Semantic Image Analogy with a Conditional Single-Image GAN

Jiacheng Li, Zhiwei Xiong, Dong Liu, Xuejin Chen, Zheng-Jun Zha
University of Science and Technology of China
zwxiong@ustc.edu.cn

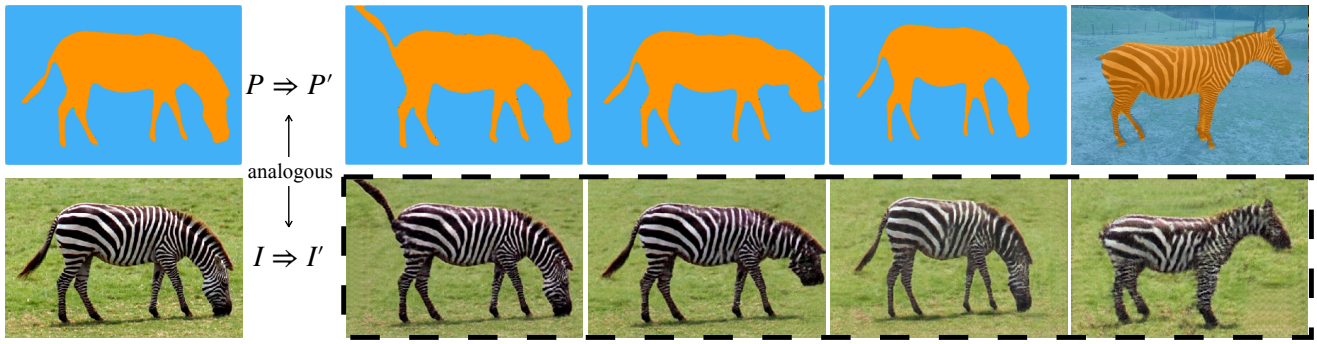

Figure 1: *Semantic Image Analogy*: given a source image I and its segmentation map P, along with another target segmentation map P', synthesizing a new image I' that matches the appearance of the source image as well as the semantic layout of the target segmentation. The transformations from P to P' and from I to I' are semantically "analogous". P' can be obtained by editing P (the first three cases) or from another image with a similar context (the last case).

ABSTRACT

Recent image-specific Generative Adversarial Networks (GANs) provide a way to learn generative models from a single image instead of a large dataset. However, the semantic meaning of patches inside a single image is less explored. In this work, we first define the task of *Semantic Image Analogy*: given a source image and its segmentation map, along with another target segmentation map, synthesizing a new image that matches the appearance of the source image as well as the semantic layout of the target segmentation. To accomplish this task, we propose a novel method to model the patch-level correspondence between semantic layout and appearance of a single image by training a single-image GAN that takes semantic labels as conditional input. Once trained, a controllable redistribution of patches from the training image can be obtained by providing the expected semantic layout as spatial guidance. The proposed method contains three essential parts: 1) a self-supervised training framework, with a progressive data augmentation strategy and an alternating optimization procedure; 2) a semantic feature translation module that predicts transformation parameters in the image domain from the segmentation domain; and 3) a semantics-aware patch-wise loss that explicitly measures the similarity of two images in terms of patch distribution. Compared with existing solutions, our method generates much more realistic results given arbitrary semantic labels as conditional input.

CCS CONCEPTS

• **Computing methodologies** → **Image manipulation**.

KEYWORDS

image analogies; generative adversarial network; semantic manipulation

ACM Reference Format:
Jiacheng Li, Zhiwei Xiong, Dong Liu, Xuejin Chen, Zheng-Jun Zha. 2020. Semantic Image Analogy with a Conditional Single-Image GAN. In *Proceedings of the 28th ACM International Conference on Multimedia (MM '20), October 12–16, 2020, Seattle, WA, USA.* ACM, New York, NY, USA, 9 pages. https://doi.org/10.1145/3394171.3413601

MM '20, October 12–16, 2020, Seattle, WA, USA

ACM ISBN 978-1-4503-7988-5/20/10…$15.00
https://doi.org/10.1145/3394171.3413601

1 INTRODUCTION

Generative models like Variational Autoencoders (VAEs) [21] and Generative Adversarial Networks (GANs) [12] have made great progress on modeling the distribution of natural images in a generative way. With additional signals such as class labels [27], text [44], edges [11], or segmentation maps [30, 39] as input, conditional generative models can generate photorealistic samples in a controllable manner, which is useful in a number of multimedia applications such as interactive design [10, 11, 30] and artistic style transfer [8, 43].

