Improved Representation Learning for Unconstrained Face Recognition

Nithin Gopalakrishnan Nair[1*], Kartik Narayan[1*], Maitreya Suin[1], Ram Prabhakar Kathirvel[1], Jennifer Xu[2], Soraya Stevens[2], Joshua Gleason[2], Nathan Shnidman[2], Rama Chellappa[1] and Vishal M. Patel[1]

[1] Johns Hopkins University, Baltimore, Maryland, USA

[2] Systems Technology Research, USA

***Abstract*— Face recognition is a widely studied problem where the aim is to design a robust network that assigns higher similarity to the same face and reduces similarity between dissimilar faces. Previous research utilizing margin-based loss functions has achieved near-perfect accuracies on high-quality face recognition datasets. However, the same networks fail to perform well on low-quality images due to the degradation of facial attributes necessary for distinguishing different faces. In this paper, we tackle the problem of low-quality face recognition. We base our analysis on an observation that the change of loss functions produce marginal changes in performance for low-quality face recognition. Hence, rather than following the traditional approach of defining problem-specific regularized functions, we take a closer look at the nature of data in low resolution datasets and redefine paradigms in terms of model choice, data input pipeline and fine-tuning schemes. With the accumulated effect of all our design choices, we achieve state-of-the-art results in medium-quality benchmarks (IJB-B, IJB-C) as well as multiple challenging benchmarks for unconstrained face recognition (Tinyface, IJB-S and BRIAR), thereby opening up a new avenue of research in the area. The pretrained model are publicly available in https://github.com/Kartik-3004/PETALface**

I. INTRODUCTION

Face recognition (FR) is a well researched problem owing to its utility in security and surveillance [4], [30]. A large number of works have tackled the high-resolution face recognition problem owing to its practical relevance and have achieved near-perfect accuracies [7], [33], [16], [27]. Most of these approaches utilize deep networks with different variants of margin-based losses [7], [33] that are well-suited for separating different facial clusters along a hypersphere. Low-quality face recognition [16], [14], on the other hand, refers to developing a method capable of recognizing faces while being robust to degradation in image quality. These degradations may include low-resolution images, compression artifacts, color jitter, atmospheric turbulence, or a complex non-linear combination of multiple of these, hence making low-quality face recognition an unsolved and challenging problem to date.

This research is based upon work supported in part by the Office of the Director of National Intelligence (ODNI), Intelligence Advanced Research Projects Ac- tivity (IARPA), via [2022-21102100005]. The views and conclusions contained herein are those of the authors and should not be interpreted as necessarily representing the official policies, either expressed or implied, of ODNI, IARPA, or the U.S. Government. The US Government is authorized to reproduce and distribute reprints for governmental purposes notwithstanding any copyright an- notation therein.

∗ Equal contribution

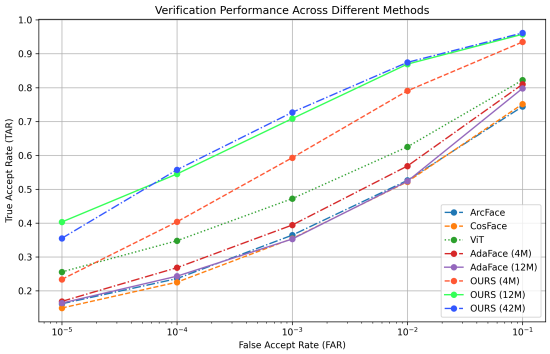
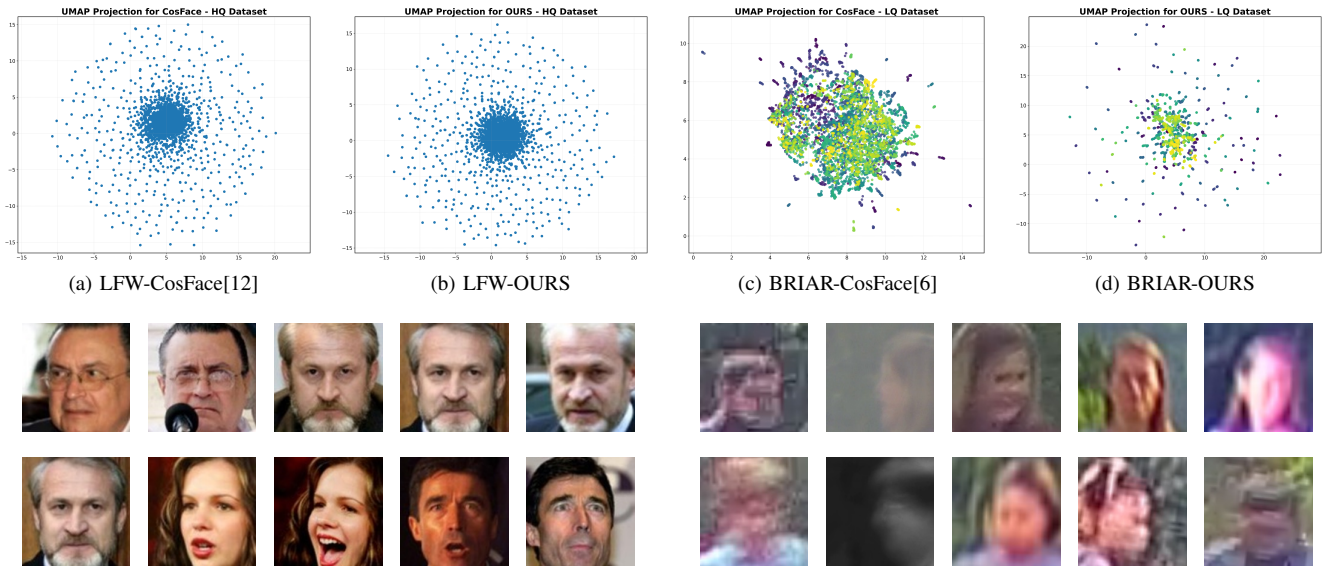


Fig. 1: We illustrate the verification performance of different methods on the BRIAR Protocol 3.1. Our proposed method when trained with WebFace4M [OURS (4M)], outperforms other existing methods utilizing different loss functions such as AdaFace, ArcFace and CosFace. Additionally, our model trained on WebFace42M [OURS (12M)] achieves state-of-the-art performance. We are able to achieve a much better performance boost over existing loss-based methods through careful design choices utilizing the CosFace loss function.

Early research in face recognition utilized handcrafted features like Haar [32] and SIFT features [24] to find similarity between different facial images. However, due to the lack of scalability of these approaches, researchers introduced deep networks to find a practical solution that can scale to large datasets comprising millions of identities. Margin-based loss functions [7], [33], [16], [27] revolutionized face recognition by providing a practical solution to learn features that are best suited to measure the similarity between faces at the same time separating different identities along a hypersphere utilizing a fixed margin. Margin-based losses can extract expressive details from facial images, enabling the formation of representative deep features. Low-quality face datasets [16], [5], [14], however, lack these clear facial features, causing models trained on high-quality datasets to fail on them.

