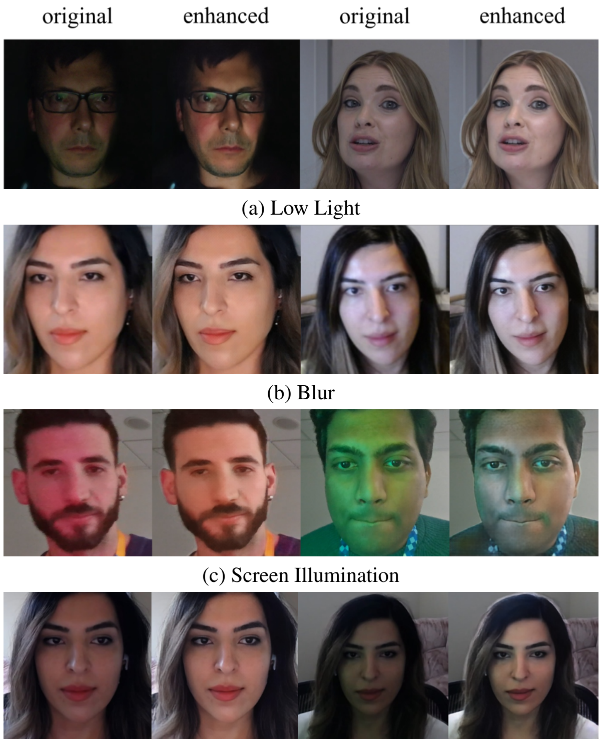
FRR-Net: A Real-Time Blind Face Restoration and Relighting Network

Samira Pouyanfar, Sunando Sengupta, Mahmoud Mohammadi, Ebey Abraham
Brett Bloomquist, Lukas Dauterman, Anjali Parikh, Steve Lim, and Eric Sommerlade
Microsoft
{sapouyan,susengup,mahmoha,ebeyabraham,brettbl,ludauter,anjalip,stlim,ersomme}@microsoft.com

Abstract

Face restoration models that mitigate low light, mixed lighting, poor camera quality conditions can benefit various applications, including video conferencing, image capture apps, among other uses. Many different models exist to address this problem. Although recent models generate impressive and high-fidelity faces, several important challenges remain, such as model efficiency, realistic texture and facial components, low-light environments, and screen illumination on the face. To tackle these challenges, we propose a simple, yet effective model called Face Restoration and Relighting Network (FRR-Net). The FRR-Net architecture includes an encoder-decoder model with a parallel distortion classifier which predicts the distortion types during training. This model is systematically scaled to balance network depth and width for better performance and efficiency trade-off. In addition, to generate the enhanced facial region, FRR-Net also utilizes a facial segmentation mask during the training, which not only helps the model performance but can also be used for further post-production uses. Furthermore, this work integrates a wide range of data degradation techniques to generate data for training to tackle both face enhancement and relighting. We demonstrate the effectiveness of our method by comparing it with several recent face restoration models. FRR-Net is computationally efficient and can perform inference at 13ms per frame on a low-powered Neural Processing Unit making it suitable for real-time face restoration applications.



(a) Low Light

(b) Blur

(c) Screen Illumination

(d) Low Light+Screen Illumination

Figure 1. Example input images captured under varying illumination conditions using a custom webcam and the corresponding enhanced version obtained by FRR-Net

1. Introduction

Enhancing the quality of faces in videos and images significantly improves the user experience in different applications (e.g., video conferencing and mobile apps). There are different conditions that may affect the image quality of the facial region including light/exposure (e.g., dark rooms, windows, lamps, etc.), camera focus blur, distance from the camera, screen illumination on the face, etc. This is further impacted by the quality of the attached camera hardware, resulting in a poor experience during video communication.

The goal of blind face restoration techniques is to recover face quality from unknown distortions. There are many studies on image super resolutions [24, 27, 28, 31, 45, 53, 55, 57] and face restoration [10, 13, 43, 44, 48]. However, there exist several challenges including: 1) How realistic the generated images are and/or the number of artifacts added to the face? 2) Are these models real-time and applicable to video applications? 3) How do these models work in different lighting environments (low light, high exposure, screen illumination)? In this paper, we address these challenges by developing an efficient full-refiner model that provides a great trade-off between accuracy (face quality) and computational efficiency.

Some work attempts to recover images from a specific single degradation (e.g., noise [8, 35, 58], blur [7, 23, 34], low-light [9, 11, 50], etc.), while more recent work focuses on multi-degradation [10, 32, 43, 44, 48]. The latter mainly utilizes some kind of priors to enhance the quality and fidelity of faces. These priors can be classified as facial or geometric priors [4, 5, 54], reference priors [6, 25, 26], and more recently generative priors [3, 32, 43, 49] or a combination of different priors [63].

One way to simulate real-world image degradation is to synthetically and randomly apply multi-degradation techniques to High-Quality (HQ) images during the training and try to recover the degraded Low-Quality (LQ) images as close as possible to the corresponding HQ images. Recent studies have followed [25, 26, 43] and adopted a degradation model as follows:

$$LQ = [(HQ \circledast k_\sigma) \downarrow_r + N_\delta]_{JPEG_q} \quad (1)$$

Where k, $\downarrow$, N, and $JPEG$ are blur kernel, down-sampling, noise, and JPEG compression respectively.

The above degradation model is designed to recover very low-quality images and therefore applies large-scale noise and down-sampling which sometimes completely destroys the image. This may not be applicable in many real-world scenarios such as video conferencing where the user is usually close to the camera, but there might be some noise, movement, or light conditions that affect the overall quality of the face. Therefore, we modified the current degradation model to generate more real-world scenarios. Specifically, we incorporate light/exposure and screen illumination distortion together with other common image degradation operations. In addition, we utilize the face region mask during training and also predict the mask as one of the outputs of the model. This helps the model to only focus on the facial region and achieve better performance.

Instead of facial and GAN priors which sometimes add artifacts and make the face/texture unrealistic (as shown in our experiments and also mentioned by [10]), we used a classifier to learn the distortion types applied to each image which is later integrated to the encoder features. This classifier is trained in parallel with the autoencoder model to further guide the decoder on how to recover from each specific distortion.

Another big challenge of face restoration is having a reasonable low-compute model that works in real-time on many devices. Many existing works are very large and only applicable for single image restoration and may not be suitable for real-time video applications [10, 44, 55]. There are also image enhancement models developed for mobile and low-computational cases but they are not designed to handle multi-degradation scenarios [9]. For this purpose, we propose a light yet effective model to enhance different types of distortions such as light, illumination, down-sampling, noise, etc. in real-time. Figure 1 shows a few real-world sample images enhanced by our model. The main contributions of this paper are as follows:

- We propose a computationally efficient model based on depth to space and dense blocks, which is capable of handling various distortions effectively. This model is carefully designed by examining different depth and width scaling factors controlling output channels and inner dense layers in each dense block, respectively, achieving 13ms inference times on a low-power Neural Processing Unit (NPU) [1], making it feasible for real-time applications and low-power devices.

