PETAL*face*: Parameter Efficient Transfer Learning for Low-resolution Face Recognition

Kartik Narayan[1], Nithin Gopalakrishnan Nair[1], Jennifer Xu[2], Rama Chellappa[1], Vishal M. Patel[1]

[1]Johns Hopkins University, [2]Systems and Technology Research

{knaraya4, ngopala2, rchella4, vpatel36}@jhu.edu, jennifer.xu@str.us

Abstract

Pre-training on large-scale datasets and utilizing margin-based loss functions have been highly successful in training models for high-resolution face recognition. However, these models struggle with low-resolution face datasets, in which the faces lack the facial attributes necessary for distinguishing different faces. Full fine-tuning on low-resolution datasets, a naive method for adapting the model, yields inferior performance due to catastrophic forgetting of pre-trained knowledge. Additionally the domain difference between high-resolution (HR) gallery images and low-resolution (LR) probe images in low resolution datasets leads to poor convergence for a single model to adapt to both gallery and probe after fine-tuning. To this end, we propose PETALface, a Parameter-Efficient Transfer Learning approach for low-resolution FACE recognition. Through PETALface, we attempt to solve both the aforementioned problems. (1) We solve catastrophic forgetting by leveraging the power of parameter efficient fine-tuning(PEFT). (2) We introduce two low-rank adaptation modules to the backbone, with weights adjusted based on the input image quality to account for the difference in quality for the gallery and probe images. To the best of our knowledge, PETALface is the first work leveraging the powers of PEFT for low resolution face recognition. Extensive experiments demonstrate that the proposed method outperforms full fine-tuning on low-resolution datasets while preserving performance on high-resolution and mixed-quality datasets, all while using only 0.48% of the parameters. The code and models will be made publicly available after the review process.

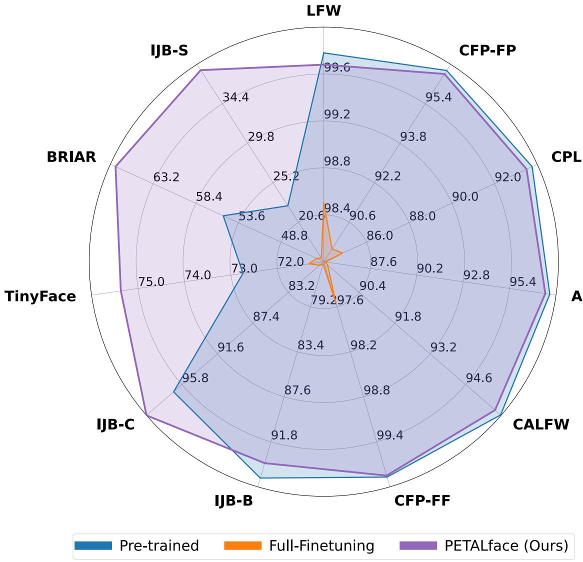
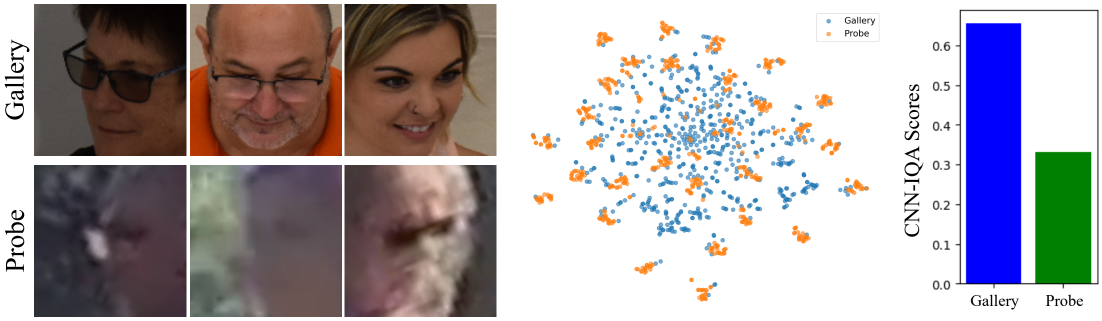


Figure 1. The proposed PETAL*face*: a parameter efficient transfer learning approach adapts to low-resolution datasets beating the performance of pre-trained models with negligible drop in performance on high-resolution and mixed-quality datasets. PETAL*face* enables development of generalized models achieving competitive performance on high-resolution (LFW, CFP-FP, CPLFW, AgeDB, CALFW, CFP-FF) and mixed-quality datasets (IJB-B, IJB-C) with big enhancements in low-quality surveillance quality datasets (TinyFace, BRIAR, IJB-S).

1. Introduction

Face recognition (FR) is one of the primal tasks in biometrics and has been extensively studied for decades due to its importance in device authentication, banking, finance, healthcare, social media, entertainment, retail, marketing, border control, security and surveillance. Early face recognition methods are evaluated using high-quality evaluation datasets and existing state-of-the-art face recognition methods have saturated these benchmarks, with several works achieving over 98% verification accuracy on high-resolution face recognition datasets like LFW [18], CFP-FP [44], CALFW [62] and AgeDB [42]. Recent efforts in face recognition [7, 10, 21] have shifted to low-quality face recognition because of its widespread use in surveillance-related applications. Moreover, analysis and generalizability of current methods in low-resolution face-datasets give



(a) (b) (c)

Figure 2. (a) An illustration of the gallery and probe images from low-resolution dataset (BRIAR). Gallery images usually are high quality compared to the probe images. (b) t-SNE plot for the gallery and probe images of the BRIAR dataset. (c) Average CNN-IQA scores of gallery and probe images for 50 identities of the BRIAR dataset.

a measure of the robustness of the recognition algorithm.

Low-resolution datasets [7, 26] contain images with poor clarity as shown in Figure 2(a), making it challenging to extract meaningful discriminative features essential for face recognition and verification. Common degradations include images with low resolution, compression artifacts, motion blur, occlusion, lighting variations, and atmospheric turbulence. Consequently, deep networks trained on high-resolution datasets perform poorly on low-resolution datasets. Moreover, low-resolution datasets are usually small, with a limited number of subjects, as curating them requires significant time, effort, and investment. There are very few low-resolution face recognition datasets that exist, most of which are private. Therefore, low-resolution face recognition remains an unsolved problem with significant room for improvement. Although, there has been a transition to datasets such as IJB-B [53] and IJB-C [40], and very-low resolution datasets like TinyFace [7], BRIAR [8] and IJB-S [26], research efforts remain focused on margin-based loss functions that create separable identity clusters on the hypersphere.

Existing methods [20, 30] force the learning of high-resolution and low-resolution face images in a single encoder, failing to account for the domain differences between them, which contradicts our belief. We claim that high-resolution and low-resolution images have distinct distributions and require separate encoders to extract meaningful features for classification. Figure 2(b) shows t-SNE visualization of the BRIAR dataset, where the clusters of gallery (high-resolution) and probe images (low-resolution) images are clearly separated, highlighting the domain difference between them. The bar plot shown in Figure 2(c) further supports this claim showing a clear difference in CNN-IQA scores between the gallery and probe images. This validates our claim that high-quality gallery images and low-quality probe images belong to distinct domains. A straightforward solution is to train two separate encoders for high-resolution and low-resolution data, but this creates misalignment in the embedding space as the two encoders do not share a common final layer.

A naive approach to adapting pre-trained models to low-resolution datasets is supervised full fine-tuning on these datasets. However, as mentioned, low-resolution datasets are small in size, and updating a model with a large number of parameters on a small low-resolution dataset results in poor convergence. This makes the model prone to catastrophic forgetting and we see a drop in performance on high-resolution and mixed-quality datasets. We illustrate this phenomenon and highlight the drop in performance in Figure 1. Existing methods perform poorly in low-resolution face recognition due to the following issues: 1) small training sets of low-resolution datasets, 2) domain differences between low-resolution and high-resolution data, and 3) catastrophic forgetting while fine-tuning for low-resolution datasets.

To address the above challenges, we propose a parameter-efficient transfer learning technique called PETAL*face*, which utilizes low-rank adaptation (LoRA) [17] of attention layers to adapt the pretrained model to low-resolution datasets. We introduce two low-rank adaptation modules that are constrained during training and act as separate proxy encoders for high-resolution and low-resolution data, respectively, with a common final embedding layer that helps avoid misalignment in the embedding space. The final output of the model depends on the weightage of these two modules, which is determined based on the image-quality scores of the input images. These scores are provided by an off-the-shelf NR-IQA network and passed to the model along with the images. The use of LoRA ensures that only a small number of parameters are added and trained, drastically reducing the training time. Low-rank adaptation preserves the feature extraction capabilities learned from the pre-training dataset and maintains performance on high-resolution and mixed-quality datasets, resulting in an efficient transfer to low-resolution datasets. The key contributions of our work are summarized below:

- We introduce the use of the LoRA-based PETL technique to adapt large pre-trained face-recognition models to low-resolution datasets.
- We propose an image-quality-based weighting of LoRA modules to create separate proxy encoders for high-resolution and low-resolution data, ensuring effective extraction of embeddings for face recognition.
- We demonstrate the superiority of PETAL*face* in adapting to low-resolution datasets, outperforming other state-of-the-art models on low-resolution benchmarks while maintaining performance on high-resolution and mixed-quality datasets.

