ABSTRACT

Object removal and image inpainting in facial images is a task in which objects that occlude a facial image are specifically targeted, removed, and replaced by a properly reconstructed facial image. Two different approaches utilize U-net-based generator and modulated approach, and they respectively have been widely endorsed but notwithstanding each method’s disadvantages of low generative capability and low reconstruction power. Here, we propose a Semantics-Guided Inpainting Network (SGIN), which is the invention of a desirable trade-off between those two methods that can be applied to any form of occluding mask while maintaining a consistent style and preserving high-fidelity details of the original image. By using the guidance of a semantic map, our model is capable of manipulating facial features and styles which grants direction to the one-to-many problem for further practicability.

Index Terms— Semantic image inpainting, Object removal, Image-to-image translation

1. INTRODUCTION

Object removal and image inpainting in facial images is a task of removing objects that block the foreground human facial area and reconstructing the occluded facial features. Generally, this task is comprised of three distinct challenges: 1) generalization in novel masks, 2) style consistency, and 3) preservation of known pixels. Object remover modules should be able to specifically erase any occluding objects regardless of their types, shapes, and sizes. The reconstructed image must maintain the style consistent with the unoccluded regions, while at the same time details of unoccluded regions should be preserved in high pixel-by-pixel fidelity.

The most popular methods for this task are the U-net-based generator and modulated-generator-based approach. U-net-based generator focuses on completing only the masked region, and conventionally the rest part of an image is directly copy-and-pasted from the conditioned image. However, as it does not train to construct the whole image, when U-net-based generators are confronted with novel masks which are not seen during training, they tend to significantly underperform. Figure 4 shows the failure cases of the U-net-based generator when tested on the different mask types unseen during the training. Modulated generative approach such as [1] is one of the recent advances among the conditional GANs, which regards an input as a random constant and each convolution layer adjusts the intermediate latent vectors with denormalization factors (e.g. scale and bias). Modulated approach has further advanced the generative capability and edit-ability. However, there is information loss when a conditioned image is compressed to a low-dimensional latent vector. [2] The lost information is mostly high-frequency details or the infrequent information such as background, and this results in the prediction being largely different from the conditioned input.

Due to these trade-offs, conventional object removal methods ([3]) are limited to certain types and shapes of occluding objects, limiting their applicability for real-world data. We aimed to construct an inpainting model which can deal with wide variety of masks, preserving fine details of the unmasked region. With the help of semantic map predictor, our model provides additional semantic knowledge that embodies user’s intention for image inpainting. For the restoration of high-frequency information of the unmasked region, we adopt self-distillation loss and the fusion feedback network. Furthermore, our work provides a broad range of control over the disentangled semantic region, enabling various image manipulation including semantic manipulation and style swapping.

Fig. 1. Our method can provide various predictions by manipulating the semantic map, or the style latent vector. Prediction 3 is the example of referencing the style from right below image when using the semantic map 2.
Our method consists of three different modules: 1) Occlusion Detector 2) Semantic Map Predictor 3) Semantics-Guided Image Inpainter (SGIN). Fig. 2 illustrates the overall architecture of our framework and we describe the details of each module below.

**Occlusion Detector** Given the input \( X_{\text{occ}} \), our occlusion detector detects the occluding objects and predicts the binary mask \( M \) (1: occlusion / 0: non-occlusion). The masked input is denoted as \( X_{\text{masked}} = X_{\text{occ}} \odot (1 - M) \) where \( 1 \) is an all-one image and \( \odot \) denotes element-wise product. We built the network upon a simple ResNet generator architecture constructed with five-layers of convolution blocks and five-layers of deconvolution blocks.

**Semantic Map Predictor** Our below-mentioned inpainting module, SGIN, requires the semantic map to produce guidance for large holes. Although requiring the semantic labeling can accompany much efforts, we overcome this by using a semantic map predictor which enables obtaining the semantic label in an on-the-fly manner so that we can neglect the need of human labeling. It is important to note that the semantic map predictor is a pre-trained network trained with a separate non-overlapping dataset with the SGIN’s training data, and fortunately it generalizes well to the SGIN’s training data.