- Our model includes a distortion-guided classifier that predicts the degradation type and uses that class information as a prior in the autoencoder

- We also incorporate a face segmentation mask and dice loss during training to only focus on the face region (limit the restoration region) and avoid the background.

- We propose a new degradation model that not only combines previous degradation techniques but also use light/exposure and illuminations distortions. To the best of our knowledge, this is the first work that combines all these degradation into one single model.

2. Related Work

Face Restoration. Face restoration methods learn a mapping from the input low-quality image to a high-quality one for the various sub-tasks (deblurring, denoising, artifact removal, etc.). These methods use natural images for training and can accurately improve images that closely match the training instance used [4, 25, 48]. Blind face distortion methods use artificial perturbations to introduce degradation into the training data [4, 10, 43]. They ease the need for costly data collection effort by allowing many different degradations to be applied to images.

With the increasing use of Generative Adversarial Networks (GANs) for generating synthetic faces, GANs can be used to create face restoration models. These methods involve using pre-trained GAN models such as StyleGAN2 [21] and fine-tuning it into an autoencoder model [3, 32, 49]. These methods struggle to create realistic-looking faces with fine-grained details. Panini-Net [44] addresses this issue by forcing the model to learn a degradation-aware feature representation that encodes features from the degraded image into the GAN representation [44]. It trains the model on a set of different degradations to learn features specific to each. GFP-GAN uses a two-step process, first using an autoencoder to remove the degradation, then using the features from the decoder along with the GAN to produce a high-quality face [43]. The combination allows spatial features from the decoder and GAN features to contribute to the output independently. Our method uses a degradation prior to augmenting the latent representation going into the decoder.

Geometric priors can be added to the latent information passed into the decoder [4,5,54]. VQFR [10] is a blind face restoration technique that uses a VQ code-book along with a parallel decoder model [10]. The VQ codebook works to remove the degradation and the parallel decoders create an accurate restoration.

Model Shrinking. Developing computationally affordable neural networks for use cases with limited resources has been rising in recent research. These efforts can be categorized mainly into model compression methods and small model architectures. The model compression techniques including quantization [37, 47, 56], model pruning [12, 29], and knowledge distillation [15, 37] mainly focus on pre-trained models to achieve sparse or reduced versions of the trained model before deploying to the production environment. Designing a small model, on the other hand, addresses the accuracy and efficiency trade-off by introducing efficient building blocks and high-level architectural parameters to shrink/expand the model based on target environment restrictions and requirements. Depthwise convolution layers [19, 60, 61], Fire Module [17], neural architecture search [42, 46, 51], and layer dimensions multipliers [16, 38, 41] mainly focus on the architectural design of neural networks. In this work, we extend the idea of model shrinking/scaling [16, 41] and carefully design a network to balance the network's depth and width which leads to better performance and speed.

3. Methodology

The architecture of the proposed FRR-NET framework is illustrated in Figure 2. First, we extract a video input frame. In this example, the face is not very clear and the room is dark. The facial region to be enhanced is determined using an existing facial landmark estimation method. The detected face area as well as the generated facial mask (using Mask-RCNN [14]) is used as the input of the FRR-Net model. This model consists of an autoencoder, distortion classifier, and mask generation. Pre-trained features from VGGFace model [36] are extracted and used as the classifier input. The output of the model includes an enhanced face and the corresponding face segmentation mask. Thus, the model discards the background and only focuses on enhancing the face area while learning how to improve segmenting the face area. These two outputs are blended with the original image (combining the enhanced face area and the original background) to generate the final output frame.

3.1. Degradation Model

Existing face restoration models can restore the face from moderately to extremely noisy or poor-quality photos. Our observations, however, demonstrate that some models produce artifacts, identity shifts, inconsistent eye coloration, or unnatural facial textures in real-world faces. A few examples are shown in the experimental section and supplementary material. This may not be acceptable in applications where the intent is to overcome difficult environmental lighting, camera limitations, and other distortions while preserving the individual's appearance. Thus, we propose a modified version of the degradation model in this paper. Figure 3 depicts a few samples from existing degradation models compared to our degradation model. The previous model usually contains highly down-scaled or very noisy images while we used a smoother, but more diverse, type of distortion to train our model. Similar to Equation 2, the proposed degradation model can be formalized as:

$$LQ = [(((HQ) \downarrow_r \xi_e \eta_j C_\gamma) + N_\delta) \circledast k_\sigma]_{JPEG_q} \qquad (2)$$

Where $\downarrow$, ξ, η, C, N, k, and $JPEG$ are down-sampling, exposure, color jitter (brightness, contrast, saturation, hue), chromatic, noise, blur, and JPEG compression, respectively. We randomly sample r, e, j, γ, δ, σ, and q.

Algorithm 1 shows the detailed steps of our degradation model. The input of our degradation model includes the list of HQ images for each batch, the $Percent_{dist}$ which is the percentage of images that will be distorted per batch, and distortion ranges (r, e, j, γ, δ, σ, and q) for each distortion type. For each image, we first check if we reach the distortion percentage limit for that batch, if not we randomly apply one or more distortions to the image, such as downscaling, exposure change per RGB channel, color jitters, chromatic, additive white Gaussian noise, and Gaussian blur convolution. Lastly, we apply JPEG compression artifacts. If the percentage of images exceeds the $Percent_{dist}$ limit, we simply use the same image without any distortion as the target image. The goal is to let the model see both LQ and HQ images during training and add some robustness to the system, where we can ensure that a good image is not over-enhanced, avoiding unnecessary overcompensated, unnatural artifacts. While we apply each degradation on HQ images, we also generate the ground truth class labels for the distortion classifier depending on what types of distortion are presented. We categorize the distortions into three general classes: noise (Gaussian noise, JPEG, and chromatic), blur (Gaussian blur and downscale), and relight (exposure or color/illumination). In our multi-label classification method, each of the three class labels has a value of 0

