A Coarse-to-Fine Dual Attention Network for Blind Face Completion

Stefan Hörmann * Zhibing Xia * Martin Knoche Gerhard Rigoll
Technical University of Munich

***Abstract*— In the area of face completion, the missing information within an occluded area is estimated, yielding a realistic face of the same identity. In most previous works, the mask describing the occluded region is known, limiting the scope of application. To alleviate this limitation, we propose a coarse-to-fine network trained as a conditional generative adversarial network. While the coarse network predicts the mask and generates a rough estimation of the semantic content, the subsequent fine network refines the rough prediction into a realistic and identity-persevering reconstruction. This is achieved by incorporating adversarial loss and using features from a pretrained face feature extractor. Unlike previous approaches, we employ two parallel attention mechanisms: 1) a patch-wise cross-attention module to substitute information within the occluded patches with patches from the non-occluded region; 2) a pixel-wise global self-attention to allow information exchange within the entire feature map.**

Our exhaustive analysis, including reconstruction quality and face recognition metrics, shows that our approach outperforms the state of the art in blind face completion, improving the true positive identification rate at rank 1 on the MegaFace benchmark from $36.55\,\%$ to $42.48\,\%$. This represents a substantial step towards closing the gap between occluded ($29.34\,\%$) and non-occluded faces ($52.32\,\%$). In terms of reconstruction quality, we obtain a structural similarity of 0.9639 compared to 0.8526 and 0.9563 for occluded faces and the state of the art, respectively. In addition to previous approaches, we provide an in-depth analysis of the influence of the position, size, and sparsity of the occlusion and use facial landmark prediction to measure reconstruction quality.

I. Introduction

Nowadays, with the vast amount of images distributed over the internet, images are edited more and more frequently. This includes watermarks to protect ownership or text providing additional information to the image. Moreover, subtitles in movies often occlude faces and mitigate face recognition results. In addition to text occlusion, rectangular occlusions are often used to mark areas of unwanted information (e.g., a foreground object or accessories like a face mask covering parts of the face), which shall be removed to obtain a clear view of the entire face.

After the occlusion removal, we expect the resulting face to be realistic. For faces, this involves additional challenges since details such as eye color and make-up must be consistent within the face. While even recent approaches [31], [36] struggle in this regard, our qualitative results in Fig. 1 (left) illustrate that our approach provides satisfying results.

With the help of image inpainting techniques [4], [6], [8], [9], [14]–[17], [21], [26]–[31], [35], [36], which are adapted to face completion [14], [35], [36], occluded faces with a given mask can be reconstructed. However, most approaches rely on information about which pixels to reconstruct in the form of an additional mask at the input, thereby limiting the scope of applicability and requiring the development of blind face completion methods. Moreover, following a blind approach, the network is not susceptible to errors occurring during manual mask annotation.

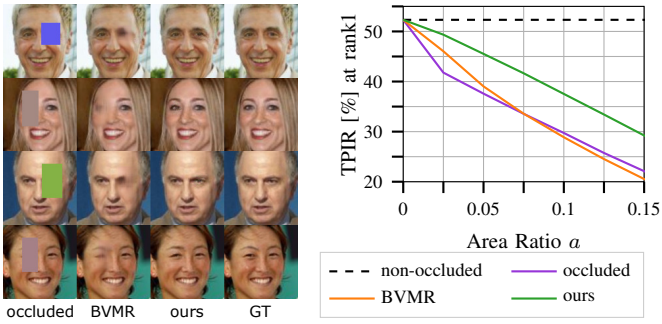

Fig. 1. Left: Qualitative results of our approach compared with BVMR [6] for rectangular occlusion. Right: Robustness of a ResNet50-v2 [5] trained for face recognition against rectangular occlusion with the occluded area in relation to the image area a evaluated on MegaFace benchmark [13] with 1 M distractors.

An occluded face is prone to cause errors for face recognition systems, as we show in Fig. 1 (right). The true positive identification rate (TPIR) for non-occluded faces of $52.32\,\%$ substantially drops for an increasing area of occlusion. However, we mitigate the drop in TPIR when reconstructing the face using our approach as the TPIR is on average $7.7\,\%$ higher compared to occluded faces. Simultaneously, the TPIR is identical for non-occluded faces ($a = 0$), which proves that no unwanted artifacts deteriorate the reconstruction. This clearly highlights the benefits of incorporating a face reconstruction module before recognizing the face, which is in accordance with [19], [33].

To cope with the challenging task of reconstructing the rich information within the human face in the presence of occlusions with varying shape, position, size, color, and

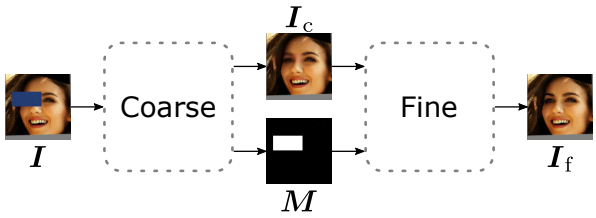

Fig. 2. Overview of our approach to blind face completion. For an occluded image $\boldsymbol{I}$, the coarse network estimates a rough reconstruction $\boldsymbol{I}_c$ and a mask $\boldsymbol{M}$. In the fine network, $\boldsymbol{I}_c$ is refined using $\boldsymbol{M}$ providing the final reconstructed image $\boldsymbol{I}_f$.

* These authors contributed equally to this work.

978-1-6654-3176-7/21/$31.00 ©2021 IEEE

2021 16th IEEE International Conference on Automatic Face and Gesture Recognition (FG 2021) | 978-1-6654-3176-7/21/$31.00 ©2021 IEEE | DOI: 10.1109/FG52635.2021.9666985

form, we propose a coarse-to-fine network as depicted in Fig. 2. While the coarse network creates a rough reconstruction of the face, the fine network refines it with the help of two parallel attention mechanisms. The latter enable the network to search for similar patches within the non-occluded face parts to substitute the occluded areas and utilize global pixel-wise relationships within a face. Training the coarse-to-fine network as part of a generative adversarial network (GAN) allows the refined faces to become realistic, whereas supervision by identity losses ensures that identity information is preserved during the reconstruction.

Our main contributions are summarized as follows:

- We propose a coarse-to-fine network embedded into a GAN combined with two parallel attention modules to ensure realistic reconstruction using the entire face's information.
- Our exhaustive analysis shows how shape, position, size, and form of the occlusions affect face recognition and reconstruction performance.
- We identify and investigate the trade-off between reconstruction quality and recognition performance.

II. Related Work

A. Image Inpainting

Most image inpainting methods incorporate a U-Net structure [6], [15]–[17], [28], [35], [36]. While there exist methods using convolutional neural networks (CNNs) [6], [15]–[17] without GANs, the majority [4], [8], [9], [14], [21], [26]–[31], [35], [36] employs the GAN-structure to obtain a more realistic image. A crucial part for well-performing image inpainting is increasing the receptive field of vanilla convolutions to allow global information exchange. This can be achieved via dilated convolutions [4], [9], [21], [29]. To compensate the sparse kernel of the dilated convolution, Hui et al. [8] proposed the dense multi-scale fusion block (DMFB), which contains multiple dilated convolutions with different dilation rates in parallel. Another popular way to increase the receptive field is to integrate an attention mechanism [28]–[31].