Given the masked image \( X_{\text{masked}} \), the semantic map predictor predicts the semantic label map \( L = \{l_1, \cdots, l_C\} \). Each \( l_c, c \in [C] \), indicates the binary class label map for eleven regions (i.e., \( C = 11 \)). We trained BiseNet [4] for our generalized semantic map predictor.

**Semantic Style Encoding** We chose Feature Pyramid Network (FPN) [5] as our encoder, which generates latent codes through multi-scaled hierarchical features. Style representations from the latent code are fully determined by the masked image \( X_{\text{masked}} \) and the semantic label map \( L_n, n \in [N] \), where \( N \) indicates the number of layers in the FPN’s feature map. We first concatenate the semantic label \( L_n \) and the masked image \( X_{\text{masked}} \) channel-wise. Then each pyramid network produces \( F_n \), where \( H_n \) and \( W_n \) indicate the spatial dimension of height and width for each layer. We expand \( F_n \in \mathbb{R}^{H_n \times W_n \times 512} \) to \( F_{\text{exp}_n} \in \mathbb{R}^{H_n \times W_n \times 512 \times C} \) by broadcasting along the dimension of binary class map. Also, the semantic label map \( L_n \in \mathbb{R}^{H_n \times W_n \times C} \) is broadcasted along the dimension of feature map channels, producing \( L_{\text{exp}_n} \in \mathbb{R}^{H_n \times W_n \times 512 \times C} \). We also harness the well-known contextual attention module [6] in between the feature pyramids in order to provide additional attention-wise information in the masked region.

**Region-wise Average Pooling (RAP)** As the label map is binary, we can extract semantic latent codes which contain activations for each semantic region by multiplying the latent code with the semantic label map. Each activation is spatially pooled, and then denormalized by the semantic label map. Region-wise Average Pooling (RAP) [7] in an on-the-fly manner so that we can neglect the need of human labeling. It is important to note that the semantic style encoding approach tend to lose high-frequency details because of their lossy data compression. Making up for this loss is the key to best achieving our method’s goal towards high fidelity generation.

After passing through the convolution blocks, the ‘coarse’ output data is compared with the original images overlaid by a mask (conditioned images) and the differences in the corresponding pixels are denoted in the form of MSE loss.

**Fusion Feedback Module**

As mentioned previously, the latent vectors of the modulation generative approach tend to lose high-frequency details because of their lossy data compression. Making up for this loss is the key to best achieving our method’s goal towards high fidelity generation.

To tackle these problems, we introduce the Fusion Feedback Module, which uses a lightweight encoder-decoder network that compares pixel values of the initially reconstructed image with those of the original masked image from the input to retrieve lost details, without any need for additional modules. After the generator draws its first ‘coarse image’, which
is deprived of finer features of the original image, the $L_2$ difference between the generated image and the ground truth is calculated, and this crude generated image along with $L_2$ difference is injected to the middlemost layer of the generator module, which is experimentally shown to generate images with the highest fidelity.

**Loss Function** We introduce the concept of self-distillation loss, which provides the feature-level supervision directly to the generator for preserving high-fidelity details of the input. We devised an information flow that the generator is fed with its own first coarse image along with the loss calculated from the comparison between the feature map of the ground truth and the predicted output. The details of the calculation are as follows: The generator is forwarded with the ground truth and the predicted output. The details of the calculation from the comparison between the feature map of the ground truth and the predicted image, as well as an adversarial loss, which provides the feature-level supervision directly to the generator for preserving high-fidelity details of the input.

Finally, the self distillation loss is defined as

$$L_{sd} = \sum_{i=1}^{K} ||f_i(X_{gt}) - f_i(X_{m})||_2,$$

where $K$ denotes the number of SGI blocks. The advantageous effect of using self-distillation loss can be found in the ablation study section.