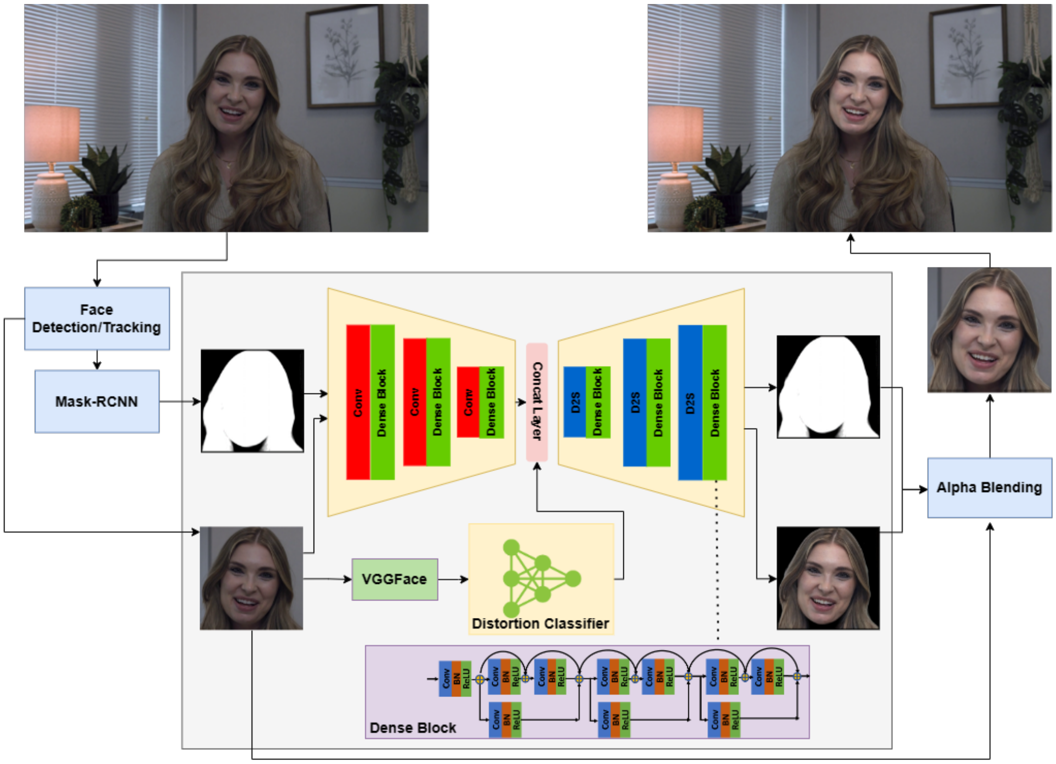
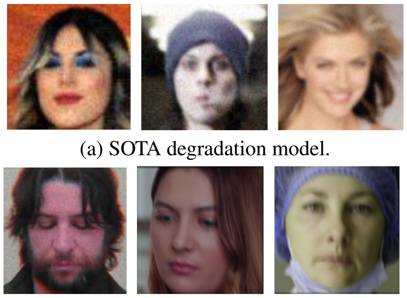


Figure 2. Overview of the FRR-NET framework.


(a) SOTA degradation model.

(b) Our degradation model.

Figure 3. Sample comparisons between an existing degradation model [43] vs ours.

or 1, so an image with poor exposure and noise is labeled as 101, while an image with noise and blur is labeled as 110.

3.2. Model Architecture

Auto Encoder Network: As shown in Figure 2, FRR-Net consists of an encoder and decoder (generator). The encoder gets the output from the degradation model (LQ images) and goes through several Conv layers followed with dense blocks [62]. Each dense block includes several residual connections as shown in Figure 2 and extracts rich local features via dense residual connections.

To help the decoder correctly restore the image, a distortion classifier predicts the types of degradation present in the input image. For this, we first take the LQ image features from the pre-trained VGGFace model [36] and pass them along with the label produced by the degradation model to the classifier. We defined three main classes (noise, blur, and exposure) as presented in Algorithm 1 to train the classifier to anticipate these primary types of distortions in the image. The autoencoder model and the classifier are both being trained simultaneously. The classifier's output is combined with the encoder's final output.

The generator gets the encoder output and applies a series of depth-to-space transformations, rearranging the data from depth (channel) to space (weight and height), followed by dense blocks (Figure 2). Finally, we apply two CNN layers with channel sizes three and one to generate enhanced output and facial mask, respectively. Facial segmentation is used to let the model only focus on enhancing the facial region while predicting the face boundary. These output layers integrate the original input to generate the final frame with an enhanced face.

Model Shrinking: Convolution layers, as the basic building blocks of image-processing neural networks, can be represented as a function of channels (C_i), height (H_i), and width (W_i). The Encoder/Generator components $(\mathcal{N})$ of the FRR-Net shown in (Figure 2) are presented as lists

```
Algorithm 1: Degradation algorithm
  Data: $HQ, Percent_{dist}, r, e, \gamma, \delta, \sigma, q$
  Result: $LQ$
  $Percent_{accum} \leftarrow 0$;
  $Count_{dist} \leftarrow 0$;
  $N \leftarrow len(HQ)$;
  $LQ \leftarrow \emptyset$ ;
  for $X \in HQ$ do
      $L \leftarrow \{0, 0, 0\}$ ;
      $Percent_{accum} \leftarrow Count_{dist}/N$;
      if $Percent_{accum} \leq Percent_{dist}$ then
          $Y, is\_downsacle \leftarrow DownScale(X, r)$;
          $Y, is\_exposure \leftarrow RGBExposure(Y, e)$;
          $Y, is\_chromatic \leftarrow Chromatic(Y, \gamma)$;
          $Y, is\_noisy \leftarrow Y + Noise(Y, \delta)$;
          $Y, is\_blurry \leftarrow Y \circledast Blur(Y, \sigma)$;
          $Y, is\_jpeg \leftarrow JPEG(Y, q)$;
          $Y, is\_color \leftarrow ColorJitter(Y, f_b, f_s, f_h)$;
          $Count_{dist} \leftarrow Count_{dist} + 1$;
          if $is\_noisy$ or $is\_jpeg$ or $is\_chromatic$
            then
              $L\{0\} \leftarrow 1$;
          end
          if $is\_blurry$ or $is\_downsacle$ then
              $L\{1\} \leftarrow 1$;
          end
          if $is\_exposure$ or $is\_color$ then
              $L\{2\} \leftarrow 1$;
          end
      else
          $Y \leftarrow X$;
      end
      $LQ \leftarrow LQ + \{Y\}$ ;
      $Label \leftarrow Label + \{L\}$ ;
  end
```