B. Face Completion

Face inpainting (or completion) is considered one of the most challenging image inpainting tasks, as the objective of reconstruction not only focuses on realism but also on maintaining the identity of the person. By employing a symmetry-consistent CNN, Li et al. [14] transfer brightness-adjusted information from one half-face to the other and separately reconstruct pixels missing in both half-faces. Zhang et al. [35] propose a domain embedded generative model to guide the reconstruction using the mask, face part, and facial landmark information. To allow the network to capture the vast details for face recognition, Zhou et al. [36] use a dual spatial attention module together with an oracle supervision signal to learn the correlations between facial textures at multiple scales efficiently. Jam et al. [11] focus on high-resolution face inpainting leveraging symmetric skip connections and a Wasserstein-Perceptual loss function.

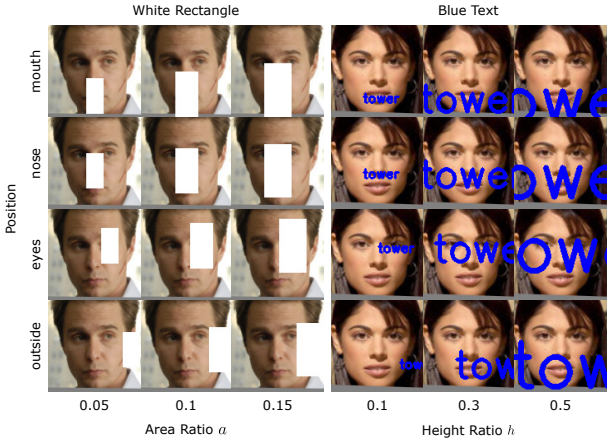

Fig. 3. Examples of occluded faces with different occlusions.

C. Blind Inpainting

In contrast to the default image inpainting task, for blind inpainting, the occluded area is unknown and needs to be estimated by the model. Liu et al. [16] use the residual learning algorithm and leverage gradients to estimate the details in the occluded regions. The VCNet [26] disentangles mask prediction and robust inpainting, which are trained in an adversarial manner. Hertz et al. [6] propose the blind visual motif removal (BVMR) model, which uses a U-Net architecture with three decoders to predict the reconstructed image, mask, and motif. As an extension to BVMR, Cun et al. [3] add a subsequent refinement network.

III. Methodology

A. Generating Occlusions

The task of our network is to remove occlusion and reconstruct the hidden face areas. To consider a multitude of occlusion types, we differ between form, color, size, and position. Fig. 3 depicts two examples of our augmentation scheme. We consider two geometric forms, a rectangle and text. While a rectangle represents a closed surface, the text illustrates an arbitrary shape with holes. To increase generalization, we vary the aspect ratio of the rectangle within $[0.5; 2]$ and select a word from a list of 2048 English words[1]. The color can be adjusted freely within the RGB color space. For rectangles, the size represents its occlusion's area ratio a, whereas, for text occlusion, the size illustrates the height ratio h to be independent of word length. In order to evaluate the influence of the occlusions' position on the performance, we divide the face into four regions: 1) mouth; 2) nose; 3) eyes; 4) outside.

To obtain an occluded image $\boldsymbol{I}$ at the input of our network, we synthetically generate a binary mask $\boldsymbol{M}_{\text{gt}}$ with 1 denoting areas affected by occlusion, which is applied on a ground-truth image $\boldsymbol{I}_{\text{gt}}$:

$$\boldsymbol{I} = (\mathbf{1} - \boldsymbol{M}_{\text{gt}}) \odot \boldsymbol{I}_{\text{gt}} + \boldsymbol{M}_{\text{gt}} \odot \boldsymbol{c} \tag{1}$$

with $\odot$ being the Hadamard product (including broadcasting) and $\boldsymbol{c}$ the vector describing the occlusion's color.

[1] https://raw.githubusercontent.com/sindresorhus/mnemonic-words/master/words.json

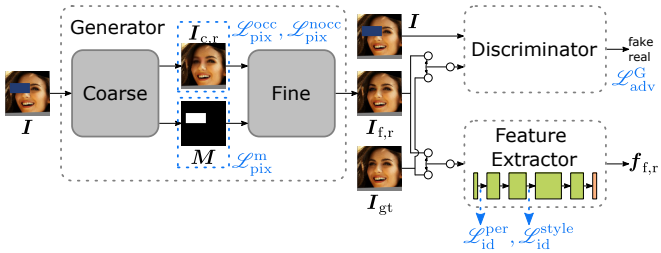

Fig. 4. Our approach to blind face reconstruction: A coarse-to-fine generator reconstructs the occluded face $\boldsymbol{I}_{\mathrm{f,r}}$ and predicts the mask $\boldsymbol{M}$. To ensure a high degree of realism, a discriminator with adversarial loss is employed together with pixel-wise $L1$ losses for guidance. A feature extractor further allows losses to enforce similar identity features of the reconstructed face $\boldsymbol{I}_{\mathrm{f,r}}$ and the ground-truth face $\boldsymbol{I}_{\mathrm{gt}}$.

B. Network Architecture

Fig. 4 depicts an overview of our architecture, which can be considered a conditional GAN. While the generator tries to generate a face $\boldsymbol{I}_{\mathrm{f,r}}$, which is indistinguishable from $\boldsymbol{I}_{\mathrm{gt}}$, the discriminator's objective is to judge whether the face is reconstructed (fake) or is real. By employing a pretrained feature extractor, we not only obtain a feature vector encoding the reconstructed face's identity $\boldsymbol{f}_{\mathrm{f,r}}$, but also allow identity losses to ensure that the $\boldsymbol{I}_{\mathrm{f,r}}$ resembles $\boldsymbol{I}_{\mathrm{gt}}$ with respect to its identity. In the following subsections, the details are described.

1) Generator: Inspired by [3], [17], [29], we incorporate a coarse-to-fine network as illustrated in Fig. 5. While the coarse network yields a rough estimate $\boldsymbol{I}_{\mathrm{c}}$ and generates a binary mask $\boldsymbol{M}$, the fine network refines the reconstructed coarse face $\boldsymbol{I}_{\mathrm{c,r}}$ yielding a face with a high degree of realism and detail $\boldsymbol{I}_{\mathrm{f,r}}$. Both networks follow the popular U-Net architecture [24] with several adaptions for our task.