Specifically, segmentation maps provide dense pixel-level guidance to generative models and enable users to control the expected instances spatially [30, 39], which is much more flexible than image-level guidance like class labels [27] or styles [19]. Generally, a large training dataset is needed to map the segmentation labels to various patch appearance across the dataset. However, the appearance of

instances of a certain label in the generated images is limited to what that label looks like in the training dataset, which thus limits the generalization capability of these models in the wild.

Along the other line, recent efforts on image-specific GANs show the possibility to learn a generative model from the internal patch distribution of a single image. Progresses are made in a number of tasks, e.g., retargeting [35], super-resolution [4], unconditioned image generation and harmonization [34]. While these image-specific GANs are dataset independent and generate promising results, the semantic meaning of patches inside a single image remains less explored.

In this work, we try to combine the advantages from both worlds: with a conditional single-image GAN, we train a generative model that generates semantically controllable images through segmentation labels in the own context of a source image instead of external datasets. We show that a natural image is semantically self-contained and it is feasible to find the patch-level semantic correspondence between a single image and its segmentation map. We name this task *Semantic Image Analogy*, as a variant of Image Analogies [13] and define it as below.

Problem ("SEMANTIC IMAGE ANALOGY"): Given a source image I and its corresponding semantic segmentation map P, along with some additional semantic segmentation map P', synthesizing a new target image I' such that

$$P \Rightarrow P' :: I \Rightarrow I'.$$

As illustrated in Figure 1, the target image I' should match both the appearance of the source image I and the layout of the target segmentation P'. Different from Image Analogies [13] which learns a location-based filter shared between P to I and P' to I', our task setting aims to find an "analogous" transformation from I to I' in the same way transforming P to P'. Furthermore, we suggest to evaluate the quality of the generated images from *Semantic Image Analogy* models with two metrics: a patch-level distance and a semantic alignment score. The former restricts that the original image I is the only source for patches of the generated image I', while the latter enforces that the generated image I' must have an aligned semantic layout with the target segmentation map P'.

We cast the *Semantic Image Analogy* task as a patch-level distribution matching problem with the guidance of transformation in the semantic segmentation domain. To this end, we need to address three major challenges: the source of paired data for training a generative model from a single image, the condition scheme for providing guidance from the segmentation domain to the image domain, and the suitable supervision for the generated samples. To accomplish this task, we propose a novel method integrating the following three essential parts:

(1) We design a self-supervised training framework with a progressive data augmentation strategy. By alternating optimization with the augmented segmentation and the original one, we successfully train a conditional GAN from a single image, which generalizes well on unseen transformations.
(2) We design a Semantic Feature Translation module that translates the transformation parameters from the segmentation domain to the image domain.
(3) We design a semantics-aware *Patch Coherence Loss*, which encourages that the transformed image only contains patches from the source image. Together with the semantic alignment constraint, it enables our generator to produce realistic images with the target semantic layout.

In practice, we can either edit the source segmentation map P or provide another image with a similar context to obtain the target segmentation map P'. Our generator can then produce the semantic-aligned image I' from the source image I analogous to the way we obtain P' from P. Comparisons with existing methods show the superiority of our method in terms of both quantitative and qualitative evaluations. Thanks to our flexible task setting, the proposed method can easily extend to various applications including object removal [14, 41, 42], face editing [22], and sketch-to-image synthesis [11] for images in the wild.