2. Related Work

Face Recognition. Face recognition has significantly progressed from using hand-crafted features [1, 2] to utilizing deep learning models [11, 43, 50, 59]. Several works [9, 35, 51, 52] propose different variants of margin-based loss functions for face recognition that show impressive performance on high-resolution benchmarks [18, 42, 45]. However, much less attention has been given to low-resolution unconstrained face recognition benchmarks [7, 8, 26], which contain face images that are sometimes unidentifiable due to extreme degradations. To address this, some approaches [20, 30] incorporate adaptiveness in their training or loss functions to effectively leverage the low-quality images in large datasets [14, 64], based on the utility and quality of the low-resolution face images.

Low Resolution Face-Recognition. The main challenge in low-resolution face recognition is the domain difference between high-resolution gallery images captured in controlled environments and degraded probe images from surveillance cameras. [48, 56] use super-resolution (SR) models to upscale low-resolution images to high-resolution images to close the domain gap between gallery and probe images. However, several other works [23, 32, 58] suggest that this approach causes identity hallucination. Many studies [16, 47, 54, 55] have followed, relating recognition to visual quality. However, this is infeasible as it requires paired high-resolution and low-resolution images of the same subject, which are mostly unavailable in low-resolution datasets. [39, 63] use knowledge distillation to transfer knowledge from the high-resolution domain to the low-resolution domain. [13] utilizes a teacher-student configuration with help of synthetically degraded samples. [12] achieves cross-resolution distillation by employing an additional network between student and teacher network. [19] proposed distribution distillation loss and [37, 38] introduced augmentations to mitigate the performance gap between high-resolution and low-resolution samples. In our work, we adapt high-resolution model to low-resolution images by parameter efficient transfer learning, employing low-rank adaptation modules weighted based on the image quality of the input.

Parameter-Efficient Transfer Learning. Parameter-efficient transfer learning was initially introduced in the field of NLP [15, 17, 28, 31, 57]. It aims to achieve competitive performance with full fine-tuning by training only a small fraction of the total number of parameters. Recently, it has been adopted in the field of computer vision for various applications [5, 6, 22, 24]. VPT [22] appends learnable prompts to frozen transformer layer. Adapter [15] employs feedforward-down and feedforward-up blocks to adapt the pre-trained model. LoRA [17] leverages the low-rank nature of attention weights and performs matrix decomposition for parameter efficiency. Several variants of LoRA have been proposed since then. DyLoRA [49] truncates the up-projection and down-projection matrices in the objective, further reducing the number of trainable parameters. ResLoRA [46] adds a residual path for stable training. DoRA [34] decomposes pre-trained weight into magnitude and direction components, and efficiently updates the direction component. NOAH [60] and GLoRA [4] introduce Neural Architecture Search (NAS) to combine different methods. SSF [33] proposes a scale and shift learnable transformation on features of the pre-trained model. FacT [25] uses a tensorization-decomposition framework to break down the weight increments into lightweight factors. In our work, we use LoRA to fine-tune a pre-trained model on low-resolution face images, resulting in improved performance on low-resolution benchmarks while preserving the knowledge of the pre-trained model.

3. Proposed Work

In this section, we provide the necessary background on the sub-modules utilized in our method, namely LoRA [17], followed by a detailed explanation of the proposed fine-tuning procedure: PETAL*face*.

3.1. Low-Rank Adaptation

Low-rank adaptation (LoRA) [17], is a technique that was first introduced to adapt large language modules to low data regime, while retaining the original knowledge learned by a network. To achieve this, additional low-rank parameter modules are added in parallel to the pre-trained weights. During fine-tuning, the original pre-trained weights are kept frozen, and only the LoRA blocks are updated. For a pre-trained weight matrix $W_0 \in \mathbb{R}^{m \times n}$ of a dense layer in the network, LoRA appends a weight update $\Delta W \in \mathbb{R}^{m \times n}$, utilizing a low-rank decomposition

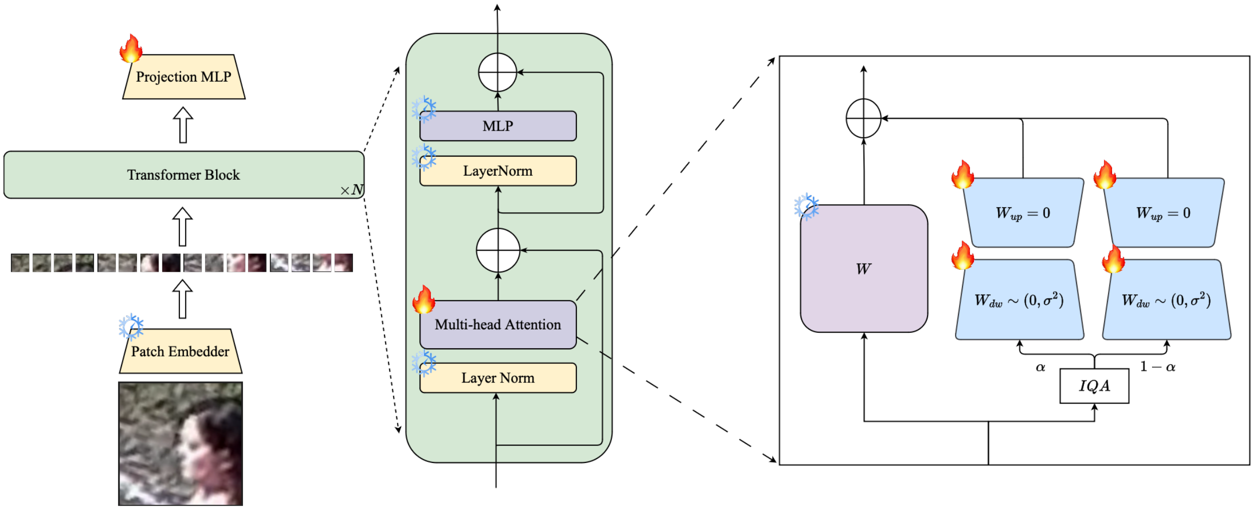


Figure 3. Overview of the proposed **PETAL*face***. We include an additional trainable module in linear layers present in attention layers and the final feature projection MLP. The trainable module is highlighted on the right. Specifically, we add two LoRA layers, where the weightage α is decided based on the input-image quality, computed using an off-the-shelf image quality assessment network (IQA).

such that $\Delta W = W_{up}W_{dw}$, where $W_{up} \in \mathbb{R}^{m\times r}$, $W_{dw} \in \mathbb{R}^{r\times n}$, and $r \ll \min(m, n)$. Here, r is the rank hyper-parameter which controls the bottleneck dimension of the low-rank decomposition. The output x_{out} of the dense layer with input x_{in} can be represented as:

$$x_{out} = W_0 x_{in} + \Delta W x_{in} = W_0 x_{in} + \alpha W_{up} W_{dw} x_{in}$$

Here α is a constant scale hyper-parameter. In the initial work [17], W_{up} matrix is initialized with zeros, and the W_{dw} matrix is initialized as a Gaussian distribution with zero mean and standard deviation $1/r$. The zero initialization ensures that during the start of the fine-tuning process, the default configuration corresponds to the pre-trained one. Therefore, any further training should improve the performance over the pre-trained results.

To the best of our knowledge, PETAL*face* is the first to explore parameter-efficient transfer learning methods, such as LoRA, for adapting to low-resolution face recognition datasets. A naive LoRA over a pre-trained transformer network trained for face recognition adds trainable parameters parallel to the attention layers, while the final layer that outputs the embeddings is shared. This shared layer enables the alignment of high-resolution and low-resolution features in the embedding space. It helps alleviate the issue of embedding misalignment that happens while training two separate encoders for gallery and probe. Further, LoRA can be treated as a plug-in module, which could be turned on or off, hence it preserves the pre-trained knowledge of models, performing on par with high-resolution and mixed-quality datasets and on plugging in improves the performance on low-resolution datasets. However, such a naive implementation of LoRA suffers a disadvantage that gallery and probe images share the same parameters even though in most cases gallery images are easier to recognize than probe images. Please refer to Figure **??**. Consequently, if common weights are fit to both datasets, this might lead to an average fit between probe and gallery images. To address this issue of domain difference between gallery and probe images, we propose PETAL*face*, which employs twin LoRA modules weighted based on the input image quality during training. This approach further boosts performance on low-resolution datasets while maintaining performance on high-resolution datasets.

