FurryGAN: High Quality Foreground-aware Image Synthesis

Jeongmin Bae, Mingi Kwon, and Youngjung Uh⋆

Yonsei University
{jaymin.bae, kwonmingi, yj.uh}@yonsei.ac.kr

Fig. 1: **Example images and the corresponding foreground masks. Both are simultaneously *generated* by our model.** FurryGAN learns not only to generate realistic images, but also to synthesize alpha masks with fine details such as hair, fur, and whiskers in a fully unsupervised manner (left). Our model also can be trained on various datasets (right).

Abstract. Foreground-aware image synthesis aims to generate images as well as their foreground masks. A common approach is to formulate an image as an masked blending of a foreground image and a background image. It is a challenging problem because it is prone to reach the trivial solution where either image overwhelms the other, i.e., the masks become completely full or empty, and the foreground and background are not meaningfully separated. We present FurryGAN with three key components: 1) imposing both the foreground image and the composite image to be realistic, 2) designing a mask as a combination of coarse and fine masks, and 3) guiding the generator by an auxiliary mask predictor in the discriminator. Our method produces realistic images with remarkably detailed alpha masks which cover hair, fur, and whiskers in a fully unsupervised manner. Project page: https://jeongminb.github.io/FurryGAN/

⋆ Corresponding author

1   Introduction

As the quality of images from generative adversarial networks (GANs) improves [9,15,16,13,14], discovering the semantics in their latent space is useful to control the generation process [27,2,28,40] or to edit real images through latent inversion [25,26,42,4]. Localizing the semantics in the latent space is another important research direction for understanding how GANs work. Some methods tackle local editing by separating parts in the intermediate feature maps [7,17].

Meanwhile, a few recent works tackle foreground-aware image synthesis by modeling an image as a composition of foreground and background images according to a mask. While previous methods achieve some success, they explicitly prepare a background distribution by removing foreground with an off-the-shelf object detector [29], assume that images with shifted foreground objects should look real [5,37], or require multi-stage training with dataset-tailored hyperparameters [1]. These ingredients are obstacles that block general solutions for foreground-aware synthesis.

In this paper, we propose FurryGAN which learns to synthesize images with the explicit understanding of the foreground given only a collection of images. Intuitions in our method include the following. 1) We encourage the foreground images and the composite images to resemble the training distribution. It prevents the foreground from losing the objects. 2) We introduce coarse and fine masks. The coarse mask captures the rough shape, and the fine mask captures details such as whiskers and hair. 3) We introduce an auxiliary task for the discriminator to predict the mask from the generated image so that the generator produces the foreground image aligned with the mask.

Compared to the previous works, our method does not require off-the-shelf networks, the assumption for perturbation, multi-stage training, or careful early stopping. Experiments demonstrate the superiority of our framework compared to previous methods regarding high quality alpha masks. We also provide thorough ablation studies to justify each component of our method.

Fig. 1 shows example synthesized images and the corresponding alpha masks. They catch unprecedented levels of fine details, especially in hair and whiskers. Consequently, the detailed masks enable natural composition of the foreground part and any background (Fig. 5). As a byproduct, GAN inversion on our method achieves unsupervised object segmentation with the same level of details.

2   Related Work

GANs and semantic interpretation. GANs [15,16,14] synthesize astonishingly high quality images from random latent codes. Understanding semantic interpretation of the latent codes is an important research topic so that users can control the generation process or edit real images through latent inversion [27,42,30,26,35]. Instead, we focus on teaching GANs spatial understanding of foreground objects.

Foreground-aware GANs. Although semantic interpretation has some correlation with spatial separation, incorporating the notion of foreground objects has been tackled in the orthogonal direction, mostly by modeling an image as a combination of foreground and background according to a mask. PSeg [5] and improved layered GAN [37] rely on an assumption that the composite image with spatially transformed foreground should still be realistic. However, the parameters for the transformation should be determined for each dataset, and the assumption does not hold when the foreground region touches a border of the image. In [32,20,33], they identify the latent directions in a pretrained generator for changing the background to separate the foreground and background. Labels4Free [1] trains an alpha mask network that produces masks for combining foregrounds and backgrounds, generated by pretrained StyleGAN2 and pretrained pseudo-background StyleGAN2, respectively. Whereas it requires multi-stage training with tailored hyperparameters, our framework is trained in end-to-end fashion and produces remarkably fine details in the masks.

Unsupervised segmentation. Early image segmentation methods rely on the clustering of color and coordinates [3,8]. In order to cluster the regions regarding semantics, learning deep networks for maximizing mutual information within the cluster [12,24] or for contrasting different instances [31] have been successful. These objectives assume multiple classes and are not straightforward to be applied in foreground-background separation. Given a generator for foreground-aware image synthesis, inverting a real image to the latent space inherently leads to unsupervised foreground segmentation. Thus we focus on a better understanding of foreground in GANs.

3D-aware GANs. 3D-aware GANs based on NeRF [22] represent a scene as a neural network which receives 3D coordinates and outputs their color or feature vector with occupancy. Furthermore, recent approaches divide the scene into foreground and background by a depth threshold [10,41] or separate feature fields [23]. However, they aim to understand the 3D geometry of the scene and do not explicitly learn to generate high quality foreground alpha masks.

3   Method

In this section, we overview our framework (§ 3.1), describe the networks (§ 3.2), and explain their training techniques including loss functions (§ 3.3). To begin with, we briefly introduce a common formulation.

Common formulation. We follow the common formulation [1,5,29,37] for generating images: an image is a masked combination of a foreground image $\mathbf{x}_{\mathrm{fg}}$ and a background image $\mathbf{x}_{\mathrm{bg}}$, according to an alpha mask $\mathbf{m}$. Formally,

$$\mathbf{x}_{\mathrm{comp}} = \mathbf{m} \odot \mathbf{x}_{\mathrm{fg}} + (1 - \mathbf{m}) \odot \mathbf{x}_{\mathrm{bg}}, \tag{1}$$

where $\odot$ denotes pixel-wise multiplication.