Figure 1 shows an example where ArcFace [7], an approach that achieves more than 98% accuracy in high-quality datasets like LFW [12] and AgeDB [29], barely achieves a verification accuracy of 40% at 1e-3 for low-quality BRIAR dataset [6]. To understand why this happens, we visualize



(a) LFW-CosFace[12] (b) LFW-OURS (c) BRIAR-CosFace[6] (d) BRIAR-OURS

Fig. 2: Top: UMAP plots of embeddings extracted from Low quality (LQ) and High quality (HQ) datasets. The first two plots show embeddings for the LFW dataset (HQ). Existing works extract separable embeddings, but with decreased quality as in BRIAR-CosFace, embeddings merge, reducing accuracy. Bottom: Corresponding images from LFW and BRIAR dataset, illustrating the ease of distinguishing LFW images compared to the LQ BRIAR images.

the UMAP embeddings [26] for ArcFace features on images[1] from the LFW dataset and the BRIAR dataset [6] in Figure 2. As can be seen, high-quality embeddings from different identities form distinct clusters, making it feasible to separate them into different classes along a hypersphere. In contrast, those from BRIAR [6] merge across classes, hence explaining the failure of a network trained to recognize high-quality faces. One can postulate that this happens because of training ArcFace with a high-quality dataset like MS1MV2 [3] and argue that training on a low-quality dataset will solve this issue. However, previous works have already found that training a network naively with low-quality data will only lead to a drop in performance rather than a boost [37]. The only work designed specifically to tackle low-quality face recognition (AdaFace) [16] learns an adaptive margin loss where the margin is learned based on image quality. Following this approach, we trained networks utilizing different variants of margin-based losses but found only marginal improvement in low-quality verification with the inclusion of these losses. Hence, this prompted us to reconsider the problem from a different perspective to find a more robust solution for low-quality face recognition. Moreover, while designing the approach, we need to keep in mind the following points: (1) there are very few labeled low-quality face datasets [5], [6]; (2) different variants of margin-based losses provide marginal improvement; (3) previous literature in face recognition rarely examined different variants of architectures for face recognition, with most works mainly aiming at finding robust loss functions [7], [33], [27], [16]; and (4) existing face recognition networks are highly sensitive to keypoint locations [7], [8] and alignment of the face with reference to the image frames.

[1]We have obtained Informed Consent from the subjects used in the paper

In this paper, we design an effective solution for low-quality face recognition. Considering the challenges, we postulate that existing methods fail in low-quality face recognition because of (1) the lack of a proper data-based approach, (2) the lack of a robust pre-trained architectures less sensitive to keypoint alignment, and (3) the lack of a step-by-step guide to adapt to smaller datasets. Taking these shortcomings into account, we present a novel framework to develop face recognition networks for low-quality datasets. Specifically, we treat low-quality face recognition as a domain adaptation problem and solve it utilizing a transfer-learning based approach. Moreover, we reveal careful design choices like fine-tuning resolution, dropout in the network and optimizer choices, which allows us to get a significant boost in performance when adapted to low resolution datasets. We perform extensive experiments across different low-resolution benchmarks (TinyFace [4], IJB-S [14], and BRIAR [6]) as well as public benchmarks for medium-quality face recognition and obtain state-of-the-art results across all the benchmark datasets. To summarize, our contributions are as follows:

- We discover that rather than the choice of loss function, the architecture and the choice of training dataset plays a key role in low-resolution face recognition performance.
- We propose a simple supervised pre-training, fine-tuning based algorithm for boosting the facial recognition performance in low resolution datasets.
- We discover that the choice of bigger crops gives a significant boost in performance for low-quality datasets.
- We further propose multiple training techniques to boost performance for low-resolution datasets.
- We obtain state-of-the-art results across multiple benchmark datasets for low-resolution and mixed-quality face recognition.

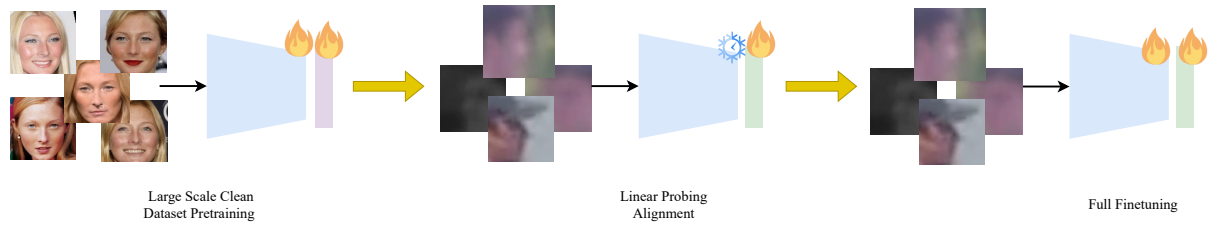


Fig. 3: An overview of the proposed method. We first perform supervised pre-training on a large scale, aligned, clean dataset. After convergence, we utilize a small-scale, low-resolution dataset for fine-tuning and initiate with linear probing for alignment, during which the backbone is frozen. Following the alignment, we perform a full-scale fine-tuning.

II. PROPOSED METHOD

In this section, we proceed to discuss the design of our proposed approach with the relevant background. The selection of Swin transformer architecture is motivated by its superior transfer learning properties compared to vision transfomers as proposed by Kim *et al*[15]. Furthermore, the Swin Transformer demonstrates enhanced performance using the same computational resources and an identical number of parameters compared to conventional Vision Transformers [9]. Specific details of the network structure are provided in the experiments section. Before delving into the specific details of our method, we discuss the relevant background concerning these design choices, how they have been previously utilized in general vision tasks, and which aspects of facial recognition rendered them unfeasible in prior face recognition works.

A. Background

1) Transformer architecture for Face recognition: Previous works have primarily utilized Convolutional Neural Network (CNN)-based architectures for face recognition tasks, a trend that has persisted over time. A major reason for this is the development of a large number of face recognition approaches prior to the year 2020 [9], [21], resulting in most baselines being established using CNNs. Additionally, as mentioned in the previous sections, Resnet-101(R-101) and ResNet-200(R-200) [11] architectures have shown near-perfect results on many existing datasets, thus diminishing the perceived need for a transformer-based approach [9]. However, keeping all these facts in mind, R-101 and R-200 architectures are bulky and slow. Moreover, they suffer from inductive biases. Inductive bias refers to the inherent modeling bias that is present in CNNs due to the design structure. As an example, consider an object that is present in the top of an image and the bottom of an image. Both of these would be treated equally in a convolutional neural network. However, in the case of faces, we argue that such a modelling procedure might not always be beneficial since most features useful for face recognition are present along keypoint locations rather than other parts like corners or the forehead/cheek part of the face. Therefore, a differential preference for different areas of the face would aid in designing a robust model. Moreover, current face recognition architectures are highly sensitive to the alignment of the 5 major keypoints comprising of eyes, nose and mouth in the face [8]. Current CNN-based frameworks are highly sensitive to these key point locations in the testing images, and even small variations to these key points cause a drastic drop in face recognition performance. In contrast, transformers with their inherent positional embeddings, exhibit less sensitivity to shifts in keypoint positions and maintain robustness even in low-resolution datasets [14], [4], [6].

