IEEE Access
Multidisciplinary | Rapid Review | Open Access Journal

Received May 19, 2020, accepted June 8, 2020, date of publication June 11, 2020, date of current version June 24, 2020.

Digital Object Identifier 10.1109/ACCESS.2020.3001649

Effective Removal of User-Selected Foreground Object From Facial Images Using a Novel GAN-Based Network

NIZAM UD DIN, KAMRAN JAVED, SEHO BAE, AND JUNEHO YI
Department of Electrical and Computer Engineering, Sungkyunkwan University, Suwon 16419, South Korea

Corresponding author: Juneho Yi (jhyi@skku.edu)

This work was supported by the National Research Foundation of Korea (NRF) Grant funded by the Korea Government (MSIT) under Grant 2020R1F1A1048438.

ABSTRACT This research features a user-friendly method for face de-occlusion in facial images where the user has control of which object to remove. Our system removes one object at a time, however, it is capable of removing multiple objects through repeated application. Although we show the effectiveness of our system on five commonly occurring occluding objects including hands, a medical mask, microphone, sunglasses, and eyeglasses, more types of object can be considered based on the proposed methodology. Our model learns to detect a user-selected, possibly distracting, object in the first stage. Then, the second stage removes the object using the object detection information from the first stage as guidance. To achieve this, we employ GAN-based networks in both stages. Specifically, in the second stage, we integrate both partial and vanilla convolution operations in the generator part of the GAN network. We show that by using this integration, the proposed network can learn a well-incorporated structure and also overcome the problem of visual discrepancies in the affected region of the face. To train our network, we produce a paired synthetic face-occluded dataset. Our model is evaluated using real world images collected from the Internet and publicly available CelebA and CelebA-HQ datasets. Experimental results confirm our model's effectiveness in removing challenging foreground non-face objects from facial images as compared to the existing representative state-of-the-art approaches.

INDEX TERMS Generative adversarial network, object removal, image editing, image completion.

I. INTRODUCTION

The goal of this work is to effectively remove user-selected foreground objects from facial images and fill in the resulting hole with plausible content. We synthesize distraction-free face results from images that originally contain multiple types of occluding objects. In this work we consider five objects: hands, a medical mask, microphone, sunglasses, and eyeglasses. However, this method can be applied to various object types. We allow the user to select the object to be removed by employing an object label encoder. Our model can remove multiple types of occlusions on the face by sequential application of the system as can be seen in Figure 1. Automatic editors of this sort can be used for further processing of faces such as segmentation, occluded face recognition [1], [2] and data augmentation [3].

The associate editor coordinating the review of this manuscript and approving it for publication was Orazio Gambino.

Removing objects and filling in the holes in facial images, especially when the resulting holes are large and irregularly shaped, is the most challenging scenario. It becomes more challenging when the damaged region involves a transition between two different segments. For example, objects like medical masks, microphones, and hands cover portions beyond the actual boundary of the face. Hands add more complexity since the textures of the hands and face are similar.

Conventional object removal works [4]–[6], remove unwanted objects and produces missing content by iteratively finding similar patches from the remainder of the image or an external database. While these non-learning-based algorithms produce smooth results, they are heavily dependent on the available image statistics. In general, these traditional methods synthesize plausible results for small objects with standard locations but fail critically for large arbitrarily located objects.

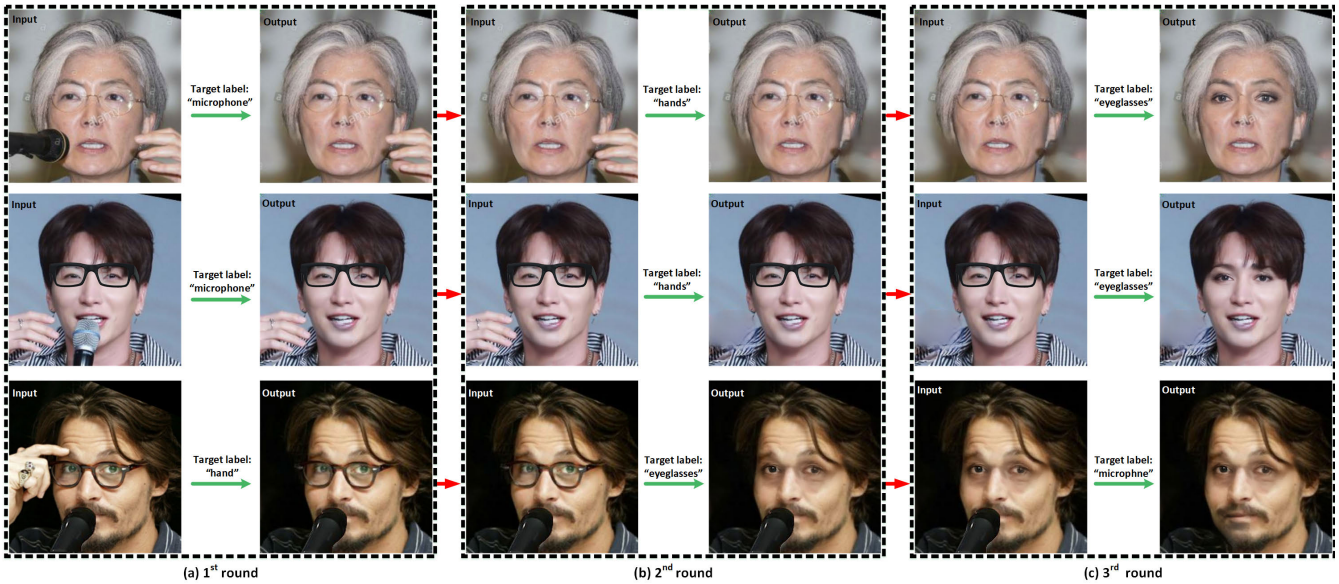

FIGURE 1. Result of our method for test samples having more than one type of occluding object on the face: (a) first round output of our model given the corresponding target object label, (b) second round output of our model given the corresponding target object label, (c) third round output of our model given the corresponding target object label.

Recent deep-learning-based image inpainting methods [7]–[14] show potential in removing unwanted objects and reconstructing the damaged regions. In [15] and [8], only centered rectangular regions are considered, making its usage limited with regard to our task. Approaches like those in [7], [10], [12] attempt to handle arbitrarily shaped objects but are unable to overcome the complexity of the task and produce artifacts. FaceD [13] proposed a 3D morphable model (3DMM) conditioned face de-occlusion algorithm to restore de-occluded faces from six common occluded non-face objects. This method produces plausible results but struggles with large, complex occlusions. Moreover, it cannot handle the removal of multiple types of objects on the face. All of these deep-learning-based methods assume that an object map or a 3D face model is given. One recent work [16] automatically detects and removes a medical mask object from facial images. This method generates better results in removing mask objects but does not generalize well for multiple types of objects.

Moreover, all aforementioned deep-learning-based methods only use vanilla convolution as the backbone of their deep-learning-based networks. This vanilla convolution applies the same filter weights throughout the image, regardless of whether the region is valid or affected. This helps in achieving well-incorporated predictions but leads to severe visual artifacts, especially at the boundaries of the valid and affected regions of the image as reported by [17] and [18]. This problem becomes more severe if a region to be filled is large or of irregular shape. In this case, many approaches such as those in [7], [9], and [10] either use extensive post-processing or an additional refinement network. In contrast, our work does not use any supplementary processing or extra refinement network.

To overcome the limitations of vanilla convolution and properly handle irregularly shaped objects, improved convolution schemes, such as partial convolution [17] and gated convolution [18] were developed. In PConv [17], convolution is masked and re-normalized to valid pixels only. After each operation, the mask is updated to compute a new region of valid pixels. On the other hand, GConv [18] generalizes partial convolution using gated convolution to learn the updated mask automatically. These methods perform better than those using vanilla convolution and overcome the problem of visual artifacts like color discrepancies and obvious edge responses surrounding the removal region. However, these methods focus more on valid regions and do not effectively consider affected regions, hence they generate unsatisfactory results when the foreground object occludes two different segments of the face. For example, most of the time, medical masks, hands, and microphones cover both the chin and part of the neck. This problem becomes more severe when the object, e.g., hands or sometimes mask, and the face segments have a similar texture.