Fig. 2: **Our framework.** consists of a foreground generator, a mask generator, a background generator, and a discriminator with a mask predictor. The alpha mask specifies the combination of the foreground image and the background image to produce the composite image. We feed both the foreground images and the composite images to the discriminator as fake images.

3.1 Framework overview and dual fake input strategy

Fig. 2 shows the framework overview. FurryGAN has three generators for the foreground, background, and mask to produce the composite images according to Eq. (1). Then the discriminator guides the generator to produce realistic images.

Dual fake input strategy. Guiding the generator to produce realistic *composite* images solely does not guarantee the separation of the foreground and background. Motivation of the dual fake input is the following. The foreground images should contain salient objects (e.g., a person in FFHQ) so that there exists a solution for the masks to produce realistic composite images including the foreground images. Otherwise, the mask will favor excluding the foreground images from the composite images. Hence, we ensure the foreground images to contain salient objects by imposing a sufficient condition: being realistic by themselves. The fake mini-batch for the discriminator consists of the foreground images and the composite images (Fig. 3(a)). Then the discriminator tries to classify them as fakes and the generator tries to produce realistic images in both the foreground and the composite images. We find that the dual fake input strategy helps prevent improper foreground separation.

3.2 Architecture

Generators. The foreground and background generators, $\mathcal{G}_{fg}$ and $\mathcal{G}_{bg}$, synthesize images $\mathbf{x}_{fg}$ and $\mathbf{x}_{bg}$ from latent codes $\mathbf{z}_{fg}$ and $\mathbf{z}_{bg}$, respectively. The two generators do not share any parameters. The mask generator $\mathcal{G}_{mask}$ synthesizes $\mathbf{m}$ from the penultimate feature maps of the foreground generator. Then,

Fig. 3: **Dual fake input strategy and mask generators.** (a) Our discriminator receives the foreground images and the composite images as fake. They evenly share a fake mini-batch. (b) Our mask generator produces an alpha mask as a combination of a coarse mask and a fine mask.

their simple alpha-blending produces the composite image $\mathbf{x}_{\text{comp}}$ (Eq. (1)). Note that the composite function causes unexpected additional degree of freedom: $\mathbf{x}_{\text{fg}} \odot \mathbf{m} = 2 \cdot \mathbf{x}_{\text{fg}} \odot 0.5 \cdot \mathbf{m}$. Thus we restrict the generators' outputs to be in range of $[-1, 1]$ by adding a tanh function at the output of the ToRGB layer.

Coarse and fine mask generator. As shown in Fig. 3 (b), our mask generator consists of a coarse mask network $\mathcal{G}_{\mathbf{m}_{\text{coarse}}}$ and a fine mask network $\mathcal{G}_{\mathbf{m}_{\text{fine}}}$. We expect the coarse mask network to cover the overall shape and the fine mask network to make up for the details missed in the coarse mask (e.g., cat whiskers, fur, and hair). Each mask is normalized to the range of $[0,1]$ by min-max normalization and their summation becomes the final alpha mask. The final mask $\mathbf{m}$ is computed as:

$$\mathbf{m}_{\text{coarse}} = \mathcal{G}_{\mathbf{m}_{\text{coarse}}}(\mathbf{f}), \quad \mathbf{m}_{\text{fine}} = \mathcal{G}_{\mathbf{m}_{\text{fine}}}(\mathbf{f}), \tag{2}$$

$$\mathbf{m} = \text{clip}(\mathbf{m}_{\text{coarse}} + \gamma \mathbf{m}_{\text{fine}}, 0, 1), \tag{3}$$

where $\mathbf{f}$ denotes the penultimate feature maps of the foreground generator. The design details are described in the appendix (§ E). For stability, we fade in the fine mask by linearly increasing γ from 0 to 1 over the first 5K iterations.

Discriminator with a mask predictor. We follow the discriminator architecture of StyleGAN2 [16] and add an auxiliary mask predictor. The mask predictor tries to reconstruct the mask of an input image given the 16×16 feature maps. It has minimal capacity for predicting the masks, i.e., two 1×1 convolutional layers and residual connections. How it guides the generators will be discussed in the following section (Eq. (4) and Eq. (5)).

3.3   Training objectives

Adversarial loss. As described in § 3.1, we impose adversarial losses on the foreground image and the composite image. We adopt non-saturating loss [9]

(a) Corner cases without mask consistency loss

(b) Illustration of the mask consistency loss

Fig. 4: **Mask consistency loss.** (a) Results without mask consistency loss show inconsistency between foreground images and composite images, e.g., cutting off long hair or adding shoulders. (b) Mask consistency loss computes discrepancy between the predicted masks of the foreground and composite images. $//$ denotes a stop gradient operator.

and lazy R1-regularization [21,16] and skip defining trivial equations $L_{\mathrm{adv}}^{\mathcal{D}}$, $L_{\mathrm{R1}}^{\mathcal{D}}$, and $L_{\mathrm{adv}}^{\mathcal{G}}$ for brevity. Adversarial losses act as the primary source for driving foreground-aware image synthesis.

Mask prediction loss. The auxiliary mask predictor $\mathcal{D}_{\mathrm{aux}}$ in the discriminator aims to regress the generated mask given the generated image:

$$L_{\mathrm{pred}} = \frac{1}{|\hat{\mathbf{m}}|}\|\texttt{Downsample}(\mathbf{m}) - \hat{\mathbf{m}}\|_2^2, \tag{4}$$

where $\hat{\mathbf{m}}$ is the output of $\mathcal{D}_{\mathrm{aux}}$ for the generated images ($\mathbf{x}_{\mathrm{fg}}$ and $\mathbf{x}_{\mathrm{comp}}$). We use bilinear interpolation for $\texttt{Downsample}$. The $16{\times}16$ prediction will be useful for guiding the generator in conjunction with Eq. (5).