2) Transfer learning in deep learning: Transfer learning [31], [34] has gained widespread popularity for both vision and language tasks. In Transfer learning, a large network trained on a big-dataset is adapted for the same or a different task on a smaller dataset. During transfer learning [10], [34], [31], the weights learned from the larger dataset are directly utilized to extract meaningful features from the smaller dataset. Transfer learning is particularly beneficial because training with limited data often fails to extract diverse, meaningful features, and during testing, the network may encounter scenarios where it struggles to extract high-quality features. Therefore, training on a large, diverse dataset and transferring the learned models to a smaller dataset is often preferred. Transfer learning has proven effective for natural language modelling and is widely used in large language models (LLMs) for instruction tuning [39], [36]. In the realm of vision tasks, transfer learning has been employed through various approaches. (1) Large-scale supervised pre-training on a large dataset followed by supervised fine-tuning on a smaller dataset. (2) Large scale self-supervised pre-training[10] using masked image modelling and supervised fine-tuning. However, for face recognition, both these approaches have not been explored. Nevertheless, given the suboptimal performance of existing methods on low-quality datasets and the typically small size of public face recognition datasets, we were motivated to explore these methodologies while developing our approach.

B. Unconstrained Face Recognition

Our method consists of three stages during the training process: supervised pre-training, linear probing alignment, and full fine-tuning stage. Additionally, we propose new design choices to enhance performance: (1) choosing larger fine-tuning crops, (2) utilizing a more effective optimizer, (3) employing dropout during pre-training. In the following sections, we will discuss these parts in greater detail.

1) Design choice: Supervised pre-training & Fine-tuning: Multiple approaches for pre-training have been proposed in the literature, including self supervised pre-training using Masked autoencoders (MAEs) [10] where a portion of the input image is masked and the network learns to reconstruct

Fig. 4: An illustration of the process of extracting face crops from face video probes for low-resolution datasets. We present the preparation of training and test probes from BRIAR dataset[6] here. First, a detection algorithm is applied to find bounding boxes of the face and landmark coordinates. Next, the landmark points are aligned and a 112×112 crop is extracted. Finally, the crop is expanded to dimension 180×180 and then resized to 120×120 for fine-tuning. While smaller crops may miss relevant details in the faces, enlarging the crops captures more details.

these masked portions to derive meaningful representations of the image. However, for facial recognition tasks, because of the availability of large scale open source labelled datasets, we opt for supervised pre-training instead. This decision is driven by two main factors: (1) Face images are typically smaller when compared to natural domain images, with dimensions of only 112×112 for public datasets. (2) While MAEs excel in reconstructive tasks, the features they learn are not inherently discriminative. Hence to learn better discriminative features, we choose supervised pre-training as the first step of the pre-training process. Moreover, given the challenges posed by low-quality face recognition, where the input test images have drastically different quality and lighting conditions compared to the pre-training images, low-quality face recognition can also be treated as a domain adaptation problem. Previous studies[15] have shown that for domain adaptive tasks, supervised pre-training is more effective than self-supervised and loss-based techniques. Thus, we choose supervised pre-training as the first stage of our training process.

Stage 1: Supervised Pre-training; we pre-train a Swin Transformer backbone with a large-scale pre-training dataset. For face recognition, supervised pre-training presents a particular challenge when followed by a fine-tuning stage. Typically, datasets used for fine-tuning in low-range face recognition are quite small, often containing only a few hundred identities. In face recognition networks, the final layer is an MLP layer (classification head) designed to perform classification tasks. Once the network is pre-trained, this final layer is discarded. A new classification head needs to be appended to the pre-trained backbone to adapt the pre-trained backbone for a smaller dataset. This newly added classification head is initialized with random weights at the beginning of the fine-tuning process.

Consider the fine-tuning process where the entire network, including the classification head, is fine-tuned from scratch. During the first iteration of inference, the MLP head encounters reasonable features; however, due to its random initialization, it produces random outputs. Furthermore, during the initial phases of backpropagation, a significant portion of the pre-trained weights undergoes alteration. Hence, one might have to fine-tune with an extremely low learning rate to prevent the loss of pre-trained weights. To prevent this forgetting of weight space often referred to as *catastrophic forgetting* [19], we employ a two-stage fine-tuning process. In the first step, we perform linear probing by training only the MLP head for a limited number of epochs. This allows the MLP head to align with the input features from the pre-training backbone and effectively classify different identities in the fine-tuning dataset. Once this step has converged, further training integrates the adjusted MLP head with the overall network.

Stage 2: Linear Probing [1], we fine-tune on a low-resolution dataset by performing linear probing. During the full fine-tuning process, two critical details must be considered: (1) pre-training datasets are typically large, encompassing millions of identities, and (2) fine-tuning datasets usually contain only a few hundred identities. Therefore, if one were to fine-tune the entire model for large number of epochs, catastrophic forgetting could occur. Due to this, the features, initially learned to discriminate among millions of identities might be overwritten to fit to just a few hundred, causing the network to lose its generalization capability. Additionally, low-resolution datasets contain very diverse images with significant variations in lighting conditions, image resolution, and image quality. If one were to use a large batch size for gradient updates, this variation would be averaged out to a large extent. Thus, we opt for a smaller batch size during fine-tuning. Furthermore, to prevent catastrophic forgetting [19], we restrict training to a limited number of iterations and maintain a low learning rate.

Stage 3: Full fine-tuning for a small number of iterations with a small batch size and low learning rate. We illustrate the overall training process in Figure 3. Initially, we perform supervised pre-training on the transformer backbone using a dataset of million-scale identities. Subsequently, we employ linear probing to align the MLP head with the smaller dataset. Finally, we fine-tune the entire model on the low-resolution dataset for a small number of training iterations, utilizing a small batch size and a low learning rate.

2) Design choice: Bigger fine-tuning crops: Traditionally, face recognition has been performed using 112×112 crops, because meaningful discriminative features can be extracted from these regions for clear faces. However, as shown in the 4, most faces in low-resolution datasets do not retain these features, leading to a scenario where even humans

might struggle to distinguish between different faces due to significant blurring of facial regions. Furthermore, while some previous works have suggested that discarding extremely hard samples during the training process can enhance performance [37], this is often impractical in low-range face recognition due to the small size of the training datasets. This is because discarding images could reduce the dataset to just a few hundred images, which is insufficient for effective training. Hence, we adopt an alternative approach. Before presenting our solution, it is crucial to understand the common challenges in low-quality face recognition as depicted in Figure 4. In low-quality face recognition, the dataset often includes images/videos of scenes captured from a significant distance from the subjects. An off-the-shelf detector is employed to crop faces from these broader scene, which are then processed through a facial recognition system. In our case, however, we take advantage of having access to the entire scene, allowing for additional contextual information to be utilized.