To address the limitations of the aforementioned methods, we propose a novel GAN-based model to effectively remove a user-selected non-face object and generate sharp content for the removed region in facial images. Our model consists of two stages: 1) an object detection module, and 2) an image completion module. In the object detection module, we automatically detect the user-labeled object and generate a binary segmentation map of the object. To allow the user to select the target object, we employ an object label encoder. We detect

only one object at a time, taking into consideration accurate and computationally efficient detection. In the image completion module, we remove the object and fill the resulting region with plausible content using the input image and the output of the first stage, i. e., the object binary segmentation map. To achieve this, we use a GAN-based network with two discriminators. We took the approach of gradually adding both discriminators to the network to achieve a coarse-to-fine effect using a single network instead of using an additional refinement network [9], [10]. In addition, we integrate both vanilla and partial convolution operations into a single network to effectively remove the complex unwanted object and produce well-incorporated and sharp content at the affected region. To train our system, we created a paired synthetic face-occluded dataset by editing images from publicly available [19] and CelebA-HQ [20] datasets.

To summarize, the main contributions of this work are:

- We propose an effective method for face de-occlusion in facial images where the user has control of which object to remove. Our method is capable of removing multiple objects one-by-one through repeated application of our network.
- By employing a combined operation of vanilla and partial convolutions in a single network, we generate well-incorporated and visual-artifact-free content.
- To overcome the data scarcity problem, we created a large synthetic face-occluded paired dataset using publicly available CelebA and CelebA-HQ datasets.
- Experimental results demonstrate that, although trained on a synthetic face-occluded dataset, our method effectively removes non-face objects and produces structurally and perceptually plausible facial content in challenging real images.

II. RELATED WORK

In this section we will review relevant work in more detail in the context of object removal and object detection in an image.

A. OBJECT REMOVAL

Object removal is the task of removing an object from an image and reconstructing the region affected by removal of the object. Image editing/inpainting is a typical method used to accomplish this task. Conventional methods [4], [6], [21] remove objects in an image and then inpaint the damaged part by propagating similar pixels from the neighboring regions or other source images using an iterative search approach. Patchmatch [22] achieves better results and has better computation speed than [4], [6], [21]. However, it fails to generate consistent and semantically aware content for large arbitrarily shaped damaged regions, especially in facial images.

In recent years, neural network has garnered tremendous success in a variety of application domains such as sub-space clustering [23], deep clustering [24], face recognition [1], [2], image segmentation [25], [26], image manipulation [7], [8], [27] and so on. We only review some state-of-the-art deep learning based representative work related to image manipulation. Over the past few years, deep learning based GAN networks [28] have shown significant improvements in image manipulation applications [7]–[10], [12], [27]. Bau *et al.* [27] dissects their GAN network used to identify well-localized neurons which help control the manipulation of specific objects across the image. They achieve good results for adding or removing objects by controlling those identified neurons in their GAN network. However, they fill the removed region randomly and usually copy content from a remaining portion of the image. This can help in scene-level images but cannot help in removing objects and generating complex semantics in facial images.

On the other hand, GAN based image inpainting approaches [7]–[10], [12], [13] have shown potential for application in removing objects and reconstructing complex damaged regions in an image. Iizuka *et al.* (GL) [7] use a GAN setup with two discriminators to remove an object and recover the damaged region with fine details and global coherency. GICA [10] and MRGAN [9] take a coarse-to-fine approach. For this, they use a two-stage network, producing coarse content for the damaged region in the first stage and then refining the coarse output in the second stage. Although GL, GICA, and MRGAN produce plausible results, they are heavily dependent on post-processing like Poisson image blending (GL) or an extra refinement network (GICA and MRGAN) to overcome the problem of visual artifacts. EdgeConnect [12] produces better results without using any post-processing or extra refinement network by using edge map information from the image along with a binary segmentation map of the object. However, their system is unable to generate a reasonable edge map for regions occluded by large objects.

FaceD [13] utilizes 3D morphable model information and a GAN-based network with two discriminators which are gradually added to the network. We also use two discriminators along with the generator by gradually adding them to the network in the image completion module. In contrast to FaceD, our local discriminator looks at the generated region instead of the whole face region. Our network enhances the content in the affected region explicitly by focusing on the generated region while, in FaceD, the effect of the local discriminator is similar to that of a global discriminator. Moreover, FaceD uses 3DMM as guidance information which can be helpful in reconstructing the facial geometry but does not provide explicit information about the occluded object. These limitations result in FaceD generating severe visual artifacts when removing large and complex nature objects, especially hands and masks.

Din *et al.* [16] recently proposed a two-stage network to automatically detect and remove mask objects from facial images. While impressive results were produced in removing medical masks, their network is incapable of automatically detecting and removing multiple types of complex objects.

B. OBJECT DETECTION

The task of locating different objects with respect to the background of an image is called object detection. OverFeat [29] is one of the first deep-learning-based object detection algorithms. This is a convolution neural network (CNN) based algorithm to simultaneously classify, locate, and detect objects in an image. This model has been replaced by the Regional [30]–[33] and the YOLO family [34]–[36]of models for real-time detection.

All of these methods produce impressive results for various object types, however, they require a large number of training samples and computational power. Because we are concerned with the small number of object types that occur in facial images, we employ a simple GAN-based segmentation network with more user control focusing on the desired non-face object, taking both accurate segmentation and computational efficiency into consideration.

In the fully convolutional neural network (FCN) [25] a fully CNN-based segmentation network is proposed that consists of convolution, pooling, and up-sampling layers only. A skip connection architecture is introduced to overcome the problem of resolution reduction. Ronneberger *et al.* [26] proposed a U-Net architecture built upon FCN which yields better segmentation results with less training data. The main modifications are (1) U-Net has a symmetric shape, and (2) the skip connections between the encoder (contracting path) and decoder (expansive path) apply a concatenation operator instead of a sum. Skip connections help provide local information to the global information. A symmetric shape allows a network to have a large number of feature maps in the expansive path, facilitating the transfer of more information. Due to its simple architecture and better performance, other work has also exploited the U-Net architecture with minor modifications for image inpainting [17], multiple sketch styles generation [37], image unmosaicing [11], [38], object removal [9] and object detection in facial images [16]. Din *et al.* [16] employed a simple U-Net architecture to efficiently detect medical masks in facial images. Some models [9], [11], [37], [38] improved performance by using a discriminator along with the U-Net architecture. We also use the U-Net architecture along with a discriminator and target object label encoder to detect multiple types of non-face objects in facial images. This helps our model detecting the non-face object with reasonably efficient computational power as compared to previous state-of-the-art methods.

III. OUR APPROACH

Given a facial image partially covered by a non-face object, the intent of this work is to automatically remove the object and complete the image, providing a visually plausible appearance. The overall structure of our framework is illustrated in Figure 2. It consists of two main modules, the object detection module and the image completion module.

A. OBJECT DETECTION MODULE

The goal of the object detection module is to generate a binary segmentation map of the non-face object in the input image. Given the input occluded image, I_{input}, we aim to detect and generate a binary segmentation map, I_{seg}, for the occluded object. I_{seg}, is a binary segmentation map with 1 indicating the non-face object and 0 indicating the remaining image pixels.

1) ARCHITECTURE

The object detection module consists of a generator and a discriminator network. The generator G_1 is composed of a U-Net-like [26] architecture. Unlike U-Net, it has one additional encoder, known as the object label encoder, as shown in Figure 2. The input encoder consists of five blocks of convolution layers. Here, each block is a convolution layer followed by a *LRelu* activation function and an *instant_norm* layer, except for the first layer of the encoder. The object label encoder is a much shallower network compared to the input encoder. Both encoders receive different inputs: the input encoder takes an input image, I_{input}, while the object encoder receives a target object label, l_{label}. Motivated by [37], [39], we have represented target object label as one one-hot vector. We use separate encoders for the input image and target object label. If we use one encoder (input encoder) for both the input image and target object label, it can encode the input image information well, but ignores the object label information (which is usually due to its deep network architecture). Hence, the input image and object label are first encoded through their respective encoders. The output of both encoders are concatenated and fed into the decoder. The decoder architecture is a mirror copy of the input encoder except that convolution is replaced by a deconvolution layer. The last layer of the decoder uses a *tanh* activation function without a normalization layer.

We also use discriminator, D_{label}, along with generator G_1, to produce plausible results. The discriminator architecture is the same as the architecture used for discriminator in [40]. This penalizes dissimilar structures at the patch scale of 70×70. Unlike the discriminator used in [40], our discriminator is not a binary classifier (real or fake) but classifies the considered target object label. It has an output of a single $t + 1$ dimensional vector logits. Logits from 1 to t are given the label of "target object", and $t + 1$ is the logit for fake object, I_{label_fake}.

