Show Your Face: Restoring Complete Facial Images from Partial Observations for VR Meeting

Zheng Chen[1*] Zhiqi Zhang[2] Junsong Yuan[3] Yi Xu[2] Lantao Liu [1]
[1]Indiana University [2]OPPO US Research Center [3]University at Buffalo
{zc11, lantao}@iu.edu {zhiqi.zhang, yi.xu}@oppo.com jsyuan@buffalo.edu

Abstract

Virtual Reality (VR) headsets allow users to interact with the virtual world. However, the device physically blocks visual connections among users, causing huge inconveniences for VR meetings. To address this issue, studies have been conducted to restore human faces from images captured by Headset Mounted Cameras (HMC). Unfortunately, existing approaches heavily rely on high-resolution person-specific 3D models which are prohibitively expensive to apply to large-scale scenarios. Our goal is to design an efficient framework for restoring users' facial data in VR meetings. Specifically, we first build a new dataset, named Facial Image Composition (FIC) data which approximates the real HMC images from a VR headset. By leveraging the heterogeneity of the HMC images, we decompose the restoration problem into a local geometry transformation and global color/style fusion. Then we propose a 2D light-weight facial image composition network (FIC-Net), where three independent local models are responsible for transforming raw HMC patches and the global model performs a fusion of the transformed HMC patches with a pre-recorded reference image. Finally, we also propose a stage-wise training strategy to optimize the generalization of our FIC-Net. We have validated the effectiveness of our proposed FIC-Net through extensive experiments.

1. Introduction

Visual contact is important in human social communications because it can increase the level of presence and comfort, and promote close interaction among communicators. Visual contact is not only needed in the real world but also demanded by the next generation of communication mediums in Virtual Reality (VR), which relies on the headset to launch the virtual world. However, using a VR headset can also block the user's face, leading to the invisibility of the real appearance. This might be problematic for VR meetings where people want to see each other's real faces. See Figure 1 for an example.

Figure 1. An illustration of a video meeting in the VR world. There are four people — Jack, Tom, Arlen, and Molly having a video meeting with VR headsets. Instead of meeting with their avatars in the VR world, they want to see each other's real face, which is unfortunately blocked by the headset (a). In this work, we aim to reveal the 2D face behind the headset and enable people to have a real *face-to-face* video meeting (b) in the same manner as in the real world. Picture credit: [4].

To tackle this issue, a possible solution is to capture local areas of the human face by mounting several small InfraRed (IR) cameras, e.g., headset mounted cameras (HMC), to capture partial facial regions inside and around the headset. However, this solution is faced with a few major challenges: (1) IR cameras are of limited field of view and can only capture local facial regions (e.g., left eye, right eye, and mouth) due to the field of view and physical space in the headset. Therefore, restoring the complete face with sufficient details from these partial observations is non-trivial. (2) The positions of IR cameras deviate from the front view by a large angle. Closer objects thus have large non-linear distortions. Traditional photo geometric parametric models cannot work well under these conditions, and advanced restoration models can be too computationally costly to the mobile device with limited memory and computation resources; (3) IR cameras only detect thermal energy emitted from objects at dark environment inside the headset. The thermal images need further processing to recover color. It is difficult to make the appearance restoration work well for all faces of

*This work was done when Zheng Chen was an intern at OPPO US Research Center.

different skin colors and texture patterns.

To overcome the above challenges, we propose a 2D Facial Image Composition network (FIC-Net) that integrates local transformation models and a global composition model. Given three HMC images from the left eye, right eye, and mouth, respectively, we first transform local HMC images into a frontal view through a novel view synthesis. Three independent local models are built for images captured from three different facial parts. To better recover the color and appearance of the face, we provide a reference RGB image of the same user at the next global composition stage. Once the local patches are corrected, they are merged with the colorful reference image, the result of which is then used as the input to the global model. Finally, the global model will learn a global color transfer and image fusion supervised with the complete ground truth image.

To summarize, our contributions include:

- We propose a novel 2D image restoration model for restoring complete facial images from partial observations. The model works progressively from local patches to the global face, where local transformations are corrected first, following which the global color and texture pattern are restored.
- We propose a novel stage-wise training strategy so that the generalization of both local and global models can be maximized.
- We build a new 2D image composition dataset for restoring the complete 2D face and conduct extensive experiments to reveal the advantages of our proposed method. The comparisons over several other baselines validate the effectiveness of our method, which can generalize very well to new face images.

2. Related Work

Facial Expression Restoration: Existing approaches [5] [18] [15] [20] [8] [28] for restoring a complete face using partial observations in VR headset aim to drive an avatar using images from HMC. Those methods usually require a pre-captured personalized 3D neutral mesh of the user's head. After that, target textures and geometry are encoded from the HMC images. The encoded geometry is further injected into the 3D neutral model to generate the target 3D mesh. Then the target texture and the target 3D mesh is combined to render an avatar that represents the photo-realistic identity of the user. Although the existing 3D-based methods show excellent performance on restoration for VR headset users, they still suffer from several limitations. First, they heavily rely on person-specific 3D head models, which require extremely expensive data collection and computation, especially considering a unique 3D model is required for each training and each testing person. Second, the model complexity is high due to the use of 3D models, resulting in a heavy computational burden for practical deployment, particularly on mobile devices.