In addition, we applied several conventionally used loss functions in the literature of image inpainting. The discriminator computes the $L_{feat}$, which is the $L_1$ loss between the discriminator features for the $X_{gt}$ and the predicted image, as well as an adversarial loss $L_{adv}$. Also, we used the $L_{per}$, which is the perceptual loss between the features of $X_{gt}$ and $X_{masked}$ extracted from a VGG-19 network [8]. $L_{adv}$ and $L_{per}$ are defined as follows:

$$L_{adv} = \mathbb{E}_X[\log D(X)] + \mathbb{E}_X[\log(1 - D(G(X_{masked}|L)))]$$

$$L_{per} = ||\text{Vgg}(G(X_{masked}|L)) - \text{Vgg}(X)||_2.$$  

Here, $G(\cdot|L)$ is the generator conditioned on the semantic map $L$, while $D(\cdot)$ denotes the discriminator. The overall loss is as follows:

$$L = \lambda_{sd}L_{sd} + \lambda_{feat}L_{feat} + \lambda_{per}L_{per} + \lambda_{adv}L_{adv}.$$  

We uniformly set the loss weights $\lambda$’s to 10.

### 3. Experiments

To generate diverse occluded facial images, we used Naturalistic Occlusion Generation (NatOcc) [9] to overlay human facial images from HELEN [10] and CelebA-HQ [11] with occluding objects and create naturalistic synthetic images. As for the occluding objects, we used 128 objects across 20 categories from Microsoft Common Objects in Context (COCO) and 200 hands from EgoHands [12]. Note that for the training of our semantic map predictor, HELEN-derived occlusion images are used and its evaluation is done using CelebA-HQ images. Different from this, for the SGIN, we only used CelebA-HQ. We split CelebAMask-HQ-derived images into 22,300 training images and 2,800 validation images.

![Fig. 4. Qualitative Analysis of seven inpainting models. Please zoom-in to check the details of the results of model comparisons.](image-url)

**Comparison with baseline models** We compared our SGIN with various image-inpainting models different in their types and schemes. For the U-net architecture, we chose Deepfill-v2 [13] and Crfill [14], and for the modulated generator architecture, we chose PsP [5] and E4E [15]. We also included SEAN [7] in that it also uses semantic maps, and MAT [16], the current state-of-art (SOTA) inpainting module which is based on a transformer model. For a fair comparison, all of these baseline models are retrained with the same NatOcc datasets with the same masks, except for MAT whose large computational cost is unaffordable in our devices. Alternatively, we made our comparison based on the pretrained CelebA-HQ MAT model uploaded at the author’s GitHub repository and used the same masks as ours.

**Quantitative Evaluation** We employed six metrics that can shed light on the different aspects of the quality of reconstruction: PSNR, SSIM [17], MS-SSIM [18], RMSE, LPIPS [19], and FID [20]. We evaluated the average scores for all of the validation samples.

Note that U-net generators and the SOTA model MAT conventionally copy-and-paste the rest of the unmasked region from the original image unlike modulated generators that try to reconstruct the whole image. This means that when loss is calculated by comparing the output with its ground truth, they enjoy unfair advantage in the loss score as the unmasked region of the their output is always the same as the ground truth. Therefore, we measured two different scores: one of which is measured from the whole generated image and the other measured with the masked region only, where the unmasked region has pixel value of 0. In Table 1, each cell contains two values, where the left is calculated with the whole image, while the other at the right is calculated only with the masked region. As the table shows, our model scored...
Table 1. Quantitative comparison on CelebA-HQ with 256 × 256 resolution. The color red designates proper scores for models which copy-and-paste the unmasked region and the color blue designates proper scores for models which generate the whole image. It is appropriate to compare the scores in the same color.