2) LOSS FUNCTION

The loss function we use to train the object detection module is a combination of L_{l_1} loss and the GAN objective function. L_{l_1} loss measures the pixel distance between the predicted binary segmentation map, I_{pre_seg} and the corresponding target map, I_{gt_seg}, while the GAN objective function for both D_{label} and G_1 are as follows.

$$\begin{aligned}\mathcal{L}_{D_{label}} &= \mathbb{L}_{adv}(D_{label}((I_{pre\_seg}, I_{gt\_seg}), I_{label}) \\ &\quad + \mathbb{L}_{adv}(D_{label}((I_{input}, I_{pre\_seg}), I_{label\_fake}) \quad (1)\\ \mathcal{L}_{G_1} &= \mathbb{L}_{adv}(D_{label}((I_{input}, I_{pre\_seg}), I_{label}) \quad (2)\end{aligned}$$

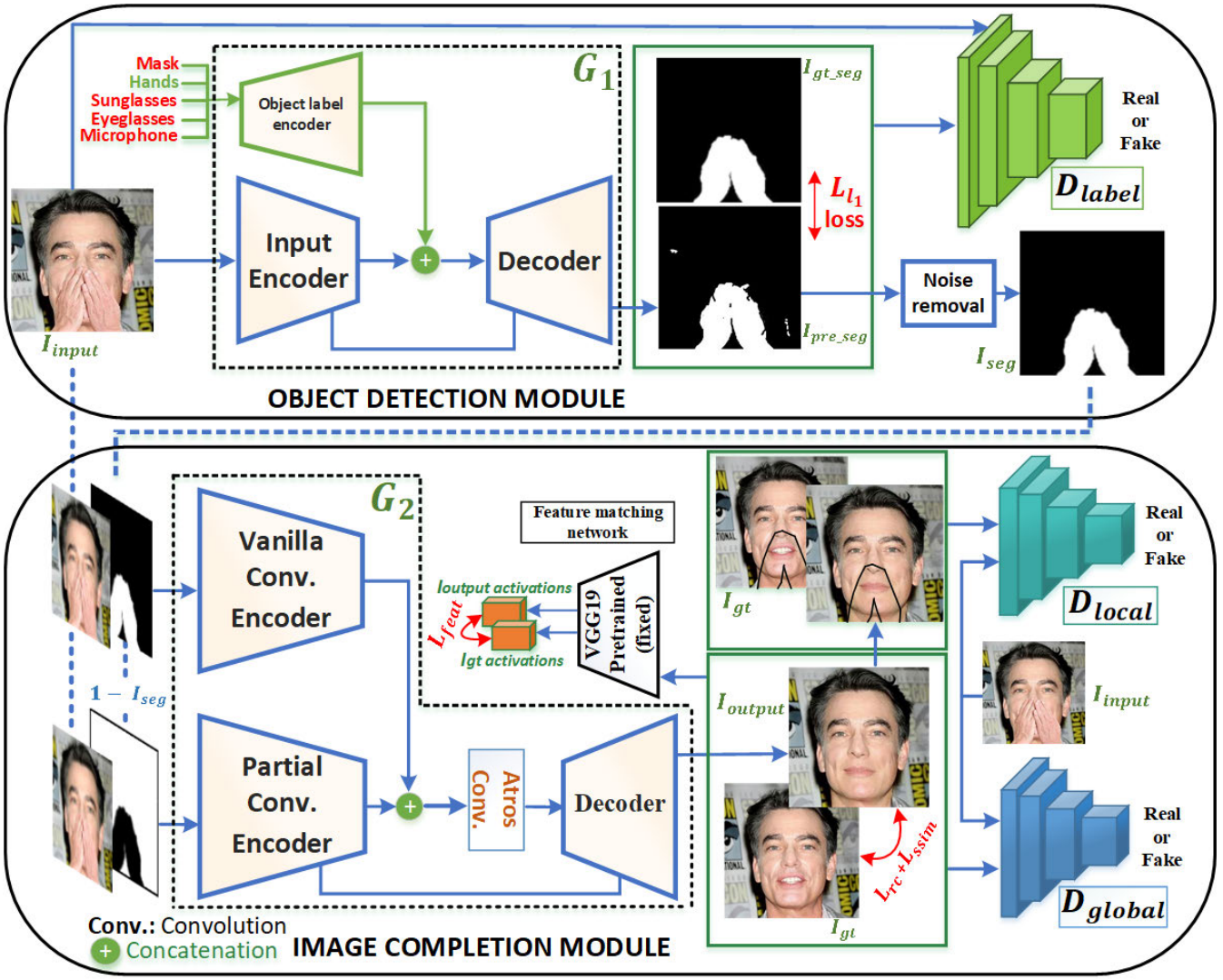

FIGURE 2. The overall architecture of our framework.

Our discriminator, D_{label}, is not only a real or fake classifier but also classifies the target objects considered.

To obtain a clean mask, I_{seg}, we use a noise removal module as shown in Figure 2. The noise removal module uses the simple morphological image processing operations of erosion and dilation. We face the problem of noise in the form of holes left in the segmentation map rather than isolated pixels. To fill the holes, we first dilate the binary segmentation map and then erode it with a disc of size 3 pixels for both operations.

B. IMAGE COMPLETION MODULE

The goal of this module is to remove the foreground non-face object and fill the region left behind with plausible content. The main blocks of this module are the completion generator, discriminators, and perceptual network. The generator, G_2, takes the input image, I_{input} (occluded image), along with the object segmentation map, I_{seg}, as a combined input and generates an occlusion-free image, I_{output}. The two discriminators D_{global} and D_{local}, force generator, G_2, to produce visually plausible and naturalistic looking images by determining the I_{output}, as real or fake face. Additionally, the perceptual network helps in content preservation of the generated image, I_{output}.

1) COMPLETION GENERATOR

The completion generator, G_2, has the same architecture as the segmentation map generator G_1. Unlike in G_1, two parallel encoders are used (the vanilla convolution. encoder and partial convolution encoder), as shown in Figure 2. The vanilla convolution encoder is the same as the input encoder used in G_1, while the partial convolution encoder is the one used in PConv [17]. The vanilla convolution encoder takes in the input image and the binary segmentation map. It uses a standard convolution network, which applies the same filter to the whole image regardless of valid and affected regions, and produces a feature map after each convolution operation.

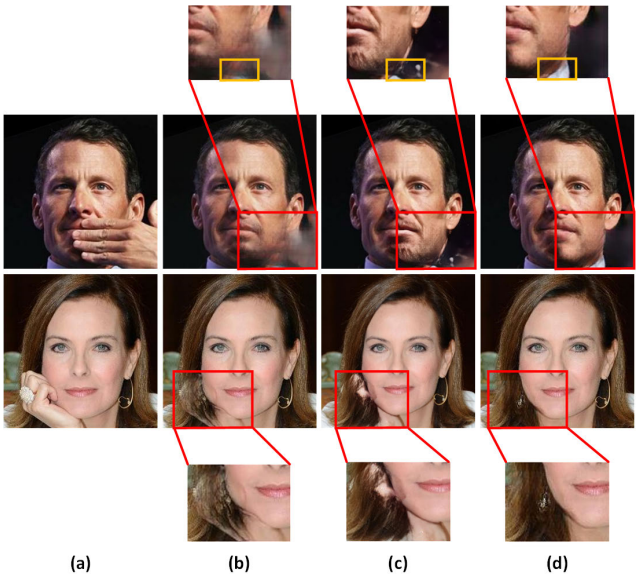

FIGURE 3. Effect of each encoder (vanilla conv. and partial conv.) and of using both in parallel in the image completion module. a) Input image, b) results of our model using only the vanilla conv. encoder, c) results using only partial conv. encoder, and d) results using both encoders in parallel.

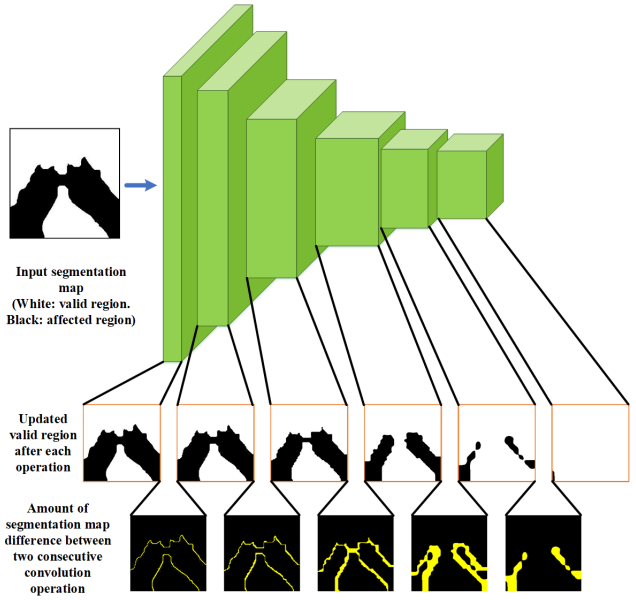

FIGURE 4. Visualization of segmentation map update in partial conv. encoder.

