety filters.
    - Datasets that have been scraped from the Internet could pose safety risks. The authors should describe how they avoided releasing unsafe images.
    - We recognize that providing effective safeguards is challenging, and many papers do not require this, but we encourage authors to take this into account and make a best faith effort.

12. **Licenses for existing assets**

    Question: Are the creators or original owners of assets (e.g., code, data, models), used in the paper, properly credited and are the license and terms of use explicitly mentioned and properly respected?

    Answer: [Yes]

    Justification: The original papers were appropriately cited in the Introduction, Methods, and Results sections.

    Guidelines:

    - The answer NA means that the paper does not use existing assets.
    - The authors should cite the original paper that produced the code package or dataset.
    - The authors should state which version of the asset is used and, if possible, include a URL.
    - The name of the license (e.g., CC-BY 4.0) should be included for each asset.

- For scraped data from a particular source (e.g., website), the copyright and terms of service of that source should be provided.
- If assets are released, the license, copyright information, and terms of use in the package should be provided. For popular datasets, `paperswithcode.com/datasets` has curated licenses for some datasets. Their licensing guide can help determine the license of a dataset.
- For existing datasets that are re-packaged, both the original license and the license of the derived asset (if it has changed) should be provided.
- If this information is not available online, the authors are encouraged to reach out to the asset's creators.

13. **New Assets**

    Question: Are new assets introduced in the paper well documented and is the documentation provided alongside the assets?

    Answer: [NA]

    Justification: The paper does not release new assets.

    Guidelines:

    - The answer NA means that the paper does not release new assets.
    - Researchers should communicate the details of the dataset/code/model as part of their submissions via structured templates. This includes details about training, license, limitations, etc.
    - The paper should discuss whether and how consent was obtained from people whose asset is used.
    - At submission time, remember to anonymize your assets (if applicable). You can either create an anonymized URL or include an anonymized zip file.

14. **Crowdsourcing and Research with Human Subjects**

    Question: For crowdsourcing experiments and research with human subjects, does the paper include the full text of instructions given to participants and screenshots, if applicable, as well as details about compensation (if any)?

    Answer: [Yes]

    Justification: Complete instructions were provided to the human subjects included in our dataset.

    Guidelines:

    - The answer NA means that the paper does not involve crowdsourcing nor research with human subjects.
    - Including this information in the supplemental material is fine, but if the main contribution of the paper involves human subjects, then as much detail as possible should be included in the main paper.
    - According to the NeurIPS Code of Ethics, workers involved in data collection, curation, or other labor should be paid at least the minimum wage in the country of the data collector.

15. **Institutional Review Board (IRB) Approvals or Equivalent for Research with Human Subjects**

    Question: Does the paper describe potential risks incurred by study participants, whether such risks were disclosed to the subjects, and whether Institutional Review Board (IRB) approvals (or an equivalent approval/review based on the requirements of your country or institution) were obtained?

    Answer: [Yes]

    Justification: We obtained IRB approval and it is clearly stated in the Method section.

    Guidelines:

    - The answer NA means that the paper does not involve crowdsourcing nor research with human subjects.

- Depending on the country in which research is conducted, IRB approval (or equivalent) may be required for any human subjects research. If you obtained IRB approval, you should clearly state this in the paper.
- We recognize that the procedures for this may vary significantly between institutions and locations, and we expect authors to adhere to the NeurIPS Code of Ethics and the guidelines for their institution.
- For initial submissions, do not include any information that would break anonymity (if applicable), such as the institution conducting the review.