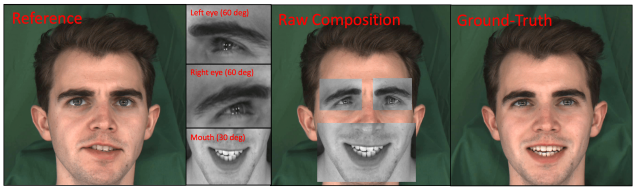

Figure 2. One example in the newly built FIC dataset. We use the frontal-view image at each frame as the ground-truth (gt) image, and use the complete face in another frame as the reference image. The HMC crops are captured from angled images with the same frame index as the gt image.

Table 1. FIC Dataset overview (splits).

Training	36545
Val	200
Weak Testing (Seen person; Unseen expression)	200
Strong Testing (Unseen person)	200

GAN Inversion for Face Editing: GAN inversion [26] is a powerful image editing technique. The general idea of GAN inversion for face editing is to encode the input face into a latent space where the desired manipulation/editing is applied to the latent code. Then the new face can be obtained by decoding the manipulated latent code through a decoder/generator. The manipulation of the latent code can be embeddings of geometric facial landmarks that are usually concatenated to the embedding of the image to form a new latent code [19]. Semantic segmentation is also treated as one latent space where changes can be made to the dense semantic image, from which a new real image is generated [13] [14] [21]. Another popular latent space for editing might be the latent style space of Style-GAN2. Along this direction, many representative works have been published including leveraging the disentanglement of the latent style space [25] [2], region-based semantic factorization of the style space by a dual optimization [30], exploring the local-rank property of latent subspace [29], and discovering the underlying variation factors controlling semantics [22]. In addition, Latent-Composition [1] and StyleMapGAN [11] are proposed to composite images. Both methods use StyleGAN-2 as the generator. The difference is Latent-Composition fixes the weights of the generator while StyleMapGAN trains the generator together with the image encoder. GAN-inversion-based methods are able to generate natural human faces, but fail to maintain the identity and details of expressions. Our FIC-Net is also based on GAN and focus on 2D image composition/synthesis task. Among all GAN-inversion-based methods, the closest work to ours is the StyleMapGAN. However, an important difference is that our method can maintain the expected identity and expression for the input image.

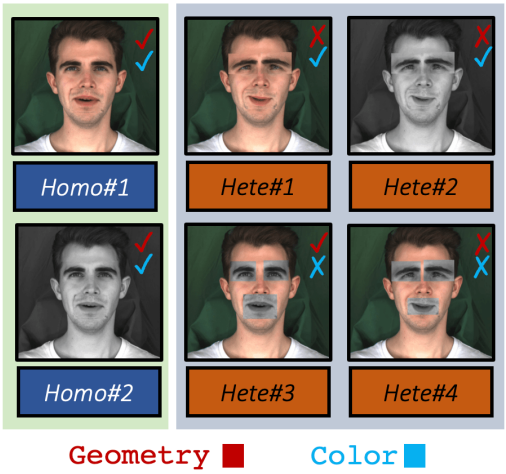

Figure 3. Examples of homogeneous and heterogeneous images. The homogeneity/heterogeneity is classified based on geometry and color. We use ✓and ✗representing consistency and inconsistency in terms of geometry (red) and color (cyan). Best viewed in color.

3. Dataset

To support the task of 2D face restoration, we build a new dataset — FIC dataset to approximate the real user case of a VR headset. Our newly built dataset is based on the MEAD dataset [23] which collects video corpuses featuring different people talking with eight differing expressions at three distinct intensity levels. For each person, the video clips are captured from seven different perspective angles, i.e., left-30 (meaning the camera is placed in 30 degrees to the left of the person); left-60; right-30; right-60; top-30; down-30; front, simultaneously in a controlled environment. Data in the MEAD dataset provides sufficient materials to approximate the VR scenarios, e.g., a cropped left eye patch in the left-60 image represents the image from left HMC; a cropped mouth in the top-30 image represents the image from the mouth HMC. To maximize the restoration performance, we also provide a reference image that can provide critical information about the color of the face and the texture of the out-of-local (outside local patches) regions. In real scenarios, the reference image could be captured before the headset is used.

We select 14 representative identities in the MEAD dataset and use all video clips of those persons to make our FIC dataset. To crop the local patches, we first use an existing landmark detection method [6] [7] [31] to detect 68 standard landmarks in a complete face. Then we use the index to classify the groups of landmarks and use the range (with proper margins) of the corresponding landmark group to determine the crop regions — left (including left eye and left eyebrow); right (including right eye and right eyebrow); and mouth (including most part of cheek). We show one example of our built data in Figure 2.

In the FIC dataset, to fully examine the model generalization to different data, we particularly split the testing data into two categories — Weak Testing (WT) and Strong Testing (ST) (see Table 1). We split the testing data according to the visibility of the expression or the identity during training. WT are images whose identities are seen while the expressions are not seen during training. ST represents images that the model has never seen. It is fair to claim that both WT and ST are not seen by the model during training, but they represent different levels of novelty to the model — WT data already leak the identity information into the training process, and this makes the generalization easier, while ST data are completely new (except for the background) to the model, which makes the generalization harder.

4. Methodology

4.1. Data Analysis

The data in FIC dataset is different from the data in conventional image editing tasks, where both the reference image and target image share the same geometric view and color pattern. We call those types of image data homogeneous images. On the contrary, FIC data consists of images with different geometric views and color patterns. For example, the local patches — left, right, and mouth patches are captured from three different non-frontal camera angles and all images are grayscale. The reference image instead is captured from a frontal camera angle and is in full RGB color. During inference, the reference image can be captured before the VR headset is used. Our model can learn to match the local patches with the provided reference image. During training, the reference image is the corresponding ground-truth image. Note that we only use the non-local regions that exclude the left eye, the right eye, and the mouth from the reference image during training. We call our FIC data heterogeneous images.