Using only the vanilla convolution encoder helps to achieve well-incorporated predictions but also leads to visual artifacts such as blurriness, especially at the boundaries of valid and affected regions as can be seen in Figure 3 (b).

The partial convolution encoder takes in both the input image and the inverse segmentation map (uncovered regions are treated as valid pixels for rule-based mask updating). A partial convolution layers network and a segmentation map update function are used jointly. The partial convolution applies the convolution operation only to valid regions to produce a feature map, while the map update function produces a slightly extended valid region after each convolution operation. Therefore, as we apply the partial convolution operation, the segmentation map gets thinner and thinner, or in other words, the valid region of the image increases as can be seen in Figure 4. This makes the partial convolution encoder very effective in handling irregular affected regions and in producing sharp content. However, partial convolution focuses more on operating in valid regions and does not effectively handle affected regions, resulting in artifacts and the missing of minor details (highlighted with yellow box), especially at the boundaries of valid and affected regions of the image as can be seen in Figure 3 (c). Hence, using both encoders (vanilla and partial) in parallel not only successfully removes the object, but also generates well-incorporated and sharp content with fine details compared to using either encoder alone (see Figure 3 (d)).

Additionally, we use a squeeze and excitation (SE) block [41] at the output of the first three blocks of the vanilla convolution encoder only. The SE blocks help improve performance by learning the weights for each channel of the feature vector. The encoded information from both encoders is concatenated and fed into an atrous convolution block. The atrous convolution block consists of four layers of atrous convolutions (rate: 2, 4, 8, 16) [42]. This helps capture the large fields of view, making the generated missing portion coherent with rest of the face. The decoder takes the output of the atrous convolution block as input, then up-samples to predict an output image, I_{output}, without a non-face object.

$$I_{output} = G_2(I_{input}, I_{seg}). \tag{3}$$

The decoder architecture is the same as that used in the object detection module. However, instead of using a skip connection between the vanilla convolution encoder and decoder, we use skip connections between the partial convolution encoder and decoder. This provides significant/updated information to the decoder because after each convolution operation, the segmentation map is updated in the partial convolution encoder, while the segmentation map in the vanilla convolution encoder remains the same throughout the training iterations. Figure 4 provides a visualization of the segmentation map (valid region) update in the partial convolution encoder.

2) DISCRIMINATORS

We use two discriminators, D_{global} and D_{local}, as shown in Figure 2. The architecture of both discriminators is the same as the discriminator in pix2pix [40]. Discriminator D_{global}, penalizes dissimilar structures at the patch scale of 70×70, while D_{local} penalizes at the smaller patch scale of 40×40. The role of both discriminators is to force the completion generator to produce visually plausible and semantically consistent images. To obtain a coarse-to-fine effect, we add both discriminators gradually to the network [13], [16].

Discriminator D_{global} looks at the entire generated image while D_{local} focuses more on the generated missing region.

3) FEATURE MATCHING NETWORK

The feature matching network is a pre-trained VGG-19 fixed network [43]. The purpose of this network is to encourage the generator output, I_{output}, to have feature representation similar to the ground truth, I_{gt}.

4) LOSS FUNCTION

To force the completion generator to produce realistic and perceptually correct missing content, we use similarity loss which is an amalgam of L_{l_1} loss, structural similarity loss *SSIM* [44] and feature matching loss $\mathcal{L}_{feat}$. This is expressed as:

$$\mathcal{L}_{sim} = \mathcal{L}_{l_1} + \mathcal{L}_{ssim} + \mathcal{L}_{feat}. \tag{4}$$

where, $\mathcal{L}_{l_1}$, is the pixel difference between the generated image, I_{output} and the ground truth, I_{gt}, while $\mathcal{L}_{SSIM}$ measures the structural similarity at the patch level of 11×11 between I_{output} and I_{gt} as following:

$$\mathcal{L}_{ssim} = 1 - SSIM(I_{output}, I_{gt}). \tag{5}$$

On the other hand, feature matching loss $\mathcal{L}_{feat}$ [45] penalize any perceptually unreasonable generated output I_{output} by computing the distance from the intermediate layers activation maps of I_{output} and I_{gt}, from a pre-trained network (VGG-19 [43]). Let f_i be the activation map of the i^{th} layer of the VGG-19 network, then the feature matching loss is defined as:

$$\mathcal{L}_{feat} = \sum_i ||f_i(I_{output}) - f_i(I_{gt})|| \tag{6}$$

We exploit the intermediate convolution layer feature maps (*conv_3*, *conv_4* and *conv_5*) of the VGG-19 network to obtain rich structural information, which helps in recovering a plausible structure for the face semantics.

Generator, G_2, learns to produce real-looking synthesized content by incorporating feedback from the two discriminators (D_{global}, D_{local}). To train the model in a GAN setup, the generator tries to minimize the following function while the discriminators try to maximize it:

$$\begin{aligned}\mathcal{L}_{adv}^{global} &= \min_{G_2} \max_{D_{global}} \mathbb{E}[\log(D_{global}(I_{output}, I_{gt})) \\ &\quad + \log(1 - D_{global}(G_2(I_{input}, I_{seg})))] \end{aligned} \tag{7}$$

$$\begin{aligned}\mathcal{L}_{adv}^{local} &= \min_{G_2} \max_{D_{local}} \mathbb{E}[\log(D_{local}(I_{seg} \otimes I_{output}, I_{seg} \otimes I_{gt})) \\ &\quad + \log(1 - D_{local}(I_{seg} \otimes G_2(I_{input}, I_{seg})))] \end{aligned} \tag{8}$$

Here, $\otimes$ denotes element-wise multiplication.

The joint loss function used to train the image completion module is defined as:

$$\mathcal{L}_{joint} = \lambda_1 \mathcal{L}_{sim} + \lambda_2 \mathcal{L}_{adv}^{global} + \lambda_3 \mathcal{L}_{adv}^{local} \tag{9}$$

We have set the weight parameters as $\lambda_1 = 100$, $\lambda_2 = 0.3$ and $\lambda_3 = 0.7$.

IV. EXPERIMENTS

This section describes the synthetic face-occluded dataset creation and training details of our model.

A. SYNTHETIC FACE OCCLUDED DATASET

There is no publicly available dataset that contains facial image pairs with and without occlusion objects. We constructed a synthetic face-occluded dataset using the publicly available CelebFaces Attributes Dataset (CelebA) [19] and CelebA-HQ dataset [20]. Both CelebA and CelebA-HQ are large-scale face attribute datasets with more than 200k and 30k celebrity images, respectively. Each face image in the aforementioned dataset is cropped and roughly aligned by eye position. We synthesized occlusions caused by the five most common non-face objects, hands, a mask, sunglasses, eyeglasses, and a microphone. We used more than 40 different types of each object which varied in size, shape, color, and structure. We randomly placed non-face objects on the faces. Then we generated a corresponding binary segmentation map, I_{gt_seg}, of the objects using Adobe Photoshop CC 2018. Some examples of the occlusion objects used are shown in Figure 6. Figure 5 shows examples from our face-occluded synthetic dataset. We created a total of 60k training samples and all samples were resized to 272×272. Our test data consisted of 4,000 real images collected from the CelebA dataset [19], CelebA-HQ dataset [20] and the Internet.

B. IMPLEMENTATION AND TRAINING DETAILS

To train the map module, we feed input image, I_{input}, into the network and generate a binary segmentation map, I_{seg}, that is close to the target binary map I_{gt_seg}. Generated binary map, I_{seg}, and its inverse (1 - I_{seg}), along with input image, I_{input}, are fed into the completion network and generate the final output, I_{output}. Instead of training the whole network of the image completion module at once, we train it into two steps. This allows us to obtain a coarse-to-fine-network effect using a single network instead of using two networks separately as is done in most previous work [9], [10], [18]. In first step we train the completion generator G_2, and global discriminator D_{global} for the first 150 epochs, focusing only on obtaining the global structure of the face and coarse results in the affected region. In the second step, we add the local discriminator, D_{local}, to the network and train the model for another 150 epochs. The local discriminator, D_{local}, looks at the affected region only, and hence focuses on refining the coarse results in the removed region.