Similar to [6], we reconstruct the coarse and fine faces using the predicted mask $\boldsymbol{M}$:

$$\boldsymbol{I}_{\mathrm{f,r}} = (\boldsymbol{1} - \boldsymbol{M}) \odot \boldsymbol{I} + \boldsymbol{M} \odot \boldsymbol{I}_{\mathrm{f}} \tag{2}$$

If we assume a reliable mask prediction $\boldsymbol{M} = \boldsymbol{M}_{\mathrm{gt}}$, we can use Equation 1 to obtain:

$$\boldsymbol{I}_{\mathrm{f,r}} = (\boldsymbol{1} - \boldsymbol{M}_{\mathrm{gt}}) \odot \boldsymbol{I}_{\mathrm{gt}} + \boldsymbol{M}_{\mathrm{gt}} \odot \boldsymbol{I}_{\mathrm{f}} \tag{3}$$

This shows that by splitting the task into mask prediction and face reconstruction, we can incorporate the mask in the reconstruction to ensure that pixels outside of the occlusion remain untouched. We apply the reconstruction according to Equation 2 after the coarse and the fine network.

Coarse Network. The input of our coarse network is an occluded face $\boldsymbol{I}$ with a resolution of $112 \times 112 \times 3$ pixels, which is downsampled to a resolution of 28×28 using convolutional layers with stride 2. With every reduction of the resolution, we double the number of channels from initially 64. After 6 convolutional layers with a 3×3 kernel we employ a DMFB [8] to capture information present at multiple scales and different regions, followed by another 3×3 convolution obtaining the latent feature map $\boldsymbol{L}$.

We decode $\boldsymbol{L}$ using two (almost) identical branches in parallel for the face $\boldsymbol{I}_{\mathrm{c}}$ and mask $\boldsymbol{M}$, respectively. For the decoding, we utilize upsampling followed by the concatenation of the respective feature map from the encoder and 3 convolutional layers for every resolution. Unlike [6], we incorporate nearest-neighbor interpolation together with a convolution in order to reduce checkerboard artifacts induced by transposed convolution [23]. Every branch is concluded with a final convolutional layer to obtain the desired output channels.

While we use the leaky rectified linear unit (ReLU) [18] as an activation function in the encoder, we incorporate ReLU in the decoder. To obtain the desired value range, we clip the values after the last convolution for $\boldsymbol{I}_{\mathrm{c}}$ and utilize the sigmoid activation function in the last layer for $\boldsymbol{M}$.

Fine Network. The fine network follows a similar U-Net structure [24] as the coarse network. However, due to its purpose of refining the rough prediction from the coarse network, we added several modules, namely a cross-attention and a self-attention module, to leverage similarity between different patches within the face.

As in the coarse network, we use 3×3 convolutional layers but downsample to a resolution of 56×56. Since the cross-attention module operates in a patch-wise manner and, therefore, benefits from higher resolution, it is applied at a resolution of 56×56. In contrast, the pixel-wise self-attention is employed after another downsampling step at a resolution of 28×28. A DMFB follows both attention modules. After another convolutional layer (and an upsampling block for the self-attention), we concatenate the feature maps of both attention branches and the feature map of the corresponding resolution in the encoder. To obtain a resolution of 112×112 we apply another upsampling block as in the coarse network. As in [29], we use the exponential linear unit (ELU) [2] in the fine network as the activation function.

Incorporating attention modules is widely used in image inpainting tasks in order to handle long-range dependencies within images [28]–[31]. As depicted in Fig. 5, we fuse the information of three branches: 1) the skip connection without attention; 2) the patch-wise cross-attention; 3) the pixel-wise self-attention.

The cross-attention module is designed to substitute the occluded patches with patches from different face regions while maintaining consistent and realistic values within the patch. On the other hand, the global self-attention module learns every pixel's relationship within the image and thereby allows reconstruction of face parts, which are unique within the faces.

This parallel utilization of two attention modules similar to [36] makes both modules complement each other. While the cross-attention is most effective when transferring information from a similar face part (e.g., from left eye to the right eye), the self-attention reconstructs unique face parts leveraging global information (e.g., the nose). To fuse the information of both attention modules and the skip connection from the encoder, we use a 3×3 convolution followed by a DMFB.

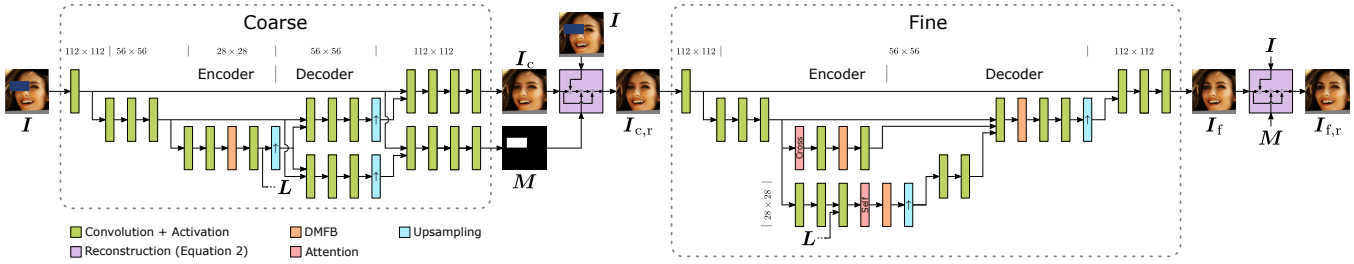

Fig. 5. Architecture of the generator. The occluded face $\boldsymbol{I}$ first passes through the coarse network, which outputs a rough reconstruction of the image $\boldsymbol{I}_{\mathrm{c}}$ and a binary mask $\boldsymbol{M}$. After reconstruction according to Equation 2, the reconstructed face $\boldsymbol{I}_{\mathrm{c,r}}$ is further refined utilizing a parallel dual attention structure in the fine network, which yields the final reconstructed face $\boldsymbol{I}_{\mathrm{f}}$.

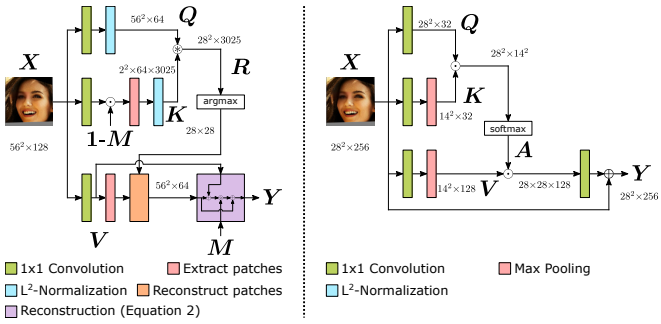

Fig. 6. Overview of the attention modules: While the patch-wise cross-attention substitutes every patch within the occluded area with the most similar patch of the non-occluded area, the global pixel-wise self-attention allows unrestricted information exchange.

Patch-wise Cross-Attention. The details of the cross-attention block [27]–[29] are depicted in Fig. 6 (left) with its input being a $56 \times 56 \times 128$ feature map $\boldsymbol{X}$. We obtain the query $\boldsymbol{Q}$, key $\boldsymbol{K}$, and value $\boldsymbol{V}$ after reducing the feature map resolution to $56 \times 56 \times 64$ with a 1×1 convolution each. For $\boldsymbol{K}$, we set all activations within the occluded area to 0 by multiplying with $\mathbf{1} - \boldsymbol{M}$. In this way, we ensure that only reliable information from the non-occluded area is utilized. Then, we extract $55^2 = 3025$ overlapping patches of 2×2 pixels and L^2-normalize every patch separately. By performing a convolution of the channel-wise L^2-normalized $\boldsymbol{Q}$ with the patches $\boldsymbol{K}$ as convolutional kernels, we obtain a tensor $\boldsymbol{R}$ containing the cosine distance as a similarity metric of the patches. Hence, at every position within 28×28 of $\boldsymbol{R}$ the 3025-dimensional vector depicts how similar the corresponding patch is with $\boldsymbol{Q}$. By applying $\arg\max$, we receive a 28×28 matrix containing the index of the most similar patch, which will later be used for reconstruction.