Mask consistency loss. We observe that object regions of foreground $\mathbf{x}_{\mathrm{fg}}$ and composite $\mathbf{x}_{\mathrm{comp}}$ can be inconsistent. For example, the mask may cut off the long hair in the foreground so that the composite image becomes a face with short hair. As another example, the missing body part of the foreground object may be supplemented from the background (both cases are shown in Fig. 4(a)).

Hence, we demand the mask predicted from the composite image to be consistent to the mask predicted from the foreground image (Fig. 4(b)):

$$L_{\mathrm{consistency}} = \frac{1}{|\hat{\mathbf{m}}_{\mathrm{comp}}|}\|\mathcal{D}_{\mathrm{aux}}(\texttt{stopgrad}(\mathbf{x}_{\mathrm{fg}})) - \hat{\mathbf{m}}_{\mathrm{comp}}\|_2^2, \tag{5}$$

where $\hat{\mathbf{m}}_{\text{comp}} = \mathcal{D}_{\text{aux}}(\mathbf{x}_{\text{comp}})$ and $\texttt{stopgrad}(\cdot)$ denotes a stop gradient operator.

As the mask predictor regresses the mask from the mask generator given a composite image, imposing consistency between the two masks encourages the foreground object region and the generated mask to resemble each other.

Coarse mask loss. We adopt binarization loss and area loss following previous methods [5,1]. The binarization loss pushes the alpha values in the masks to either 0 or 1:

$$L_{\text{binary}} = \mathbb{E}[\min(\mathbf{m}_{\text{coarse}}, 1 - \mathbf{m}_{\text{coarse}})]. \tag{6}$$

The area loss penalizes the ratio of a mask being less than ϕ_1 to promote using the foreground images more than ϕ_1, i.e., preventing the background image from taking charge of everywhere, which is a degenerate solution:

$$L_{\text{area}}^{\text{coarse}} = \max(0, \phi_1 - \frac{1}{|\mathbf{m}_{\text{coarse}}|} \sum \mathbf{m}_{\text{coarse}}), \tag{7}$$

where $|\mathbf{m}|$ denotes the number of pixels in the mask image and ϕ_1 is set to 0.35 for all experiments (unless otherwise noted). The final coarse mask loss is:

$$L_{\mathbf{m}_{\text{coarse}}} = L_{\text{binary}} + L_{\text{area}}^{\text{coarse}}. \tag{8}$$

Fine mask loss. The fine mask aims to capture details like hair, fur, and whiskers. Such a thin body becomes transparent due to the property of light. Hence, we do not use the binarization loss to free the masks to bear medium values between 0 and 1. Instead, we impose an inverse area loss to prevent the fine mask from taking charge of too large area:

$$L_{\mathbf{m}_{\text{fine}}} = L_{\text{area}}^{\text{fine}} = \max(0, \phi_2 - \frac{1}{|\tilde{\mathbf{m}}_{\text{fine}}|} \sum (1 - \tilde{\mathbf{m}}_{\text{fine}})), \tag{9}$$

where $\tilde{\mathbf{m}}_{\text{fine}} = \mathbf{m} - \mathbf{m}_{\text{coarse}}$ to penalize the area where the fine mask actually contributes after clipping. ϕ_2 is set to 0.01 in all experiments. More details about the mask are illustrated in Appendix E.

Background participation loss. We sometimes observe that the alpha mask tries to employ foreground images excessively. As a remedy, we penalize the difference between the composite image and the background image. It indirectly removes the excessive spread of the alpha mask.

$$L_{\text{reg}} = \frac{1}{|\mathbf{x}_{\text{comp}}|} \|\mathbf{x}_{\text{comp}} - \mathbf{x}_{\text{bg}}\|_2^2 \tag{10}$$

Intuitively, an easy way to reduce the difference between the composite image and the background is to remove unnecessary foreground areas that do not harm the realism of the composite image.

Overall objective. Consequently, our full loss functions are:

$$L_{\text{total}}^{\mathcal{D}} = L_{\text{adv}}^{\mathcal{D}} + L_{\text{R1}}^{\mathcal{D}} + L_{\text{pred}}, \tag{11}$$

$$L_{\text{total}}^{\mathcal{G}} = L_{\text{adv}}^{\mathcal{G}} + L_{\text{consistency}} + \lambda_{\text{coarse}} L_{\mathbf{m}_{\text{coarse}}} + \lambda_{\text{fine}} L_{\mathbf{m}_{\text{fine}}} + L_{\text{reg}}. \tag{12}$$

4   Experiments

4.1   Implementation Detail

Our foreground generator and background generator are based on StyleGAN2[16]. For simplicity, we remove output skip connections in the synthesis network and use a shallow mapping network [13]. The foreground and background generators, $\mathcal{G}_{\text{fg}}$ and $\mathcal{G}_{\text{bg}}$, use a slightly modified StyleGAN2 structure. Number of channels of the latent codes and the feature maps in $\mathcal{G}_{\text{fg}}$ and $\mathcal{G}_{\text{bg}}$ become $\frac{3}{4}$ and $\frac{1}{4}$, respectively. As a result, the total number of parameters reduces by about half. Similar to [37], background codes are shared with foreground codes. More precisely, we borrow the front part of the $\mathbf{z}_{\text{fg}} \sim \mathcal{N}(0, I)$ and use it as $\mathbf{z}_{\text{bg}}$.