| models          | PSNR(+) | SSIM(+) | MS-SSIM(+/-) | RMSE(-) | LPIPS(-) | SSIM(+) | PSNR(+) | MS-SSIM(+/-) | RMSE(-) | LPIPS(-) | SSIM(+) | PSNR(+) | MS-SSIM(+/-) | RMSE(-) | LPIPS(-) | SSIM(+) |
|-----------------|---------|---------|--------------|---------|----------|---------|---------|--------------|---------|----------|---------|---------|--------------|---------|----------|---------|
| DeepFill-v2     | 25.34   | 0.91    | 0.90         | 0.66    | 0.94     | 0.56    | 0.92    | 0.69         | 0.91    | 0.63     | 0.87    | 0.90    | 0.51         | 0.90    | 0.87     | 0.91    |
| Crfill          | 17.18   | 12.94   | 0.66         | 0.94    | 0.69     | 0.91    | 0.63    | 0.87         | 0.91    | 0.63     | 0.87    | 0.90    | 0.51         | 0.90    | 0.87     | 0.91    |
| MAT             | 23.76   | 26.62   | 14.45        | 14.45   | 28.19    | 12.10   | 37.38   | 15.89        | 15.38   | 15.38    | 17.18   | 12.94   | 38.50        | 6.31    | 25.77    | 6.88    |
| SGIN (ours)     | 25.04   | 25.23   | 19.36        | 27.13   | 16.96    | 24.79   | 25.04   | 25.23        | 23.76   | 26.62    | 17.18   | 12.94   | 25.98        | 6.59    | 25.52    | 25.52   |

Qualitative Evaluation Figure 4 shows image inpainting results from various models against the four types of occluding masks of common objects (Object mask), human hands (Hand mask), rectangular masks occluding half of the image (Half mask), and rectangular masks at the center taking 70% of the image (Center mask). Two U-net generators (DeepFill-v2 and Crfill) still suffer from some noticeable glitches in their reconstructed images for the masks seen during training (Object and Hand mask) and can produce no acceptable reconstructions for other types of masks, which points out the known problem in the U-net generator’s generalization capability. For modulated generator models (PsP and E4E) and SEAN, their output image is very natural and works relatively well for all four types of masks, but fine features like the orientation of the eyes and other details such as the background are not retained. Our model can faithfully regenerate the occluded regions of the face and match the overall performance of MAT, even though ours require far less computational resources.

4. VARIOUS APPLICATIONS OF SGIN

Lots of conventional image inpainting models copy-and-paste the unmasked region so the generator only learns how to construct the masked region. It deters the generator from flexible reconstruction of an image, so the model can only control the masked region. Different from the conventional models, our inpainting module has a control over not only the masked region, but also the unmasked regions. Our model has universal applicability in terms of broad controllability.

Semantic map manipulation Our SGIN module receives two separate inputs, the semantic map and the latent vectors encoded for style and spatial information. The output is constructed in accordance with its semantic map, and by changing it, the user can manipulate the outcome of reconstruction as the user desires. Fig. 5 shows some examples of semantic map manipulation. Our model is capable of creating extra facial features such as glasses and lifted eyes, and altering facial features to have larger eyes, closed lips, shortened eyebrows and opened mouth.

Style Swapping The use of SEAN at the normalization blocks enables simultaneous control over both the spatial and style information. By referring to the style from another image or using a user-defined style, it is possible to change the outcome of the reconstruction. Fig. 5 shows the examples of style swapping, done by swapping the extracted style latent vectors from the style encoder of the target image and the source image.

5. CONCLUSION

We demonstrated that the integration of semantic map, self-distillation loss, and fusion feedback module not only retains the mask-independence of the modulated generator, but also guides it to reconstruct the output image in high consistency with the unmasked image. As our model uses a semantic map of a human face, it also enables users to directly control the reconstructed image by feeding the network with the facial semantic map that includes desired features. We expect that more generalized architecture with the same rationale will usher the advent of an inpainting model with broad applicability.

* Acknowledgement: This work was supported by NRF (2021R1A2C3006659) and IITP (2021-0-00537) of Korean Government.
6. REFERENCES

[1] 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, 2019, pp. 4401–4410.

[2] Claude E Shannon et al., “Coding theorems for a discrete source with a fidelity criterion,” IRE Nat. Conv. Rec., vol. 4, no. 142-163, pp. 1, 1959.