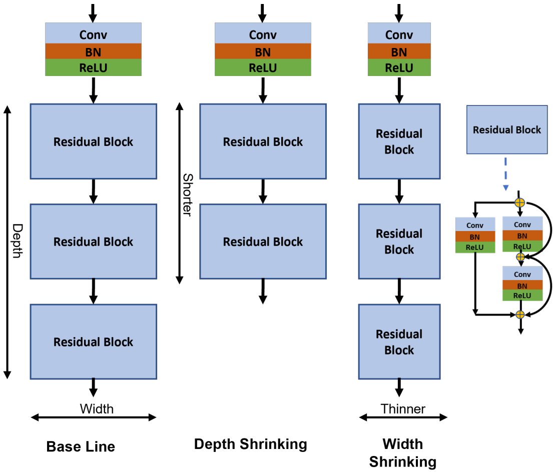


Figure 4. Using Depth and Width parameters to build various smaller versions of the base model

(L_i) of dense blocks $(\mathcal{D})$ composed of multiple residual layers. Our focus is to adjust the input/output channels of the convolution layers as well as the number of residual layers in each dense block expressed as the width and depth of the model correspondingly. We use the width (w) and depth (d) parameters to generate various lightweight versions of the base model as shown in (Figure 4). Following the notation from [41], this can be formulated as:

$$\mathcal{N}(d, w) = \bigodot_{i=1,2,\ldots} \mathcal{D}_i^{d.L_i}(Conv(w.C_i, H_i, W_i)) \quad (3)$$

We use the same width and depth values for both the encoder and generator. We apply the width values $w \in [8, 12, 16]$ to the first dense block, and as all the blocks are connected sequentially, it will thin all the blocks uniformly. The maximum number of residual layers in each dense block is 3 and we use the depth values of $d \in [1, 2, 3]$ to generate dense blocks with shorter depths. Using the depth and width parameter, we can control the computational cost of each convolution layer (and dense blocks) expressed by the Multiply-Add Cumulation (MAC), which has a direct impact on the inference time of the model. Our experiments using various versions of our baseline model are shown in the experimental section.

3.3. Model Objectives

Reconstruction Losses: We employed the reconstruction loss ($L_{rec} = L_{L1} + L_{hub}$) using the widely-used L1 loss $\mathcal{L}_{L1} = ||y - \hat{y}||_1$ and Channel-wise Huber loss to measure how far the enhanced images ($\hat{y}$) are from the ground truth (y). Channel-wise Huber loss is more robust than the L1 and not as sensitive to outliers as L2. This enables the Huber loss to be effective in reducing the "averaging problem" [2] and generating higher quality images in cases of noise degradation [30].

$$\mathcal{L}_{hub} = \begin{cases} \frac{1}{2}(y - \hat{y})^2 & |(y - \hat{y})| < \alpha \\ \alpha(|(y - \hat{y})| - \frac{1}{2}\alpha) & otherwise \end{cases} \quad (4)$$

Color Enhancement Loss: Similar to [39, 52], to preserve the color constancy between the enhanced image and the ground truth we applied the Angular loss as well.

$$\mathcal{L}_{ang} = \arccos \frac{y.\hat{y}}{||y||\ ||\hat{y}||} \quad (5)$$

Perceptual Loss: Similar to [18, 43], we used the MSE loss $\mathcal{L}_{per} = ||\theta(y) - \theta(\hat{y})||_2$ where Θ is the features extracted from the information distilled in layers 3, 8, and 15 of a pre-trained VGGFace model [36] immediately before the average and max pooling layers.

Style Loss: We also applied style loss to generate a more realistic image texture with a style similar to the ground truth.

$$\mathcal{L}_{style} = ||Gram(\theta(y)) - Gram(\theta(\hat{y}))||_2 \tag{6}$$

here $Gram$ is the Gram matrix which estimates the feature of style across all layers.

Dice Loss: We used Dice loss as a soft approximation of the Dice metric to penalize the overlapping between the predicted and ground truth images [33, 40]:

$$\mathcal{L}_{dice}(m, \hat{m}) = 1 - \frac{2 * (\sum m * \hat{m})}{\sum m^2 + \sum \hat{m}^2 + \epsilon} \tag{7}$$

here m and $\hat{m}$ are mask and predicted mask, respectively, and ϵ is added to avoid dividing by zero and to provide more computing stability.

The overall model objective is a weighted summation of the above losses:

$$\mathcal{L}_{total} = \lambda_{rec}.\mathcal{L}_{rec} + \lambda_{ang}.\mathcal{L}_{ang} + \lambda_{per}.\mathcal{L}_{per} + \lambda_{style}.\mathcal{L}_{style} + \lambda_{dice}.\mathcal{L}_{dice} \tag{8}$$

4. Experiments

4.1. Datasets and Implementations

For training, we used FFHQ [20] and synthetic data generated by StyleGAN2 [21][1]. In total, around 65,000 HQ images are used for training. All images are resized to 336x336. We applied our degradation model with the following factors: $r = \{2 : 4\}$, $e = \{0.4 : 1.5\}$, $j = \{0.1 : 0.2\}$, $\gamma = \{1 : 1.5\}$, $\delta = \{0.01 : 0.11\}$, $\sigma = \{0.2 : 5\}$, $q = \{0 : 100\}$. $Percent_{dist}$ is also assigned to 0.85 (85% of the time degradation is applied, while 15% of the time the original image is used during the training). We used two datasets to evaluate the performance of the FRR-Net. 1) A new dataset generated from StyleGAN2 (there is no overlap between training and testing data) consisting of 4,000 images. For this data, we applied our degradation model with the same factors as training. 2) We also evaluated our model on 3,000 CelebA-HQ test data following the previous work in face restoration [43]. Although our model is not trained on the same level of image degradation as CelebA-Test, we still want to validate how our model works in those scenarios compared to the most recent state-of-the-art models.

The training batch size is set to 8. Layers 3, 8, and 15 of the pre-trained VGGFace [36] are utilized for face feature extraction and used for L_{per} and L_{style}. We trained FRRNet for a total of 500k iterations with the Adam optimizer [22], learning rate is set to $2e-4$, and loss hyper-parameters are set as follows: $\lambda_{rec} = 1$, $\lambda_{ang} = 1$, $\lambda_{per} = 0.04$, $\lambda_{style} = 4e-5$, $\lambda_{dice} = 1$. Our model is implemented in both TensorFlow and PyTorch. We used Azure Machine Learning for hyper-parameter tuning and an NVIDIA TITAN RTX 24 GB RAM for training.

Width	Depth	PSNR ↑	LPIPS ↓	MAC (GFlops)	NPU (ms) ↓
16	3	30.19	0.22	40.42	18.50
12	**2**	30.00	0.23	**16.20**	**13.70**
8	1	26.91	0.30	11.26	10.70

Table 1. Inference time on NPU for various Width-Depth versions.

Our evaluation metrics include Peak Signal-to-Noise Ratio (PSNR) and Structural Similarity (SSIM) as pixel-wise metrics as well as Learned Perceptual Image Patch Similarity (LPIPS [59]) as a perceptual metric.