We train our model on a single RTX-3090 for a period of about 100 hours. In all experiments, we trained our model for 300K iterations with a batch size of 16. We follow training parameters from StyleGAN2 but do not use mixing regularization. Mask consistency loss and background participation loss update the model every other iteration. We set $\lambda_{\text{coarse}} = \lambda_{\text{fine}} = 5$. The coefficient of binarization loss is linearly reduced to 0.5 over the first 5K iterations.

4.2   Setup

Datasets. We evaluate our model on FFHQ[15] and AFHQv2-Cat[6,14]. FFHQ has 70,000 high-quality images of human faces. It has faces of various races and poses and also has good coverage of accessories such as eyeglasses, hats, etc. AFHQv2-Cat contains 5000 images of cat faces. The rebuilt (v2) dataset has higher quality due to proper resizing and compression. We also trained our model on unaligned datasets such as LSUN-Object [39], and CUB[34] (see Appendix F for details and results). All models are trained at 256×256 resolution.

Pseudo ground truth masks. We evaluate the generated mask quality to show foreground-background separation performance. Because the generated images do not contain ground truth masks for evaluating the generated foreground masks, we adopt TRACER [19] to prepare pseudo ground truth masks. It provides detailed masks including hair and whiskers, which are not captured by segmentation networks used in PSeg [5] and Labels4Free [1]. Please refer to Appendix B for their comparison.

(a) Generated foreground on generated background   (b) Generated foreground on real background

Fig. 5: **Composite images.** The same person is placed on the vertical axis, and the same background is placed on the horizontal axis.

Fig. 6: **Latent space interpolation.** We show the mask changes naturally as the image changes.

Metrics. To quantitatively measure the quality of images, we compute Fréchet Inception Distance (FID)[11] between generated foreground images and all training images. Unless otherwise specified, all results were obtained with 50,000 generated images following [13,14]. To quantitatively measure the quality of masks, we employ intersection over union (IoU) for the foreground and background, and their mean (mIoU). IoU and mIoU measure the overlap between prediction masks and ground truth masks. Furthermore, we report standard segmentation metrics: precision, recall, F1 score, and segmentation accuracy following [1].

4.3   Experiments about masked foregrounds

Thanks to the high quality mask generated by our model, the masked foreground object can be naturally combined with various backgrounds, as shown in Fig. 5. We sample random background latent codes for the backgrounds used in Fig. 5 (a). Fig. 6 shows that the interpolation in the foreground latent space not only changes the image but also changes the shape of masks correspondingly.

Fig. 7: **Ablation of dual fake input strategy.** Each row shows the early results as training proceeds. (a)without dual fake input, foreground object has shown in background image. (b)With dual fake input, it is separated naturally.

4.4   Ablation study

Dual fake input strategy. Fig. 7(a) shows that the foreground and mask generator fail to synthesize meaningful foreground and mask, respectively, without the dual fake input strategy. We suppose that it is easier for the generators to focus on the background to synthesize realistic composite images because the foreground generator has a more complicated task: producing foreground images and the masks with more parameters. On the other hand, with the dual fake input strategy, the foreground images have clear objects as they should resemble the training images (Fig. 7(b)).

Ablation of losses. Fig. 8 visually compares the results without one component at a time and our full method. Without background participation loss (Eq. (10)), the masks tend to be wider than the foreground object area. Without mask consistency loss (Eq. (5)), the mask does not align correctly with the foreground object region. Without the fine mask network, the fine details in the mask tend to be less accurate, especially on the region between hair and background. With all components combined, our method produces fine masks aligned to the foreground object region.

Table 1 provides quantitative ablation study. Decrease in quality of masks show the necessity of background participation loss and mask consistency loss. Ablating any of the components harms FID, implying that spatial understanding in the generators is important for the quality of images. We suppose that the influence of the fine mask generator is negligible in the metrics for the masks because the metrics are not sensitive enough to reflect changes in small area.

Fig. 8: **Ablation of our methods.** Three columns show the result without one component. (a) The masks have background area. (b) The masks are not align correctly with the foreground object region. (c) The masks bring the surrounding background. (d) our method produces fine-grained masks.

Table 1: Quantitative comparison of ablation study on FFHQ.

Setting	IoU(fg/bg)	mIoU	recall	precision	F1	Accuracy	FID
w/o Fine Mask Network	**0.93/0.83**	**0.88**	0.95	**0.98**	**0.96**	**0.93**	9.48
w/o BG Participation	0.91/0.77	0.84	**0.97**	0.94	0.95	0.91	9.79
w/o Mask Consistency	0.92/0.81	0.86	0.93	0.98	**0.96**	0.92	9.53
Full ours	**0.93**/0.82	**0.88**	0.95	**0.98**	**0.96**	**0.93**	**8.72**

4.5   Comparisons

Competitors. We choose PSeg[1] [5] and Labels4Free[2] (L4F in short, [1]) as our competitors. For a fair comparison, we trained L4F in FFHQ and AFHQ under the same conditions, i.e., batch size, data augmentations, training iterations. We pretrain StyleGAN2[3] for 256×256 resolution and then train the alpha network with its official setting. As Labels4Free does not conduct experiments on AFHQ, we train their alpha network for the same number of iterations on FFHQ (=1K), and manually find the working hyperparameters: $\lambda_2 = 3$ and $\phi_2 = 0.2$[4]. For PSeg, we add additional layers to their networks for 256×256 resolution since PSeg conducts their experiments in 128×128 resolution. When training Pseg, we followed the default setting reported in the paper. We do not include FineGAN [29] because it does not focus on foreground-background separation and it requires an external pretrained object detector for supervision.

---

[1] https://github.com/adambielski/perturbed-seg
[2] https://github.com/RameenAbdal/Labels4Free
[3] https://github.com/rosinality/stylegan2-pytorch
[4] Without setting ϕ_2, all masks of Labels4Free saturate to 1.