3.2. PETAL*face*

PETAL*face* introduces a novel approach for adapting to low-resolution face images using two LoRA (Low-Rank Adaptation) blocks in each adaptation layer of the network. These blocks are constrained during training to ensure that one acts as a proxy encoder for high-resolution images and the other for low-resolution images. We achieve this by assigning different weights to these blocks based on the input image quality, effectively creating proxy encoders tailored to the quality of the input. This dynamic weight assignment allows PETAL*face* to better handle varying input qualities, enhancing overall performance. For PETAL*face*, we add two LoRA blocks parallel to the attention layer, which are weighted by a parameter in $(0, 1)$ depending on the input-image quality. The use of two LoRA blocks, along with the backbone network, enables meaningful extraction of features from both high-resolution and low-resolution images, which is difficult to achieve with a single encoder due to the domain difference, as previously discussed. Additionally, we add a LoRA block parallel to the last layer to ensure that the final embeddings are aligned even after adaptation to the domain of the low resolution dataset.

Specifically, let the two LoRA blocks parallel to a pre-trained weight matrix $W_0 \in \mathbb{R}^{m \times n}$ of a dense layer be $W_1 \in \mathbb{R}^{m \times n}$ and $W_2 \in \mathbb{R}^{m \times n}$. Given a batch of p input images $X = \{x_i \mid 0 \leq i < p\}$, PETAL*face* calculates the image quality score using an image quality estimator $\phi(x)$, represented as $Q = \{q_i \mid 0 \leq i < p \ni q_i = \phi(x_i), \forall\ 0 \leq i < p\}$. For each dataset we fine-tune on, we sample a random l number of samples $x_1, x_2, \ldots, x_l$ and calculate an estimate of the mean μ and the standard deviation σ for the quality score.

$$\mu = \frac{1}{l}\sum_{i=1}^{l}\phi(x_i), \quad \sigma = \sqrt{\frac{1}{l}\sum_{i=1}^{l}(\phi(x_i) - \mu)^2}$$

We set a threshold $t = \mu + \sigma$ for the whole dataset and then transform the quality scores q_i of each sample into weightage α_i for the LoRA blocks, using the following equation:

$$\alpha_i = \begin{cases} 0.5 & \text{if } q_i = t \\ 0.5 - (t - q_i) & \text{if } q_i < t \\ 0.5 + (q_i - t) & \text{if } q_i > t \end{cases}$$

This transformation regularizes the weightages per sample, ensuring that the weightages given to the LoRA blocks are continuous rather than discrete 0 and 1. It also stabilizes the training of PETAL*face* and leads to smooth convergence of the loss. The final output is given by:

$$x_{out} = W_0(x) + \alpha W_1(x) + (1 - \alpha)W_2(x)$$

Here, based on the weightage α,

$$\hat{W}(x) = W_0(x) + \alpha W_1(x) + (1 - \alpha)W_2(x)$$

acts as a proxy encoder for high-resolution images as well as low-resolution images. The proposed method allows for separate encoders for different resolutions to exist within a single backbone, differing only by a few low-rank parameters. This approach achieves the dual objectives of resolution-specific encoders and an aligned embedding space, enhancing performance in low-resolution face recognition and preserving pre-trained knowledge.

The low-rank blocks can be added in parallel at various locations. For finding the most suitable layers in a transformer based recognition network to add LoRA blocks, we tested different LoRA placements, as shown in Table 3, and chose the best performing configuration. Specifically, we found that the most effective layers are the attention (qkv) linear weights along with the final feature layer. Additionally, we ablated over various ranks for low-rank decomposition and set it to 8, which delivered superior performance. The rank of a LoRA block is generally set low because the attention matrix has an intrinsic low-rank [17], which also helps minimize the number of trainable parameters.

4. Experiments

4.1. Datasets

We employ WebFace4M and WebFace12M [64] as our pre-training datasets, which include about 4M and 12M million images, with approximately $205,000$ and $617,000$ distinct identities, respectively. To adapt the model to low-resolution images, we fine-tune it on the training sets of TinyFace [7] and BRIAR [8]. We evaluate the fine-tuned models on the test sets of TinyFace [7], IJB-S [26], and BRIAR [8], demonstrating the superiority of our proposed fine-tuning procedure. TinyFace [7] comprises $169,403$ low-resolution images of $5,139$ identities, with a training subset containing $7,804$ images of $2,570$ identities. IJB-S [26] is a surveillance video-based face dataset consisting of 398 videos and 202 identities. It is employed under three protocols: *Surveillance-to-Surveillance*, *Surveillance-to-Single*, and *Surveillance-to-Booking*. *Surveillance* refers to surveillance videos, *Single* indicates high-quality enrollment images, and *Booking* includes multiple enrollment images captured from various angles. The BRIAR [8] training set consists of $550,000$ images from 577 unique identities. For evaluation on BRIAR, we adhere to BRIAR Protocol 3.1 (face included treatment) [8]. This protocol includes a gallery of $86,958$ controlled images representing 615 identities, and a probe set comprising $5,435$ clips from $3,441$ unique field videos representing 260 identities. Additionally, we show that PETAL*face* adapts to low-resolution face images without forgetting the pre-trained knowledge by evaluating it on six high-resolution datasets: LFW [18], CFP-FP [44], CPLFW [61], AgeDB [42], CALFW [62], and CFP-FF [44], as well as two mixed-quality datasets: IJB-B [53] and IJBC [40].

4.2. Evaluation Setup & Metrics

To validate our proposed claims, we organize our experiments into two protocols. In **Protocol 1**, we fine-tune our models on the training set of TinyFace and evaluate them on its test set. Additionally, we evaluate the models on high-resolution and mixed-quality datasets. This protocol aims to highlight the capability of PETAL*face* to adapt to low-resolution datasets while maintaining performance on high-resolution and mixed-quality datasets. In **Protocol 2**, we fine-tune the models on the BRIAR dataset and evaluate them using BRIAR Protocol 3.1 and on IJB-S. We show that PETAL*face* performs better than full fine-tuning and naive LoRA. We evaluate the models on high-resolution and mixed-quality datasets using 1:1 verification accuracy and TAR@FAR at different thresholds, respectively. Rank retrieval (Rank-1, Rank-5, and Rank-10) is used for TinyFace. We report TAR@FAR at different thresholds and closed-set rank retrieval (Rank-1, Rank-5, and Rank-20) for BRIAR. For IJB-S, we report open-set TPIR@FPIR=1%/10% and

Training	**Loss**	**Dataset**	**Arch.**	**High-Resolution**						**Mixed-Quality**		**Low-resolution**		
				LFW [18]	CFP-FP [44]	CPLFW [61]	AgeDB [42]	CALFW [62]	CFP-FF [44]	IJB-B [53]	IJB-C [40]	TinyFace [7]		
				Verification Accuracy						TAR@FAR=0.01%		Rank-1	Rank-5	Rank-10
Pre-trained	CosFace [51]	WBF4M	R50	99.68	96.83	93.28	96.88	95.63	99.70	94.09	96.01	72.71	76.36	78.99
Pre-trained	ArcFace [9]	WBF4M	R50	99.67	96.71	93.41	96.81	95.71	99.75	94.02	95.99	73.04	76.85	79.45
Pre-trained	AdaFace [30]	WBF4M	R50	99.78	97.14	93.81	97.26	95.98	99.81	94.95	96.67	73.49	76.60	79.07
Pre-trained	CosFace [51]	WBF4M	ViT-B	99.73	97.30	94.31	97.51	95.95	99.87	95.18	96.87	73.57	76.95	78.94
Pre-trained	ArcFace [9]	WBF4M	ViT-B	99.82	97.23	93.68	97.53	95.91	99.80	94.91	96.64	72.74	76.28	78.13
Pre-trained	AdaFace [30]	WBF4M	ViT-B	99.76	97.00	93.75	96.85	95.71	99.80	94.90	96.52	74.03	77.22	79.37
Pre-trained	CosFace [51]	WBF4M	Swin-B	99.78	96.75	93.76	97.65	95.98	99.87	95.18	96.79	72.74	76.79	79.18
Pre-trained	ArcFace [9]	WBF4M	Swin-B	99.76	96.77	93.93	97.35	95.83	99.87	94.87	96.66	73.31	76.68	79.23
Full-FT	CosFace [51]	WBF4M	Swin-B	98.50	89.52	84.88	85.10	89.15	97.55	75.22	79.47	71.32	76.42	79.45
Full-FT	ArcFace [9]	WBF4M	Swin-B	98.31	88.94	84.00	83.45	88.33	97.14	71.84	76.10	71.11	76.63	79.96
LoRA	CosFace [51]	WBF4M	Swin-B	99.65	96.61	93.38	97.35	95.75	99.84	93.57	95.63	75.37	78.88	**82.02**
LoRA	ArcFace [9]	WBF4M	Swin-B	**99.73**	96.28	93.20	96.71	95.68	99.74	93.38	95.28	75.64	78.99	81.43
PETAL*face*	CosFace [51]	WBF4M	Swin-B	99.68	**96.61**	**93.50**	**97.40**	**95.76**	**99.85**	**93.79**	**95.67**	75.45	**79.05**	81.19
PETAL*face*	ArcFace [9]	WBF4M	Swin-B	99.66	96.37	93.18	96.45	95.61	99.80	93.29	95.27	**75.72**	78.86	81.70
PETAL*face*	ArcFace [9]	WBF12M	Swin-B	99.76	97.31	94.25	98.08	95.80	99.91	95.17	96.87	76.66	79.64	81.38

Table 1. Results of Protocol 1: The models are fine-tuned on train set of TinyFace and tested on several high-resolution, mixed-quality and TinyFace dataset. PETAL*face* adapts to the low-resolution data achieving SOTA results, preserving its performance on other datasets. [**BLUE**] indicates the best results for models trained on WebFace4M [64].

closed-set rank retrieval (Rank-1, Rank-5, and Rank-10).