2 RELATED WORK

2.1 Conditional GANs

GANs [12] have made a great success in image synthesis [5, 18, 19]. Conditional GANs synthesize images based on given conditions, which can be class labels [27], text [18], edges [11, 16], or semantic segmentation maps [30, 39]. Isola et al. show the power of conditional GANs on generating images given dense condition signals including sketches and segmentation maps [16]. Wang et al. extend the above framework with a coarse-to-fine generator and multi-scale discriminators to generate images with high-resolution details [39]. Park et al. propose a spatially-adaptive normalization technique (SPADE) [30] that uses the semantic maps to predict affine transformation parameters for modulating the activations in normalization layers. Liu et al. extend SPADE by introducing the conditional convolution block, which predicts convolutional kernels, and the FPSE discriminator which injects segmentation maps into discriminators for semantic alignment of target labels and generated images [24]. Bau et al. apply a generative image prior to semantic photo manipulation by adapting its image prior to the statistics of the source image [3]. Albahar and Huang adopt the FiLM [32] modulation layer into a spatial-varying manner and allow a bidirectional condition between the guidance and the source image [1]. Exemplar-based models [25, 38] align the translated results with the exemplar domain both in style and semantic meaning. However, these models are limited to the semantic meaning of the training dataset, which are difficult to generalize to images in the wild.

2.2 Single-Image GANs

Recently, image-specific GANs reveal the power of image priors learned from the internal similarity of a single image instead of a large external dataset [46]. InGAN [35] defines the transformation of resizing and trains a generative model to capture the internal patch statistics in the task of retargeting. SinGAN [34] utilizes a multi-stage training scheme for unconditioned image generation which produces images of arbitrary size from noise. KernelGAN [4] uses a deep linear generator and constrains it to learn an image-specific degradation kernel for blind super-resolution. In our method, we train a single-image GAN with the segmentation map as dense conditional input, which explores the semantic meaning of patches inside a single image.

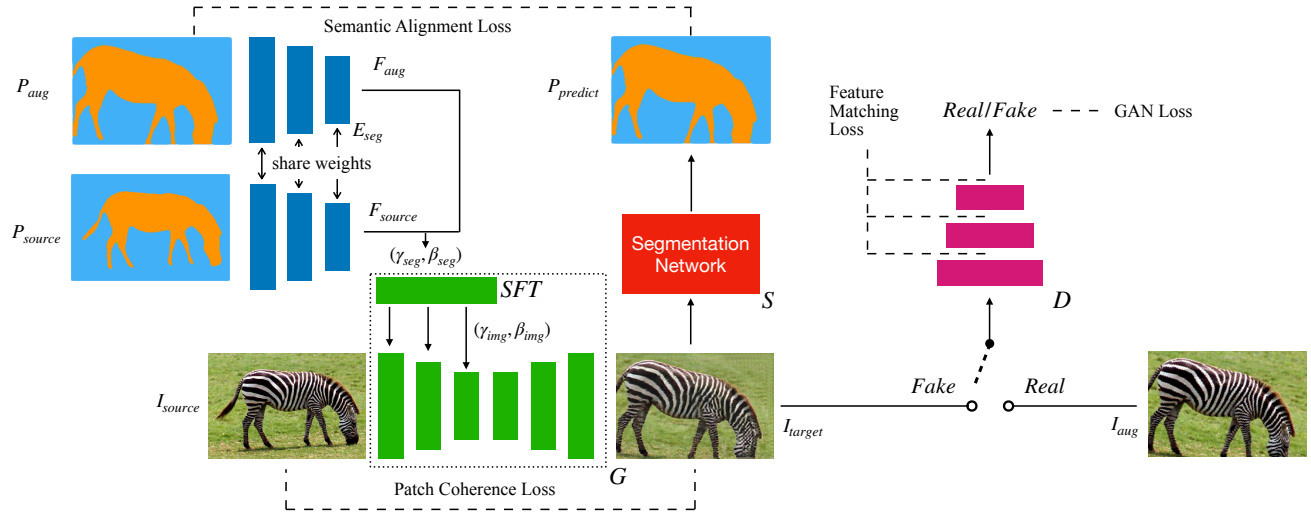

Figure 2: The proposed self-supervised training framework for our conditional GAN. At first, augmentation operations are applied on the source image I_{source} and the source segmentation map P_{source} to obtain I_{aug} and P_{aug} as psuedo lables. Then, the encoder E_{seg} extracts features F_{source} and F_{aug} from P_{source} and P_{aug}. The Semantic Feature Translation (SFT) module predicts transformation parameters $(\gamma_{img}, \beta_{img})$ from F_{source} and F_{aug}. Finally, the generator G maps I_{source} to the fake image I_{target} under the guidance of $(\gamma_{img}, \beta_{img})$. At the same time, the discriminator D tries to distinguish I_{aug} and I_{target}. The auxiliary classifier S predicts the semantic segmentation label $P_{predict}$ of I_{target}. The *Semantic Alignment Loss* between P_{aug} and $P_{predict}$ and the *Patch Cohernece Loss* between I_{source} and I_{target} are calculated for self-supervision.