Fig. 9: Qualitative comparison of image composition results on FFHQ and AFHQv2-Cat.

Qualitative Results. Fig. 9 provides a qualitative comparison between the methods. PSeg rarely succeeds in synthesizing proper foreground images and mostly draws objects on the background images. We suppose the reason to be the unmet assumption: for faces, the foreground cannot help touching the edges, thus shifting the foreground will not be realistic. Labels4Free somewhat successes separating foregrounds. However, their masks are not accurate enough and the composition leaves artifacts on the boundaries. In contrast, our method produces masks that accurately capture the foreground object, even including hair, fur, and whiskers. Uncurated samples can be found in Appendix A

Quantitative Results. Table 2 reports how well the masks align with the foreground object region. The pseudo ground truth masks are obtained by feeding the foreground images to TRACER [19]. Our method consistently outperforms the competitors in all settings: different levels of truncation and datasets. Appendix C provides the results with other choices of pseudo ground truth.

Table 3 quantitatively compares the visual quality of the generated images. Our method achieves FIDs comparable to Labels4Free whose foreground generator equals the pretrained StyleGAN2 while drastically improving the masks.

Table 2: Quantitative comparison of alpha mask results on FFHQ and AFHQv2-Cat. We report the result with/without truncation(ψ=1.0, 0.7) and the threshold for the mask is 0.5(Ours, PSeg) and 0.9(L4F).

	ψ	method	IoU(fg/bg)	mIoU	recall	precision	F1	Accuracy
FFHQ	1.0	PSeg	0.05/0.23	0.14	0.05	0.18	0.07	0.05
		L4F	0.87/0.70	0.78	0.92	0.94	0.93	0.87
		Ours	**0.93/0.82**	**0.88**	**0.95**	**0.98**	**0.96**	**0.93**
	0.7	PSeg	0.01/0.23	0.12	0.01	0.04	0.01	0.01
		L4F	0.91/0.79	0.85	0.94	0.97	0.95	0.91
		Ours	**0.95/0.88**	**0.91**	**0.95**	**0.99**	**0.97**	**0.95**
AFHQv2-Cat	1.0	PSeg	0.06/0.23	0.15	0.06	0.16	0.07	0.06
		L4F	0.91/0.80	0.86	0.93	**0.98**	0.95	0.91
		Ours	**0.94/0.82**	**0.88**	**0.98**	0.96	**0.97**	**0.94**
	0.7	PSeg	0.01/0.19	0.10	0.01	0.12	0.01	0.01
		L4F	0.91/0.79	0.85	0.94	**0.97**	0.95	0.91
		Ours	**0.95/0.87**	**0.91**	**0.98**	**0.97**	**0.97**	**0.95**

Table 3: Quantitative comparison of generated foreground images on FFHQ and AFHQv2-Cat. Foreground generator of L4F equals to the pretrained StyleGAN2.

	FID	
	FFHQ	AFHQv2-Cat
Pseg	62.44	12.71
Labels4Free (=StyleGAN2)	**6.51**	**5.19**
Ours	8.72	6.34

4.6   Segmenting real images.

In addition, we demonstrate an extension of our method for segmenting real images. Following Labels4Free, we use 1K images and their ground truth segmentation masks from CelebAMask-HQ dataset [18] for evaluation. We employ the original inversion method from StyleGAN2. While Table 4 shows that our method achieves similar performance, Fig. 10 shows that our method produces much more accurate and finer masks.

5   Conclusion and discussion

Understanding spatial semantics in the synthesized images is an important research problem in GANs. In this paper, we proposed a GAN framework for foreground-aware image synthesis, generating images as a combination of foreground and background according to a mask. Our method achieves dramatic improvement in the fine details of the masks without any supervision or dataset-tailored assumption. Our model also can be trained on unaligned datasets such as LSUN, indicating that our method generally works well.

Table 4: Quantitative comparison of alpha masks from inverted real images on CelebAMask-HQ. We report the result with the original inversion method from StyleGAN2 and the threshold for the mask is 0.5(Ours) and 0.9(Labels4Free).

	IoU(fg/bg)	mIoU	recall	precision	f1	accuracy
Labels4Free	**0.93**/0.81	**0.87**	**0.97**	0.95	**0.96**	**0.93**
Ours	0.92/**0.81**	**0.87**	0.95	**0.97**	**0.96**	0.92

Fig. 10: Visual comparison on segmenting real images.

Fig. 11: Various kinds of failures in our model (foreground-mask pair).

However, we observe exceptional cases where the mask generator struggles in Fig. 11. We suggest one of the main reasons to be the ambiguity of the task itself. In CompCars [36], the road below the vehicles is often marked as foreground. It is a reasonable choice because the road is physically close to the vehicles. Using a minimal amount of human supervision for resolving such ambiguity would be a sensible research direction, e.g., specifying foreground or background by scribbles on one or a few images. In some cases, the mask misses a small portion of the object area. This might be because the composite image is natural enough, even if the mask is inappropriate. We hope that our success in the common datasets in GAN literature sheds light on foreground-aware image synthesis.