Hence, rather than restricting our method to a crop size of 112×112 as existing methods do, we opt for bigger crops of 180×180. These bigger crops capture the entire head and hair outline of a person. Such regions can potentially reveal additional descriptions about the person, such as the head shape, hair color, gender, hairline, ear shape, etc., which may not have been present in the pre-training dataset. We observed a significant boost in performance by including these additional features utilizing the larger crop size. To accommodate the larger crops during fine-tuning, we resize the input images and pre-train at a size of 120×120, an increase from the traditional 112×112. The bigger crop size, while fine-tuning, allows for a slight enlargement of the head in the images. The initial resolution of 180 was chosen since this resolution captures the enlarged the whole head shape without including much background details. Moreover resizing the 180×180 crops to 120×120 allows to retain almost same amount of information in the crops while reducing the memory overhead by a factor of 2.4 because of quadratic memory complexity in transformers.

We ensure that the cropped face aligns well while fine-tuning by utilizing new landmark alignment coordinates. These new coordinates are chosen such that the center of the image, i.e., the nose portion of the face, aligns between the fine-tuning and pre-training datasets. In order to prevent the effect of warping and upscaling artifacts, we don't perform any upscaling of the newly cropped face; instead, we expand the borders along the crop to get additional features.

Step 2: During fine-tuning, we utilize larger crops of size 180×180***, capturing the whole head of the person.***

3) Design choice: Dropout in penultimate layer: We discovered that including dropout during pre-training serves as a major enabler for adaptation performance when fine-tuned on a smaller out-of-distribution dataset. We detail these findings in the experiments section, where we compare a Swin Transformer [21] backbone trained with and without dropout, subsequently fine-tuned on a low-resolution dataset. Our experiments reveal that although utilizing dropout brings almost no difference in the performance of the pre-trained models, the performance over low-quality datasets after fine-tuning gains a significant boost when the model is pre-trained with dropout on the penultimate layer compared to its counterpart.

Step 3: Step 3: During pre-training, we include dropout in the penultimate layer, which leads to a boost in performance after fine-tuning.

4) Design choice: Annealing-based scheduler: The problem of local optima is well researched in the deep learning community[20], [2], [18], [23]. Multiple works have proposed different variants of optimizers and learning rate schedulers [20] separately designed to work well for CNNs and transformers [9], [21]. The choice of optimizer and schedulers is crucial to ensure that during the optimization process, the model does not get stuck at local optima and the training process occurs smoothly. In the case of face recognition, Multiple works have previously found that for ResNet-based architectures, stochastic gradient descent is the best-suited optimizer, and AdamW [18] to be the best working for transformer-based backbones [9], [21]. Regarding the choice of learning rate scheduler, recent face recognition methods employ a Polynomial Decay Scheduler with a warmup phase [28].

During our experiments, we found that using a polynomial decay scheduler with a warmup during the pre-training phase for the Swin architecture leads to the model outputting sub-optimal results. Upon further analysis, we identified that the learning rate decreased significantly, reaching a very low value at nearly half the total number of iterations. This causes the model to become stuck at a local optimum, from which it struggles to recover and achieve optimal performance. To address this issue, we propose a modification: the use of annealed optimizers [22]. A particularly effective choice is the Cosine Annealing scheduler. In our work, we designed an annealed Poly optimizer that resets the learning rate to half of its initial value at the mid-stage of the training process. The use of such an annealed training process leads to a significant boost in performance as detailed in the experiments section. Additionally, we experimented with a multi-step LR strategy similar to that in AdaFace [16]. However, we found that without a warmup phase, the Swin Transformer encounters convergence issues in face recognition training, rendering this approach impractical.

Step 4: During pre-training, we employ a Polynomial decay scheduler with a warmup phase and perform annealing at the mid-stage of the training process.

III. EXPERIMENTS

A. Implementation Details

All our experiments are conducted on 8 NVIDIA A5000 GPUs. As mentioned before, we employ a Swin transformer backbone [21] as our recognition network. The choice of the Swin transformer, along with other reasons mentioned in previous sections, is due to its best out-of-distribution (OOD) generalization capability compared to other transformer-based backbones. Instead of using the publicly available Swin

Method	Dataset	Architecture	IJBB			IJBC			AgeDB	CALFW	CFP-FF	CFP-FP	CPLFW	LFW
			e-5	e-4	e-3	e-5	e-4	e-3						
ArcFace [7]	WbF4M	R-50	89.62	94.02	96.15	93.61	95.99	97.48	96.81	95.71	99.75	96.71	93.41	99.66
CosFace [33]	WbF4M	R-50	89.70	94.09	96.22	93.57	96.01	97.53	96.88	95.63	99.70	96.82	93.28	99.68
AdaFace [16]	WbF4M	R-50	90.78	94.95	96.68	90.61	94.70	96.67	97.26	95.98	99.81	97.14	93.81	99.78
ArcFace [7]	WbF4M	ViT-B	89.81	94.91	96.70	94.33	96.64	97.84	97.53	95.91	99.80	97.22	93.68	99.81
CosFace [33]	WbF4M	ViT-B	90.76	95.18	96.94	94.67	96.87	98.09	97.51	95.95	99.87	97.30	94.31	99.73
AdaFace [16]	WbF4M	ViT-B	89.91	94.90	96.71	94.04	96.52	97.91	96.85	95.71	99.80	97.00	93.75	99.76
AdaFace [16]	WbF4M	R-101	92.27	95.69	97.09	95.34	97.09	98.07	97.85	96.01	99.88	97.21	94.21	99.75
AdaFace [16]	WbF12M	ViT-B	92.17	96.10	97.25	96.01	97.59	98.31	98.18	95.93	99.85	97.47	94.65	**99.83**
AdaFace [16]	WbF12M	R-101	93.14	96.30	97.30	95.94	97.54	98.24	98.01	96.00	99.81	97.47	94.30	99.78
OURS	WbF4M	Swin-B	90.99	95.41	97.09	94.68	96.99	98.09	97.76	95.88	99.81	96.74	93.61	99.75
OURS	WbF12M	Swin-B	93.16	96.27	97.48	96.31	97.70	98.46	98.35	96.03	99.87	97.57	94.66	99.78
OURS	WbF42M	Swin-B	**94.27**	**96.71**	**97.58**	**97.11**	**98.01**	**98.54**	**98.38**	**96.13**	**99.90**	**97.70**	**94.98**	99.78

TABLE I: ROC values for face recognition on the IJB-B and IJB-C datasets at different TAR@FAR thresholds. Face verification accuracy on high-quality datasets such as AgeDB, CALFW, CFP-FF, CFP-FP, CPLFW, LFW.