We have implemented our model in tensorflow [46]. We train the two modules alternatively instead of in an end-to-end manner because: 1) Each module achieves optimal results at different amount of training steps, and 2) we train the image completion module in two steps to pursue a coarse-to-fine strategy, which does not suit well an end-to-end training scheme. We used 60k training samples sized at 272×272 from our synthetic face-occluded dataset. To train our model, we randomly cropped these samples to 256×256. We used

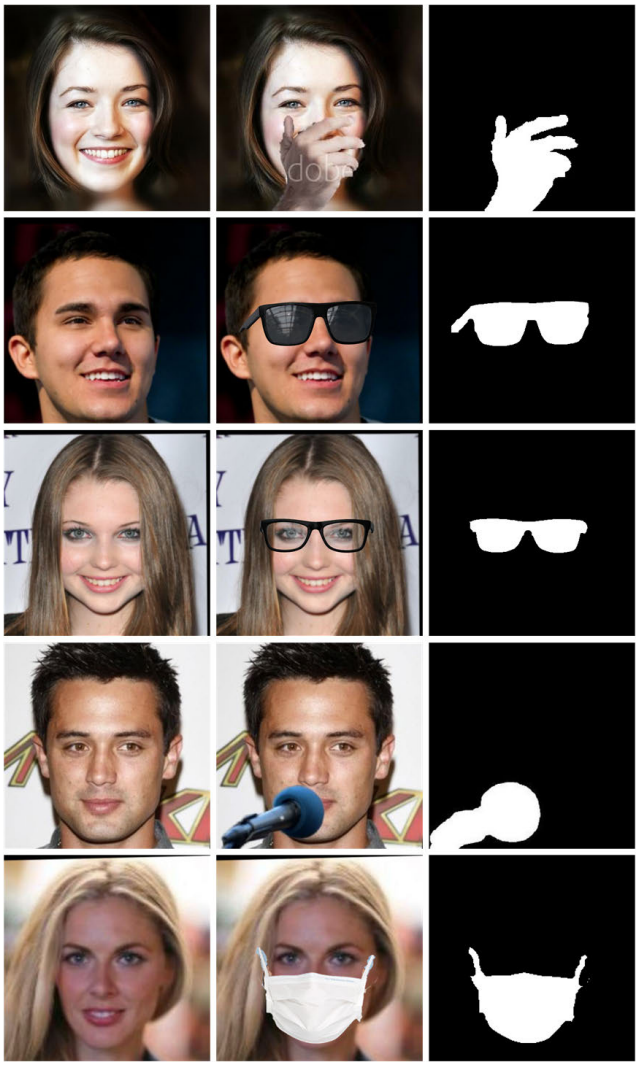

FIGURE 5. Example images from our synthetic face occluded dataset.

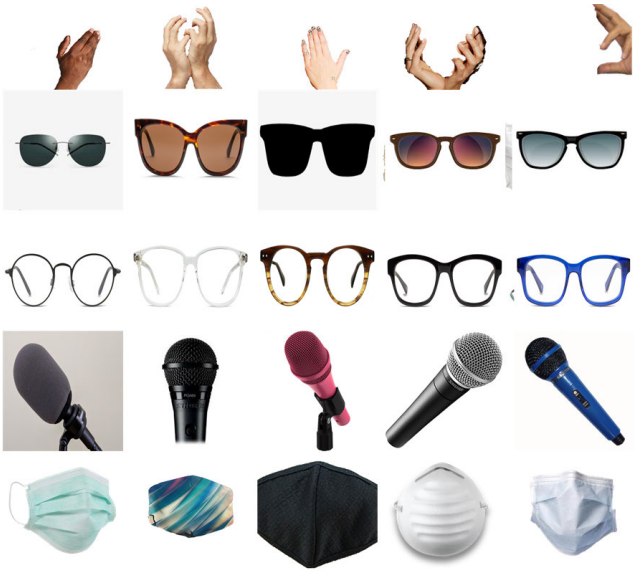

FIGURE 6. Example images of occlusion objects used in our synthetic face-occluded dataset.

a batch size of 10 and trained the model for 300 epochs using a single NVIDIA GeForce 2080Ti GPU.

V. RESULTS AND COMPARISON

Using real world images, we compare our model both qualitatively and quantitatively to five different methods: (1) GICA [10], (2) EdgeConnect [12], (3) PConv [17], (4) GConv [18], and (5) FaceD [13]. Our method is denoted as ''Ours''. For fair comparison, we have retrained all the other models, except for FaceD [13] on our synthetic face occluded dataset. We have also provided an accurate object binary segmentation map of the object along with the input image at both the training and inference stages because the tested methods assume that an exact object binary segmentation map is given. We have used the object binary segmentation map for our system training and testing generated by our object detection module. FaceD does not provide the open-source code of their work and dataset (assuming 3DMM synthesis of the face). Hence for fair comparison, we have used test images obtained from the relevant paper [13] and made a comparison with their results.

Qualitative Comparison: Figure 7 shows a comparison using real world samples produced by our model (''Ours'') and the other state-of-the-art approaches: GICA [10], EdgeConnect [12], PConv [17] and GConv [18]. Methods using vanilla convolution as the backbone operation of their networks (GICA and EdgeConnect) show visual discrepancies such as blurriness and failure to generate complex face semantics as can be seen in Figure 7 (b) and (c). PConv and GConv use partial convolution and gated convolution, respectively, and produce sharp results compared to methods using vanilla convolution, but still show apparent artifacts, especially at segments of the face overlapped by the occluding object as can be seen in Figure 7 (d) and (e). On the other hand, our method based on both vanilla and partial convolution obtains results that are visually pleasing and removed complex objects like hands and masks with seamless boundary transitions.

Figure 8 shows a comparison between our model and FaceD [13] for challenging face-occluded cases with complex backgrounds, poses, and illuminations. Exploiting a 3D approach helps FaceD to reconstruct the 3D face geometry and to remove small objects plausibly (e. g., microphones and glasses) as shown in the last four examples of Figure 8 (b). However, it suffers in synthesizing large and complex missing regions of the face due to the lack of explicit information about the occluded object as compared to our model.

All state-of-the-art methods we compare with our model use a GAN setup, except for PConv which instead uses an encoder-decoder architecture scheme without any discriminator. For fair comparison and to validate that our

FIGURE 7. Qualitative results comparison of our model with other state-of-the art image editing models on real world test images. a) Input image, b) GCA [10], c) EdgeConnect [12], d) PConv [17], e) GConv [18], and f) Ours. Note: There is no ground truth since all samples are real world images collected from the Internet.

encoder-decoder architecture scheme works better than PConv, we have trained our model as an auto-encoder by dropping both discriminators while keeping the other settings the same as in our full model. Figure 9 (c) shows that our model (auto-encoder setup) performs better than PConv full model (see Figure 9 (b)). This is because integrating vanilla

FIGURE 8. Qualitative comparison of our model with FaceD [13]: a) test images from FaceD [13], b) result copied from main paper of FaceD work [13], c) Ours.

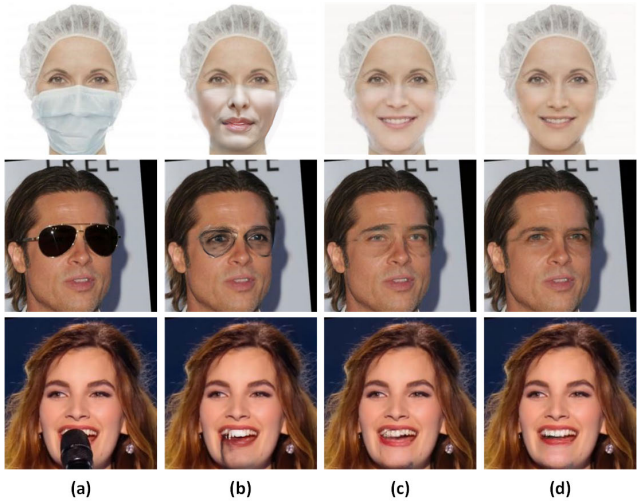

FIGURE 9. Results comparison of our model (trained with and without discriminators) with PConv [17]: (a) input, (b) results of PConv full model, (c) results of our model without discriminators, and (d) results of our full model.

and partial convolution encoders helped our model learning well-incorporated structure and also sharp contents at the affected region. Moreover, adding discriminators (adversarial training) to our auto-encoder setup results in producing more naturalistic outputs than both the PConv full model and our model without an adversarial training scheme as can be seen Figure 9 (d).