To reconstruct the feature map, we use the previously computed positions and take 2×2 patches extracted from $\boldsymbol{V}$. We further ensure that the information is only substituted within the occluded area by applying Equation 2, yielding the $56 \times 56 \times 64$ output $\boldsymbol{Y}$ of the cross-attention module. In this way, the module is capable of transferring realistic and detailed textures from the non-occluded to the occluded area, while the information in the non-occluded area is left untouched.

Pixel-wise Self-Attention. Similar to the cross-attention module, the self-attention, as illustrated in Fig. 6 (right) [32], uses 1×1 convolutions to reduce the number of channels and obtain query $\boldsymbol{Q}$, key $\boldsymbol{K}$, and value $\boldsymbol{V}$. Moreover, we employ max pooling to reduce the resolution to 14×14 for $\boldsymbol{K}$ and $\boldsymbol{V}$. Via matrix multiplication followed by a softmax activation $\mathrm{softmax}\left(\boldsymbol{Q} \cdot \boldsymbol{K}^{\mathrm{T}}\right)$, we obtain the attention map $\boldsymbol{A}$. To compute the output feature map $\boldsymbol{Y}$ of the self-attention module, we add the input $\boldsymbol{X}$ to the output of another 1×1 convolution with $\boldsymbol{A} \cdot \boldsymbol{V}$.

In this way, the module is capable of refining $\boldsymbol{X}$ with pixel-wise information while maintaining faster convergence due to the skip connection. In contrast to [36], we do not restrict the attention area to the occluded area. Hence, our self-attention is considered global.

2) Discriminator: Similar to other GAN approaches in the domain of faces [4], [35], [36], we employ multiple discriminators - a global discriminator combined with six fully convolutional patch discriminators [10] focusing on different face regions. The input of the global and one patch discriminator consists of the entire face region 112×112 pixels, whereas the remaining patch discriminators contain 28×28 crops around the left/right eye, mouth, nose. As depicted in Fig. 4, we concatenate the (cropped) generator's output $\boldsymbol{I}_{\mathrm{f,r}}$ with the (cropped) occluded face $\boldsymbol{I}$, which eases the discriminator to focus on the occluded regions. Moreover, we propose to use another patch discriminator with the concatenation of the left eye and the horizontally flipped right eye as the input to ensure that eye colors are consistent, which resulted in a critical issue in [31], [36].

All global and patch discriminator types use four 3×3 convolutions with leaky ReLU [18] activation - the former two with stride 2 - to reduce the resolution to $28 \times 28 \times 1$ and $7 \times 7 \times 1$, respectively. While the global discriminator utilizes a fully connected layer followed by a sigmoid activation, the patch discriminators employ the pixel-wise sigmoid and use the average probability as patch probability. We dispense with normalization layers as their removal increases performance and reduces computational complexity in various related tasks, including image inpainting [29], super-resolution [25], and deblurring [20].

3) Feature Extractor: In order to obtain a well generalizing face feature extractor, we incorporate a ResNet50-v2 architecture [5] with a 256-dimensional bottleneck layer. Since we pretrain the feature extractor on the VGGFace2 dataset [1], the network is concluded with a 8631-dimensional logits layer.

C. Loss Functions

Our model is trained by a weighted sum $\mathscr{L}_{\text{G}}$ of several losses, which are described in the following:

$$\begin{aligned}\mathscr{L}_{\text{G}} = \lambda_{\text{pix}}^{\text{m}}\mathscr{L}_{\text{pix}}^{\text{m}} + \lambda_{\text{pix}}^{\text{occ}}\mathscr{L}_{\text{pix}}^{\text{occ}} + \lambda_{\text{pix}}^{\text{nocc}}\mathscr{L}_{\text{pix}}^{\text{nocc}} + \\ \lambda_{\text{id}}^{\text{per}}\mathscr{L}_{\text{id}}^{\text{per}} + \lambda_{\text{id}}^{\text{style}}\mathscr{L}_{\text{id}}^{\text{style}} + \lambda_{\text{adv}}^{\text{G}}\mathscr{L}_{\text{adv}}^{\text{G}}\end{aligned} \tag{4}$$

where $\lambda_{(\cdot)}^{(\cdot)}$ are scalars to balance the losses.

1) Pixel-wise Similarity Loss $\mathscr{L}_{pix}$: Predicting a reliable mask is crucial for an overall robust reconstruction as it ensures that only occluded pixels are changed (c.f. Equation 3). Since the mask is binary, we can interpret the mask prediction as a binary pixel-wise classification and therefore use binary cross-entropy loss:

$$\mathscr{L}_{\text{pix}}^{\text{m}} = -\frac{1}{112^2}\sum_{i=1}^{112^2} \boldsymbol{M}_{\text{gt}}^i \log \boldsymbol{M}^i + (1-\boldsymbol{M}_{\text{gt}}^i)\log(1-\boldsymbol{M}^i), \tag{5}$$

where $\boldsymbol{M}_{\text{gt}}^i$ and $\boldsymbol{M}^i$ denote the i-th pixel of $\boldsymbol{M}_{\text{gt}}$ and $\boldsymbol{M}$, respectively.

As the purpose of the coarse network, in contrast to the fine network, is to generate a rough prediction of the occluded area, we incorporate pixel-wise L^1 losses for the occluded $\mathscr{L}_{\text{pix}}^{\text{occ}}$ and non-occluded area $\mathscr{L}_{\text{pix}}^{\text{nocc}}$ to allow an individual weighting:

$$\mathscr{L}_{\text{pix}}^{\text{occ}} = \frac{\left\|(\boldsymbol{I}_{\text{c}} - \boldsymbol{I}_{\text{gt}}) \odot \boldsymbol{M}_{\text{gt}}\right\|_1}{\left\|\boldsymbol{M}_{\text{gt}}\right\|_1} \tag{6}$$

$$\mathscr{L}_{\text{pix}}^{\text{nocc}} = \frac{\left\|(\boldsymbol{I}_{\text{c}} - \boldsymbol{I}_{\text{gt}}) \odot (\boldsymbol{1} - \boldsymbol{M}_{\text{gt}})\right\|_1}{\left\|\boldsymbol{1} - \boldsymbol{M}_{\text{gt}}\right\|_1} \tag{7}$$

The normalization is essential as otherwise, the area of occlusion affects the relevance of the loss.