4.3. Implementation Details

We re-trained all baseline models, consisting of configurations with different backbones (R50, ViT-B, and Swin-N) and loss functions (CosFace, ArcFace, and AdaFace). To ensure a fair comparison, we tested all models on the same cropped and aligned test sets. We fine-tuned the models using the AdamW optimizer with a weight decay of 0.1. A Polynomial LR scheduler was employed with an initial learning rate of $4e^{-5}$ during training. We sampled $l = 1000$ images from the fine-tuning dataset to calculate the mean μ and variance σ, which were used to determine the threshold t for calculating weightages for the LoRA blocks. When fine-tuning on TinyFace, we utilized 2 warm-up epochs and trained the model for 40 epochs with a batch size of 8. For BRIAR, we used 1 warm-up epoch and fine-tuned for 10 epochs with the same batch size. We applied a rank of 8 for low-rank decomposition on TinyFace and 32 for BRIAR. This difference is due to TinyFace having a relatively smaller training set of approximately $\approx$ 7000 images, compared to BRIAR's $\approx$ 300k images. The larger rank for BRIAR increases the number of trainable parameters to accommodate the larger train set. We employ CNN-IQA [27]as our NR-IQA model to classify the images as low-resolution or high-resolution. All code was written in PyTorch, and the models were trained on eight A5000 GPUs, each with 24GB of memory. The detailed implementation is provided in the supplementary document.

5. Results

In this section, we showcase PETAL*face*'s superiority in transferring to low-resolution datasets maintaining competitive performance on high-resolution and mixed-quality datasets, and compare it with other baselines. We also analyse the benefits of the proposed fine-tuning procedure.

5.1. Results on Tinyface Dataset-(Protocol-1)

The results for Protocol-1 are summarized in Table 1. From the pre-trained models, we observe that different loss functions and backbone architectures result in only minor differences in final performance. We choose the Swin-B [36] architecture for our experiments due to its ability to adapt to out-of-domain distributions [29]. We select ArcFace [9] for all our experiments as it shows better performance when coupled with Swin-B. Full fine-tuning of pre-trained face recognition models does not lead to performance improvement; instead, we observe a performance decrease from 73.31 to 71.11. Additionally, the performance on high-resolution and mixed-quality datasets also dropped after adaptaion to the low resolution dataset, as can be seen from Table 1. When a model is fully fine-tuned for low-resolution face recognition, it is typically pre-trained on large datasets with millions of identities and then updated based on low-resolution datasets with only a few hundred identities. Due to the domain differences between low-resolution and high-resolution images, the model encounters large gradient updates initially, deviating from the original pre-trained weights suitable for recognition over a large collection of images. This can lead to poor convergence as can be seen from the full fine-tuning results in Table1. These large gradient updates result in catastrophic forgetting of the pre-trained knowledge, explaining the performance drop for high-resolution and mixed-quality datasets.

PETAL*face* addresses the problem of catastrophic forgetting and achieves rank-retrieval accuracies of 75.72%, 78.86%, and 81.70% for rank-1, rank-5, and rank-10, respectively. It significantly boosts the performance of pre-trained models while maintaining performance on high-

Training	Dataset	Arch.	BRIAR Protocol 3.1 [8]						IJB-S (Surveillance to Surveillance) [26]				
			TAR@FAR			Rank Retrieval			TPIR@FPIR		Rank Retrieval		
			0.01%	0.1%	1%	Rank-1	Rank-5	Rank-20	1%	10%	Rank-1	Rank-5	Rank-10
Pre-trained	WBF4M [64]	R50	22.55	35.43	52.20	45.43	54.54	65.13	3.67	9.09	33.62	49.40	54.92
Pre-trained	WBF4M [64]	ViT-B	34.29	47.41	62.81	55.44	64.32	73.46	2.58	8.12	25.76	40.69	47.15
Pre-trained	WBF4M [64]	Swin-B	33.77	45.93	61.17	55.31	63.29	72.76	2.11	7.45	22.52	37.97	44.93
Full-FT	WBF4M [64]	Swin-B	11.62	29.68	58.66	44.81	59.88	74.73	1.72	5.95	16.44	31.58	38.65
PETAL*face*	WBF4M [64]	Swin-B	**35.12**	**55.35**	**75.43**	**67.42**	**76.74**	**85.20**	**12.25**	**25.28**	**38.32**	**51.50**	**57.05**
PETAL*face*	WBF12M [64]	Swin-B	44.29	63.01	81.86	74.49	82.82	90.12	15.28	30.40	42.30	54.33	58.31

Table 2. Results of Protocol 2: The models are fine-tuned on the BRIAR dataset and tested using BRIAR Protocol 3.1 and the IJB-S dataset. [**BLUE**] indicates the best results for models trained on WebFace4M [64].

resolution and mixed-quality datasets. We attribute this improvement to the twin low-rank modules added parallel to the attention weights, which are weighted adaptively based on input image quality and extract meaningful features based on the quality of image. The two low-rank modules serve as proxy encoders for high-resolution and low-resolution data, respectively. The adaptive LoRA modules perform better than static LoRA modules, with a performance increase from 75.64% to 75.72%. Furthermore, as we fine-tune the model by adding weights parallel to the model, they still share the common final embedding layer. This ensures that the feature space for high-resolution and low-resolution data is aligned. As shown in Table 3, we gain a further boost in performance by adding a LoRA module parallel to the final projection MLP. The LoRA module in the projection MLP layer ensures that the embedding space stays aligned by adjusting according to the weight updates of the LoRA modules parallel to the attention layer. Additionally, the low-rank decomposition keeps the trainable parameters to a minimum and makes the fine-tuning process efficient. Finally, we observe that the proposed approach provides better performance metrics when the pre-training dataset is scaled from WebFace4M to WebFace12M, with rank-1 accuracy increasing from 75.72% to 76.66%.

5.2. Results on BRIAR and IJB-S datasets-(Protocol-2)

With this protocol, we aim to highlight the effectiveness of the proposed PETAL*face* on datasets that have a clear domain difference between gallery (high-resolution) and probe (low-resolution) images. The samples in the TinyFace dataset have similar distributions of gallery and probe images, with a mean CNN-IQA [27] score of 60.26 and a standard deviation of approximately 15.44. In contrast, the BRIAR and IJB-S datasets have samples with CNN-IQA scores ranging from 20 to 90. This highlights that IJB-S and BRIAR are more challenging datasets, demanding a better feature extractor. The results of this protocol are summarized in Table 2. PETAL*face* shows significant improvements in performance on BRIAR, with a FAR of 35.12, 55.35, and 75.43 at TAR of 0.01%, 0.1%, and 1%, respectively. It achieves a rank-1 accuracy of 67.42%, which is a phenomenal 12.11% improvement. The same trend follows for rank-5 and rank-20 accuracies, with improvements of 13.45% and 12.44%, respectively. Again, we see that full-finetuning does not lead to performance improvements, as discussed in Section 5.1. The large gradient updates in the initial iterations lead to poor convergence. However, PETAL*face* provides significant improvements because of the separate proxy encoders for high-resolution and low-resolution images. It provides meaningful discriminative features for both domains, and the results reiterate the same.

We validate the generalization ability of the proposed PETAL*face* by evaluating it on the IJB-S dataset. We fine-tune the models on the BRIAR train set and test them on IJB-S to gauge the generalization of PETAL*face*. It provides significant improvements in TPIR and rank-retrieval accuracies. It achieves a TPIR of 12.25% and 25.28% at FPIRs of 1% and 10%, respectively. The rank-1, rank-5, and rank-10 retrieval accuracies are 38.32%, 51.50%, and 57.05%, respectively. We also see improvements in the *Surveillance-to-Single* and *Surveillance-to-Booking* evaluation settings of the IJB-S dataset, whose results are included in the supplementary document. One common observation across both datasets is that performance improves as we scale up the pre-training dataset size.