4.2. Evaluating Model Shrinking

We evaluated different versions of the baseline model to find the most efficient model architecture satisfying the inference time budget. We deployed these model versions on commodity hardware to obtain the amount of speedup in different environments. We used the SNPE toolkit [1] to convert a PyTorch/ONNX model to an intermediate representation called DLC, which is then deployed to the Qualcomm Snapdragon processor, which has an NPU optimized for running 8-bit quantized convolutional neural networks. Table 1 shows the quality metrics and inference time of various versions of the baseline model generated from different combinations of depth $\in \{1, 2, 3\}$ and width $\in \{16, 12, 8\}$. The baseline model has a width of 16 and a depth of 3 (W16, D3). While the trained model with (W8, D1) has the best inference time, its low-quality LPIPS and PSNR metrics values prevent it from being selected as the optimum model. The model with (W12, D2) depicts accuracy similar to the baseline and simultaneously has a reduced 13.7ms inference time. This accuracy and efficiency trade-off makes this lightweight version a candidate model we selected as the primary model for all of our experiments in this work.

4.3. Comparison with State-of-the-art Models

We compared FRR-Net with several state-of-the-art multi-degradation face restoration models including GFP-GAN [43], Panini-Net [44], and VQFR [10] as the benchmark. To the best of our knowledge, there is no work that integrates both relighting and face restoration; however, GFP-GAN applies jitter color distortion and uses color prior from generative facial prior to color/light enhancement.

StyleGAN Data: As mentioned in Section 3.1, the proposed degradation model not only generates noisy, blurry, down-scaled faces but also incorporates different exposures and screen illuminations to adjust the light on the face. For this purpose, we apply our degradation model to the new

[1] Collected from https://thispersondoesnotexist.com/

Model	PSNR ↑	LPIPS ↓	SSIM ↑
GFP-GAN	27.36	0.32	0.85
VQFR	27.45	0.32	0.83
Panini-Net	26.65	0.49	0.80
FRR-Net	**30.00**	**0.23**	**0.87**

Table 2. Comparison results on StyleGAN validation data.

Model	PSNR ↑	LPIPS ↓	SSIM ↑
GFP-GAN	**26.12**	0.43	**0.78**
VQFR	25.75	**0.42**	0.75
Panini-Net	24.87	0.50	0.67
FRR-Net	25.96	0.43	0.70

Table 3. Comparison results on CelebA validation data.

synthetic data generated from StyleGAN2. Table 2 shows the quantitative results of this experiment. From these results, we can see FRR-Net obtained the best performance regarding all three metrics, compared to the other models, by a large margin. More specifically, our model improves both pixel-wise and human perceptual metrics on this dataset. GFP-GAN and VQFR perform very close to each other on this dataset and Panini-Net has the lowest performance on all three metrics. Figure 5 shows several samples from the StyleGAN data. Although all models remove the blur/noise, we can see some models change the eye color (e.g., VQFR in the first sample), add artifacts (e.g., Panini-Net results in first row), or generate a different face texture (e.g., GFPGAN in the first row) as compared to the ground truth. Overall, our model generates competitive results compared to these top models while enhancing the light and color.

CelebA-Test: Although our degradation model is designed to generate various distortions such as noise, blur, exposure, screen illumination, etc., FRR-Net is not trained on extremely degraded faces such as CelebA-Test data. However, to have a fair comparison with the existing work, we evaluated this data and compared it with the benchmark models. The quantitative results are presented in Table 3. From this, GFP-GAN achieves the best performance regarding PSNR and SSIM, while VQFR has a slightly better LPIPS value than GFP-GAN and our FRR-Net. The performance results of FRR-Net on this dataset are still very competitive compared to the state-of-the-art models. FRR-Net performs better than Panini-Net regarding all three metrics and has very close PSNR and LPIPS values to GFP-GAN. These results demonstrate that the proposed model achieves competitive performance on extremely distorted images while addressing other challenges such as speed, low to zero artifact, realistic texture, etc.

Real-world Data: Figure 6 demonstrates the qualitative results on a few real-world datasets[2]. According to these results, GFP-GAN generates the most high-fidelity faces compared to all other models, however, it adds artifacts and makeup such as lashes (third row) or eye color change (second row), and also generates unrealistic textures. Panini-Net and VQFR did not perform well on real-world data and destroyed eyes or lips along with adding many artifacts. Our FRR-Net smoothly enhances the face quality while improving the low-light or screen illumination conditions and does not change the identity nor add any artifacts to the face. Thus, it is a better candidate for real-world applications with moderate distortion such as video conferencing.

Model	CPU (ms)	GPU (ms)	parameters (M)	size (MB)
GFP-GAN	840	50	76.2	587
VQFR	14000	250	76.3	293
Panini-Net	NA	104	131	1500
FRR-Net	**220**	**15**	**7.2**	**29**

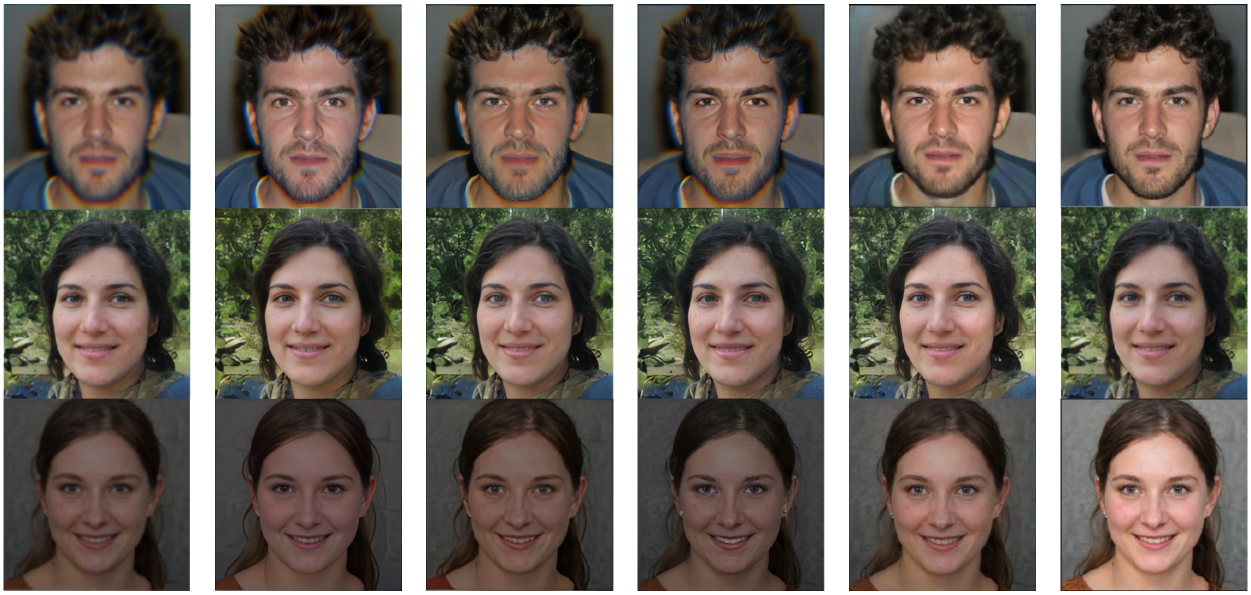
Table 4. Comparison of inference time and computational cost of FRR-Net compared to top selected face restoration models.