[3] Nizam Ud Din, Kamran Javed, Seho Bae, and Juneho Yi, “Effective removal of user-selected foreground object from facial images using a novel gan-based network,” IEEE Access, vol. 8, pp. 109648–109661, 2020.

[4] Bisenet Yu, J Wang, C Peng, C Gao, G Yu, N Sang, and Bisenet, “Bilateral segmentation network for real-time semantic segmentation,” in Proceedings of the European Conference on Computer Vision (ECCV), 2018, p. 325.

[5] Elad Richardson, Yuval Alaluf, Or Patashnik, Yotam Nitzan, Yaniv Azar, Stav Shapiro, and Daniel Cohen-Or, “Encoding in style: a stylegan encoder for image-to-image translation,” in Proceedings of the IEEE/CVF conference on computer vision and pattern recognition, 2021, pp. 2287–2296.

[6] Jiahui Yu, Zhe Lin, Jimei Yang, Xiaohui Shen, Xin Lu, and Thomas S Huang, “Generative image inpainting with contextual attention,” in Proceedings of the IEEE conference on computer vision and pattern recognition, 2018, pp. 5505–5514.

[7] Peihao Zhu, Rameen Abdal, Yipeng Qin, and Peter Wonka, “Sean: Image synthesis with semantic region-adaptive normalization,” in Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2020, pp. 5104–5113.

[8] Karen Simonyan and Andrew Zisserman, “Very deep convolutional networks for large-scale image recognition,” arXiv preprint arXiv:1409.1556, 2014.

[9] Kenny TR Voo, Liming Jiang, and Chen Change Loy, “Delving into high-quality synthetic face occlusion segmentation datasets,” in Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2022, pp. 4711–4720.

[10] Vuong Le, Jonathan Brandt, Zhe Lin, Lubomir Bourdev, and Thomas S Huang, “Interactive facial feature localization,” in European conference on computer vision. Springer, 2012, pp. 679–692.

[11] 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, 2020, pp. 5549–5558.

[12] Sven Bambach, Stefan Lee, David J Crandall, and Chen Yu, “Lending a hand: Detecting hands and recognizing activities in complex egocentric interactions,” in Proceedings of the IEEE International conference on computer vision, 2015, pp. 1949–1957.

[13] Jiahui Yu, Zhe Lin, Jimei Yang, Xiaohui Shen, Xin Lu, and Thomas S Huang, “Free-form image inpainting with gated convolution,” in Proceedings of the IEEE/CVF international conference on computer vision, 2019, pp. 4471–4480.

[14] Yu Zeng, Zhe Lin, Huchuan Lu, and Vishal M Patel, “Cr-fill: Generative image inpainting with auxiliary contextual reconstruction,” in Proceedings of the IEEE/CVF International Conference on Computer Vision, 2021, pp. 14164–14173.

[15] Omer Tov, Yuval Alaluf, Yotam Nitzan, Or Patashnik, and Daniel Cohen-Or, “Designing an encoder for stylegan image manipulation,” ACM Transactions on Graphics (TOG), vol. 40, no. 4, pp. 1–14, 2021.

[16] Wenbo Li, Zhe Lin, Kun Zhou, Lu Qi, Yi Wang, and Jiayu Jia, “Mat: Mask-aware transformer for large hole image inpainting,” in Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2022, pp. 10758–10768.

[17] 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, vol. 13, no. 4, pp. 600–612, 2004.

[18] Zhou Wang, Eero P Simoncelli, and Alan C Bovik, “Multiscale structural similarity for image quality assessment,” in The Thirty-Seventh Asilomar Conference on Signals, Systems & Computers, 2003. Ieee, 2003, vol. 2, pp. 1398–1402.

[19] 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, 2018, pp. 586–595.

[20] Martin Heusel, Hubert Ramsauer, Thomas Unterthiner, Bernhard Nessler, and Sepp Hochreiter, “Gans trained by a two time-scale update rule converge to a local nash equilibrium,” Advances in neural information processing systems, vol. 30, 2017.