6. Ablation Studies

We conduct all ablation studies using a Swin-B model trained on the WebFace4M dataset with CosFace loss. For these experiments, we used static LoRA modules instead of the adaptive LoRA model. **Effect of applying LoRA to different layers in the network:** We experimented with adding low-rank decomposition modules at various position within the transformer block. [17] proposed adding LoRA parallel to the attention layers. From our experiments shown in Table 3, we observe that adding LoRA to the final

Layers	**TinyFace [7]**			**Total Model Params**	**Trainable Params**
	Rank-1	**Rank-5**	**Rank-10**		
Pretrained	72.74	76.79	79.18	213.67M	213.67M
Full Finetuning	71.32	76.42	79.45	213.67M	213.67M
Attention	75.59	78.83	82.13	214.23 M	730k
Attention + MLP + Proj + Feature	74.89	78.64	81.35	216.24M	2737k
Attention + MLP + Proj + Patch Reduction + Feature	75.16	78.64	81.59	215.96M	2455k
Attention + MLP + Feature	75.21	78.72	81.06	215.96M	2455k
Attention + Feature	**75.64**	**78.86**	**81.59**	**214.54M**	**1041k**

Table 3. Performance of Swin-B models when LoRA is added at different positions in the transformer network.

Rank	**TinyFace [7]**			**Total Model Params**	**Trainable Params**
	Rank-1	**Rank-5**	**Rank-10**		
2	75.61	79.02	81.73	213.88M	384k
4	75.56	79.05	81.59	214.10M	603k
8	**75.64**	**78.86**	**81.59**	**214.54M**	**1041k**
16	75.26	79.15	81.46	215.41M	1918k
32	75.45	78.94	81.81	217.17M	3671k
64	75.05	78.72	81.08	220.67M	7177k
128	75.24	78.70	81.22	227.69M	14.19M

Table 4. Performance of Swin-B models fine-tuned using LoRA modules of varying ranks.

Training	**TinyFace [7]**		
	Rank-1	**Rank-5**	**Rank-10**
ViT Backbone with CosFace [51] Loss Function			
Pretrained	73.57	76.95	78.94
Full Finetuning	71.08	76.09	79.42
LoRA	73.92	77.11	79.15
PETAL*face*	**74.14**	**77.22**	**79.56**
Ablation using different IQA networks			
Pretrained	72.74	76.79	79.18
Full Finetuning	71.32	76.42	79.45
PETAL*face* (BRISQUE) [41]	75.16	78.46	80.90
PETAL*face* (CR-FIQA) [3]	75.34	78.75	81.30
PETAL*face* (CNN-IQA) [27]	**75.64**	**78.86**	**81.59**

Table 5. Results using ViT Backbone and PETAL*face* performance using different image quality estimators.

feature layer along with the attention layer leads to superior performance. This adjustment in the final feature layers help align the extracted features based on the updated attention layers, resulting in better overall performance. Moreover, we don't see a drastic increase in the number of trainable parameters, which increased from 730k to 1041k, representing only a 0.48% increase of total parameters. **Effect of LoRA rank on performance:** We ablate over different ranks for low-rank decomposition, with the results summarized in Table 4. Our findings indicate that rank-8 yields the best performance, and thus we adopted this rank for training all our models. [17] shows that the attention matrix has an intrinsic low rank, often resulting in better performance with smaller ranks. This is corroborated by the results in Table 4, where ranks 2, 4, and 8 outperform ranks 32, 64, and 128. **Effect of different backbones on performance:** We conduct experiments with various backbones to demonstrate the broad applicability of PETAL*face*. The results, summarized in Table 5, show that PETAL*face* with the ViT backbone follows a similar trend, outperforming full finetuning of the models. **Effect of Image quality assessment on performance:** We experimented with two lightweight NR-IQA models: BRISQUE [41] and CNN-IQA [27], and one face image quality assessment network CR-FIQA [3]. We present the corresponding results in Table 5. The performance on TinyFace when using BRISQUE as the IQA yielded rank-1, rank-5, and rank-10 retrieval accuracies of 75.16%, 78.46%, and 80.90%, respectively. While CR-FIQA [3] showed competitive results, its performance was slightly lower than CNN-IQA, likely because it is trained on MS1MV2 [9] that doesn't contain diverse range of degradation that are present in challenging evaluation datasets like TinyFace, BRIAR and IJB-S. CNN-IQA demonstrated superior performance, leading us to select CNN-IQA as our IQA for all subsequent experiments. We emphasize that this boost in performance is due to the robustness of a CNN-IQA method arising due to its training.

7. Limitation and Future work

As described in section 3, PETAL*face* utilizes an off the shelf IQA module to model the parameter α defining the strength of the chosen LoRA module. However, most image quality assessment networks are not accurate. Hence, research on better IQA models would enable a boost in the performance of PETAL*face*. Moreover, we define a manually selected heuristic for the choice of parameter α for choosing the LoRA module. However, one may perform a more sophisticated heuristic selection by a parameter sweep over a validation set. We leave these challenges as open problems to be addressed in future work.

8. Conclusion

In this paper, we propose PETAL*face*, a new method that harnesses the power of parameter-efficient fine-tuning to address the challenging problem of low-resolution face recognition. To achieve this, we introduce a novel image quality assessment based twin LORA module, which significantly enhances the model's ability to handle varying image qualities. By adopting this design choice, we effectively tackle two major issues prevalent in existing works: catastrophic forgetting and the domain difference between gallery and probe images. We conduct extensive experiments across multiple benchmarks on low-resolution datasets and achieve state-of-the-art results across various metrics. Notably, our approach also preserves performance on high-resolution and mixed-quality datasets. Models finetuned using PETAL*face* demonstrates versatility and can serve as a generalized model capable of handling a wide range of image resolutions, making it highly suitable for real-world deployment and practical applications.

References

[1] Timo Ahonen, Abdenour Hadid, and Matti Pietikainen. Face description with local binary patterns: Application to face recognition. *IEEE transactions on pattern analysis and machine intelligence*, 28(12):2037–2041, 2006. 3

[2] Peter N. Belhumeur, Joao P Hespanha, and David J. Kriegman. Eigenfaces vs. fisherfaces: Recognition using class specific linear projection. *IEEE Transactions on pattern analysis and machine intelligence*, 19(7):711–720, 1997. 3

[3] Fadi Boutros, Meiling Fang, Marcel Klemt, Biying Fu, and Naser Damer. Cr-fiqa: face image quality assessment by learning sample relative classifiability. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 5836–5845, 2023. 8

[4] Arnav Chavan, Zhuang Liu, Deepak Gupta, Eric Xing, and Zhiqiang Shen. One-for-all: Generalized lora for parameter-efficient fine-tuning. *arXiv preprint arXiv:2306.07967*, 2023. 3

[5] Shoufa Chen, Chongjian Ge, Zhan Tong, Jiangliu Wang, Yibing Song, Jue Wang, and Ping Luo. Adaptformer: Adapting vision transformers for scalable visual recognition. *Advances in Neural Information Processing Systems*, 35:16664–16678, 2022. 3

[6] Z Chen, Y Duan, W Wang, J He, T Lu, J Dai, and Y Qiao. Vision transformer adapter for dense predictions. arxiv 2022. *arXiv preprint arXiv:2205.08534*, 4, 2022. 3

[7] Zhiyi Cheng, Xiatian Zhu, and Shaogang Gong. Low-resolution face recognition. In *Computer Vision–ACCV 2018: 14th Asian Conference on Computer Vision, Perth, Australia, December 2–6, 2018, Revised Selected Papers, Part III 14*, pages 605–621. Springer, 2019. 1, 2, 3, 5, 6, 8

[8] David Cornett, Joel Brogan, Nell Barber, Deniz Aykac, Seth Baird, Nicholas Burchfield, Carl Dukes, Andrew Duncan, Regina Ferrell, Jim Goddard, et al. Expanding accurate person recognition to new altitudes and ranges: The briar dataset. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*, pages 593–602, 2023. 2, 3, 5, 7

[9] Jiankang Deng, Jia Guo, Niannan Xue, and Stefanos Zafeiriou. Arcface: Additive angular margin loss for deep face recognition. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 4690–4699, 2019. 3, 6, 8

[10] Muskan Dosi, Chiranjeev Chiranjeev, Shivang Agarwal, Jyoti Chaudhary, Sunny Manchanda, Kavita Balutia, Kaushik Bhagwatkar, Mayank Vatsa, and Richa Singh. Seg-dgdnet: Segmentation based disguise guided dropout network for low resolution face recognition. *IEEE Journal of Selected Topics in Signal Processing*, 17(6):1264–1276, 2023. 1

[11] Yueqi Duan, Jiwen Lu, and Jie Zhou. Uniformface: Learning deep equidistributed representation for face recognition. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 3415–3424, 2019. 3

[12] Shiming Ge, Kangkai Zhang, Haolin Liu, Yingying Hua, Shengwei Zhao, Xin Jin, and Hao Wen. Look one and more: Distilling hybrid order relational knowledge for cross-resolution image recognition. In *Proceedings of the AAAI conference on artificial intelligence*, volume 34, pages 10845–10852, 2020. 3

[13] Shiming Ge, Shengwei Zhao, Chenyu Li, and Jia Li. Low-resolution face recognition in the wild via selective knowledge distillation. *IEEE Transactions on Image Processing*, 28(4):2051–2062, 2018. 3

[14] Yandong Guo, Lei Zhang, Yuxiao Hu, Xiaodong He, and Jianfeng Gao. Ms-celeb-1m: A dataset and benchmark for large-scale face recognition. In *Computer Vision–ECCV 2016: 14th European Conference, Amsterdam, The Netherlands, October 11-14, 2016, Proceedings, Part III 14*, pages 87–102. Springer, 2016. 3