Method	Dataset	Arch.	e-4	e-3	e-1	Rank-1	Rank-5	Rank-20
ArcFace [7]	WbF4M	R-50	23.60	36.42	52.60	45.53	54.56	65.13
CosFace [33]	WbF4M	R-50	22.55	35.43	52.20	45.43	54.54	65.13
ViT(CosFace) [9]	WbF4M	ViT-B	34.78	47.20	62.50	55.59	63.44	72.76
Adaface [16]	WbF4M	R-101	24.32	35.28	52.34	51.28	59.11	67.77
Adaface [16]	WbF12M	R-101	26.82	39.41	56.84	51.94	59.53	68.78
CAFace [17]	WbF4M	R-101	33.41	41.95	51.31	-	-	-
CONAN [13]	WbF4M	R-101	36.52	46.14	56.32	-	-	-
OURS	WbF4M	Swin-B	40.37	59.29	79.04	70.35	79.76	88.22
OURS	WbF12M	Swin-B	54.50	70.84	86.93	79.60	87.00	92.27
OURS	WbF42M	Swin-B	**55.73**	**72.69**	**87.45**	**81.38**	**87.87**	**92.48**

TABLE II: Quantitative metrics for face recognition on the BRIAR dataset [6]. We evaluate using the ROC values at different TAR@FAR thresholds as well as CMC scores.

Method	Dataset	Architecture	Rank-1	Rank-5	Rank-10
ArcFace [7]	WebFace4M	ResNet-50	73.04	76.85	79.45
CosFace [33]	WebFace4M	ResNet-50	72.71	76.36	78.99
ArcFace [7]	WebFace4M	ViT-B	74.08	77.19	79.10
CosFace [33]	WebFace4M	ViT-B	72.74	76.28	78.13
AdaFace [16]	WebFace4M	ViT-B	74.03	77.22	79.37
Official reported numbers					
Adaface [16]	WebFace4M	ResNet-101	72.02	74.51	76.58
Adaface [16]	WebFace12M	ResNet-101	72.31	74.97	76.87
Reproduced results with our alignment					
Adaface [16]	WebFace4M	ResNet-101	74.75	77.52	79.50
Adaface [16]	WebFace12M	ResNet-101	75.13	77.65	79.31
Finetuned on TinyFace					
ArcFace [7]	WebFace4M	ViT-B	69.20	74.91	78.94
CosFace [33]	WebFace4M	ViT-B	71.08	76.09	79.42
AdaFace [16]	WebFace4M	ViT-B	42.59	50.69	57.00
AdaFace [16]	WebFace12M	ViT-B	42.40	51.01	57.75
AdaFace [16]	WebFace4M	ResNet-101	68.53	74.59	78.46
AdaFace [16]	WebFace12M	ResNet-101	68.72	75.08	78.62
OURS	WebFace4M	Swin-B	74.94	78.08	80.52
OURS	WebFace12M	Swin-B	77.46	80.12	**82.10**
OURS	WebFace42M	Swin-B	**77.38**	**79.96**	81.46

TABLE III: Quantitative metrics for face recognition on the TinyFace dataset [5]. We evaluate using the CMC scores. Adaface-ft are results obtained after fine-tuning Adaface.

Transformer backbone, which is configured for a resolution of 224×224, we train a new backbone from scratch at a resolution of 120×120 for the WebFace4M [38] and WebFace12M datasets. We perform downsampling twice. The overall depth across different layers is $[4, 4, 14]$. We utilize a patch size of 9×9 and a window size of 5 across all models. The trained models have a hidden embedding dimension of 256. We use a batch size of 128 for training the models. For all the models, we use AdamW optimizer [23] with a learning rate of 0.001 and a Polynomial Scheduler with warmup and annealing [28], [22]. We implement partial-fc with a sample rate of 0.3 to increase the batch size while training our models. In the pre-training stage, the WebFace4M models are trained for 26 epochs with a warmup of 1 epoch. The WebFace12M models undergo training for 20 epochs with a warmup of 1 epoch. For fine-tuning, we implement a two-stage process. For the BRIAR [6] dataset, in the first stage of linear probing, we utilize a learning rate of $1e^{-3}$ and a batch size of 512. In full fine-tuning stage, the batch size is set to 8 per GPU with a learning rate of $5e^{-6}$. In both stages, we train for 10 epochs with 2 warmup epochs. For the TinyFace [6] dataset, in the linear probing stage, we utilize a learning rate of $1e^{-3}$, a batch size of 128 and implement partial-fc with sample rate of 0.6. We train the model for 10 epochs with 2 warmup epochs. In the full fine-tuning stage, we train the model for 40 epochs with a warmp up of 4 epochs, a batch size of 8 per GPU, and a learning rate of $5e^{-6}$.

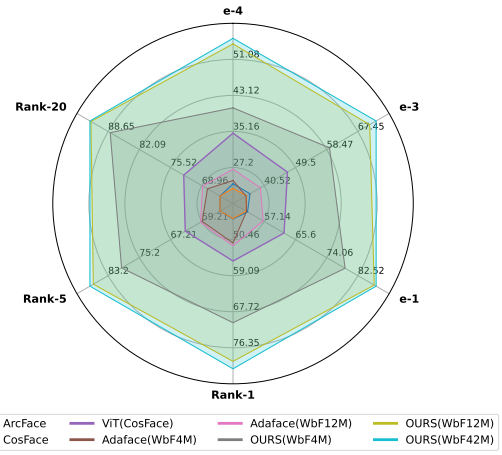


Fig. 5: Results on BRIAR dataset [6]

B. Testing Protocols

We organize our experiments into two protocols to highlight the efficacy of our architectural design choices and fine-tuning procedure. In **Protocol 1**, we evaluate models pre-trained on WebFace4M, WebFace12M and WebFace42M [38], showcasing the superiority of our proposed architectural design choices. We perform the comparison on mixed-quality datasets, IJBB [35] and IJBC [25] for TAR@FAR $= 0.001\%$, TAR@FAR $= 0.01\%$, and

TAR@FAR $= 0.1\%$, to demonstrate our models' enhanced performance. In **Protocol 2**, we assess the performance of the models fine-tuned on TinyFace (Train set) and BRIAR using the procedure described in Section II-B. This protocol emphasizes the effectiveness of the proposed fine-tuning procedure and points out the shortcomings of existing models that cannot be adapted to other low-quality datasets. We employ low-quality datasets such as TinyFace [5], IJBS [14], and BRIAR [6] for comparison. For BRIAR and TinyFace, we report the rank-1, rank-5 and rank-20 CMC scores along with the TAR at different FAR thresholds. For IJB-S, we report open-set TPIR@FPIR=1%/10% and closed-set rank retrieval (Rank-1, Rank-5 and Rank-10). CMC scores are used to evaluate the performance of identification systems and represent the probability that the correct match of an individual's face sample is found within the top N candidates provided by the system.

IV. Results and Analysis

This section summarizes the results of the experiments we perform to evaluate the proposed fine-tuning procedure. As discussed in Section III-B, we evaluate the proposed approach using two protocols. For evaluations, we train transformer based models from scratch and perform quantitative analysis with it. Although Insightface has proposed results from different models with ViT backbone, these models are not released.