We show the comparison of different levels of heterogeneities in Figure 3, where the left block contains examples of homogeneous images while the right block contains heterogeneous ones. Homogeneous images are consistent with respect to the geometry view (*e.g.* each pixel is captured from the same camera angle) and color pattern (*e.g.* all pixels are grayscale or RGB) across the whole image. Heterogeneous images can show different levels of heterogeneities. For example, the heterogeneity of *Hete#1* and *Hete#2* is only from the geometric views as they are consistent in regard to the color pattern — either RGB or grayscale. The heterogeneity of *Hete#3* is only from the color pattern as all pixels share the same geometric view. *Hete#4* has the largest heterogeneity as pixels are varying with respect to both geometric view and color patterns.

Our FIC data is similar to the *Hete#4*, and have different heterogeneities. Our task is to generate homogeneously real faces from heterogeneous input images. To achieve this, we decompose the large heterogeneity into smaller ones and adopt a local-to-global structure, where the local part is re-

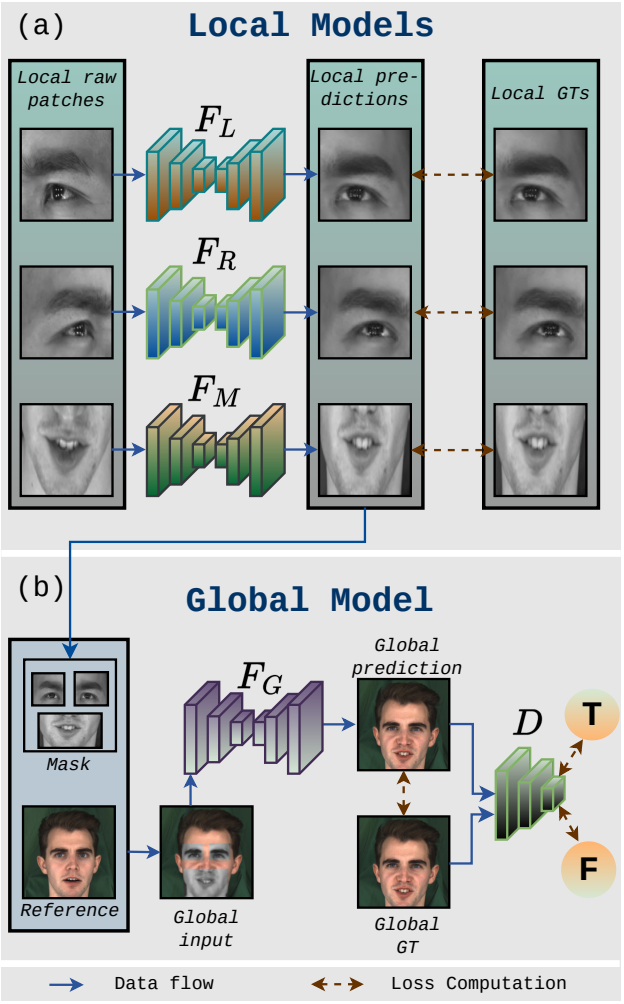

Figure 4. Overview of the FIC-Net structure. F_L, F_R, and F_M: Neural nets for transforming left eyes, right eyes, and mouth, respectively. F_G: Neural net for the global fusion. The discriminator D is responsible for distinguishing whether the global prediction is real data (True) or fake data (False).

sponsible for only geometric heterogeneity and the global part is responsible for only color pattern heterogeneity. The reasons for this decomposition are two-fold. First, it might be easier for the model to learn single heterogeneity than a complex combination of heterogeneities. Second, local models should not attempt to predict color as the user might have different skin colors, and no clue about the color of the user is provided in the HMC images. We cannot assume any prior information about color during local transformation. Valid information about color can only be learned in the global phase, where the reference RGB image of the user is provided. We further decompose the geometric heterogeneity into independent ones as different local patches have different geometric views, i.e., three independent models handle three independent patches.

4.2. Overview of the Proposed Model

By using the decomposition discussed in Section 4.1, we propose our FIC-Net. The overall structure of the FIC-Net can be seen in Figure 4. Two hierarchies are adopted — one is the local-to-global hierarchy while the other one is the independence of different local models. FIC-Net first frontalizes different local patches separately. Then an intermediate image which consists of the reference background and frontalized local patches is generated. The global model takes this intermediate image as the input and eliminates the heterogeneity of the color pattern. We introduce the local transformation models in Section 4.3; the global fusion model in Section 4.4; and the training algorithm of the whole FIC-Net in Section 4.5.

4.3. Local Transformation Models

We use three independent models to perform the local transformation for three local patches separately. The three models share the same structure but do not share weights because different patches have different transformations. All local models use a simple autoencoder (AE) as the backbone. In the FIC dataset, we provide the ground-truth for different patches. Training for all local models is fully supervised.