[15] Neil Houlsby, Andrei Giurgiu, Stanislaw Jastrzebski, Bruna Morrone, Quentin De Laroussilhe, Andrea Gesmundo, Mona Attariyan, and Sylvain Gelly. Parameter-efficient transfer learning for nlp. In *International conference on machine learning*, pages 2790–2799. PMLR, 2019. 3

[16] Chih-Chung Hsu, Chia-Wen Lin, Weng-Tai Su, and Gene Cheung. Sigan: Siamese generative adversarial network for identity-preserving face hallucination. *IEEE Transactions on Image Processing*, 28(12):6225–6236, 2019. 3

[17] 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. 2, 3, 4, 5, 7, 8

[18] Gary B Huang, Marwan Mattar, Tamara Berg, and Eric Learned-Miller. Labeled faces in the wild: A database forstudying face recognition in unconstrained environments. In *Workshop on faces in'Real-Life'Images: detection, alignment, and recognition*, 2008. 1, 3, 5, 6

[19] Yuge Huang, Pengcheng Shen, Ying Tai, Shaoxin Li, Xiaoming Liu, Jilin Li, Feiyue Huang, and Rongrong Ji. Improving face recognition from hard samples via distribution distillation loss. In *Computer Vision–ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part XXX 16*, pages 138–154. Springer, 2020. 3

[20] Yuge Huang, Yuhan Wang, Ying Tai, Xiaoming Liu, Pengcheng Shen, Shaoxin Li, Jilin Li, and Feiyue Huang. Curricularface: adaptive curriculum learning loss for deep face recognition. In *proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 5901–5910, 2020. 2, 3

[21] Bhavin Jawade, Deen Dayal Mohan, Dennis Fedorishin, Srirangaraj Setlur, and Venu Govindaraju. Conan: Conditional neural aggregation network for unconstrained face feature fusion. In *2023 IEEE International Joint Conference on Biometrics (IJCB)*, pages 1–10. IEEE, 2023. 1

[22] Menglin Jia, Luming Tang, Bor-Chun Chen, Claire Cardie, Serge Belongie, Bharath Hariharan, and Ser-Nam Lim. Visual prompt tuning. In *European Conference on Computer Vision*, pages 709–727. Springer, 2022. 3

[23] Junjun Jiang, Yi Yu, Jinhui Hu, Suhua Tang, and Jiayi Ma. Deep cnn denoiser and multi-layer neighbor component embedding for face hallucination. *arXiv preprint arXiv:1806.10726*, 2018. 3

[24] Shibo Jie and Zhi-Hong Deng. Convolutional bypasses are better vision transformer adapters. *arXiv preprint arXiv:2207.07039*, 2022. 3

[25] Shibo Jie and Zhi-Hong Deng. Fact: Factor-tuning for lightweight adaptation on vision transformer. In *Proceedings of the AAAI conference on artificial intelligence*, volume 37, pages 1060–1068, 2023. 3

[26] Nathan D Kalka, Brianna Maze, James A Duncan, Kevin O'Connor, Stephen Elliott, Kaleb Hebert, Julia Bryan, and Anil K Jain. Ijb–s: Iarpa janus surveillance video benchmark. In *2018 IEEE 9th international conference on biometrics theory, applications and systems (BTAS)*, pages 1–9. IEEE, 2018. 2, 3, 5, 7

[27] Le Kang, Peng Ye, Yi Li, and David Doermann. Convolutional neural networks for no-reference image quality assessment. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 1733–1740, 2014. 6, 7, 8

[28] Rabeeh Karimi Mahabadi, James Henderson, and Sebastian Ruder. Compacter: Efficient low-rank hypercomplex adapter layers. *Advances in Neural Information Processing Systems*, 34:1022–1035, 2021. 3

[29] Donghyun Kim, Kaihong Wang, Stan Sclaroff, and Kate Saenko. A broad study of pre-training for domain generalization and adaptation. In *European Conference on Computer Vision*, pages 621–638. Springer, 2022. 6

[30] Minchul Kim, Anil K Jain, and Xiaoming Liu. Adaface: Quality adaptive margin for face recognition. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 18750–18759, 2022. 2, 3, 6

[31] Brian Lester, Rami Al-Rfou, and Noah Constant. The power of scale for parameter-efficient prompt tuning. *arXiv preprint arXiv:2104.08691*, 2021. 3

[32] Pei Li, Loreto Prieto, Domingo Mery, and Patrick J Flynn. On low-resolution face recognition in the wild: Comparisons and new techniques. *IEEE Transactions on Information Forensics and Security*, 14(8):2000–2012, 2019. 3

[33] Dongze Lian, Daquan Zhou, Jiashi Feng, and Xinchao Wang. Scaling & shifting your features: A new baseline for efficient model tuning. *Advances in Neural Information Processing Systems*, 35:109–123, 2022. 3

[34] Shih-Yang Liu, Chien-Yi Wang, Hongxu Yin, Pavlo Molchanov, Yu-Chiang Frank Wang, Kwang-Ting Cheng, and Min-Hung Chen. Dora: Weight-decomposed low-rank adaptation. *arXiv preprint arXiv:2402.09353*, 2024. 3

[35] Weiyang Liu, Yandong Wen, Zhiding Yu, Ming Li, Bhiksha Raj, and Le Song. Sphereface: Deep hypersphere embedding for face recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 212–220, 2017. 3

[36] Ze Liu, Yutong Lin, Yue Cao, Han Hu, Yixuan Wei, Zheng Zhang, Stephen Lin, and Baining Guo. Swin transformer: Hierarchical vision transformer using shifted windows. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 10012–10022, 2021. 6

[37] Cheng-Yaw Low and Andrew Beng-Jin Teoh. An implicit identity-extended data augmentation for low-resolution face representation learning. *IEEE Transactions on Information Forensics and Security*, 17:3062–3076, 2022. 3

[38] Cheng-Yaw Low, Andrew Beng-Jin Teoh, and Jaewoo Park. Mind-net: A deep mutual information distillation network for realistic low-resolution face recognition. *IEEE Signal Processing Letters*, 28:354–358, 2021. 3

[39] Fabio Valerio Massoli, Giuseppe Amato, and Fabrizio Falchi. Cross-resolution learning for face recognition. *Image and Vision Computing*, 99:103927, 2020. 3

[40] Brianna Maze, Jocelyn Adams, James A Duncan, Nathan Kalka, Tim Miller, Charles Otto, Anil K Jain, W Tyler Niggel, Janet Anderson, Jordan Cheney, et al. Iarpa janus benchmark-c: Face dataset and protocol. In *2018 international conference on biometrics (ICB)*, pages 158–165. IEEE, 2018. 2, 5, 6

[41] Anish Mittal, Anush Krishna Moorthy, and Alan Conrad Bovik. No-reference image quality assessment in the spatial domain. *IEEE Transactions on image processing*, 21(12):4695–4708, 2012. 8

[42] Stylianos Moschoglou, Athanasios Papaioannou, Christos Sagonas, Jiankang Deng, Irene Kotsia, and Stefanos Zafeiriou. Agedb: the first manually collected, in-the-wild age database. In *proceedings of the IEEE conference on computer vision and pattern recognition workshops*, pages 51–59, 2017. 1, 3, 5, 6

[43] Florian Schroff, Dmitry Kalenichenko, and James Philbin. Facenet: A unified embedding for face recognition and clustering. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2015. 3

[44] Soumyadip Sengupta, Jun-Cheng Chen, Carlos Castillo, Vishal M Patel, Rama Chellappa, and David W Jacobs. Frontal to profile face verification in the wild. In *2016 IEEE winter conference on applications of computer vision (WACV)*, pages 1–9. IEEE, 2016. 1, 5, 6

[45] S. Sengupta, J.C. Cheng, C.D. Castillo, V.M. Patel, R. Chellappa, and D.W. Jacobs. Frontal to profile face verification in the wild. In *IEEE Conference on Applications of Computer Vision*, February 2016. 3

[46] Shuhua Shi, Shaohan Huang, Minghui Song, Zhoujun Li, Zihan Zhang, Haizhen Huang, Furu Wei, Weiwei Deng, Feng Sun, and Qi Zhang. Reslora: Identity residual mapping in low-rank adaption. *arXiv preprint arXiv:2402.18039*, 2024. 3

[47] Maneet Singh, Shruti Nagpal, Richa Singh, and Mayank Vatsa. Derivenet for (very) low resolution image classification. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 44(10):6569–6577, 2021. 3

[48] Maneet Singh, Shruti Nagpal, Mayank Vatsa, Richa Singh, and Angshul Majumdar. Identity aware synthesis for cross resolution face recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition workshops*, pages 479–488, 2018. 3

[49] Mojtaba Valipour, Mehdi Rezagholizadeh, Ivan Kobyzev, and Ali Ghodsi. Dylora: Parameter efficient tuning of pretrained models using dynamic search-free low-rank adaptation. *arXiv preprint arXiv:2210.07558*, 2022. 3