Inference Time Comparison: FRR-Net is specifically designed to enhance faces for video applications. Therefore, it is important that it performs fast enough on different devices. Table 4 shows the comparison of speed and size between the benchmark models and our proposed model. We tested the results on the NVidia GeForce RTX 2080 GPU and Intel Xeon Gold 5218 CPU @ 2.30GHz. Our model is almost 20 times smaller than GFP-GAN and is 3 to 4 times faster than GFP-GAN on GPU and CPU. Panini-Net is the largest model (1.5GB) and VQFR is the slowest on both CPU and GPU devices. FFR-Net is small and efficient and can recover faces in real-time (about 50 fps on GPU).

As previously shown in Table 1, we also evaluate our model on low power Qualcomm NPU devices (8cx gen3) by quantizing the model using the Snapdragon SDK and computed inference times for DSP execution. For both variety of GPU and NPU hardware accelerators, the proposed model has potential of being used for real-time video conferencing applications.

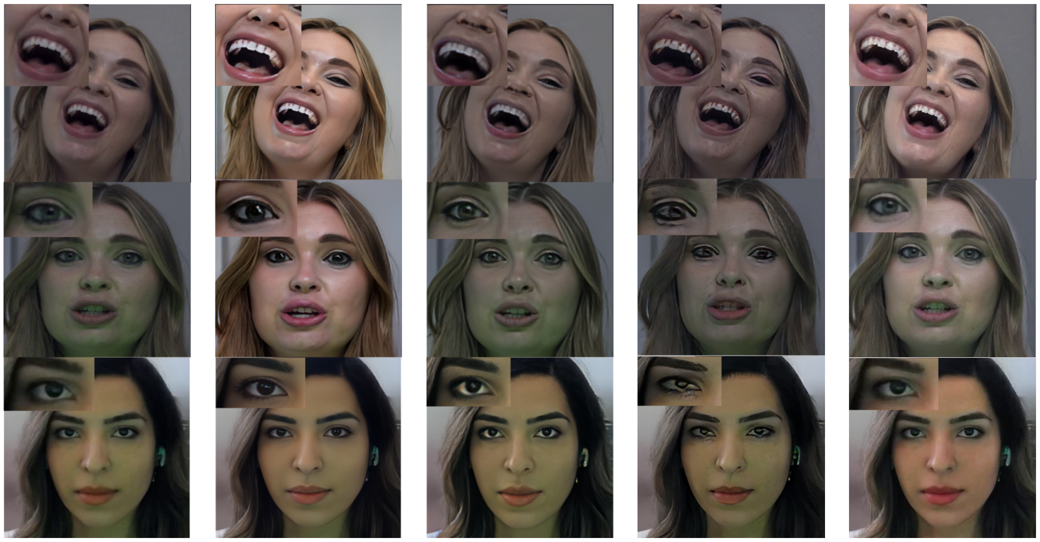
For more details on ablation studies regrading the importance of classifier, losses, segmentation, and image size, as well as more visualization results, please refer to the supplementary materials.

[2] These samples are collected internally following CVPR ethics guidelines, and all users agree



(a) Input (b) GPFGAN (c) VQFR (d) Panini-Net (e) FRR-Net (Ours) (f) Ground Truth

Figure 5. Comparisons with SOTA face restoration models: GFP-GAN, VQFR, Panini-Net on real-world images on StyleGAN data.



(a) Input (b) GPFGAN (c) VQFR (d) Panini-Net (e) FRR-Net (Ours)

Figure 6. Comparisons with SOTA face restoration models: GFP-GAN, VQFR, Panini-Net on real-world images (zoom for better view).

5. Conclusion

In this paper, we propose FRR-Net for face restoration and relighting. The novelty of the FRR-Net model includes: 1) a new autoencoder utilizing Depth to Space following with Dense layers, with a parallel distortion classification and facial segmentation mask, 2) a comprehensive distortion model containing noise, blur, exposure, and screen illumination 3) an efficient network design with significantly reduced parameters that achieves real-time performance on various battery-powered devices. The experimental results show that FRR-Net is competitive compared to the state-of-the-art models in face restoration and provides an excellent balance of accuracy and latency, making it suitable for real-time image/video face restoration/relighting across CPU, GPU, and NPU hardware. In the future, we will look at expanding our model's operating range by incorporating more extreme and realistic distortions to improve handling of problems such as very low illumination and motion compression, just to name a few.

References

[1] Snapdragon neural processing engine sdk. https://developer.qualcomm.com/sites/default/files/docs/snpe/. 2, 6

[2] Yousef Atoum, Mao Ye, Liu Ren, Ying Tai, and Xiaoming Liu. Color-wise attention network for low-light image enhancement. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops*, pages 506–507, 2020. 5

[3] Kelvin CK Chan, Xintao Wang, Xiangyu Xu, Jinwei Gu, and Chen Change Loy. GLEAN: Generative latent bank for large-factor image super-resolution. In *Proceedings of the IEEE/CVF Conference on Computer Vision and pattern recognition*, pages 14245–14254, 2021. 2

[4] Chaofeng Chen, Xiaoming Li, Lingbo Yang, Xianhui Lin, Lei Zhang, and Kwan-Yee K Wong. Progressive semantic-aware style transformation for blind face restoration. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 11896–11905, 2021. 2, 3

[5] Yu Chen, Ying Tai, Xiaoming Liu, Chunhua Shen, and Jian Yang. FSRNET: End-to-end learning face super-resolution with facial priors. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 2492–2501, 2018. 2, 3

[6] Berk Dogan, Shuhang Gu, and Radu Timofte. Exemplar guided face image super-resolution without facial landmarks. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops*, pages 0–0, 2019. 2

[7] Jiangxin Dong, Stefan Roth, and Bernt Schiele. Deep wiener deconvolution: Wiener meets deep learning for image deblurring. *Advances in Neural Information Processing Systems*, 33:1048–1059, 2020. 2