Protocol 1: In this protocol, we present the results on mixed-quality datasets IJB-B and IJB-C, as shown in Table I. When trained on the WebFace42M dataset, our model achieves a True Acceptance Rate (TAR) of 94.27, 96.71, and 97.58 for IJB-B, and 97.11, 98.01, and 98.54 for IJB-C at False Acceptance Rates (FAR) of 0.001%, 0.01%, and 0.1%, respectively. With the WebFace12M and WebFace4M training dataset, the TARs are 93.16, 96.27, 97.48, and 90.99, 95.41, 97.09 for IJB-B, and 96.31 97.70, 98.46 and 94.68, 96.99, and 98.09 for IJB-C at similar FAR thresholds. We observe that our proposed architectural design choices enhances performance with increase in dataset size which is not the case for the current SOTA model. Additionally, we see that the proposed architectural design choices deliver competitive performance while requiring significantly less training time ($4\times$ less). The results detailed in Table I demonstrate that the proposed architectural choices lead to stable training and consistent model convergence. Moreover, these choices optimizes GPU memory usage, enabling training with larger batches and easily scaling to large datasets. Notably, our approach achieves state-of-the-art (SOTA) performance on the IJB-B and IJB-C benchmarks. Our model, based on the Swin-B architecture and trained on WebFace4M, WebFace12M, and WebFace42M, achieves exceptional face verification accuracy on high-quality datasets, as demonstrated in Table I, reaffirming the efficacy of our design choices.

Protocol 2: The results of Protocol 2 are summarized in Table II, Table III and Table IV. In Table II, we observe that the proposed fine-tuning procedure results in a significant improvement of approx. 13% (on avg.) over CONAN [13] at various thresholds when trained on WebFace4M dataset. When compared with current SOTA models trained on the WebFace12M dataset for face recognition, our model outperforms previous baselines by a staggering 30% (on avg.) improvement at various thresholds, along with a 26 point increment in the CMC scores. Our model trained on WebFace42M establishes a new SOTA, achieving a TAR of 55.73%, 72.69%, and 87.45% at FAR of 0.01%, 0.1%, and 1%, respectively. The Rank-1, Rank-5 and Rank-20 retrieval accuracies are 81.38%, 87.87%, and 92.48%, respectively.

Full fine-tuning of a model often leads to *catastrophic forgetting*, as can be seen in the case of AdaFace when fine-tuned on TinyFace (see Table III, rows 8 and 9). In contrast, our method achieves superior performance on the TinyFace dataset, with Rank-1, Rank-5, and Rank-10 accuracies of 77.46%, 80.12%, and 82.10%, respectively. This outperforms the second-best method, AdaFace, which obtains accuracies of 75.13%, 77.65%, and 79.31%. These results highlight the effectiveness of our two-step fine-tuning procedure for transferring to low-resolution data. Furthermore, our design enables the model to scale and improve as the dataset size increases. Specifically, the Rank-1 accuracy of our method improves by 2.52% when scaling the dataset from 4M to 12M, while AdaFace shows a marginal increase of just 0.38%. Notably, when further scaling to the WebFace42M dataset, our model continues to excel, achieving Rank-1, Rank-5, and Rank-10 accuracies of 77.38%, 79.96%, and 81.46%, respectively.

As discussed earlier, full fine-tuning often leads to *catastrophic forgetting*, resulting in a significant drop in performance. We portray these results in Table III. As we can see, for all comparison methods, a naive full fine-tuning process leads to a performance drop whereas including out alignment technique results in a boost of performance. The catastrophic forgetting happens particularly when adapted for specific settings like low-resolution or surveillance-quality data. The results on the IJB-S dataset, presented in Table IV, validate this point, showing an average drop of 30% in CMC scores. This highlights the efficacy of our proposed two-step fine-tuning procedure. We evaluate the models fine-tuned on the BRIAR train set, on IJB-S as both contain surveillance-quality videos. Notably, our model shows a substantial improvement in the Surveillance-to-Surveillance setting compared to other settings, aligning with its fine-tuning focus. However, the model trained on WebFace4M underperforms in the Surveillance-to-Single and Surveillance-to-Booking settings, likely due to its limited ability to extract discriminative features for enrollment images. Nevertheless, this limitation can be mitigated by scaling the pre-training dataset size, which our architectural design effectively accommodates. As seen in Table IV, the performance improves significantly when the pre-training dataset is scaled from WebFace4M to WebFace12M. Specifically, the Rank-1 accuracy increases from 43.68% to 57.42% in the Surveillance-to-Single setting, from 50.14% to 61.84% in the Surveillance-to-Booking setting, and from 42.94% to 49.22% in the Surveillance-to-Surveillance setting. Ad-

Method	Dataset	Surveillance to Single					Surveillance to Booking					Surveillance to Surveillance				
		Rank-1	Rank-5	Rank-10	1%	10%	Rank-1	Rank-5	Rank-10	1%	10%	Rank-1	Rank-5	Rank-10	1%	10%
ArcFace [7]	WebFace4M	31.88	45.01	50.38	15.12	23.32	41.46	52.39	58.24	19.94	29.79	34.58	50.71	55.68	3.61	8.71
CosFace [33]	WebFace4M	32.01	45.72	51.25	16.14	25.18	43.82	55.75	61.28	20.02	30.90	33.62	49.40	54.92	3.67	9.09
ViT(CosFace) [9]	WebFace4M	45.54	55.64	60.49	28.48	39.83	55.96	65.75	70.38	32.59	46.54	38.88	52.93	56.58	5.40	14.83
Adaface [16]	WebFace4M	46.89	54.30	58.85	27.38	38.37	52.65	60.54	65.09	29.20	42.22	36.95	51.94	57.34	4.14	9.97
Adaface [16]	WebFace12M	47.23	55.78	60.43	13.73	35.03	53.87	62.95	67.58	21.58	41.59	37.86	52.43	57.66	4.49	10.73
Pre-trained models																
OURS	WebFace4M	46.52	55.78	59.83	27.48	39.07	53.71	63.56	68.48	31.36	42.75	40.43	55.15	58.92	5.11	11.83
OURS	WebFace12M	51.15	58.95	62.97	36.32	46.09	60.09	68.21	71.60	41.56	52.54	42.82	55.94	59.77	5.66	13.47
OURS	WebFace42M	52.97	61.41	66.52	37.08	45.88	63.87	72.00	75.60	45.54	56.02	44.69	56.29	59.68	5.37	14.38
Full finetuning of models																
OURS	WebFace4M	14.97	26.30	32.78	3.70	8.68	18.76	30.53	36.96	5.01	10.38	43.00	55.24	58.95	10.89	28.07
OURS	WebFace12M	25.03	37.04	43.45	8.27	17.79	28.07	42.31	49.72	10.61	19.55	46.21	57.05	60.21	14.29	33.52
OURS	WebFace42M	26.67	36.72	42.97	9.90	19.02	31.48	46.83	52.92	9.98	20.51	44.25	55.36	59.68	12.99	32.70
OURS	WebFace4M	43.68	56.94	63.21	19.40	32.22	50.14	64.32	69.33	22.14	35.39	42.94	54.80	58.63	17.80	32.92
OURS	WebFace12M	57.42	67.63	71.63	34.86	47.82	61.84	72.71	77.16	36.11	49.37	49.22	58.86	61.61	23.49	39.91
OURS	WebFace42M	**59.72**	**68.48**	**72.47**	**32.51**	**52.19**	**63.00**	**73.59**	**77.63**	**37.06**	**53.67**	**50.68**	**59.10**	**61.81**	**23.71**	**40.71**

TABLE IV: Quantitative metrics for face recognition on the IJB-S dataset[14] under three settings - *Surveillance-to-Single*, *Surveillance-to-Booking* and *Surveillance-to-Surveillance*. We evaluate using the TPIR@FPIR=1%/10% values as well as CMC scores. For both these metrics , higher the value better the result.