[50] Feng Wang, Xiang Xiang, Jian Cheng, and Alan Loddon Yuille. Normface: L2 hypersphere embedding for face verification. In *Proceedings of the 25th ACM international conference on Multimedia*, pages 1041–1049, 2017. 3

[51] Hao Wang, Yitong Wang, Zheng Zhou, Xing Ji, Dihong Gong, Jingchao Zhou, Zhifeng Li, and Wei Liu. Cosface: Large margin cosine loss for deep face recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 5265–5274, 2018. 3, 6, 8

[52] Yandong Wen, Kaipeng Zhang, Zhifeng Li, and Yu Qiao. A discriminative feature learning approach for deep face recognition. In *Computer vision–ECCV 2016: 14th European conference, amsterdam, the netherlands, October 11–14, 2016, proceedings, part VII 14*, pages 499–515. Springer, 2016. 3

[53] Cameron Whitelam, Emma Taborsky, Austin Blanton, Brianna Maze, Jocelyn Adams, Tim Miller, Nathan Kalka, Anil K Jain, James A Duncan, Kristen Allen, et al. Iarpa janus benchmark-b face dataset. In *proceedings of the IEEE conference on computer vision and pattern recognition workshops*, pages 90–98, 2017. 2, 5, 6

[54] Xi Yin, Ying Tai, Yuge Huang, and Xiaoming Liu. Fan: Feature adaptation network for surveillance face recognition and normalization. In *Proceedings of the Asian Conference on Computer Vision*, 2020. 3

[55] Xin Yu, Basura Fernando, Richard Hartley, and Fatih Porikli. Super-resolving very low-resolution face images with supplementary attributes. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 908–917, 2018. 3

[56] Linwei Yue, Huanfeng Shen, Jie Li, Qiangqiang Yuan, Hongyan Zhang, and Liangpei Zhang. Image super-resolution: The techniques, applications, and future. *Signal processing*, 128:389–408, 2016. 3

[57] Elad Ben Zaken, Shauli Ravfogel, and Yoav Goldberg. Bitfit: Simple parameter-efficient fine-tuning for transformer-based masked language-models. *arXiv preprint arXiv:2106.10199*, 2021. 3

[58] Kaipeng Zhang, Zhanpeng Zhang, Chia-Wen Cheng, Winston H Hsu, Yu Qiao, Wei Liu, and Tong Zhang. Super-identity convolutional neural network for face hallucination. In *Proceedings of the European conference on computer vision (ECCV)*, pages 183–198, 2018. 3

[59] Xiao Zhang, Rui Zhao, Yu Qiao, Xiaogang Wang, and Hongsheng Li. Adacos: Adaptively scaling cosine logits for effectively learning deep face representations. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 10823–10832, 2019. 3

[60] Yuanhan Zhang, Kaiyang Zhou, and Ziwei Liu. Neural prompt search. *arXiv preprint arXiv:2206.04673*, 2022. 3

[61] Tianyue Zheng and Weihong Deng. Cross-pose lfw: A database for studying cross-pose face recognition in unconstrained environments. *Beijing University of Posts and Telecommunications, Tech. Rep*, 5(7):5, 2018. 5, 6

[62] Tianyue Zheng, Weihong Deng, and Jiani Hu. Cross-age lfw: A database for studying cross-age face recognition in unconstrained environments. *arXiv preprint arXiv:1708.08197*, 2017. 1, 5, 6

[63] Mingjian Zhu, Kai Han, Chao Zhang, Jinlong Lin, and Yunhe Wang. Low-resolution visual recognition via deep feature distillation. In *ICASSP 2019-2019 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 3762–3766. IEEE, 2019. 3

[64] Zheng Zhu, Guan Huang, Jiankang Deng, Yun Ye, Junjie Huang, Xinze Chen, Jiagang Zhu, Tian Yang, Jiwen Lu, Dalong Du, et al. Webface260m: A benchmark unveiling the power of million-scale deep face recognition. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 10492–10502, 2021. 3, 5, 6, 7

PETAL*face*: Parameter Efficient Transfer Learning for Low-resolution Face Recognition
Supplementary Document

The supplementary document is organized into the following sections. First, we discuss additional implementation details. Next, we present the results on other evaluation settings of the IJB-S dataset. Also, we present a gradient analysis of PETAL*face* and compare it to full fine-tuning to highlight that the proposed approach leads to stable convergence. Finally, we provide a failure case analysis of PETAL*face*.

A. Implementation Details

All deployment codes were implemented in PyTorch framework and executed it on eight A5000 GPUs, each equipped with 24GB of memory.The models are trained using the AdamW optimizer and a polynomial learning rate (LR) scheduler, with an initial learning rate of $5e^{-4}$ and a weight decay set to 0.1. We fine-tuned for 40 epochs on TinyFace [1] dataset , utilizing a warm-up of 2 epochs and a batch size of 8. For the BRIAR [2] dataset, we fine-tuned for 10 epochs with one warm-up epoch, also using a batch size of 8. We utilized a low-rank decomposition of 8 for the TinyFace dataset and 32 for the BRIAR dataset. We employed CNN-IQA [4] as our NR-IQA network to assign weightages to the LoRA modules. We present the implementation code modules for the adaptive weight estimated and Adaptive LoRA in the below code fragments. The weightage for the twin LoRA modules is calculated using `generate_alpha`. The final output is calculated as shown in `adaptive_lora`. The complete PETAL*face* training framwork for a single layer is outlined in Algorithm 1.

Algorithm 1 PETAL*face* Training Framework for a single layer

```
Given: Pre-trained weight matrix $W_0 \in \mathbb{R}^{m \times n}$, LoRA blocks $W_1 \in \mathbb{R}^{m \times n}$ and $W_2 \in \mathbb{R}^{m \times n}$, Input images $X = \{x_i \mid 0 \leq i < p\}$, Image quality estimator $\phi(x)$
for each dataset do
    Sample a random $l$ number of samples $x_1, x_2, \ldots, x_l$
    Calculate the mean $\mu$ and standard deviation $\sigma$ of the quality scores:
        $\mu = \frac{1}{l} \sum_{i=1}^{l} \phi(x_i), \quad \sigma = \sqrt{\frac{1}{l} \sum_{i=1}^{l} (\phi(x_i) - \mu)^2}$
end for
Set the threshold $t = \mu + \sigma$
for each sample $x_i$ in X do
    $q_i = \phi(x_i)$
    The weightage $\alpha_i$ is calculated using $q_i$ by:
        $\alpha_i = \begin{cases} 0.5 & \text{if } q_i = t \\ 0.5 - (t - q_i) & \text{if } q_i < t \\ 0.5 + (q_i - t) & \text{if } q_i > t \end{cases}$
end for
We obtain image quality scores $Q = \{q_i \mid 0 \leq i < p \ni q_i = \phi(x_i), \forall\, 0 \leq i < p\}$
The final output $x_{out}^{i}$ is calculated as:
    $x_{out}^{i} = W_0(x_i) + \alpha_i W_1(x_i) + (1 - \alpha_i) W_2(x_i)$
```

Image quality based weight assignment

```
!pip install pyiqa
iqa = pyiqa.create_metric('cnniqa').cuda()

def generate_alpha(img, iqa, threshold):
    device = img.device
    BS, C, H, W = img.shape
    alpha = torch.zeros((BS, 1), dtype=torch.float32, device=device)

    score = iqa(img)
    for i in range(BS):
        if score[i] == threshold:
            alpha[i] = 0.5
        elif score[i] < threshold:
            alpha[i] = 0.5 - (threshold - score[i])
        else:
            alpha[i] = 0.5 + (score[i] - threshold)
    return alpha
```