2.3 Image Analogies

Image Analogies is first introduced in [13], where a new "analogous" image B that relates to B' in the same way as A relates to A'. The target image B is computed by searching the best-matching patches for each image location between A' and B', and then implying the patch appearance for B from the corresponding patch A. Cheng et al. extend the above algorithm with Markov Random Field and image quilting to ensure the global and local consistency [9]. Reed et al. study the inferring of the relationship like rotation between a pair of images with the constraint inspired by the Natural Language Model in synthetic datasets of geometry shapes and avatars [33]. Liao et al. propose Deep Image Analogy [23] that utilizes a pre-trained Network as the feature extractor and use PatchMatch [2] to find feature-level correspondence between image A and B', and then produce A' and B which transfers the visual attributes of original images. Park et al. replace the CNN regression with the Gaussian process to adjust the feature vectors which are used in creating a filtered image [29]. In the task setting of *Semantic Image Analogy*, we aim to learn an "analogous" relationship between transformations, instead of a shared filter as in Image Analogies [13].

3 SEMANTIC IMAGE ANALOGY

3.1 Problem Formulation

We formulate the task setting of *Semantic Image Analogy* as follows. Given a source image I_{source} and its segmentation map P_{source}, along with another target segmentation map P_{target}, synthesizing a new target image I_{target} such that the transformation from I_{source} to I_{target}, denoted as $\mathcal{T}_{img}$, is "analogous" as the transformation from P_{source} to P_{target}, denoted as $\mathcal{T}_{seg}$. We model this process in a generative way, *i.e.*, we aim to find an optimal generator G such that

$$\begin{aligned} I_{target} &= \mathcal{T}_{img}(I_{source}) = G(I_{source}|\mathcal{T}_{seg}) \\ s.t. \quad P_{target} &= \mathcal{T}_{seg}(P_{source}). \end{aligned} \tag{1}$$

Specifically, we model $\mathcal{T}_{img}$ with a conditional single-image GAN, where the generator G maps I_{source} to I_{target} given $\mathcal{T}_{seg}$ as conditional input.

To translate transformation $\mathcal{T}_{seg}$ in the segmentation domain to $\mathcal{T}_{img}$ in the image domain, we first extract features from segmentation maps and images with two Convolutional Neural Networks (CNNs), by assuming that transformation $\mathcal{T}^f$ is linear in the feature space $\mathcal{F}$ of either segmentation or image domain. We define $\mathcal{T}^f$ as

$$\mathcal{T}^f : y = \gamma x + \beta, \mathcal{T}^f \subset \{\mathcal{F}^{W\times H\times C} \mapsto \mathcal{F}^{W\times H\times C}\}, \tag{2}$$

where W, H, and C denote the size of feature tensors x and y in either segmentation or image domain. γ can be regarded as a scaling factor and β a shifting factor. Next, we introduce a unique Semantic Feature Translation (SFT) module, which converts the scaling and shifting factors from the feature space of the segmentation domain to that of the image domain

$$(\gamma_{img}, \beta_{img}) = SFT(\gamma_{seg}, \beta_{seg}). \tag{3}$$

The transformation parameters $(\gamma_{img}, \beta_{img})$ are then applied to the features of I_{source} to obtain the features of I_{target}. Finally, the features of I_{target} are mapped back to the image domain to produce I_{target} with another CNN.

In the rest of this section, we describe three essential parts of the proposed method in detail: the self-supervised learning framework, the SFT module, and the loss terms.