[8] Majed El Helou, Ruofan Zhou, and Sabine Süsstrunk. Stochastic frequency masking to improve super-resolution and denoising networks. In *European Conference on Computer Vision*, pages 749–766. Springer, 2020. 2

[9] Zhicheng Fu, Miao Song, Chao Ma, Joseph Nasti, Vivek Tyagi, Grant Lloyd, and Wei Tang. An efficient hybrid model for low-light image enhancement in mobile devices. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 3057–3066, 2022. 2

[10] Yuchao Gu, Xintao Wang, Liangbin Xie, Chao Dong, Gen Li, Ying Shan, and Ming-Ming Cheng. VQFR: Blind face restoration with vector-quantized dictionary and parallel decoder. In *Proceedings of the European Conference on Computer Vision*, 2022. 1, 2, 3, 6

[11] Chunle Guo, Chongyi Li, Jichang Guo, Chen Change Loy, Junhui Hou, Sam Kwong, and Runmin Cong. Zero-reference deep curve estimation for low-light image enhancement. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 1780–1789, 2020. 2

[12] Jinyang Guo, Wanli Ouyang, and Dong Xu. Multi-dimensional pruning: A unified framework for model compression. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 1508–1517, 2020. 3

[13] Jingwen He, Wu Shi, Kai Chen, Lean Fu, and Chao Dong. GCFSR: a generative and controllable face super resolution method without facial and gan priors. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 1889–1898, 2022. 1

[14] Kaiming He, Georgia Gkioxari, Piotr Dollár, and Ross Girshick. Mask R-CNN. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 2961–2969, 2017. 3

[15] Geoffrey Hinton, Oriol Vinyals, and Jeff Dean. Distilling the knowledge in a neural network. *arXiv preprint arXiv:1503.02531*, 2015. 3

[16] Andrew G Howard, Menglong Zhu, Bo Chen, Dmitry Kalenichenko, Weijun Wang, Tobias Weyand, Marco Andreetto, and Hartwig Adam. MobileNets: Efficient convolutional neural networks for mobile vision applications. *arXiv preprint arXiv:1704.04861*, 2017. 3

[17] Forrest N Iandola, Song Han, Matthew W Moskewicz, Khalid Ashraf, William J Dally, and Kurt Keutzer. SqueezeNet: AlexNet-level accuracy with 50x fewer parameters and¡ 0.5 mb model size. *arXiv preprint arXiv:1602.07360*, 2016. 3

[18] Justin Johnson, Alexandre Alahi, and Li Fei-Fei. Perceptual losses for real-time style transfer and super-resolution. In *European Conference on Computer Vision*, pages 694–711. Springer, 2016. 5

[19] Lukasz Kaiser, Aidan N Gomez, and Francois Chollet. Depthwise separable convolutions for neural machine translation. *arXiv preprint arXiv:1706.03059*, 2017. 3

[20] Tero Karras, Samuli Laine, and Timo Aila. A style-based generator architecture for generative adversarial networks. In *Proceedings of the IEEE/CVF Conference on Computer Vision and pattern recognition*, pages 4401–4410, 2019. 6

[21] Tero Karras, Samuli Laine, Miika Aittala, Janne Hellsten, Jaakko Lehtinen, and Timo Aila. Analyzing and improving the image quality of StyleGAN. In *Proceedings of the IEEE/CVF Conference on Computer Vision and pattern recognition*, pages 8110–8119, 2020. 2, 6

[22] Diederik P Kingma and Jimmy Ba. Adam: A method for stochastic optimization. In *3rd International Conference on Learning Representations, ICLR 2015*, 2015. 6

[23] Orest Kupyn, Volodymyr Budzan, Mykola Mykhailych, Dmytro Mishkin, and Jiří Matas. DeblurGAN: Blind motion deblurring using conditional adversarial networks. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 8183–8192, 2018. 2

[24] Hunsang Lee, Hyesong Choi, Kwanghoon Sohn, and Dongbo Min. KNN local attention for image restoration. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 2139–2149, 2022. 1

[25] Xiaoming Li, Chaofeng Chen, Shangchen Zhou, Xianhui Lin, Wangmeng Zuo, and Lei Zhang. Blind face restoration via deep multi-scale component dictionaries. In *European Conference on Computer Vision*, pages 399–415. Springer, 2020. 2

[26] Xiaoming Li, Ming Liu, Yuting Ye, Wangmeng Zuo, Liang Lin, and Ruigang Yang. Learning warped guidance for blind

face restoration. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pages 272–289, 2018. 2

[27] Jingyun Liang, Jiezhang Cao, Guolei Sun, Kai Zhang, Luc Van Gool, and Radu Timofte. SwinIR: Image restoration using swin transformer. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 1833–1844, 2021. 1

[28] Ding Liu, Bihan Wen, Yuchen Fan, Chen Change Loy, and Thomas S Huang. Non-local recurrent network for image restoration. *Advances in Neural Information Processing Systems*, 31, 2018. 1

[29] Jian-Hao Luo, Jianxin Wu, and Weiyao Lin. ThiNet: A filter level pruning method for deep neural network compression. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 5058–5066, 2017. 3

[30] Ryo Matsuoka, Shunsuke Ono, and Masahiro Okuda. Transformed-domain robust multiple-exposure blending with huber loss. *IEEE Access*, 7:162282–162296, 2019. 5

[31] Yiqun Mei, Yuchen Fan, and Yuqian Zhou. Image super-resolution with non-local sparse attention. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 3517–3526, 2021. 1

[32] Sachit Menon, Alexandru Damian, Shijia Hu, Nikhil Ravi, and Cynthia Rudin. PULSE: Self-supervised photo upsampling via latent space exploration of generative models. In *Proceedings of the ieee/cvf Conference on Computer Vision and pattern recognition*, pages 2437–2445, 2020. 2

[33] Fausto Milletari, Nassir Navab, and Seyed-Ahmad Ahmadi. V-Net: Fully convolutional neural networks for volumetric medical image segmentation. *2016 Fourth International Conference on 3D Vision (3DV)*, Oct 2016. 6

[34] Yuesong Nan, Yuhui Quan, and Hui Ji. Variational-EM-based deep learning for noise-blind image deblurring. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 3626–3635, 2020. 2

[35] Tongyao Pang, Huan Zheng, Yuhui Quan, and Hui Ji. Recorrupted-to-recorrupted: unsupervised deep learning for image denoising. In *Proceedings of the IEEE on Computer Vision and Pattern Recognition*, pages 2043–2052, 2021. 2