Figure 10 shows the performance comparison on image completion for test samples having more than one type of occluding object on the face, where we have replaced the second stage (image completion module) by the state-of-the-art inpainting method GCA [10], EdgeConnect [12], PConv [17], GConv [18]. We can see that all previous state-of-the-art methods fail to produce plausible results for complex scenarios.

Quantitative Comparison: Table 1 shows a quantitative comparison of our model with state-of-the art methods. As mentioned in [8] and [38], there is no good quantitative metric available to evaluate image editing results because the goal of these methods is to not to generate exact content, but to produce realistic-looking content. Nonetheless, we measure the quantitative performance using four popular metrics: 1) Structural SIMilarity (SSIM) [44], 2) Peak Signal to Noise Ratio (PSNR), 3) Naturalness Image Quality Evaluator (NIQE) [47], and 4) Blind/Referenceless Image Spatial Quality Evaluator (BRISQUE) [48]. SSIM and PSNR are calculated using the results for the synthetic test dataset since we do not have ground truth for real images, while NIQE and BRISQUE are evaluated using the results from the real test samples. It can be seen in Table 1 that our method outperforms all state-of-the-art methods. Our model performance for the eye/sunglasses and microphone are slightly higher than for mask and hands occlusions because masks and hands occlude large areas of the face which are also complex and difficult to reconstruct, such as the boundaries of the chin and neck.

Additional Results and Limitations: We have further tested our network on occlusions that do not exist in our synthetic face-occluded training dataset (mobile phone, card, scarf, and apple, etc.). Figure 11 shows these results which were obtained by providing segmentation maps of the objects manually. The results show that our model produces reasonable results for those that do not exist in our training dataset. The results confirm that the proposed model can remove different types of complex and challenging face occlusions. However, to warrant a robust performance regardless of occluding object category it is necessary to develop a novel approach that can work in an object-agnostic manner.

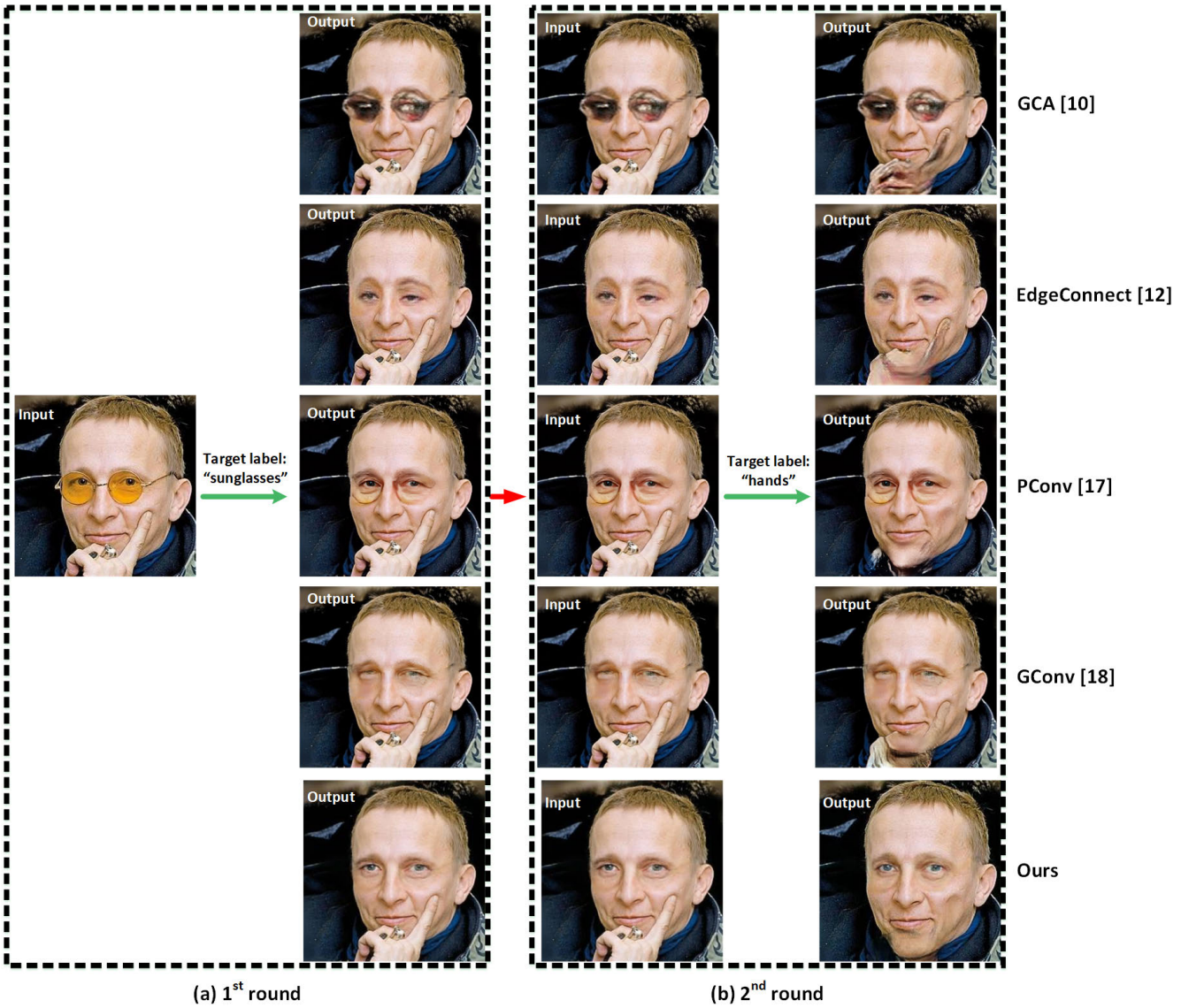

FIGURE 10. Qualitative results comparison of our model with other state-of-the art image editing models for test samples having more than one type of occluding object on the face: (a) first round output of our model and other state-of-the art image editing models given the corresponding target object label, (b) second round output of our model and other state-of-the art image editing models given the corresponding target object label.

Failure cases occur when more than 70% of the face is occluded, especially in the case of occluding hands. In these instances, object detection and image completion modules fail to differentiate between the face and hands due to the large portion of the face being occluded and the similar texture of the hands and face.

A. ABLATION STUDY

We have analyzed the effects of different object detection module settings. Figure 12 (b) shows the results from the object detection module when we fed both the input image and the target object label through the input encoder, while 12 (c) shows the results for using separate encoders for both the input image and the target object label. We can see that using a single encoder for both inputs does not produce reasonable results compared to using separate encoders for both inputs. For example, in the first row we want to generate the segmentation map of the microphone but due to its small size, the microphone object is ignored and the segmentation map for glasses is produced. Also, in the second row of Figure 12, the network generated a segmentation map of a microphone along with a segmentation map of the glasses, although there is no microphone object in the input. However, these issues are not seen in the output of the setting of our method.

These problems occur because the input encoder usually ignores the target object label information, which is a one-hot vector, due to its deep network architecture. On the other

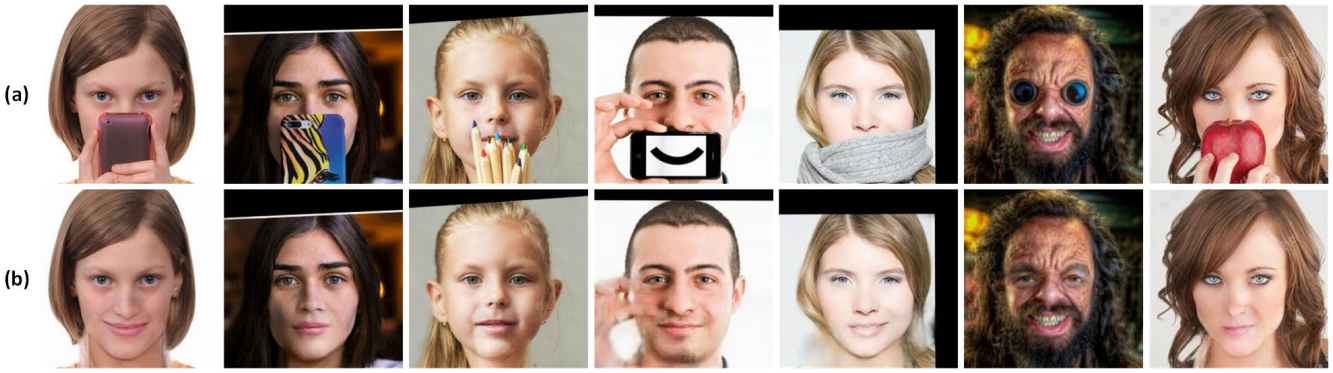

FIGURE 11. Our model results for occlusions that do not exist in our training dataset: (a) input, (b) output. Note: We provided the manual segmentation map of the object for all test samples.