Arcface	Cosface	180	Dropout	Optimizer	Rank-1	Rank-5	Rank-20
	✓	✓	✓	✓	64.12	74.77	85.00
✓		✓	✓	✓	66.00	76.35	85.97
	✓		✓	✓	33.73	45.88	62.26
	✓	✓		✓	33.31	46.74	61.04
	✓	✓	✓		61.32	73.33	83.78

TABLE V: Ablation Analysis: Analyzing the impact of different design choices on face recognition performance.

ditionally, when further scaling the pre-training dataset to WebFace42M, we observe even greater improvements, with Rank-1 accuracies reaching 59.72%, 63.00%, and 50.68% in the Surveillance-to-Single, Surveillance-to-Booking, and Surveillance-to-Surveillance settings, respectively. These results demonstrate that our fine-tuning approach and model design can effectively leverage larger datasets, enhancing performance in unconstrained face recognition tasks.

Protocol 3. We choose high-quality benchmarks as Protocol 3. We want to highlight that our training process is beneficial in high-quality face recognition as well. For this, we experiment we evaluate our pretrained model at a resolution 120×120 with the proposed training settings in Section II. Please note that for high-quality fine-tuning datasets, we do not perform any fine-tuning other than the pertaining process. We present the results in six high-quality datasets in Table 1. As can be seen, our 42M model achieves state-of-the-art results for 5 out of the 6 datasets and achieve very close to SOTA performance in the sixth dataset. Moreover, it can be seen as well that scaling up the pertaining dataset size is beneficial for high-quality face recognition as well.

V. Ablation Study

We demonstrate the impact of various design choices on performance in Table V. We observe that the effect of loss functions on model performance is relatively small (see Table V, rows 1 & 2), compared to other design choices. A larger crop size of 180×180 during fine-tuning, along with the addition of dropout in the penultimate layer, are the two design choices that significantly boost performance. The experiments for the ablation study were conducted using a smaller Swin Transformer model on the BRIAR [6] dataset to amplify the importance of different design choices during training on performance.

Large crop-size of 180×180: We fine-tune the model on low-quality dataset using larger crops of size 180×180, thereby including additional information about the identity, such as head shape, hair color, ear etc. We observe that the inclusion of such information significantly improves the Rank-1 accuracy from 33.73% to 64.12%.

Effect of dropout in the penultimate layer: The inclusion of dropout in the penultimate layer enables the model to better adapt to smaller, low-quality, out-of-distribution dataset. Excluding dropout during training results in a significant drop in model performance on low-quality datasets, with Rank-1 accuracy dropping from 64.12% to 33.31%. This highlights the critical importance of our design choice for improved transfer to low-resolution data.

Optimizer choice - AdamW with annealed polynomial scheduler: To address local minima during training, we implemented an annealing-based polynomial scheduler that resets the learning rate mid-training, to help the model in escaping local minima and enhancing performance. As shown in Table V, adding AdamW along with annealed polynomial scheduler leads in a performance jump of 2.8%, increasing the Rank-1 accuracy from 61.32% to 64.12%.

VI. Conclusion

Previous efforts in face recognition have revolved around margin-based and adaptive loss functions that fail to produce discriminative features robust enough for low-quality, unconstrained face recognition. Diverging from the current trend, in this paper, we demonstrate that conscious data and architectural design choices, coupled with a two-step fine-tuning procedure, provides a significant boost over the current state-of-the-art. We show that our architectural choices result in the least training time and enables scaling to large, million-scale datasets. We ablate each of our choices and highlight the importance of each component within the training framework. Through extensive experiments on mixed-quality (IJB-B and IJB-C) and low-quality datasets (TinyFace, IJB-S and BRIAR), we showcase the superiority of our proposed architectural design choices and two-step fine-tuning procedure.

VII. Ethical Impact Statement

In this research, we have carefully addressed the ethical implications surrounding face recognition technology, particularly focusing on issues of privacy, surveillance, and potential biases. Our model was trained on publicly available datasets: WebFace4M, WebFace12M, and WebFace42M [38], acquired through signing the official license agreement. For benchmarking, we utilized IJB-B [35], IJB-C [25], IJB-S [14], BRIAR [6], and TinyFace [5], which contain diverse, mixed-quality, and low-resolution images from real-world settings. These datasets were obtained through official repositories and websites, ensuring adherence to ethical standards. Informed consent for publication was acquired for all subjects depicted in the paper, supporting ethical data use.

This research offers significant benefits within authorized security contexts, where accurate low-resolution face recognition enhances identification capabilities in challenging environments. When applied responsibly, these advancements contribute to security and enable legitimate monitoring efforts. Importantly, the model's design and training process adhere to standards that do not introduce risks beyond those inherent in traditional face recognition systems. However, we acknowledge the potential for misuse in unauthorized surveillance, profiling, or privacy infringements if deployed outside controlled, ethical frameworks. Our work aims to support face recognition for responsible use within authorized security settings, while recognizing that unintended applications or misinterpretations could lead to societal issues, such as privacy erosion or biased treatment of certain groups. By proactively addressing these considerations, we seek to mitigate risks associated with the model's deployment and advocate for ethical oversight to prevent misuse.

Ethical considerations for human subjects and data usage were fully respected. This research relies solely on existing datasets and no new consent was required. These datasets are approved for research use, ensuring adherence to ethical data standards. No individuals were recruited which eliminates the need for compensation. The datasets do not predominantly include vulnerable populations, such as minors, elderly individuals, or other at-risk groups, instead representing a standard demographic spectrum. Given our commitment to ethical standards, this research presents minimal risk to individuals while advancing low-resolution face recognition technology.

VIII. Ethical Impact Checklist

1) Yes, we read the Ethical Guidelines document.
2) Yes, it is approved by a valid ethical review board.
3) Yes, the ethical impact statement discusses the potential risks of individual harm and negative impacts associated with the research.
4) Yes, we advocate for ethical oversight as risk-mitigation strategy to prevent misuse of the proposed research.
5) Yes, the benefits and potential positive impact outweighs the potential risks of the proposed low-resolution face recognition model.
   a) Yes, informed consent was obtained for publication and is mentioned in the main paper.
   b) No, but we state in the ethical impact statement that we use publicly available datasets collected in adherence to ethical data use standards.
   c) No, we mention in the ethical impact statement that no individuals were recruited, eliminating the need for compensation.
   d) No, the study does not involve special or vulnerable populations. The datasets do not predominantly include any special populations, and instead represent a standard demographic spectrum.

References

[1] G. Alain and Y. Bengio. Understanding intermediate layers using linear classifier probes. *arXiv preprint arXiv:1610.01644*, 2016.