We use the MSE and LPIPS [27] to compute the supervised loss for training local models. The MSE loss is computed between the predicted local patch and the gt patch

$$\mathcal{L}_{\text{mse}} = \frac{1}{N} \left\| p_t - \hat{x} \right\|_2^2, \tag{1}$$

where N is the total number of pixels in the image, $\hat{p}_t$ is the prediction while $\hat{x}$ is the local gt image, see Table 2 for more information about notations. For the convenience of notations, we omit the subscript for local patch names, e.g., l for the left patch. We denote the LPIPS loss [27] as $\mathcal{L}_{\text{lpips}}$. The total loss for a local model is

$$\mathcal{L}_{\text{local}} = \mathcal{L}_{\text{mse}} + \mathcal{L}_{\text{lpips}}. \tag{2}$$

4.4. Global Fusion Model

The global fusion model is responsible for eliminating the heterogeneity of color patterns. We build the global fusion model using the GAN framework, as shown in Figure 4. The generator is still an autoencoder. We use the provided complete face ground-truth image as the supervision of the generator. Similar to local models, we use MSE and LPIPS [27] to compute the supervised loss between the global prediction and the ground-truth image. We denote the MSE and LPIPS [27] loss for global model as $\mathcal{L}'_{\text{mse}}$ and $\mathcal{L}'_{\text{lpips}}$.

We also have an extra adversarial loss for the global model. The loss we use is the non-saturating loss [3] together with the R1 regularization [16]. We denote the adversarial loss as $\mathcal{L}_{\text{adv}}$. Then the total loss for the global model

Table 2. Notations used for representing different data.

Symbol	Description	Symbol	Description	Symbol	Description
x	Frontal RGB complete image	$\hat{x}$	Frontal gray complete image		
x_l	Frontal RGB left gt	x_r	Frontal RGB right gt	x_m	Frontal RGB mouth gt
$\hat{x}_l$	Frontal gray left gt	$\hat{x}_r$	Frontal gray right gt	$\hat{x}_m$	Frontal gray mouth gt
$\hat{x}_{lt}$	Tilted gray left patch	$\hat{x}_{rt}$	Tilted gray right patch	$\hat{x}_{mt}$	Tilted gray mouth patch
$\hat{p}_{lt}$	Final predicted gray left	$\hat{p}_{rt}$	Final predicted gray right	$\hat{p}_{mt}$	Final predicted gray mouth

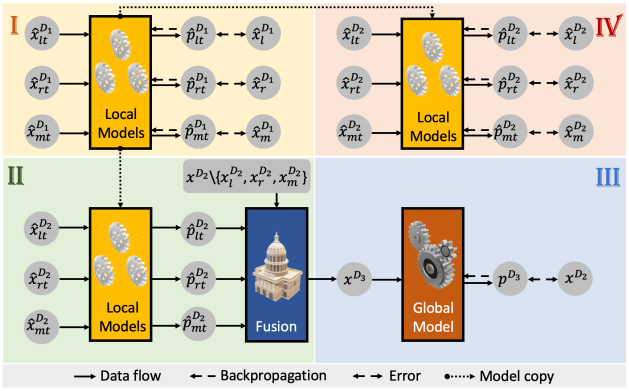

Figure 5. Training stages for FIC-Net.

becomes

$$\mathcal{L}_{\text{global}} = \mathcal{L}'_{\text{mse}} + \mathcal{L}'_{\text{lpips}} + \mathcal{L}_{\text{adv}}. \quad (3)$$

4.5. Stage-wise Model Training

To clearly describe our training strategy, notations for representing different image data are listed in Table 2.

We propose a four-stage training strategy for efficiently training the FIC-Net. The overview of the training strategy can be seen in Figure 5. There are several different domains of data involved during the whole training process. We use the superscript of $D_{\#}$ to represent different domains.

Stage-I: We first split the training data into two sets, $\mathcal{D}_1$ and $\mathcal{D}_2$. In this stage, we train local models only using the data in $\mathcal{D}_1$.

Stage-II: We use local models trained in Stage-I to predict frontalized patches for the raw patches of $\mathcal{D}_2$. Then for each data sample in $\mathcal{D}_2$, we use the provided mask in the FIC dataset to crop the out-of-local (background) region in the reference image and fill the frontalized patches in. This fusion operation generates a new domain of data, $\mathcal{D}_3$, where all pixels of each image share the same geometric view.

Stage-III: The newly generated data $\mathcal{D}_3$ is used as the data for training the global model. As we build the $\mathcal{D}_3$ based on $\mathcal{D}_2$, we can directly use the gt images in $\mathcal{D}_2$ to supervise the training of the global model.

Stage-IV: We resume the training of local models using the data in $\mathcal{D}_2$, as the union of $\mathcal{D}_1$ and $\mathcal{D}_2$ is the complete training data. We expect the final local models to be trained thoroughly using all the training data.

Why do we split the data into $\mathcal{D}_1$ and $\mathcal{D}_2$? We split the data into two sets because training local models using all data in the Stage-I might do harm to the training of global model in the Stage-III. Suppose we have trained the local models thoroughly in Stage-I, then in Stage-II, we might want to predict the local patches which have been seen by the model during Stage-I, leading to all images in $\mathcal{D}_3$ having near-perfect local patches. This will cause strong bias to the input data of the global model and demolish the generalization as the global model might need to deal with non-perfect local patches from local models. To diminish the domain shift of training data and testing data for global model, we need to train the global model using non-perfect local patches. To achieve this, we have to use the trained local models to predict patches they never saw during Stage-I, therefore we need to pre-split the training data into two sets such that images in $\mathcal{D}_2$ is novel to the trained local models, making data in $\mathcal{D}_3$ similar to the inference phase.