3.2 Self-supervised Learning Framework

As shown in Figure 2, we design a self-supervised framework for training a conditional GAN from a single image. During each training iteration, we apply random augmentation like flipping and rotation on I_{source} and P_{source} to obtain a pair of augmented image I_{aug} and segmentation map P_{aug} as pseudo labels. We increase the randomness of augmentation progressively during the training process. Since our generator is an endomorphism, the source image should be well reconstructed when P_{target} is the same as P_{source}. Thus, we split our optimization procedure into two alternating modes: sampling and reconstruction. In the sampling mode, the generator takes the augmented transformation as guidance to produce a target image with the same appearance as I_{aug} and the same semantic layout as P_{aug}. In the reconstruction mode, the generator tries to reconstruct the source image, given that its conditional input $\mathcal{T}_{seg}$ is an identity mapping.

3.2.1 The Alternating Optimization. In the sampling mode, given a source image I_{source} and its segmentation map P_{source}, we first perform random augmentation to obtain I_{aug} and P_{aug}. Next, P_{source} and P_{aug} are fed into the same encoder E_{seg} to extract features F_{source} and F_{aug} respectively. Then, the SFT module predicts the transformation parameters $(\gamma_{img}, \beta_{img})$ in the image domain from extracted feature tensors F_{source} and F_{aug}. Finally, the generator maps I_{source} into I_{target} under the guidance of $(\gamma_{img}, \beta_{img})$. The discriminator D takes I_{aug} as a real sample and I_{target} as a fake sample. Meanwhile, the generated image I_{target} is also fed into the auxiliary classifier S to predict its segmentation map $P_{predict}$. In the reconstruction mode, we set P_{source} and P_{aug} to be the same. The transformation $\mathcal{T}_{seg}$ becomes an identity mapping and the generator learns to reconstruct the source image.

3.2.2 Generator and Encoder. Our generator G adopts the Encoder-Decoder architecture. The encoder E_{seg} uses the same structure as the encoder part of the generator G. E_{seg} takes segmentation maps P_{source} and P_{aug} as input to extract their features for inferring transformation parameters $(\gamma_{seg}, \beta_{seg})$ in the segmentation domain, which are then translated to $(\gamma_{img}, \beta_{img})$ in the image domain by the SFT module. In each downsampling stage of the encoder part in the generator G, we apply an affine transformation with parameters $(\gamma_{img}, \beta_{img})$ to image features extracted from I_{source}. Then the decoder decodes these transformed image features to generate the target image I_{target}.

3.2.3 Discriminator and Auxiliary Classifier. Our discriminator D is a fully convolutional PatchGAN [16], which predicts a score map to distinguish real and fake samples. We use a light version of the DeepLab V3 [7] architecture in our auxiliary classifier S for semantic segmentation. In the sampling mode, S is trained with the augmented image I_{aug} as input and the corresponding segmentation map P_{aug} as label. Then, we predict the segmentation map $P_{predict}$ of the generated image I_{target} with S.

3.3 Semantic Feature Translation

We explicitly translate the transformation parameters from the segmentation domain to the image domain through the SFT module, as shown in Figure 3. We model the transformation $\mathcal{T}_{seg}$ from P_{source} to P_{target} as a linear transformation at the feature level. Thus, after extracting features F_{source} and F_{aug} from segmentation maps P_{source} and P_{aug} via the encoder E_{seg}, we perform element-wise division and difference on the feature tensors and define a feature scaling tensor and a feature shifting tensor as

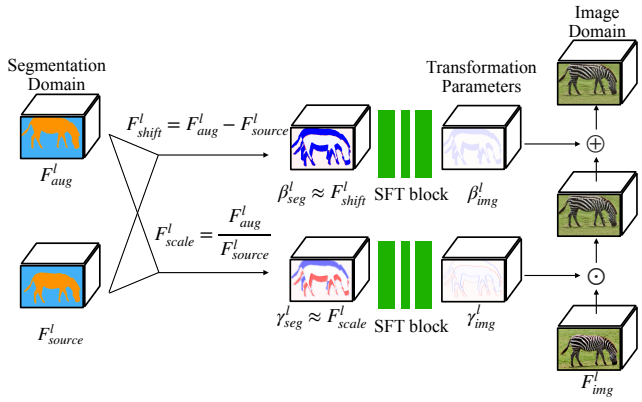

Figure 3: The Semantic Feature Translation (SFT) module. The scaling factor γ_{img} is learned from the result of element-wise division of the augmented segmentation features F_{aug} and the source segmentation features F_{source}, while the shifting factor β_{img} is learned from the result of element-wise difference. Then, we apply an affine transformation with $(\gamma_{img}, \beta_{img})$ on features in the image domain.