[2] L. Bottou. Large-scale machine learning with stochastic gradient descent. In *Proceedings of COMPSTAT'2010: 19th International Conference on Computational StatisticsParis France, August 22-27, 2010 Keynote, Invited and Contributed Papers*, pages 177–186. Springer, 2010.

[3] D. Cao, X. Zhu, X. Huang, J. Guo, and Z. Lei. Domain balancing: Face recognition on long-tailed domains. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 5671–5679, 2020.

[4] J.-C. Chen, V. M. Patel, and R. Chellappa. Unconstrained face verification using deep cnn features. In *2016 IEEE winter conference on applications of computer vision (WACV)*, pages 1–9. IEEE, 2016.

[5] Z. Cheng, X. Zhu, and S. 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.

[6] D. Cornett, J. Brogan, N. Barber, D. Aykac, S. Baird, N. Burchfield, C. Dukes, A. Duncan, R. Ferrell, J. 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.

[7] J. Deng, J. Guo, N. Xue, and S. 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.

[8] J. Deng, J. Guo, Z. Yuxiang, J. Yu, I. Kotsia, and S. Zafeiriou. Retinaface: Single-stage dense face localisation in the wild. In *arxiv*, 2019.

[9] A. Dosovitskiy, L. Beyer, A. Kolesnikov, D. Weissenborn, X. Zhai, T. Unterthiner, M. Dehghani, M. Minderer, G. Heigold, S. Gelly, et al. An image is worth 16x16 words: Transformers for image recognition at scale. *arXiv preprint arXiv:2010.11929*, 2020.

[10] K. He, X. Chen, S. Xie, Y. Li, P. Dollár, and R. Girshick. Masked autoencoders are scalable vision learners. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 16000–16009, 2022.

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

[12] G. B. Huang, M. Ramesh, T. Berg, and E. Learned-Miller. Labeled faces in the wild: A database for studying face recognition in unconstrained environments. Technical Report 07-49, University of Massachusetts, Amherst, October 2007.

[13] B. Jawade, D. D. Mohan, D. Fedorishin, S. Setlur, and V. 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.

[14] N. D. Kalka, B. Maze, J. A. Duncan, K. O'Connor, S. Elliott, K. Hebert, J. Bryan, and A. 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.

[15] D. Kim, K. Wang, S. Sclaroff, and K. Saenko. A broad study of pre-training for domain generalization and adaptation. In *European Conference on Computer Vision*, pages 621–638. Springer, 2022.

[16] M. Kim, A. K. Jain, and X. 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.

[17] M. Kim, F. Liu, A. K. Jain, and X. Liu. Cluster and aggregate: Face recognition with large probe set. *Advances in Neural Information Processing Systems*, 35:36054–36066, 2022.

[18] D. P. Kingma and J. Ba. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*, 2014.

[19] J. Kirkpatrick, R. Pascanu, N. Rabinowitz, J. Veness, G. Desjardins, A. A. Rusu, K. Milan, J. Quan, T. Ramalho, A. Grabska-Barwinska, et al. Overcoming catastrophic forgetting in neural networks. *Proceedings of the national academy of sciences*, 114(13):3521–3526, 2017.

[20] Z. Li and S. Arora. An exponential learning rate schedule for deep learning. *arXiv preprint arXiv:1910.07454*, 2019.

[21] Z. Liu, Y. Lin, Y. Cao, H. Hu, Y. Wei, Z. Zhang, S. Lin, and B. Guo. Swin transformer: Hierarchical vision transformer using shifted windows. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 10012–10022, 2021.

[22] I. Loshchilov and F. Hutter. Sgdr: Stochastic gradient descent with warm restarts. *arXiv preprint arXiv:1608.03983*, 2016.

[23] I. Loshchilov and F. Hutter. Decoupled weight decay regularization. *arXiv preprint arXiv:1711.05101*, 2017.

[24] D. G. Lowe. Distinctive image features from scale-invariant keypoints. *International journal of computer vision*, 60:91–110, 2004.

[25] B. Maze, J. Adams, J. A. Duncan, N. Kalka, T. Miller, C. Otto, A. K. Jain, W. T. Niggel, J. Anderson, J. Cheney, et al. Iarpa janus benchmark-c: Face dataset and protocol. In *2018 international conference on biometrics (ICB)*, pages 158–165. IEEE, 2018.

[26] L. McInnes, J. Healy, and J. Melville. Umap: Uniform manifold approximation and projection for dimension reduction. *arXiv preprint arXiv:1802.03426*, 2018.

[27] Q. Meng, S. Zhao, Z. Huang, and F. Zhou. Magface: A universal representation for face recognition and quality assessment. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 14225–14234, 2021.

[28] P. Mishra and K. Sarawadekar. Polynomial learning rate policy with warm restart for deep neural network. In *TENCON 2019-2019 IEEE Region 10 Conference (TENCON)*, pages 2087–2092. IEEE, 2019.

[29] S. Moschoglou, A. Papaioannou, C. Sagonas, J. Deng, I. Kotsia, and S. Zafeiriou. Agedb: the first manually collected, in-the-wild age database. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshop*, volume 2, page 5, 2017.

[30] S. Sengupta, J.-C. Chen, C. Castillo, V. M. Patel, R. Chellappa, and D. 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.

[31] L. Torrey and J. Shavlik. Transfer learning. In *Handbook of research on machine learning applications and trends: algorithms, methods, and techniques*, pages 242–264. IGI global, 2010.

[32] P. Viola and M. Jones. Rapid object detection using a boosted cascade of simple features. In *Proceedings of the 2001 IEEE computer society conference on computer vision and pattern recognition. CVPR 2001*, volume 1, pages I–I. Ieee, 2001.

[33] H. Wang, Y. Wang, Z. Zhou, X. Ji, D. Gong, J. Zhou, Z. Li, and W. 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.

[34] K. Weiss, T. M. Khoshgoftaar, and D. Wang. A survey of transfer learning. *Journal of Big data*, 3:1–40, 2016.

[35] C. Whitelam, E. Taborsky, A. Blanton, B. Maze, J. Adams, T. Miller, N. Kalka, A. K. Jain, J. A. Duncan, K. 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.

[36] S. Zhang, L. Dong, X. Li, S. Zhang, X. Sun, S. Wang, J. Li, R. Hu, T. Zhang, F. Wu, et al. Instruction tuning for large language models: A survey. *arXiv preprint arXiv:2308.10792*, 2023.

[37] C. Zhou, P. Liu, P. Xu, S. Iyer, J. Sun, Y. Mao, X. Ma, A. Efrat, P. Yu, L. Yu, et al. Lima: Less is more for alignment. *Advances in Neural Information Processing Systems*, 36, 2024.

[38] Z. Zhu, G. Huang, J. Deng, Y. Ye, J. Huang, X. Chen, J. Zhu, T. Yang, J. Lu, D. 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.

[39] D. M. Ziegler, N. Stiennon, J. Wu, T. B. Brown, A. Radford, D. Amodei, P. Christiano, and G. Irving. Fine-tuning language models from human preferences. *arXiv preprint arXiv:1909.08593*, 2019.