5. Experiments

5.1. Baselines and Implementation Details

We compare our proposed model (FIC-Net) with several related methods, including the recently released StyleMapGAN (SMG) [12] which aims at local editing using a StyleGAN [9]; Poisson Blending (PB) [17] that fuses local patches using a geometric method. To support the prediction, we use our trained local models to predict frontalized local patches. PB is only responsible for global fusion. We also compare our proposed local-to-global model with a single autoencoder model (AE) that takes as input a mixture of raw HMC patches and the out-of-local regions in the reference image. The input image for the AE model can be seen as the *Raw Composition* in Figure 2.

The local models and the global model share the same AutoEncoder (AE) structure. We show the details of our AE backbone in Table 3. Our AE backbone contains three downsamplings and three upsamplings. The difference between the local model and the global model lies in the channels for layer `conv0` and `conv5`, where 1 is used for lo-

Table 3. AutoEncoder network structure in our FIC-Net.

Layer	Filter Size	Output Size
conv0	3 × 3/1	w × h × 64
conv1	3 × 3/2	w/2 × h/2 × 128
conv2	3 × 3/2	w/4 × h/4 × 256
conv3	3 × 3/2	w/8 × h/8 × 512
deconv0	3 × 3/2	w/4 × h/4 × 256
deconv1	3 × 3/2	w/2 × h/2 × 128
deconv2	3 × 3/2	w × h × 64
conv4	3 × 3/1	w × h × 64
conv5	3 × 3/1	w × h × 3

Table 4. Quantitative comparison for weak testing data.

Method	Runtime (s) (↓)	MSE_g (↓)	MSE_{in} (↓)	MSE_{out} (↓)	SSIM (↑)	PSNR (↑)	LPIPS (↓)
StyleMapGAN [12]	0.0202	0.0570	0.1534	0.0287	0.7773	20.2123	0.1319
Poisson Blending (PB) [17]	0.0214	0.0119	0.0462	0.0017	0.9559	24.0310	0.1280
AutoEncoder (AE)	0.0098	0.0053	0.0210	0.0009	0.9132	27.7129	0.1264
FIC	0.0105	0.0048	0.0190	0.0001	0.9544	28.6264	0.0996

Table 5. Quantitative comparison for strong testing data.

Method	Runtime (s) (↓)	MSE_g (↓)	MSE_{in} (↓)	MSE_{out} (↓)	SSIM (↑)	PSNR (↑)	LPIPS (↓)
StyleMapGAN [12]	0.0196	0.0209	0.0480	0.0120	0.8389	21.8831	0.1843
Poisson Blending (PB) [17]	0.0224	0.0051	0.0185	0.0007	0.9565	27.8583	0.1711
AutoEncoder (AE)	0.0101	0.0081	0.0389	0.0010	0.9258	25.7609	0.1021
FIC	0.0113	0.0075	0.0316	0.0005	0.9404	26.1046	0.0863

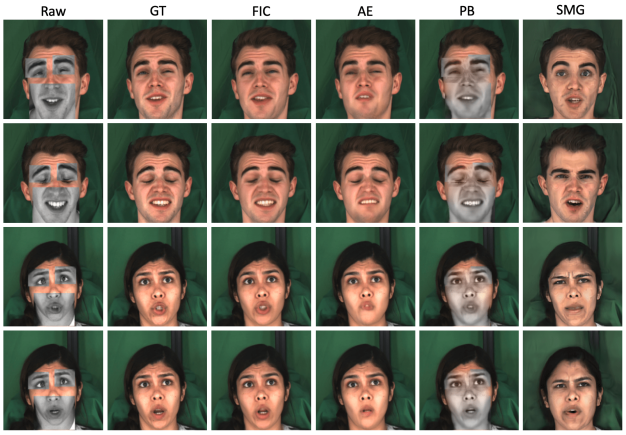

Figure 6. Qualitative comparison on weak testing data.

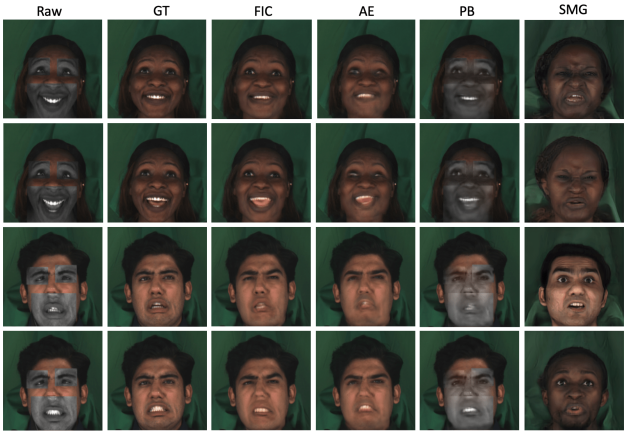

Figure 7. Qualitative comparison on strong testing data.

cal models while 3 is used for the global model. We also explore the performance of adding skip connections between the encoder and the decoder. We empirically find that adding skip connections might hurt the performance. More details about the ablation for skip connections can be found in Section 5.4.