$$F_{scale} = \frac{F_{aug}}{F_{source}}, F_{shift} = F_{aug} - F_{source}. \tag{4}$$

We use F_{scale} and F_{shift} to approximate the scaling factor γ_{seg} and the shifting factor β_{seg} of the transformation $\mathcal{T}_{seg}$. Then Eq. (3) becomes

$$(\gamma_{img}, \beta_{img}) = SFT(F_{scale}, F_{shift}). \tag{5}$$

We model the translation process from the segmentation domain to the image domain with two SFT blocks. The scaling factor γ_{img} is translated from F_{scale}, while the shifting factor β_{img} is from F_{shift}. Following [32] and [1], we apply an affine transformation with parameters $(\gamma_{img}, \beta_{img})$ in the generator G. For the l-th downsampling stage in G, the output feature tensor is computed from the input one as

$$F^{l+1}_{img} = DS^l(\gamma^l_{img} \frac{F^l_{img} - \text{mean}\left(F^l_{img}\right)}{\text{std}\left(F^l_{img}\right)} + \beta^l_{img}), \tag{6}$$

where the transformation parameters $(\gamma^l_{img}, \beta^l_{img})$ are learned from features of the l-th downsampling stage in E_{seg} and DS^l denotes the convolutional layers of the l-th downsampling stage.

3.4 Loss Terms

According to the task setting of *Semantic Image Analogy*, the generated image I_{target} should satisfy the following requirements: 1) homogeneous appearance with the source image I_{source}, and 2) aligned semantic layout with the target segmentation map P_{target}.

Thus, we introduce 1) a *Patch Coherence Loss* that measures the similarity of appearance between I_{target} and I_{source}, and 2) a *Semantic Alignment Loss* that measures the consistency between P_{target} and $P_{predict}$ which is predicted by the auxiliary classifier S from I_{target}. Next, we describe the constraints imposed in the sampling mode and the reconstruction mode respectively.

3.4.1 Constraints in Sampling Mode. Inspired by [36], we propose a *Patch Coherence Loss* to measure the similarity of appearance between the generated image I_{target} and the source image I_{source}. This constraint will penalize the generator G if it generates undesired patches not found in I_{source}. It is defined as the average of lower bounds of patch distance between I_{target} and I_{source}

$$\mathcal{L}_{patch}(G) = \frac{1}{N_{target}} \sum_{V \subset G(I_{source})} \min_{U \subset I_{source} \& U_{class}=V_{class}} d(V,U), \tag{7}$$

where N_{target} is the number of patches inside image I_{target}, U_{class} and V_{class} denote the segmentation labels of patches P and Q, and $d(\cdot)$ is a distance metric.

Our *Patch Coherence Loss* releases the location dependence of pixel-wise distances. Instead, we regard an image as a bag of visual features. For each patch V from I_{target}, we run a Nearest-Neighbour search to find the closest patch U with the same class label from I_{source} and then take the average of their distance. Unlike [36], we only search in the area with the same class label as V, which makes our search process semantics-aware.

On the other hand, we use an auxiliary classifier S to predict the segmentation map of the generated image I_{target}. Then we calculate the cross-entropy (CE) loss between the predicted segmentation map and the augmented one P_{aug}. The *Semantic Alignment Loss* for G is defined as

$$\mathcal{L}_{seg}(G) = CE(P_{aug}, S(G(I_{source}))). \tag{8}$$

We train the auxiliary classifier S with the augmented image I_{aug} as input and its segmentation map P_{aug} as label along with the generator and the discriminator in the sampling mode, using the following loss function

$$\mathcal{L}(S) = CE(P_{aug}, S(I_{aug})). \tag{9}$$

In addition, we use the Least-Square GAN loss [26] $\mathcal{L}_{GAN}(G,D)$ as the adversarial constraint, and take the features from the discriminator to calculate the feature-matching loss $\mathcal{L}_{fm}(G,D)$ [17] between the augmented image I_{aug} and the generated image I_{target}. To summarize, in the sampling mode, our total loss is

$$\begin{aligned}\mathcal{L}_{total}^{sample} = \mathcal{L}_{patch}(G) + \lambda_{seg}\mathcal{L}_{seg}(G) \\ + \lambda_{GAN}\mathcal{L}_{GAN}(G,D) + \lambda_{fm}\mathcal{L}_{fm}(G,D),\end{aligned} \tag{10}$$

where λ_{seg}, λ_{GAN}, and λ_{fm} are tradeoff parameters.