TABLE 1. Quantitative comparison of our methods to other state-of-the-art representative methods. The best result are boldfaced.

Object	Methods	SSIM	PSNR	NIQE	BRISQUE
Mask	GICA [10]	0.827	22.40	4.842	41.030
	Edge [12]	0.867	20.873	4.755	41.895
	PConv [17]	0.869	24.452	4.830	44.976
	GConv [18]	0.850	22.357	4.573	**39.676**
	Ours	**0.908**	**28.727**	**4.425**	40.883
Hands	GICA [10]	0.761	20.866	4.940	28.597
	Edge [12]	0.818	24.911	4.597	31.913
	PConv [17]	0.863	25.122	4.691	24.603
	GConv [18]	**0.885**	26.920	4.929	24.879
	Ours	0.882	**26.948**	**4.443**	**24.206**
Eye/Sunglasses	GICA [10]	0.863	24.675	4.752	37.322
	Edge [12]	0.882	25.641	4.763	**36.374**
	PConv [17]	0.896	26.678	4.509	39.680
	GConv [18]	0.889	26.289	4.598	38.358
	Ours	**0.914**	**28.878**	**4.458**	38.111
Microphone	GICA [10]	0.913	25.34	4.482	35.174
	Edge [12]	0.917	27.919	4.186	34.213
	PConv [17]	0.940	29.455	4.331	35.514
	GConv [18]	0.940	29.455	4.853	35.396
	Ours	**0.944**	**31.323**	**4.105**	**34.038**

FIGURE 12. Comparison results of image object detection module under different settings. a) Input image. b) Results of object detection module using only input encoder taking in both input image and target object label. c) Results of object detection module using separate encoder for both input image and target object label. Note: All results were obtained without using a noise removal module for both cases.

hand, a separate encoder helps both networks to focus more explicitly on each input. The comparison indicates that using separate encoders greatly improves the performance of the object detection module and results in promising guidance information for the image completion module.

VI. CONCLUSION

In this paper, we propose a user-friendly automatic face de-occlusion method. We show the effectiveness of our method on five commonly occurring occluding objects which can be extended to more types of objects. Our system removes one distracting object at a time, however, it is capable of removing multiple distracting objects through repeated application very smoothly. Our model first detects the occlusion object and generates a segmentation map of the object, then uses the segmentation map as guidance information to remove the object and fill in the empty region. We have shown that integration of vanilla and partial convolution operations significantly improves performance in challenging scenarios involving the generation of content for two different segments occluded by the object. In conclusion, our model outperforms the previous state-of-the-art approaches in the task of object-removal from facial images both in terms of qualitative and quantitative results.

REFERENCES

[1] J.-S. Park, Y. Hwa Oh, S. Chul Ahn, and S.-W. Lee, "Glasses removal from facial image using recursive error compensation," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 27, no. 5, pp. 805–811, May 2005.

[2] J. Wright, A. Y. Yang, A. Ganesh, S. Shankar Sastry, and Y. Ma, "Robust face recognition via sparse representation," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 31, no. 2, pp. 210–227, Feb. 2009.

[3] A. Shrivastava, T. Pfister, O. Tuzel, J. Susskind, W. Wang, and R. Webb, "Learning from simulated and unsupervised images through adversarial training," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jul. 2017, pp. 2107–2116.

[4] A. Criminisi, P. Perez, and K. Toyama, "Region filling and object removal by exemplar-based image inpainting," *IEEE Trans. Image Process.*, vol. 13, no. 9, pp. 1200–1212, Sep. 2004.

[5] S. Darabi, E. Shechtman, C. Barnes, D. B. Goldman, and P. Sen, "Image melding: Combining inconsistent images using patch-based synthesis," *ACM Trans. Graph.*, vol. 31, no. 4, pp. 1–82, 2012.

[6] J. Hays and A. A. Efros, "Scene completion using millions of photographs," *ACM Trans. Graph.*, vol. 26, no. 3, p. 4, Jul. 2007.

[7] S. Iizuka, E. Simo-Serra, and H. Ishikawa, "Globally and locally consistent image completion," *ACM Trans. Graph.*, vol. 36, no. 4, pp. 1–14, Jul. 2017.

[8] C. Yang, X. Lu, Z. Lin, E. Shechtman, O. Wang, and H. Li, "High-resolution image inpainting using multi-scale neural patch synthesis," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jul. 2017, pp. 6721–6729.