2) Identity Loss $\mathscr{L}_{id}$: In addition to pixel-wise losses, we further employ identity losses to capture perceptual similarities. We achieve this with the perceptual loss [12], which computes similarity for high-level feature maps extracted from a pretrained face feature extractor. The perceptual loss is defined as follows for a given image $\boldsymbol{I}_i$ at multiple depths of the feature extractor $\mathcal{D}$:

$$\mathscr{L}^{\text{per}}(\boldsymbol{I}_i, \mathcal{D}) = \sum_{d\in\mathcal{D}} \frac{1}{N\left(\boldsymbol{\Psi}_d(\boldsymbol{I}_{\text{gt}})\right)} \left\|\boldsymbol{\Psi}_d(\boldsymbol{I}_i) - \boldsymbol{\Psi}_d(\boldsymbol{I}_{\text{gt}})\right\|_1 \tag{8}$$

with $\boldsymbol{\Psi}_d(\boldsymbol{I}_i)$ being the feature map at depth d for $\boldsymbol{I}_i$ as input and $N\left(\boldsymbol{\Psi}_d(\boldsymbol{I}_{\text{gt}})\right)$ denotes the number of elements in $\boldsymbol{\Psi}_d(\boldsymbol{I}_{\text{gt}})$.

As in [15], we use the perceptual loss to compute the L^1 distances for $\boldsymbol{I}_{\text{f}}$ and $\boldsymbol{I}_{\text{f,r}}$, and utilize high-level semantic information after the first ResNet block and low-level feature information after the third ResNet block $\mathcal{D} = \{\text{'block1'}; \text{'block3'}\}$, which results in:

$$\mathscr{L}_{\text{id}}^{\text{per}} = \mathscr{L}^{\text{per}}(\boldsymbol{I}_{\text{f}}, \mathcal{D}) + \mathscr{L}^{\text{per}}(\boldsymbol{I}_{\text{f,r}}, \mathcal{D}) \tag{9}$$

In addition to $\mathscr{L}_{\text{id}}^{\text{per}}$, we incorporate the style loss $\mathscr{L}_{\text{id}}^{\text{style}}$ [12], in which every feature map $\boldsymbol{\Psi}_d(\boldsymbol{I}_i)$ in Equation 8 is replaced by its Gram matrix. For a given feature map $\boldsymbol{\Psi}_d(\boldsymbol{I}_i)$ with resolution $W \times H \times C$, we first reshape it into a matrix $\boldsymbol{\Phi}_d(\boldsymbol{I}_i)$ of size $WH \times C$. Then the corresponding Gram matrix is computed as follows:

$$G\left(\boldsymbol{\Psi}_d(\boldsymbol{I}_i)\right)) = \frac{\boldsymbol{\Phi}_d(\boldsymbol{I}_i)^{\text{T}}\boldsymbol{\Phi}_d(\boldsymbol{I}_i)}{N\left(\boldsymbol{\Psi}_d(\boldsymbol{I}_i)\right)} \tag{10}$$

The overall style loss $\mathscr{L}_{\text{id}}^{\text{style}}$ is computed analogously to Equation 9.

3) Adversarial Loss $\mathscr{L}_{adv}$: To ensure a realistic reconstruction, we use the mean adversarial loss of all seven discriminators introduced in section III-B.2, which are denoted by $D_i(\cdot)$. The adversarial loss for the generator is defined as follows:

$$\mathscr{L}_{\text{adv}}^{\text{G}} = -\frac{1}{7}\sum_{i=1}^{7} \log\left(D_i(C_i(\boldsymbol{I}_{\text{f,r}}) \oplus C_i(\boldsymbol{I}))\right) \tag{11}$$

with $C_i(\cdot)$ being a function to crop the patch corresponding to the respective discriminator and $\oplus$ denoting a concatenation along the channel axis. To train the discriminators to distinguish $\boldsymbol{I}_{\text{f,r}}$ from $\boldsymbol{I}_{\text{gt}}$, we optimize the following loss:

$$\begin{aligned}\mathscr{L}_{\text{adv}}^{\text{D}} = -\frac{1}{7}\sum_{i=1}^{7} \log\left(1 - D_i(C_i(\boldsymbol{I}_{\text{f,r}}) \oplus C_i(\boldsymbol{I}))\right) + \\ \log\left(D_i(C_i(\boldsymbol{I}_{\text{gt}}) \oplus C_i(\boldsymbol{I}))\right)\end{aligned} \tag{12}$$

IV. Results

A. Training Details

We train our feature extractor separately on the VGGFace2 dataset [1], which comprises 3.1 M images of 8631 identities, with softmax cross-entropy loss for 20 epochs. As preprocessing, we align all faces using their facial landmarks predicted by the MTCNN [34] and crop them to 112×112 pixels.

During training, we generate occlusions according to section III-A with random form, color, and position. We randomize the area and height ratio within the following intervals $a \in [0.01; 0.15]$ and $h \in [0.05; 0.5]$, respectively. Moreover, we apply horizontal flipping and align the faces as during pretraining of the feature extractor.

Training our network is divided into two steps. First, we train the coarse network for one epoch with ADAM optimizer and a learning rate of $2{\cdot}10^{-4}$ to obtain a rough reconstruction but already an accurate estimation of the mask, which is decisive for a stable GAN training afterwards. As losses, we only consider pixel-wise similarity losses with $\lambda_{\text{pix}}^{\text{occ}} = 3$ and $\lambda_{\text{pix}}^{\text{m}} = \lambda_{\text{pix}}^{\text{nocc}} = 1$ to make the coarse net focus on the occluded region. Next, we load the weights from the coarse network and continue training the whole network. While we set the learning rate of the untrained weights to 10^{-4}, the learning rate of the pretrained weights in the coarse network is $5 \cdot 10^{-5}$. As typical for GANs, we train the generator and the discriminator in an alternating manner for 5 epochs. We reduce the learning rate by a factor of 10 every epoch, use a batch size of 8 and use the following loss factors to balance the losses similar to [15]: $\lambda_{\text{pix}}^{\text{m}} = 10$, $\lambda_{\text{pix}}^{\text{occ}} = 3$, $\lambda_{\text{pix}}^{\text{nocc}} = 1$, $\lambda_{\text{id}}^{\text{per}} = 0.1$, $\lambda_{\text{id}}^{\text{style}} = 240$, and $\lambda_{\text{adv}}^{\text{G}} = 1$. All other hyperparameters and the generation of occlusion are identical as during pretraining of the coarse network.

TABLE I

COMPARISON OF OUR APPROACH FOR DIFFERENT CONFIGURATIONS WITH SEVERAL BASELINES. FACE RECOGNITION PERFORMANCE IS EVALUATED USING TPIR AT RANK 1 [%] ON THE MEGAFACE BENCHMARK [13] WITH $D = 1\,\mathrm{M}$ AND ACCURACY [%] ON THE LFW BENCHMARK [7], WHEREAS RECONSTRUCTION QUALITY IS MEASURED VIA PSNR AND SSIM ON MEGAFACE. ALL METRICS ARE AVERAGED OVER ALL FOUR POSITIONS, THREE SIZES (C.F. FIG. 3) AND FOR RANDOM COLOR. $*$ DEPICTS THAT $\mathscr{L}_{\mathrm{PIX}}^{\mathrm{OCC}}$ WAS COMPUTED ALSO ON $\boldsymbol{I}_{\mathrm{F}}$.