Adaptive LoRA

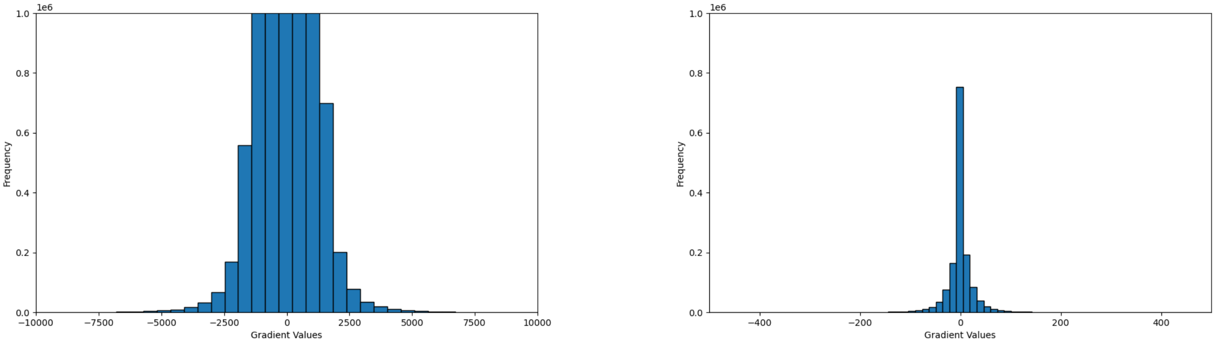
```
class AdaptiveLoRA(nn.Linear):
    def __init__(self, in_features: int, out_features: int,  r: int, scale: int, bias:
     ↪  bool=True) -> None:
        super().__init__(in_features, out_features, bias)
        # LoRA 1
        self.r_1 = r
        self.scale_1 = scale
        self.trainable_lora_down_1 = nn.Linear(in_features, self.r_1, bias=False)
        self.dropout_1 = nn.Dropout(0.1)
        self.trainable_lora_up_1 = nn.Linear(self.r_1, out_features, bias=False)
        self.selector_1 = nn.Identity()
        nn.init.normal_(self.trainable_lora_down_1.weight, std=1/self.r_1)
        nn.init.zeros_(self.trainable_lora_up_1.weight)

        # LoRA 2
        self.r_2 = r
        self.trainable_lora_down_2 = nn.Linear(in_features, self.r_2, bias=False)
        self.dropout_2 = nn.Dropout(0.1)
        self.trainable_lora_up_2 = nn.Linear(self.r_2, out_features, bias=False)
        self.scale_2 = scale
        self.selector_2 = nn.Identity()

        nn.init.normal_(self.trainable_lora_down_2.weight, std=1/self.r_2)
        nn.init.zeros_(self.trainable_lora_up_2.weight)

    def forward(self, x, alpha):
        out = F.linear(x, self.weight, self.bias)
        lora_adjustment_1 = self.scale_1*self.dropout_1(self.trainable_lora_up_1(
         ↪  self.selector_1(self.trainable_lora_down_1(x))))
        lora_adjustment_2 = self.scale_2*self.dropout_2(self.trainable_lora_up_2(
         ↪  self.selector_2(self.trainable_lora_down_2(x))))
        out = out + (1 - alpha)*lora_adjustment_1 + alpha*lora_adjustment_2
        return  out

```



(a) Full fine-tuning

(b) PETAL*face*

Figure 1. Comparison of initial gradients when (a) Full fine-tuning a model and using (b) PETAL*face* fine-tuning approach. We can see that PETAL*face* has small initial gradients which results in stable and gradual convergence. **NOTE:** The scale of the 'Gradient Values' axis for Full fine-tuning and PETAL*face* is different.

B. IJB-S Results

Training	**Dataset**	**Arch.**	**IJB-S (Surveillance to Single) [3]**			**IJB-S (Surveillance to Booking) [3]**		
			Rank-1	Rank-5	Rank-10	Rank-1	Rank-5	Rank-10
Pre-trained	WBF4M [5]	R50	32.01	45.72	51.25	43.82	55.75	61.28
Pre-trained	WBF4M [5]	Swin-B	33.23	49.85	57.63	46.22	59.40	64.93
Full-FT	WBF4M [5]	Swin-B	4.20	10.95	16.64	5.39	13.31	19.84
PETAL*face*	WBF4M [5]	Swin-B	**37.12**	**51.07**	**57.60**	**43.63**	**59.85**	**66.15**
PETAL*face*	WBF12M [5]	Swin-B	44.40	57.84	63.87	51.09	64.67	70.30

Table 1. Results on IJB-S [3] dataset in *Surveillance-to-single* and *Surveillance-to-booking* settings. The models are fine-tuned on the BRIAR train set. We report the closed-set rank retrieval (Rank-1, Rank-5 and Rank-10). [**BLUE**] indicates the best results for models trained on WebFace4M [5].

The results on the IJB-S dataset in the *Surveillance-to-single* and *Surveillance-to-booking* settings are shown in Table 1. In the *Surveillance-to-single* setting, gallery images are single high-quality images. Similarly, in *Surveillance-to-booking*, we have high-quality gallery images from different angles. The probes are of surveillance quality in both settings. This setup highlights the importance of having two proxy encoders for different resolutions within the same backbone, which are weighted based on input image quality. PETAL*face* shows improved performance with rank-1, rank-5, and rank-10 retrieval accuracies of 44.40, 57.84, and 63.87, respectively, in the *Surveillance-to-single* setting. We see similar improvements in the *Surveillance-to-booking* setting for rank-5 and rank-10 accuracies, with increases of 0.45% and 1.22%. The results demonstrate the generalization capability of the proposed fine-tuning approach. Although the model is fine-tuned on the BRIAR dataset, the knowledge of low-resolution data gained from that can be translated to other datasets such as IJB-S. Additionally, we observe a significant drop in performance when we fully fine-tune the model. We discussed the causes in the main paper and want to reiterate here. Face recognition models are pre-trained on large datasets with high-resolution images. When fine-tuning on low-resolution datasets, the model encounters a domain difference, which leads to large gradient updates initially. This deviates the model from the original pre-trained state abruptly, leading to poor convergence. We provide a gradient analysis in Section D to validate our claims.

C. Gradient Analysis

We analyze the gradients of the model backbone when fully fine-tuning the model versus when using PETAL*face* to fine-tune the model. We plot the frequency of gradient values for the first iteration of training. As shown in the Figure 1, we see that when fully fine-tuning the model, the initial gradients are very large, and even after clipping the gradients, there will be a large number of parameters that will change significantly. This is due to the domain difference between pre-trained and fine-tuned data, leading to an abrupt deviation from pre-trained weights when fully fine-tuning the model. The initial value of gradients when using the PETAL*face* fine-tuning approach results in relatively smaller gradients initially, leading to more stable and gradual convergence and improved performance. Moreover, the original weights remain frozen thereby preserving all information learned during large scale training. This demonstrates the superiority of our approach in efficiently adapting to low-resolution data.

D. Failure Case Analysis

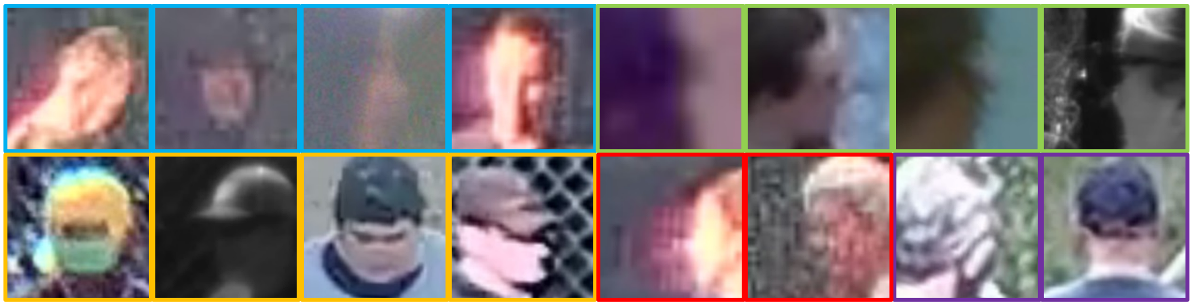


Figure 2. Failure Case Analysis of PETAL*face* on the BRIAR dataset. All the subjects are consented for publication.

We conducted a failure case analysis of the probe videos, as summarized in Figure 2, to examine the limitations of our model. We found that it struggled to recognize faces that were very low in resolution and featured extreme head poses. It also failed in cases of heavy occlusion, where faces were obscured by items like caps, masks, or sunglasses. Additionally, the model performed poorly when faces were degraded by atmospheric turbulence, making recognition difficult. Furthermore, the model failed with probe videos lacking frontal face views, as it could not identify individuals without clear frontal visibility throughout the video.

References

[1] Zhiyi Cheng, Xiatian Zhu, and Shaogang Gong. Low-resolution face recognition. In *Computer Vision–ACCV 2018: 14th Asian Conference on Computer Vision, Perth, Australia, December 2–6, 2018, Revised Selected Papers, Part III 14*, pages 605–621. Springer, 2019. 1

[2] David Cornett, Joel Brogan, Nell Barber, Deniz Aykac, Seth Baird, Nicholas Burchfield, Carl Dukes, Andrew Duncan, Regina Ferrell, Jim Goddard, et al. Expanding accurate person recognition to new altitudes and ranges: The briar dataset. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*, pages 593–602, 2023. 1

[3] Nathan D Kalka, Brianna Maze, James A Duncan, Kevin O'Connor, Stephen Elliott, Kaleb Hebert, Julia Bryan, and Anil K Jain. Ijb–s: Iarpa janus surveillance video benchmark. In *2018 IEEE 9th international conference on biometrics theory, applications and systems (BTAS)*, pages 1–9. IEEE, 2018. 3

[4] Le Kang, Peng Ye, Yi Li, and David Doermann. Convolutional neural networks for no-reference image quality assessment. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 1733–1740, 2014. 1

[5] Zheng Zhu, Guan Huang, Jiankang Deng, Yun Ye, Junjie Huang, Xinze Chen, Jiagang Zhu, Tian Yang, Jiwen Lu, Dalong Du, et al. Webface260m: A benchmark unveiling the power of million-scale deep face recognition. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 10492–10502, 2021. 3