References

1. Abdal, R., Zhu, P., Mitra, N.J., Wonka, P.: Labels4free: Unsupervised segmentation using stylegan. In: Proceedings of the IEEE/CVF International Conference on Computer Vision. pp. 13970–13979 (2021)
2. Abdal, R., Zhu, P., Mitra, N.J., Wonka, P.: Styleflow: Attribute-conditioned exploration of stylegan-generated images using conditional continuous normalizing flows. ACM Transactions on Graphics (TOG) **40**(3), 1–21 (2021)
3. Achanta, R., Shaji, A., Smith, K., Lucchi, A., Fua, P., Süsstrunk, S.: Slic superpixels compared to state-of-the-art superpixel methods. IEEE transactions on pattern analysis and machine intelligence **34**(11), 2274–2282 (2012)
4. Alaluf, Y., Patashnik, O., Cohen-Or, D.: Restyle: A residual-based stylegan encoder via iterative refinement. In: Proceedings of the IEEE/CVF International Conference on Computer Vision. pp. 6711–6720 (2021)
5. Bielski, A., Favaro, P.: Emergence of object segmentation in perturbed generative models. Advances in Neural Information Processing Systems **32** (2019)
6. Choi, Y., Uh, Y., Yoo, J., Ha, J.W.: Stargan v2: Diverse image synthesis for multiple domains. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (2020)
7. Collins, E., Bala, R., Price, B., Susstrunk, S.: Editing in style: Uncovering the local semantics of gans. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 5771–5780 (2020)
8. Comaniciu, D., Meer, P.: Mean shift: A robust approach toward feature space analysis. IEEE Transactions on pattern analysis and machine intelligence **24**(5), 603–619 (2002)
9. Goodfellow, I., Pouget-Abadie, J., Mirza, M., Xu, B., Warde-Farley, D., Ozair, S., Courville, A., Bengio, Y.: Generative adversarial nets. Advances in neural information processing systems **27** (2014)
10. Gu, J., Liu, L., Wang, P., Theobalt, C.: Stylenerf: A style-based 3d-aware generator for high-resolution image synthesis. arXiv preprint arXiv:2110.08985 (2021)
11. Heusel, M., Ramsauer, H., Unterthiner, T., Nessler, B., Hochreiter, S.: Gans trained by a two time-scale update rule converge to a local nash equilibrium. Advances in neural information processing systems **30** (2017)
12. Ji, X., Henriques, J.F., Vedaldi, A.: Invariant information clustering for unsupervised image classification and segmentation. In: Proceedings of the IEEE/CVF International Conference on Computer Vision. pp. 9865–9874 (2019)
13. Karras, T., Aittala, M., Hellsten, J., Laine, S., Lehtinen, J., Aila, T.: Training generative adversarial networks with limited data. Advances in Neural Information Processing Systems **33**, 12104–12114 (2020)
14. Karras, T., Aittala, M., Laine, S., Härkönen, E., Hellsten, J., Lehtinen, J., Aila, T.: Alias-free generative adversarial networks. Advances in Neural Information Processing Systems **34** (2021)
15. Karras, T., Laine, S., Aila, T.: A style-based generator architecture for generative adversarial networks. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition. pp. 4401–4410 (2019)
16. Karras, T., Laine, S., Aittala, M., Hellsten, J., Lehtinen, J., Aila, T.: Analyzing and improving the image quality of stylegan. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition. pp. 8110–8119 (2020)
17. Kim, H., Choi, Y., Kim, J., Yoo, S., Uh, Y.: 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. pp. 852–861 (2021)

18. Lee, C.H., Liu, Z., Wu, L., Luo, P.: Maskgan: Towards diverse and interactive facial image manipulation. In: IEEE Conference on Computer Vision and Pattern Recognition (CVPR) (2020)
19. Lee, M.S., Shin, W., Han, S.W.: Tracer: Extreme attention guided salient object tracing network. arXiv preprint arXiv:2112.07380 (2021)
20. Melas-Kyriazi, L., Rupprecht, C., Laina, I., Vedaldi, A.: Finding an unsupervised image segmenter in each of your deep generative models. arXiv preprint arXiv:2105.08127 (2021)
21. Mescheder, L., Geiger, A., Nowozin, S.: Which training methods for gans do actually converge? In: International conference on machine learning. pp. 3481–3490. PMLR (2018)
22. Mildenhall, B., Srinivasan, P.P., Tancik, M., Barron, J.T., Ramamoorthi, R., Ng, R.: Nerf: Representing scenes as neural radiance fields for view synthesis. In: European conference on computer vision. pp. 405–421. Springer (2020)
23. Niemeyer, M., Geiger, A.: Giraffe: Representing scenes as compositional generative neural feature fields. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 11453–11464 (2021)
24. Ouali, Y., Hudelot, C., Tami, M.: Autoregressive unsupervised image segmentation. In: European Conference on Computer Vision. pp. 142–158. Springer (2020)
25. Patashnik, O., Wu, Z., Shechtman, E., Cohen-Or, D., Lischinski, D.: Styleclip: Text-driven manipulation of stylegan imagery. In: Proceedings of the IEEE/CVF International Conference on Computer Vision. pp. 2085–2094 (2021)
26. Richardson, E., Alaluf, Y., Patashnik, O., Nitzan, Y., Azar, Y., Shapiro, S., Cohen-Or, D.: Encoding in style: a stylegan encoder for image-to-image translation. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 2287–2296 (2021)
27. Shen, Y., Yang, C., Tang, X., Zhou, B.: Interfacegan: Interpreting the disentangled face representation learned by gans. IEEE transactions on pattern analysis and machine intelligence (2020)
28. Shen, Y., Zhou, B.: Closed-form factorization of latent semantics in gans. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 1532–1540 (2021)
29. Singh, K.K., Ojha, U., Lee, Y.J.: Finegan: Unsupervised hierarchical disentanglement for fine-grained object generation and discovery. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 6490–6499 (2019)
30. Tov, O., Alaluf, Y., Nitzan, Y., Patashnik, O., Cohen-Or, D.: Designing an encoder for stylegan image manipulation. ACM Transactions on Graphics (TOG) **40**(4), 1–14 (2021)
31. Van Gansbeke, W., Vandenhende, S., Georgoulis, S., Van Gool, L.: Unsupervised semantic segmentation by contrasting object mask proposals. In: Proceedings of the IEEE/CVF International Conference on Computer Vision. pp. 10052–10062 (2021)
32. Voynov, A., Babenko, A.: Unsupervised discovery of interpretable directions in the gan latent space. In: International conference on machine learning. pp. 9786–9796. PMLR (2020)
33. Voynov, A., Morozov, S., Babenko, A.: Object segmentation without labels with large-scale generative models. In: International Conference on Machine Learning. pp. 10596–10606. PMLR (2021)
34. Wah, C., Branson, S., Welinder, P., Perona, P., Belongie, S.: The caltech-ucsd birds-200-2011 dataset (2011)