Losses			Attention		MegaFace $D = 1\,\mathrm{M}$				LFW			PSNR	SSIM
$\mathscr{L}_{\mathrm{id}}^{\mathrm{per}}$	$\mathscr{L}_{\mathrm{id}}^{\mathrm{style}}$	$\mathscr{L}_{\mathrm{pix}}^{\mathrm{occ}}$ & $\mathscr{L}_{\mathrm{pix}}^{\mathrm{nocc}}$	Self	Cross	non-occluded	Rectangle	Text	Avg	Rectangle	Text	Avg	Avg	avg
ResNet-50					52.32	29.81	28.88	29.34	97.65	97.18	97.41	20.42	0.8526
ResNet-50-occ					54.40	42.38	44.66	43.52	98.85	99.05	98.95		
BVMR [6]					52.32	29.48	43.61	36.55	97.75	99.04	98.40	35.18	0.9563
Coarse Network					52.32	31.63	44.89	38.26	97.95	99.08	98.52	36.30	0.9622
√	√	√	√	√	52.32	37.18	47.50	42.34	98.61	99.19	98.90	36.69	0.9638
√		√	√	√	52.32	**37.43**	**47.53**	**42.48**	98.60	99.17	98.89	36.68	0.9639
	√	√	√	√	52.32	36.98	47.35	42.17	98.62	**99.23**	**98.93**	36.68	0.9634
√			√	√	52.32	36.53	47.24	41.89	98.56	99.20	98.88	36.48	0.9632
√		√*	√	√	52.32	37.09	47.43	42.26	**98.64**	99.22	**98.93**	**37.39**	**0.9662**
√		√		√	52.32	35.05	46.91	40.98	98.44	99.17	98.80	36.53	0.9628
√		√	√		52.32	37.19	47.42	42.31	**98.64**	99.21	98.92	36.61	0.9635

B. Benchmark Details

We evaluate face recognition performance using the synthetically-occluded MegaFace benchmark [13] to compute closed-set face identification performance and the popular Labeled Faces in the Wild (LFW) benchmark [7] for face verification:

For the MegaFace benchmark, the gallery is formed of 80 identities taken from FaceScrub dataset [22] with approximately 50 images each, and $D = 1\,\mathrm{M}$ distractor images of 690 k identities. In the evaluation, every gallery image is matched with the remaining gallery and a predefined number of up to $D = 1\,\mathrm{M}$ distractors. In this way, we not only obtain the TPIR at different ranks but also for a different number of distractors. In contrast to the original benchmark, we synthetically occlude the gallery images as described in section III-A. We evaluate for different sizes and forms and ensure that the benchmark is deterministic even if the color is randomized. By only occluding the gallery image, we increase the difficulty as the distractors remain untouched. The occluded gallery images are publicly available[2].

The LFW benchmark comprises 6 k face pairs, in which we occlude both faces in a similar way. Hence, in the case of the occlusion of eyes, both faces of a pair have one eye occluded yet not necessarily the same one.

In addition to face recognition performance, we compute the peak signal-to-noise ratio (PSNR) and the structural similarity (SSIM) to measure reconstruction quality and extract 196 facial landmarks using[3] to measure how accurate the facial landmarks were reconstructed.

C. Baselines

We compare our approach to the BVMR model, for which we use the best parameters as mentioned by the authors [6] and train on the same dataset with the same augmentation as our model. Moreover, we train another ResNet-50 (denoted by ResNet-50-occ) as our face extractor, however, with occlusion as data augmentation. Even though the latter does not reconstruct the faces and, therefore, cannot be considered a face completion method, we want to compare ourselves with a rather straightforward method in terms of face recognition performance.

[2] https://github.com/stefhoer/C2FDAN

[3] https://github.com/deepinsight/insightface/tree/master/alignment/coordinateReg

D. Ablation Study

Table I depicts the results of our approach for different configurations in comparison with the baselines. First, we clearly observe that occluded faces severely affect face recognition performance, with the TPIR at rank 1 on MegaFace dropping $52.32\,\%$ to $29.34\,\%$ and the accuracy on LFW from $99.40\,\%$ to $97.18\,\%$. In comparison with the state-of-the-art in blind face completion, we see that even our pretrained coarse network outperforms BVMR model on all metrics even though it has only about $1/4$ of its parameters (4.1 M compared to 20.5 M). Since both are trained with pixel-wise losses only, we believe that this improvement is mainly due to the addition of the DMFB.

By adding another refinement network to the coarse network, we boost the performance by a considerable amount. In our ablation study, we observe a trade-off between face recognition and reconstruction performance. While our approach without $\mathscr{L}_{\mathrm{id}}^{\mathrm{style}}$ achieves the best face recognition performance on MegaFace benchmark, computing $\mathscr{L}_{\mathrm{pix}}^{\mathrm{occ}}$ also on the output image of the fine network $\boldsymbol{I}_{\mathrm{f}}$ achieves the overall best PSNR and SSIM values at the cost of lower face recognition accuracy. Moreover, we conclude that the parallel attention structure in the fine network with cross and self-attention is superior to using only one attention mechanism. Even though self-attention is crucial to the overall performance, the addition of cross-attention still improves the performance.

When comparing our approach to directly extracting features by a ResNet trained with occlusion (ResNet-50-occ), we observe that the face recognition performance is, on average, slightly superior to our approach caused by the improved performance for rectangles occlusion. This is since