[9] M. K. J. Khan, N. Ud Din, S. Bae, and J. Yi, "Interactive removal of microphone object in facial images," *Electronics*, vol. 8, no. 10, p. 1115, Oct. 2019.
[10] J. Yu, Z. Lin, J. Yang, X. Shen, X. Lu, and T. S. Huang, "Generative image inpainting with contextual attention," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, Jun. 2018, pp. 5505–5514.
[11] K. Javed, N. Ud Din, S. Bae, and J. Yi, "Image unmosaicing without location information using stacked GAN," *IET Comput. Vis.*, vol. 13, no. 6, pp. 588–594, Sep. 2019.
[12] K. Nazeri, E. Ng, T. Joseph, F. Z. Qureshi, and M. Ebrahimi, "EdgeConnect: Generative image inpainting with adversarial edge learning," 2019, *arXiv:1901.00212*. [Online]. Available: http://arxiv.org/abs/1901.00212
[13] X. Yuan and I. K. Park, "Face de-occlusion using 3D morphable model and generative adversarial network," in *Proc. IEEE/CVF Int. Conf. Comput. Vis. (ICCV)*, Oct. 2019, pp. 10062–10071.
[14] Y. Song, C. Yang, Y. Shen, P. Wang, Q. Huang, and C.-C. J. Kuo, "Spg-net: Segmentation prediction and guidance network for image inpainting," in *Proc. BMVC*, 2018, pp. 1–14.
[15] D. Pathak, P. Krahenbuhl, J. Donahue, T. Darrell, and A. A. Efros, "Context encoders: Feature learning by inpainting," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2016, pp. 2536–2544.
[16] N. Ud Din, K. Javed, S. Bae, and J. Yi, "A novel GAN-based network for unmasking of masked face," *IEEE Access*, vol. 8, pp. 44276–44287, 2020.
[17] 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 *Proc. Eur. Conf. Comput. Vis. (ECCV)*, 2018, pp. 85–100.
[18] J. Yu, Z. Lin, J. Yang, X. Shen, X. Lu, and T. Huang, "Free-form image inpainting with gated convolution," in *Proc. IEEE/CVF Int. Conf. Comput. Vis. (ICCV)*, Oct. 2019, pp. 4471–4480.
[19] Z. Liu, P. Luo, X. Wang, and X. Tang, "Deep learning face attributes in the wild," in *Proc. IEEE Int. Conf. Comput. Vis. (ICCV)*, Dec. 2015, pp. 3730–3738.
[20] T. Karras, T. Aila, S. Laine, and J. Lehtinen, "Progressive growing of GANs for improved quality, stability, and variation," 2017, *arXiv:1710.10196*. [Online]. Available: http://arxiv.org/abs/1710.10196
[21] J. Wang, K. Lu, D. Pan, N. He, and B.-K. Bao, "Robust object removal with an exemplar-based image inpainting approach," *Neurocomputing*, vol. 123, pp. 150–155, Jan. 2014.
[22] C. Barnes, E. Shechtman, A. Finkelstein, and D. Goldman, "PatchMatch: A randomized correspondence algorithm for structural image editing," *ACM Trans. Graph.*, vol. 28, no. 3, p. 24, 2009.
[23] X. Peng, J. Feng, S. Xiao, W.-Y. Yau, J. T. Zhou, and S. Yang, "Structured AutoEncoders for subspace clustering," *IEEE Trans. Image Process.*, vol. 27, no. 10, pp. 5076–5086, Oct. 2018.
[24] X. Peng, H. Zhu, J. Feng, C. Shen, H. Zhang, and J. T. Zhou, "Deep clustering with sample-assignment invariance prior," *IEEE Trans. Neural Netw. Learn. Syst.*, early access, Dec. 31, 2019, doi: 10.1109/TNNLS.2019.2958324.
[25] J. Long, E. Shelhamer, and T. Darrell, "Fully convolutional networks for semantic segmentation," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2015, pp. 3431–3440.
[26] O. Ronneberger, P. Fischer, and T. Brox, "U-net: Convolutional networks for biomedical image segmentation," in *Proc. Int. Conf. Med. Image Comput. Comput. Assist. Intervent.*, Munich, Germany, 2015, pp. 234–241.
[27] D. Bau, J.-Y. Zhu, H. Strobelt, B. Zhou, J. B. Tenenbaum, W. T. Freeman, and A. Torralba, "GAN dissection: Visualizing and understanding generative adversarial networks," in *Proc. ICLR*, 2019, pp. 1–18.
[28] I. Goodfellow, J. Pouget-Abadie, M. Mirza, B. Xu, D. Warde-Farley, S. Ozair, A. Courville, and Y. Bengio, "Generative adversarial nets," in *Proc. Adv. Neural Inf. Process. Syst.*, 2014, pp. 2672–2680.
[29] P. Sermanet, D. Eigen, X. Zhang, M. Mathieu, R. Fergus, and Y. LeCun, "OverFeat: Integrated recognition, localization and detection using convolutional networks," 2013, *arXiv:1312.6229*. [Online]. Available: http://arxiv.org/abs/1312.6229
[30] R. Girshick, J. Donahue, T. Darrell, and J. Malik, "Rich feature hierarchies for accurate object detection and semantic segmentation," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2014, pp. 580–587.
[31] R. Girshick, "Fast R-CNN," in *Proc. IEEE Int. Conf. Comput. Vis. (ICCV)*, Dec. 2015, pp. 1440–1448.
[32] S. Ren, K. He, R. Girshick, and J. Sun, "Faster R-CNN: Towards real-time object detection with region proposal networks," in *Proc. Adv. Neural Inf. Process. Syst.*, 2015, pp. 91–99.
[33] K. He, G. Gkioxari, P. Dollár, and R. Girshick, "Mask R-CNN," in *Proc. IEEE Int. Conf. Comput. Vis.*, Oct. 2017, pp. 2961–2969.
[34] J. Redmon, S. Divvala, R. Girshick, and A. Farhadi, "You only look once: Unified, real-time object detection," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2016, pp. 779–788.
[35] J. Redmon and A. Farhadi, "YOLO9000: Better, faster, stronger," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jul. 2017, pp. 7263–7271.
[36] J. Redmon and A. Farhadi, "YOLOv3: An incremental improvement," 2018, *arXiv:1804.02767*. [Online]. Available: http://arxiv.org/abs/1804.02767
[37] S. Bae, N. Ud Din, K. Javed, and J. Yi, "Efficient generation of multiple sketch styles using a single network," *IEEE Access*, vol. 7, pp. 100666–100674, 2019.
[38] K. Javed, N. U. Din, S. Bae, R. S. Maharjan, D. Seo, and J. Yi, "UMGAN: Generative adversarial network for image unmosaicing using perceptual loss," in *Proc. 16th Int. Conf. Mach. Vis. Appl. (MVA)*, May 2019, pp. 1–5.
[39] Y. Choi, M. Choi, M. Kim, J.-W. Ha, S. Kim, and J. Choo, "StarGAN: Unified generative adversarial networks for multi-domain image-to-image translation," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, Jun. 2018, pp. 8789–8797.
[40] P. Isola, J.-Y. Zhu, T. Zhou, and A. A. Efros, "Image-to-image translation with conditional adversarial networks," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jul. 2017, pp. 5967–5976.
[41] J. Hu, L. Shen, and G. Sun, "Squeeze-and-excitation networks," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, Jun. 2018, pp. 7132–7141.
[42] L.-C. Chen, G. Papandreou, F. Schroff, and H. Adam, "Rethinking atrous convolution for semantic image segmentation," 2017, *arXiv:1706.05587*. [Online]. Available: http://arxiv.org/abs/1706.05587
[43] K. Simonyan and A. Zisserman, "Very deep convolutional networks for large-scale image recognition," in *Proc. ICLR*, 2015, pp. 1–14.
[44] Z. Wang, A. C. Bovik, H. R. Sheikh, and E. P. Simoncelli, "Image quality assessment: From error visibility to structural similarity," *IEEE Trans. Image Process.*, vol. 13, no. 4, pp. 600–612, Apr. 2004.
[45] J. Johnson, A. Alahi, and L. Fei-Fei, "Perceptual losses for real-time style transfer and super-resolution," in *Proc. Eur. Conf. Comput. Vis. (ECCV)*, Amsterdam, The Netherlands, Oct. 2016, pp. 694–711.
[46] M. Abadi *et al.*, "TensorFlow: A system for large-scale machine learning," in *Proc. OSDI*, vol. 16, 2016, pp. 265–283.
[47] A. Mittal, A. K. Moorthy, and A. C. Bovik, "No-reference image quality assessment in the spatial domain," *IEEE Trans. Image Process.*, vol. 21, no. 12, pp. 4695–4708, Dec. 2012.
[48] A. Mittal, A. K. Moorthy, and A. C. Bovik, "Blind/referenceless image spatial quality evaluator," in *Proc. Conf. Rec. 45th Asilomar Conf. Signals, Syst. Comput. (ASILOMAR)*, Nov. 2011, pp. 723–727.

NIZAM UD DIN received the B.Sc. degree in electrical (computer) engineering from the COMSATS Institute of Information Technology, Pakistan, in 2013, and the M.Sc. degree in computer engineering from the University of Engineering and Technology, Taxila, Pakistan, in 2016. He is currently pursuing the Ph.D. degree with the Computer Vision Laboratory, Department of Electrical and Computer Engineering, Sungkyunkwan University, South Korea.

He was a Visiting Faculty Member with Quaid-i-Azam University, Islamabad, Pakistan, from 2016 to 2017. He is mainly interested in deep learning, especially applied to computer vision. His awards and honors include the University Scholarship for his B.Sc. degree in electrical (computer) engineering from the COMSATS Institute of Information Technology, in 2009, and the Higher Education Commission HRDI-Faculty Development of UESTPS-UETS Scholarship for his Ph.D. degree from HEC, Pakistan, in 2017.

KAMRAN JAVED received the B.Sc. degree (Hons.) in electronic engineering and the M.Sc. degree in computer engineering from the University of Engineering and Technology (UET), Taxila, Pakistan, in 2012 and 2014, respectively, and the Ph.D. degree in electronic and computer engineering from Sungkyunkwan University, South Korea, in 2020.

He was a Lecturer with the Electronic Engineering Department, University of Engineering and Technology, from 2013 to 2016. His research interest includes generative adversarial networks and its application to computer vision for image unmosaicing and object removal. His awards and honors include the Award of Honors for his B.Sc. degree in electronic engineering from UET, in 2012, the University Scholarship for his M.Sc. degree in computer engineering from UET, in 2012, and the Higher Education Commission Scholarship for his Ph.D. degree from HEC, Pakistan, in 2016.

SEHO BAE received the B.S. degree in electronic and electrical engineering from Sungkyunkwan University, Suwon, South Korea, in 2015, where he is currently pursuing the Integrated Ph.D. degree in electrical and computer engineering. He has been a member of the Computer Vision Laboratory, Sungkyunkwan University, since 2015. His current research interests include cross-modal image matching and cross-modal image synthesis using generative adversarial networks.

JUNEHO YI received the B.S. degree from Seoul National University, South Korea, in 1985, the M.S. degree from Pennsylvania State University, University Park, PA, USA, in 1987, and the Ph.D. degree from Purdue University, West Lafayette, IN, USA, in 1994, all in electrical engineering. In 1989, he was a Research Scientist with the Samsung Advanced Institute of Technology. From 1994 to 1995, he was a Research Scientist with the University of California at Riverside, Riverside. From 1995 to 1996, he was a Senior Research Scientist with the Korea Institute of Science and Technology, Seoul, South Korea. Since 1997, he has been with Sungkyunkwan University, South Korea, where he is currently a Professor with the School of Electronic and Electrical Engineering. His pioneering works include masked fake face detection and depth filtering using parametrized structured light imaging. His research interests include computer vision and statistical pattern recognition.