35. Wu, Z., Lischinski, D., Shechtman, E.: Stylespace analysis: Disentangled controls for stylegan image generation. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 12863–12872 (2021)
36. Yang, L., Luo, P., Change Loy, C., Tang, X.: A large-scale car dataset for fine-grained categorization and verification. In: Proceedings of the IEEE conference on computer vision and pattern recognition. pp. 3973–3981 (2015)
37. Yang, Y., Bilen, H., Zou, Q., Cheung, W.Y., Ji, X.: Learning foreground-background segmentation from improved layered gans. In: Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision. pp. 2524–2533 (2022)
38. Yu, C., Wang, J., Peng, C., Gao, C., Yu, G., Sang, N.: Bisenet: Bilateral segmentation network for real-time semantic segmentation. In: Proceedings of the European conference on computer vision (ECCV). pp. 325–341 (2018)
39. Yu, F., Seff, A., Zhang, Y., Song, S., Funkhouser, T., Xiao, J.: Lsun: Construction of a large-scale image dataset using deep learning with humans in the loop. arXiv preprint arXiv:1506.03365 (2015)
40. Yüksel, O.K., Simsar, E., Er, E.G., Yanardag, P.: Latentclr: A contrastive learning approach for unsupervised discovery of interpretable directions. In: Proceedings of the IEEE/CVF International Conference on Computer Vision. pp. 14263–14272 (2021)
41. Zhang, K., Riegler, G., Snavely, N., Koltun, V.: Nerf++: Analyzing and improving neural radiance fields. arXiv preprint arXiv:2010.07492 (2020)
42. Zhu, J., Shen, Y., Zhao, D., Zhou, B.: In-domain gan inversion for real image editing. In: European conference on computer vision. pp. 592–608. Springer (2020)

Supplementary Materials for
FurryGAN: High Quality Foreground-aware Image Synthesis

Jeongmin Bae, Mingi Kwon, and Youngjung Uh[⋆]

Yonsei University
{jaymin.bae, kwonmingi, yj.uh}@yonsei.ac.kr

We provide the following supplementary materials:

A Uncurated visual comparison of Labels4Free and ours
B Choice of pseudo ground truth masks
C Quantitative results with alternative pseudo ground truth masks
D Examples of style mixing
E Additional details and visualization of the mask
F Results on unaligned datasets (LSUN-Church, LSUN-Horse, and CUB)
G User study on mask quality (between Lables4Free and ours)

A   Uncurated comparison

Fig. S7-S8 (located at the end for clear spacing) present uncurated visual comparisons between Labels4Free and ours on FFHQ and AFHQv2-Cat. The columns represent foreground images, alpha masks, and composite images with generated backgrounds. While Labels4Free often misses clothes and whiskers, our method produces more accurate and detailed masks, especially on hair, fur, and whiskers. Consistency between the generated masks and the actual foreground region in the composite image also demonstrates the superiority of our method.

B   Choice of pseudo ground truth masks

In this section, we provide the grounds for choosing TRACER (TE7) [19] to prepare pseudo ground truth masks over BiSeNet [38] (in Labels4Free [1]) and Mask R-CNN[1] (in PSeg [5]). As FFHQ do not have ground truth masks, we manually annotate ten images for the evaluation. The images are broadly chosen to cover various ages, genders, ethnic groups, and accessories. Fig. S1 shows the chosen images, annotated ground truths, and the pseudo ground truths from the methods. The quantitative comparison also reveals that TRACER achieves the best performance. Note that CelebAMask-HQ does not suffice to serve as the benchmark because BiSeNet is trained on it.

Fig. S2 further contrast the performance of the methods. On FFHQ, TRACER captures even hair while BiSeNet struggles. On AFHQv2-Cat, TRACER precisely captures even long fur on the ears and the top of the heads.

[⋆] Corresponding author
[1] https://github.com/facebookresearch/maskrcnn-benchmark

Fig. S1: **Qualitative comparison of masks.** We manually annotated ground truth masks (second row). TRACER produces masks very similar to the ground truth. BiSeNet also shows acceptable performance, but it often misclassifies the background as a foreground (3rd, 9th column) and vice versa (10th column). Mask R-CNN is relatively poor in quality, especially near the borders of the mask.

method	IoU(fg/bg)	mIoU	recall	precision	F1	Accuracy
Mask R-CNN	0.92/0.85	0.88	0.97	0.94	0.96	0.92
BiSeNet	0.98/0.96	0.97	0.99	**0.99**	**0.99**	0.98
TRACER	**0.99/0.97**	**0.98**	**1.00**	**0.99**	**0.99**	**0.99**

Table S1: **Quantitative comparison of predicted masks on the ten selected FFHQ images.** We evaluate the performance of the models with ten manually annotated ground truth masks.

C   Quantitative evaluation with alternative pseudo ground truth masks

In this section, we report quantitative results with other choices of generating pseudo ground truth masks: BiSeNet for FFHQ and Mask R-CNN for AFHQv2-Cat following Labels4Free[2]. Table S2 confirms the same rankings as the ones with TRACER; our method consistently outperforms the competitors in all settings.

---

[2] Labels4Free uses Mask R-CNN for LSUN-Cat.

(a) Examples of predicted mask on FFHQ   (b) Examples of predicted mask on AFHQv2-Cat

Fig. S2: **Further comparison of TRACER and other methods.** We evaluate each model on real images from FFHQ and AFHQv2-Cat datasets.