5.2. Evaluation Metrics

We use several metrics to evaluate the distance between the generated face and the gt image. To assess the pixel-level error between the generated image and its gt, we use Mean Squared Error (**MSE**). Given that our application focuses on the synthesis of eyes and mouth, the quality of generated images inside the patches is more critical/difficult than the ones outside the patches. Therefore, we use $\mathbf{MSE_{in}}$ to evaluate the errors inside the patches and $\mathbf{MSE_{out}}$ to evaluate the errors outside the patches, and $\mathbf{MSE_g}$ to evaluate the errors of the whole images. To evaluate the faithfulness, we adopt the Structural Similarity Index Measure (**SSIM**) [24] and Peak Signal-Noise Ratio (**PSNR**). To evaluate the perceptual quality, we use Learned Perceptual Image Patch Similarity (**LPIPS**) [27].

5.3. Comparisons

Weak Testing Data We first show the qualitative comparison of different methods in Figure 6, where the column of *Raw* shows the input data in a form of raw composition, from which we can see what the reference image background and raw local patches are. Our FIC-Net shows better image composition than other baselines. From Figure 6, we can see that AE has the closest performance to our FIC-Net. However, the results from AE are inferior to ours with respect to the texture details of local patches, especially for the transformation of the mouth patch. This issue happens to AE because different local patches have different transformations, but AE mixes the learning of different transformations in one single model.

Among different presented methods in Figure 6, PB shows the visually-worst results. The biggest issue with PB is it lacks the capacity to blend images with large heterogeneity (color in Figure 6) even after we have provided the frontalized local patches. PB attempts to fuse the color and texture in the boundary of the local patches, for example, the right eye part of the first row and second row; the face of the third row. But the attempt is too weak to reach most regions inside local patches.

The SMG method shows results with a very different style compared with other methods. Although the generated images look natural and smooth, they have lost the precise correspondence with the gt image. In SMG, we encode the input image to a latent stylemap code, from which the new images are generated. The generated image might be able to maintain the general identity of the person, but the

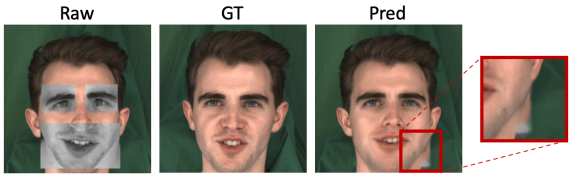

Figure 8. Artifacts in AE predictions.

expression will be out of control compared with the corresponding gt image. This is problematic for VR applications as the user might be sensitive to the inconsistency of the real expression and the generated out-of-control expressions. More detailed quantitative comparisons can be seen in Table 4. We can still see the advantage of our proposed method over other baselines on different quantitative evaluations, showing the effectiveness of our proposed model to generate images with low pixel-wise errors; low image noise; and strong structural and perceptual recovery.

One interesting thing one might notice in Table 4 is that the SSIM value for the PB method is the highest among all the presented methods. The reason for this is that SSIM ignores the color information but only focuses on the structure in the image, thus the failure of fusing color is not reflected in the SSIM value.

Strong Testing Data Similar to the analysis of the weak testing data, we show qualitative and quantitative comparisons for strong testing data in Figure 7 and Table 5, respectively. As we can see in Figure 7, our proposed FIC-Net is able to maintain a high-level performance for the data it never sees. Predictions from AE become more blurry compared with its predictions for weak testing data, *e.g.* eyes in the first row and the second row in the column of *AE*.

However, we found the evaluation values for PB are particularly abnormal — PB can be the best method if we only look at Table 5. But we can see the real performance from Figure 7, where PB still fails to fuse the color and shows the visually-worst results. The reason for this is related to the skin color of the person. RGB values of a dark-color person tend to be close to the values of grayscale local patches. In this case, staying near the original local patches — as what the PB does, will make the pixel values be close to the values of the gt image. On the contrary, other methods trying to adjust the image values of local patches will enlarge the distance between the generated image and the gt image. We leave the investigation of a better evaluation metric considering skin color as future work.

Also, we can observe from Figure 7 that SMG takes the strong testing images as input, but the output images are actually natural images in training data with novel expressions. This set of predictions from SMG validates the poor generalization of the GAN-based methods. The generator might be able to only learn the distribution of the given training data. In this case, any new data in testing will be encoded into the latent space of the training data.

Table 6. Comparison of different stage-wise training.

Stages	MSE_{in} (↓)	SSIM (↑)	LPIPS (↓)
Three-stage	0.0448	0.9389	0.0903
Four-stage	0.0316	0.9404	0.0863
Δ	-0.0132	+0.0015	-0.0040

Figure 9. Results for strong testing data if we randomly sample a reference image during training.

5.4. Ablation Studies

AE vs. FIC From the comparisons in Section 5.3, we can see that the performance of AE is the closest one to our proposed method, particularly for weak testing data. The difference between the AE and our proposed FIC is that we use a hierarchical local-to-global structure. The AE method attempts to learn all functions with one single model in one stage. This mixture of learning might be problematic for our FIC data because our data is a kind of heterogeneous data. The local patches from HMC might contain noises, *e.g.* the neck part in the raw mouth patch in Figure 8. This kind of noise will significantly affect the final image prediction, as we show in the *pred* image of Figure 8, where the neck has an obvious displacement compared with the gt image.

Different stage-wise training strategies We compare the performance of the model trained with two different stage-wise strategies in Table 6. "Three-stage" means that we do not resume the training for local models in Stage-IV(see Figure 5), while the "four-stage" means that the complete training data is applied to train local models. We can see that a complete training of local models (in the four-stage) has a better performance than the three-stage strategy. This validates that the additional training of local models can improve model generalization.