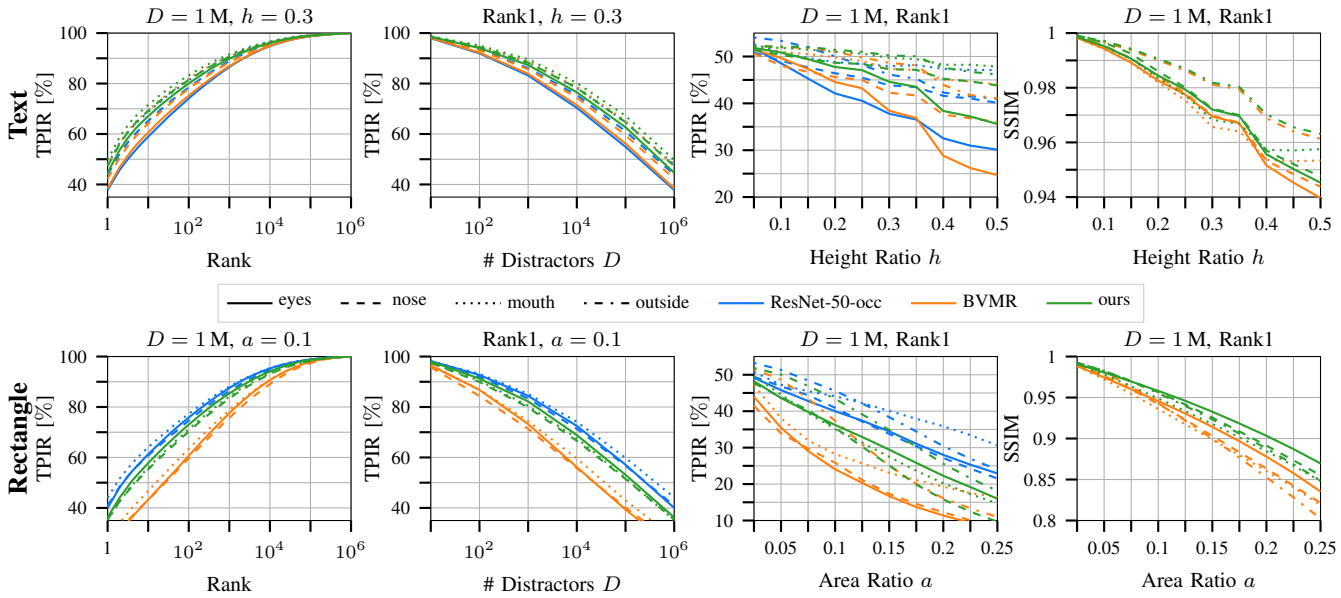

Fig. 7. Overview of the robustness of our method and the baselines in terms of rank, number of distractors D, size, and position on the MegaFace benchmark [13].

the feature extractor learned which information to rely on as it was exposed to occlusion during training. For the facial reconstruction, though, the feature extractor is presented with a reconstructed image. Since, in this case, the feature extractor was never trained with occlusion, it assumes that every pixel contains the correct information, neglecting that previously occluded (and therefore reconstructed) pixels are not equally reliable as non-occluded (and therefore ideally untouched) pixels. For text occlusion, our approach outperforms ResNet-50-occ, indicating that our approach achieves satisfying performance in reconstructing sparse occlusions, which are sparse and can span the whole image. Moreover, compared to ResNet-50-occ, our approach has the added benefit of generating a reconstructed face.

Overall, we observe that we can close the gap between occluded and non-occluded faces by a substantial amount while maintaining identical performance on non-occluded faces, ensuring that the reconstruction of non-occluded faces does not induce unwanted artifacts and they remain untouched.

E. Detailed analysis

In the following, we evaluate face recognition and reconstruction performance on the MegaFace benchmark depending on D, ranks, size, and position by comparing our best model (c.f. Table I without $\mathcal{L}_{\text{id}}^{\text{style}}$) with BVMR [6] and ResNet-50-occ (only face recognition). The analysis is depicted in Fig. 7 and Fig. 8.

Rank and Number of Distractors D. The graphs depicting the TPIR in terms of rank and D (first two columns in Fig. 7) confirm the results averaged over the size from Table I. While our approach outperforms BVMR for text and rectangular occlusion, we achieve superior results compared to ResNet-50-occ only for text occlusion. In terms of reconstruction quality (last column in Fig. 7), the larger gap between our approach and BVMR for rectangular occlusions compared to text occlusions indicates that our approach is more viable. Note that ResNet-50-occ does not perform any reconstruction. These findings are consistent for all ranks and D.

Influence of Size and Position. As expected, increased height ratio h or area ratio a leads to worse TPIR. We believe that it is difficult to draw conclusions from the text occlusion in terms of position as the inherent randomness in word length causes occlusions applied outside of the face to also cover significant parts of the faces (c.f. Fig. 3). For rectangular occlusion, we observe that the TPIR is highest when occlusions occur outside of the face. Due to the worse SSIM compared to other face parts, the good TPIR is not caused by a realistic reconstruction but rather by being unaffected by occlusions outside the face. The best reconstruction and recognition within the faces are for occlusions around the eye. This indicates that our architecture successfully transfers information from the left to the right eye and vice versa due to the cross-attention module and the additional discriminator for the eyes. The worst recognition is obtained for the nose occlusion as it covers the largest area of useful information within the face, whereas the mouth occlusion leads to the overall worst SSIM since the network does not know whether the mouth is open or closed when it is entirely covered by occlusion.

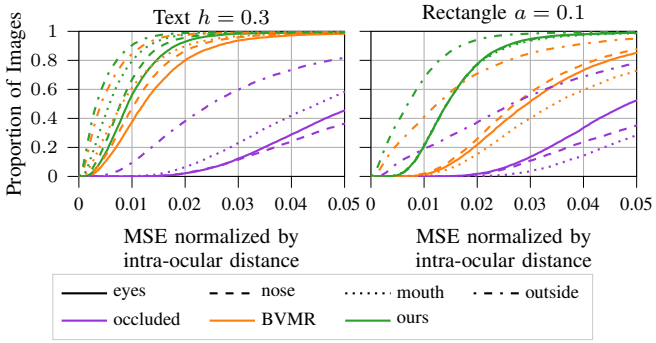

Fig. 8. Effect of occlusion and reconstruction on the accuracy of facial landmark prediction on the LFW dataset [7].

Reconstruction Quality. Fig. 1 (left) depicts qualitative results when reconstructing rectangular occlusions around the eyes. Compared to BVMR [6], our approach yields realistic sharp results. Moreover, as opposed to [31], [36], our results have consistent make-up and eye color.

We further analyze the accuracy of facial landmark prediction on the LFW dataset. For that, we take the landmarks of the non-occluded faces as ground truth and compute the mean squared error (MSE) normalized by the intra-ocular distance. As illustrated in Fig. 8, using our reconstruction substantially boosts the quality of facial landmark prediction.

V. Conclusion

In this paper, we present a novel approach for blind face reconstruction utilizing a coarse-to-fine network. Combining a parallel structure of two attention modules with adversarial loss allows the network to combine global information to yield a realistic face with sharp details. Moreover, we ensure that the face's identity is preserved by supervising the training with features from a pretrained face feature extractor. Our exhaustive analysis encompassing reconstruction quality and recognition accuracy metrics reveals the trade-off between the two. Moreover, we demonstrate that our approach outperforms the baseline for different positions, sizes, and shapes of occlusions.

References

[1] Q. Cao, L. Shen, W. Xie, O. Parkhi, and A. Zisserman. VGGFace2: A dataset for recognising faces across pose and age. In *13th IEEE International Conference on Automatic Face & Gesture Recognition (FG 2018)*, pages 67–74. IEEE, 2018.

[2] D.-A. Clevert, T. Unterthiner, and S. Hochreiter. Fast and Accurate Deep Network Learning by Exponential Linear Units (ELUs). *arXiv preprint arXiv:1511.07289*, 2015.

[3] X. Cun and C.-M. Pun. Split then Refine: Stacked Attention-guided ResUNets for Blind Single Image Visible Watermark Removal, 2020.

[4] U. Demir and G. Unal. Patch-Based Image Inpainting with Generative Adversarial Networks. *arXiv preprint arXiv:1803.07422*, 2018.

[5] K. He, X. Zhang, S. Ren, and J. Sun. Identity Mappings in Deep Residual Networks. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pages 630–645. Springer, 2016.