	ψ	method	IoU(fg/bg)	mIoU	recall	precision	F1	Accuracy
		PSeg	0.05/0.24	0.14	0.05	0.16	0.07	0.05
	1.0	L4F	0.86/0.70	0.78	0.93	0.92	0.92	0.86
FFHQ		Ours	**0.92/0.80**	**0.86**	**0.95**	**0.96**	**0.95**	**0.92**
(BiSeNet)		PSeg	0.01/0.23	0.12	0.01	0.04	0.01	0.01
	0.7	L4F	0.94/0.87	0.91	**0.96**	**0.99**	**0.97**	0.94
		Ours	**0.95/0.89**	**0.92**	**0.96**	**0.99**	**0.97**	**0.95**
		PSeg	0.06/0.21	0.13	0.06	0.17	0.07	0.06
	1.0	L4F	0.88/**0.72**	0.80	0.91	**0.97**	0.94	0.88
AFHQv2-Cat		Ours	**0.91/0.72**	**0.81**	**0.95**	0.95	**0.95**	**0.91**
(Mask R-CNN)		PSeg	0.01/0.17	0.09	0.01	0.13	0.01	0.01
	0.7	L4F	0.91/**0.77**	**0.84**	0.92	**0.98**	0.95	0.91
		Ours	**0.92/0.77**	**0.84**	**0.95**	0.96	**0.96**	**0.92**

Table S2: **Quantitative comparison of alpha masks on FFHQ and AFHQv2-Cat.** We use results of BiSeNet trained on CelebAMask-HQ as ground truth for FFHQ and results of Facebook's Detectron2 Mask R-CNN Model (R101-FPN) as ground truth for AFHQv2-Cat. We report the result with/without truncation trick (ψ=0.7, 1.0). The threshold for the alpha mask is 0.5 in ours and PSeg, and 0.9 in Labels4Free.

D   Style mixing

Our generator supports style mixing since it is based on StyleGAN2. As coarse style affects shape in StyleGAN2, the masks of the coarse source determine the masks of the mixed results in our generator (Fig. S3). Note that we do not use mixing regularization during the training.

Fig. S3: The two leftmost columns are source images denoted by A and B. The right side of the figure is the result of using the latent code of B instead of the latent code of A in the coarse (4^2-8^2), middle (16^2-32^2), and fine (64^2-256^2) layers, respectively. We demonstrate masked foreground images to show the changes in the foreground mask according to different style mixing. In addition, we provide the composite image and mask in the upper left corner of each image.

Fig. S4: **Architecture of the mask generator and the mask predictor.** Coarse and fine mask networks use the same structure shown in the upper right corner of (a). γ is defined in Eq. (3). For brevity, we omit the LeakyReLU activation function between the convolution layers of the right branch in (b).

E   Details about masks

In this section, we present the motivation for introducing fine masks and show additional mask visualizations. We assumed that the binarization loss (Eq. (6)) makes it difficult for the model to learn the matting-like details in the mask. These fine details are expected to occupy only a small part around the object boundary. Accordingly, we do not use binarization loss for the fine masks and use a very low threshold value for the inverse area loss (Eq. (9)).

We show some examples of coarse and fine masks in Fig. S5. As mentioned in Eq. (9), we penalize the area where the fine mask actually contributes to the final mask (the rightmost column of Fig. S5). Our generator can produce detailed alpha masks using the fine mask as needed. Finally, Fig. S4 illustrates architectures of the mask generator and the mask predictor.

(a) Foreground   (b) Mask   (c) Coarse Mask   (d) Fine Mask   (b) – (c)

Fig. S5: **Visualization of coarse and fine masks.** We generate a final mask by summing up coarse and fine masks and then clipping it to the range in [0,1]. Due to the clipping operation, the area where the fine mask contributes to the final mask is the difference between the final mask and the coarse mask.

F   Results on Unaligned Datasets

We also conducted training on unaligned datasets such as CUB and LSUN-Object. There are some changes in the training setting for this: 1) The coefficient of binarization loss is linearly reduced to 2.0 over the first 5K iterations. (default is 0.5). 2) We apply mask consistency loss after 5K iterations. 3) The average operation of the mask area loss is calculated for the mini-batch (not for each sample). 4) we set $\phi_1 = 0.2$ for LSUN-Object, and $\phi_1 = 0.1$ for CUB (Eq. (7)).

For LSUN-Object datasets, we use the first 100K images. We preprocess all datasets by center cropping and rescaling them to 256×256. Fig. S6 shows the results of selected samples for three unstructured datasets.

(a) CUB   (b) LSUN-Church   (c) LSUN-Horse

Fig. S6: Curated qualitative results on unaligned datasets.

G   User Study on Mask Quality

To further evaluate the mask performance of our model, we asked 50 participants to choose more precise masks between ours and Labels4Free. In Table S3 (a), we report the results for ten random matches of generated image-mask pair used in Fig. S7-S8. In Table S3 (b), we report the results for the quality of masks obtained through the inversion of 20 real images (CelebAMask-HQ). For real image segmentation, both models were trained on FFHQ. Our model outperforms Labels4Free in mask quality of generated images and segmentation results of real images.

Table S3: The reported values mean the preference rate of mask outputs from ours against Labels4Free.

	(a) Generated		(b) Real
	AFHQv2-Cat	FFHQ	CelebA-HQ
Labels4Free	15.8%	11.2%	11.8%
Ours	84.2%	88.8%	88.2%

Fig. S7: Uncurated qualitative comparison of image composition results on FFHQ, with truncation setting $\psi = 0.7$.

Fig. S8: Uncurated qualitative comparison of image composition results on AFHQ, with truncation setting $\psi = 0.7$.