How to select reference image? For the real use case, we can obtain a reference image before the VR headset is used. Intuitively, the reference image is an image that could differ from the images when using the VR headset. Based on this application setting, a straightforward way to select the reference image could be a random sampling of an image of the same person.

We show quantitative comparisons of the random sampling and ours in Table 7, where we can see the generalization performance on the strong testing data of our adopted strategy is significantly superior to using the random sampling. We further show some examples of predictions by us-

Table 7. Comparison of different reference images.

Reference source	MSE_{in} (↓)	SSIM (↑)	LPIPS (↓)
Random sampling	0.1628	0.7256	0.1529
Ground truth	0.0316	0.9404	0.0863
Δ	-0.1312	+0.2148	-0.0666

Table 8. Ablation for dataset split.

	MSE_{in} (↓)	SSIM (↑)	LPIPS (↓)
w.o Split	0.0513	0.9326	0.0902
w. Split	0.0316	0.9404	0.0863
Δ	-0.0197	+0.0078	-0.0039

ing the random sampling strategy in Figure 9, from which we can see the prediction collapses when it comes to the strong testing data.

The reason for this problem is that if we use the background face in a randomly sampled face, then the model might be confused to learn the mapping relation for the out-of-local region. Since the out-of-local region might take more than half pixels of the whole image, then the confusion about this big region might lead to the failure of predicting the whole face. If we use the background of the gt image, we do not intend to leak information but want to explicitly inject very useful knowledge into the model — the out-of-local region of the output should be maintained the same as the input. By doing so, we can reduce the complexity of the learning process and resolve the potential confusion about the model, leading to the whole training being more efficient.

Is data split of $\mathcal{D}_1$ and $\mathcal{D}_2$ necessary? We use the data split of $\mathcal{D}_1$ and $\mathcal{D}_2$ to enhance the generalization of the global model. We show the comparison of using the data split and not using it in Table 8. As we can see, the model performance can be boosted if we apply the data split to the training data.

Do we need skip connections? In this work, we also explore the use of the skip connection between the encoder and the decoder in the autoencoder backbone network. We show the comparison in Table 9, where we test different network settings. We can observe that the model can obtain better performance without using the skip connection, especially when local models remove the skip connection. The skip connection is usually used to strengthen the feature intensity for the decoder features. However, in our FIC-Net we need to learn local transformations to frontalize local patches, meaning the input of the model has a different geometric structure than the output. This implies that the features in the encoder and the decoder have different distributions. If we apply skip connections between the encoder and the decoder, we might damage the learned frontalized features in the decoder, and fail to learn transformations.

GAN Ablation In Section 5, we have shown that the GAN-based methods, *e.g.* SMG [12], cannot maintain the desired identity, especially for images that are never seen during training. In this experiment, we only use homogeneous RGB images (no geometry or color heterogeneities) from another face dataset FFHQ [10], see Figure 10. We randomly select reference images (first column, Figure 10) that provide the face background and source images (second column, Figure 10) that provide three local patches. Then we can obtain new composited images as shown in the third column. The output from the model can be seen in the last column. We can see that although the generated images are natural, the identity has been changed to different extents. It seems that the model is just trying to find the closest image to the input composited image from its learned image distribution. This phenomenon is consistent with our findings in Section 5.

Table 9. Ablation for skip connection in backbone net.

	L-S	G-S	MSE_{in} (↓)	SSIM (↑)	LPIPS (↓)
1	✓	✓	0.0417	0.9130	0.1195
2	✓	-	0.0421	0.9182	0.1227
3	-	✓	0.0355	0.9399	0.0859
4	-	-	0.0316	0.9404	0.0863

Figure 10. GAN-based composition on FFHQ.

6. Conclusion

Restoring real faces covered by VR headsets is of great significance to VR applications. However, there are two major hurdles to achieve this efficiently. The first one is existing approaches heavily rely on person-specific 3D-model-based methods, hindering large-scale applications across different platforms, especially for mobile devices. The second one is the community still lacks sufficient data to support research/engineering works related to restoring complete faces. To overcome the existing obstacles, we first build a new 2D face restoration dataset that highly approximates the real use case of a VR headset. Then we propose a 2D lightweight face restoration model that uses a local-to-global hierarchy. Our model reveals high effectiveness under a series of challenging experimental settings.

References

[1] Lucy Chai, Jonas Wulff, and Phillip Isola. Using latent space regression to analyze and leverage compositionality in gans. *arXiv preprint arXiv:2103.10426*, 2021. 2

[2] Edo Collins, Raja Bala, Bob Price, and Sabine Susstrunk. Editing in style: Uncovering the local semantics of gans. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 5771–5780, 2020. 2

[3] Antonia Creswell, Tom White, Vincent Dumoulin, Kai Arulkumaran, Biswa Sengupta, and Anil A Bharath. Generative adversarial networks: An overview. *IEEE signal processing magazine*, 35(1):53–65, 2018. 4

[4] Dreamstime. DreamsTime, online homepage. https://www.dreamstime.com/. 1

[5] Shin en Wei, Jason Saragih, Tomas Simon, Adam W. Harley, Stphen Lombardi, Michal Perdoch, Alexander Hypes, Dawei Wang, Hernan Badino, and Yaser Sheikh. Vr facial animation via multiview image translation. volume 38, 2019. 2

[6] Jianzhu Guo, Xiangyu Zhu, and Zhen Lei. 3ddfa. https://github.com/cleardusk/3DDFA, 2018. 3