[6] A. Hertz, S. Fogel, R. Hanocka, R. Giryes, and D. Cohen-Or. Blind Visual Motif Removal From a Single Image. In *Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 6858–6867. IEEE, 2019.

[7] E. G. Huang, G. B. Learned-Miller. Labeled Faces in the Wild: Updates and New Reporting Procedures. Technical Report UM-CS-2014-003, University of Massachusetts, Amherst, May 2014.

[8] Z. Hui, J. Li, X. Wang, and X. Gao. Image fine-grained inpainting. *arXiv preprint arXiv:2002.02609*, 2020.

[9] S. Iizuka, E. Simo-Serra, and H. Ishikawa. Globally and Locally Consistent Image Completion. *ACM Transactions on Graphics (ToG)*, 36(4):1–14, 2017.

[10] P. Isola, J.-Y. Zhu, T. Zhou, and A. A. Efros. Image-to-Image Translation with Conditional Adversarial Networks. In *Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 1125–1134. IEEE, 2017.

[11] J. Jam, C. Kendrick, V. Drouard, K. Walker, G.-S. Hsu, and M. H. Yap. Symmetric Skip Connection Wasserstein GAN for High-Resolution Facial Image Inpainting. *arXiv preprint arXiv:2001.03725*, 2020.

[12] J. Johnson, A. Alahi, and L. Fei-Fei. Perceptual Losses for Real-Time Style Transfer and Super-Resolution. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pages 694–711. Springer, 2016.

[13] I. Kemelmacher-Shlizerman, S. M. Seitz, D. Miller, and E. Brossard. The MegaFace Benchmark: 1 Million Faces for Recognition at Scale. In *Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 4873–4882. IEEE, 2016.

[14] X. Li, G. Hu, J. Zhu, W. Zuo, M. Wang, and L. Zhang. Learning Symmetry Consistent Deep CNNs for Face Completion. *IEEE Transactions on Image Processing*, 29:7641–7655, 2020.

[15] G. Liu, F. A. Reda, K. J. Shih, T.-C. Wang, A. Tao, and B. Catanzaro. Image Inpainting for Irregular Holes Using Partial Convolutions. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pages 85–100, 2018.

[16] Y. Liu, J. Pan, and Z. Su. Deep Blind Image Inpainting. In *International Conference on Intelligent Science and Big Data Engineering*, pages 128–141. Springer, 2019.

[17] Y. Ma, X. Liu, S. Bai, L. Wang, D. He, and A. Liu. Coarse-to-Fine Image Inpainting via Region-wise Convolutions and Non-Local Correlation. In *Proceedings of the 28th International Joint Conference on Artificial Intelligence, AAAI Press*, pages 3123–3129, 2019.

[18] A. L. Maas, A. Y. Hannun, and A. Y. Ng. Rectifier Nonlinearities Improve Neural Network Acoustic Models. In *ICML*, volume 30, page 3, 2013.

[19] J. Mathai, I. Masi, and W. AbdAlmageed. Does Generative Face Completion Help Face Recognition? pages 1–8, 2019.

[20] S. Nah, T. Hyun Kim, and K. Mu Lee. Deep Multi-scale Convolutional Neural Network for Dynamic Scene Deblurring. In *Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 3883–3891. IEEE, 2017.

[21] K. Nazeri, E. Ng, T. Joseph, F. Qureshi, and M. Ebrahimi. EdgeConnect: Structure Guided Image Inpainting using Edge Prediction. In *International Conference on Computer Vision Workshops (ICCVW)*, pages 3265–3274. IEEE, 2019.

[22] H.-W. Ng and S. Winkler. A data-driven approach to cleaning large face datasets. In *IEEE International Conference on Image Processing (ICIP)*, pages 343–347. IEEE, 2014.

[23] A. Odena, V. Dumoulin, and C. Olah. Deconvolution and Checkerboard Artifacts. *Distill*, 2016.

[24] O. Ronneberger, P. Fischer, and T. Brox. U-Net: Convolutional Networks for Biomedical Image Segmentation. In *International Conference on Medical Image Computing and Computer-Assisted Interventio (MICCAI)*, pages 234–241. Springer, 2015.

[25] X. Wang, K. Yu, S. Wu, J. Gu, Y. Liu, C. Dong, Y. Qiao, and C. Change Loy. ESRGAN: Enhanced Super-Resolution Generative Adversarial Networks. In *Proceedings of the European Conference on Computer Vision Workshops (ECCV)*, 2018.

[26] Y. Wang, Y.-C. Chen, X. Tao, and J. Jia. VCNet: A Robust Approach to Blind Image Inpainting. *arXiv preprint arXiv:2003.06816*, 2020.

[27] Y. Wang, X. Tao, X. Qi, X. Shen, and J. Jia. Image Inpainting via Generative Multi-column Convolutional Neural Networks. In *Advances in Neural Information Processing Systems (NeurIPS)*, pages 331–340, 2018.

[28] Z. Yan, X. Li, M. Li, W. Zuo, and S. Shan. Shift-Net: Image Inpainting via Deep Feature Rearrangement. In *Proceedings of the European conference on computer vision (ECCV)*, pages 1–17, 2018.

[29] J. Yu, Z. Lin, J. Yang, X. Shen, X. Lu, and T. S. Huang. Generative Image Inpainting with Contextual Attention. In *Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 5505–5514. IEEE, 2018.

[30] J. Yu, Z. Lin, J. Yang, X. Shen, X. Lu, and T. S. Huang. Free-Form Image Inpainting with Gated Convolution. In *International Conference on Computer Vision (ICCV)*, pages 4471–4480. IEEE, 2019.

[31] Y. Zeng, J. Fu, H. Chao, and B. Guo. Learning Pyramid-Context Encoder Network for High-Quality Image Inpainting. In *Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 1486–1494. IEEE, 2019.

[32] H. Zhang, I. Goodfellow, D. Metaxas, and A. Odena. Self-Attention Generative Adversarial Networks. In *International Conference on Machine Learning (ICML)*, pages 7354–7363. PMLR, 2019.

[33] H. Zhang, J. Yang, Y. Zhang, N. M. Nasrabadi, and T. S. Huang. Close the Loop: Joint Blind Image Restoration and Recognition with Sparse Representation Prior. In *IEEE International Conference on Computer Vision (ICCV)*, pages 770–777. IEEE, 2011.

[34] K. Zhang, Z. Zhang, Z. Li, and Y. Qiao. Joint Face Detection and Alignment using Multi-task Cascaded Convolutional Networks. *IEEE Signal Processing Letters*, 23(10):1499–1503, 2016.

[35] X. Zhang, X. Wang, B. Kong, Y. Yin, Q. Song, S. Lyu, J. Lv, C. Shi, and X. Li. Domain Embedded Multi-model Generative Adversarial Networks for Image-based Face Inpainting. *arXiv preprint arXiv:2002.02909*, 2020.

[36] T. Zhou, C. Ding, S. Lin, X. Wang, and D. Tao. Learning Oracle Attention for High-fidelity Face Completion. In *Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 7680–7689. IEEE, 2020.