[36] Omkar M Parkhi, Andrea Vedaldi, and Andrew Zisserman. Deep face recognition. In *British Machine Vision Association*, 2015. 3, 4, 5, 6

[37] Antonio Polino, Razvan Pascanu, and Dan Alistarh. Model compression via distillation and quantization. In *6th International Conference on Learning Representations*, 2018. 3

[38] Mark Sandler, Andrew Howard, Menglong Zhu, Andrey Zhmoginov, and Liang-Chieh Chen. MobileNetV2: Inverted residuals and linear bottlenecks. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 4510–4520, 2018. 3

[39] Oleksii Sidorov. Artificial color constancy via GoogleNet with angular loss function. *Applied Artificial Intelligence*, 34(9):643–655, 2020. 5

[40] Carole H Sudre, Wenqi Li, Tom Vercauteren, Sebastien Ourselin, and M Jorge Cardoso. Generalised dice overlap as a deep learning loss function for highly unbalanced segmentations. In *Deep learning in medical image analysis and multimodal learning for clinical decision support*, pages 240–248. Springer, 2017.

[41] Mingxing Tan and Quoc Le. EfficientNet: Rethinking model scaling for convolutional neural networks. In *International Conference on Machine Learning*, pages 6105–6114. PMLR, 2019. 3, 5

[42] Ning Wang, Yang Gao, Hao Chen, Peng Wang, Zhi Tian, Chunhua Shen, and Yanning Zhang. NAS-FCOS: Fast neural architecture search for object detection. In *proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 11943–11951, 2020. 3

[43] Xintao Wang, Yu Li, Honglun Zhang, and Ying Shan. Towards real-world blind face restoration with generative facial prior. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 9168–9178, 2021. 1, 2, 3, 4, 5, 6

[44] Yinhuai Wang, Yujie Hu, and Jian Zhang. Panini-Net: Gan prior based degradation-aware feature interpolation for face restoration. In *Proceedings of the AAAI Conference on Artificial Intelligence*, 2022. 1, 2, 6

[45] Zhendong Wang, Xiaodong Cun, Jianmin Bao, Wengang Zhou, Jianzhuang Liu, and Houqiang Li. Uformer: A general u-shaped transformer for image restoration. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 17683–17693, 2022. 1

[46] Bichen Wu, Xiaoliang Dai, Peizhao Zhang, Yanghan Wang, Fei Sun, Yiming Wu, Yuandong Tian, Peter Vajda, Yangqing Jia, and Kurt Keutzer. FBNet: Hardware-aware efficient convnet design via differentiable neural architecture search. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 10734–10742, 2019. 3

[47] Jiaxiang Wu, Cong Leng, Yuhang Wang, Qinghao Hu, and Jian Cheng. Quantized convolutional neural networks for mobile devices. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 4820–4828, 2016. 3

[48] Lingbo Yang, Shanshe Wang, Siwei Ma, Wen Gao, Chang Liu, Pan Wang, and Peiran Ren. HiFaceGAN: Face renovation via collaborative suppression and replenishment. In *Proceedings of the 28th ACM International Conference on Multimedia*, pages 1551–1560, 2020. 1, 2

[49] Tao Yang, Peiran Ren, Xuansong Xie, and Lei Zhang. Gan prior embedded network for blind face restoration in the wild. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 672–681, 2021. 2

[50] Wenhan Yang, Shiqi Wang, Yuming Fang, Yue Wang, and Jiaying Liu. From fidelity to perceptual quality: A semi-supervised approach for low-light image enhancement. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 3063–3072, 2020. 2

[51] Zhaohui Yang, Yunhe Wang, Xinghao Chen, Boxin Shi, Chao Xu, Chunjing Xu, Qi Tian, and Chang Xu. CARS: Continuous evolution for efficient neural architecture search. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 1829–1838, 2020. 3

[52] Huanglin Yu, Ke Chen, Kaiqi Wang, Yanlin Qian, Zhaoxiang Zhang, and Kui Jia. Cascading convolutional color con-

stancy. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 34, pages 12725–12732, 2020. 5

[53] Ke Yu, Xintao Wang, Chao Dong, Xiaoou Tang, and Chen Change Loy. Path-Restore: Learning network path selection for image restoration. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 44(10):7078–7092, 2022. 1

[54] Xin Yu, Basura Fernando, Bernard Ghanem, Fatih Porikli, and Richard Hartley. Face super-resolution guided by facial component heatmaps. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pages 217–233, 2018. 2, 3

[55] Syed Waqas Zamir, Aditya Arora, Salman Khan, Munawar Hayat, Fahad Shahbaz Khan, and Ming-Hsuan Yang. Restormer: Efficient transformer for high-resolution image restoration. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 5728–5739, 2022. 1, 2

[56] Dongqing Zhang, Jiaolong Yang, Dongqiangzi Ye, and Gang Hua. LQ-Nets: Learned quantization for highly accurate and compact deep neural networks. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pages 365–382, 2018. 3

[57] Kai Zhang, Yawei Li, Wangmeng Zuo, Lei Zhang, Luc Van Gool, and Radu Timofte. Plug-and-Play image restoration with deep denoiser prior. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 44(10):6360–6376, 2022. 1

[58] Kai Zhang, Wangmeng Zuo, Yunjin Chen, Deyu Meng, and Lei Zhang. Beyond a gaussian denoiser: Residual learning of deep CNN for image denoising. *IEEE Transactions on Image Processing*, 26(7):3142–3155, 2017. 2

[59] Richard Zhang, Phillip Isola, Alexei A Efros, Eli Shechtman, and Oliver Wang. The unreasonable effectiveness of deep features as a perceptual metric. In *Proceedings of the IEEE Conference on Computer Vision and pattern recognition*, pages 586–595, 2018. 6

[60] Ru Zhang, Feng Zhu, Jianyi Liu, and Gongshen Liu. Depthwise separable convolutions and multi-level pooling for an efficient spatial CNN-based steganalysis. *IEEE Transactions on Information Forensics and Security*, 15:1138–1150, 2019. 3

[61] Xiangyu Zhang, Xinyu Zhou, Mengxiao Lin, and Jian Sun. ShuffleNet: An extremely efficient convolutional neural network for mobile devices. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 6848–6856, 2018. 3

[62] Yulun Zhang, Yapeng Tian, Yu Kong, Bineng Zhong, and Yun Fu. Residual dense network for image super-resolution. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 2472–2481, 2018. 4

[63] Feida Zhu, Junwei Zhu, Wenqing Chu, Xinyi Zhang, Xiaozhong Ji, Chengjie Wang, and Ying Tai. Blind face restoration via integrating face shape and generative priors. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 7662–7671, 2022. 2