[7] Jianzhu Guo, Xiangyu Zhu, Yang Yang, Fan Yang, Zhen Lei, and Stan Z Li. Towards fast, accurate and stable 3d dense face alignment. In *Proceedings of the European Conference on Computer Vision (ECCV)*, 2020. 3

[8] Amin Jourabloo, Fernando De la Torre, Jason Saragih, Shih-En Wei, Stephen Lombardi, Te-Li Wang, Danielle Belko, Autumn Trimble, and Hernan Badino. Robust egocentric photo-realistic facial expression transfer for virtual reality. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 20323–20332, 2022. 2

[9] Tero Karras, Samuli Laine, and Timo Aila. A style-based generator architecture for generative adversarial networks. In *CVPR*, 2019. 5

[10] 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. 8

[11] Hyunsu Kim, Yunjey Choi, Junho Kim, Sungjoo Yoo, and Youngjung Uh. Exploiting spatial dimensions of latent in gan for real-time image editing. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 852–861, 2021. 2

[12] Hyunsu Kim, Yunjey Choi, Junho Kim, Sungjoo Yoo, and Youngjung Uh. Exploiting spatial dimensions of latent in gan for real-time image editing. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2021. 5, 6, 8

[13] Cheng-Han Lee, Ziwei Liu, Lingyun Wu, and Ping Luo. Maskgan: Towards diverse and interactive facial image manipulation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 5549–5558, 2020. 2

[14] Huan Ling, Karsten Kreis, Daiqing Li, Seung Wook Kim, Antonio Torralba, and Sanja Fidler. Editgan: High-precision semantic image editing. *Advances in Neural Information Processing Systems*, 34:16331–16345, 2021. 2

[15] Stephen Lombardi, Jason Saragih, Tomas Simon, and Yaser Sheikh. Deep appearance models for face rendering. *ACM Transactions on Graphics (ToG)*, 37(4):1–13, 2018. 2

[16] Lars Mescheder, Andreas Geiger, and Sebastian Nowozin. Which training methods for gans do actually converge? In *International conference on machine learning*, pages 3481–3490. PMLR, 2018. 4

[17] Patrick Perezs, Michel Gangnet, and Andrew Blake. Poisson image editing. In *SIGGRAPH*, 2003. 5, 6

[18] Hai Pham and Vladimir Pavlovic. Speech-driven 3d facial animation with implicit emotional awareness: A deep learning approach. In *cvpr*, 2017. 2

[19] Fengchun Qiao, Naiming Yao, Zirui Jiao, Zhihao Li, Hui Chen, and Hongan Wang. Geometry-contrastive gan for facial expression transfer. *arXiv preprint arXiv:1802.01822*, 2018. 2

[20] Gabriel Schwartz, Shih-En Wei, Te-Li Wang, Stephen Lombardi, Tomas Simon, Jason Saragih, and Yaser Sheikh. The eyes have it: An integrated eye and face model for photo-realistic facial animation. *ACM Transactions on Graphics (TOG)*, 39(4):91–1, 2020. 2

[21] Yujun Shen, Ceyuan Yang, Xiaoou Tang, and Bolei Zhou. Interfacegan: Interpreting the disentangled face representation learned by gans. *IEEE transactions on pattern analysis and machine intelligence*, 2020. 2

[22] Yujun Shen and Bolei Zhou. Closed-form factorization of latent semantics in gans. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 1532–1540, 2021. 2

[23] Kaisiyuan Wang, Qianyi Wu, Linsen Song, Zhuoqian Yang, Wayne Wu, Chen Qian, Ran He, Yu Qiao, and Chen Change Loy. Mead: A large-scale audio-visual dataset for emotional talking-face generation. In *ECCV*, 2020. 3

[24] Zhou Wang, Alan C Bovik, Hamid R Sheikh, and Eero P Simoncelli. Image quality assessment: from error visibility to structural similarity. *IEEE transactions on image processing*, 13(4):600–612, 2004. 6

[25] Zongze Wu, Dani Lischinski, and Eli Shechtman. Stylespace analysis: Disentangled controls for stylegan image generation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 12863–12872, 2021. 2

[26] Weihao Xia, Yulun Zhang, Yujiu Yang, Jing-Hao Xue, Bolei Zhou, and Ming-Hsuan Yang. Gan inversion: A survey. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2022. 2

[27] 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. 4, 6

[28] Yajie Zhao, Qingguo Xu, Weikai Chen, Chao Du, Jun Xing, Xinyu Huang, and Ruigang Yang. Mask-off: Synthesizing face images in the presence of head-mounted displays. In *2019 IEEE Conference on Virtual Reality and 3D User Interfaces (VR)*, pages 267–276. IEEE, 2019. 2

[29] Jiapeng Zhu, Ruili Feng, Yujun Shen, Deli Zhao, Zheng-Jun Zha, Jingren Zhou, and Qifeng Chen. Low-rank subspaces in

gans. *Advances in Neural Information Processing Systems*, 34:16648–16658, 2021. 2

[30] Jiapeng Zhu, Yujun Shen, Yinghao Xu, Deli Zhao, and Qifeng Chen. Region-based semantic factorization in gans. *arXiv preprint arXiv:2202.09649*, 2022. 2

[31] Xiangyu Zhu, Xiaoming Liu, Zhen Lei, and Stan Z Li. Face alignment in full pose range: A 3d total solution. *IEEE transactions on pattern analysis and machine intelligence*, 2017. 3