3.4.2 Constraints in Reconstruction Mode. In the reconstruction mode, we use the L_1 loss to measure the reconstruction quality of the generator as

$$\mathcal{L}_{rec}(G) = \|I_{source}, G(I_{source}|\mathcal{I})\|, \tag{11}$$

where $\mathcal{I}$ denotes the identity mapping.

With P_{source} and P_{aug} being the same, the feature scaling tensor F_{scale} would be 1 and the feature shifting tensor F_{shift} would be 0 at every location. Thus, we enforce the transformation parameters γ_{img} and β_{img} to be **1** and **0** in the image domain. This constraint encourages minimal changes for feature tensors of I_{source} in the generator. We call it *Fixed-Point Loss* since the source image is expected to be unchanged by G when the condition input $\mathcal{T}_{seg}$ is an identity mapping. This loss for the SFT module is defined as

$$\mathcal{L}_{fp}(SFT) = \|\gamma_{img} - 1\| + \|\beta_{img}\|. \tag{12}$$

We also use $\mathcal{L}_{GAN}(G,D)$ as the adversarial constraint. To summarize, in the reconstruction mode, our total loss is

$$\begin{aligned}\mathcal{L}_{total}^{rec} = \mathcal{L}_{rec}(G) + \lambda_{fp}\mathcal{L}_{fp}(SFT) \\ + \lambda_{GAN}\mathcal{L}_{GAN}(G,D),\end{aligned} \tag{13}$$

where λ_{fp} and λ_{GAN} are tradeoff parameters.

4 EXPERIMENTS AND RESULTS

4.1 Implementation Details

We implement our framework based on Pytorch [31]. The generator G is an Encoder-Decoder structure with 3 downsample blocks and 3 upsample blocks. The encoder E_{seg} shares the same structure as the encoder part of the generator G. The discriminator D is a PatchGAN [16] with 3 downsample blocks. Each block contains a 3×3 convolutional layer with stride 1, and a 4×4 convolutional layer or a transposed convolutional layer with stride 2 for downsampling or upsampling. The starting channel number is 32 and we double it during downsampling. Spectral Normalization [28], Batch Normalization [15], and Leaky ReLU activation are used in every block for encouraging stability. The auxiliary classifier S is a light version of DeepLab V3 [7] with 1/4 number of channels compared to the original version. Similar to [1], the layer in SFT blocks is composed of a bottleneck layer with 1×1 convolutions whose number of channels is half of the input features.

We train our model in the reconstruction mode once every 10 iterations and in the sampling mode otherwise. During a single iteration, we optimize the discriminator once and the generator 10 times. For augmentation, we apply random flip, resize, rotation, and crop operations to the source image and its segmentation map with the same seed. We increase the randomness of these operations linearly as the training steps. This progressive strategy helps the encoder learns the appearances of the source image in the early iterations of training.

In all experiments, we set the tradeoff parameters in Eq. (10) and Eq. (13) as 1.0. We use the ADAM [20] optimizer with a learning rate of 0.0005. For the *Patch Coherence Loss*, we find empirically that the features from a pre-trained VGG network [37] produce good results, although other feature descriptors are also applicable. We then calculate the L_1 distance between the features of two patches. We randomly choose 10 patches and 5 patches from the first two stages of VGG as V in Eq. (7), *i.e.*, N_{target} is set to 15. We train our model above 2k iterations, which takes about 1-3 days on an Nvidia Titan XP GPU, depending on the resolution of the source image.

4.2 Quantitative Results

We apply the proposed *Semantic Image Analogy* on images from different datasets, including COCO-Stuff [6], ADE20K [45], CelebAMask-HQ [22] and the Web. The results of our method along with methods

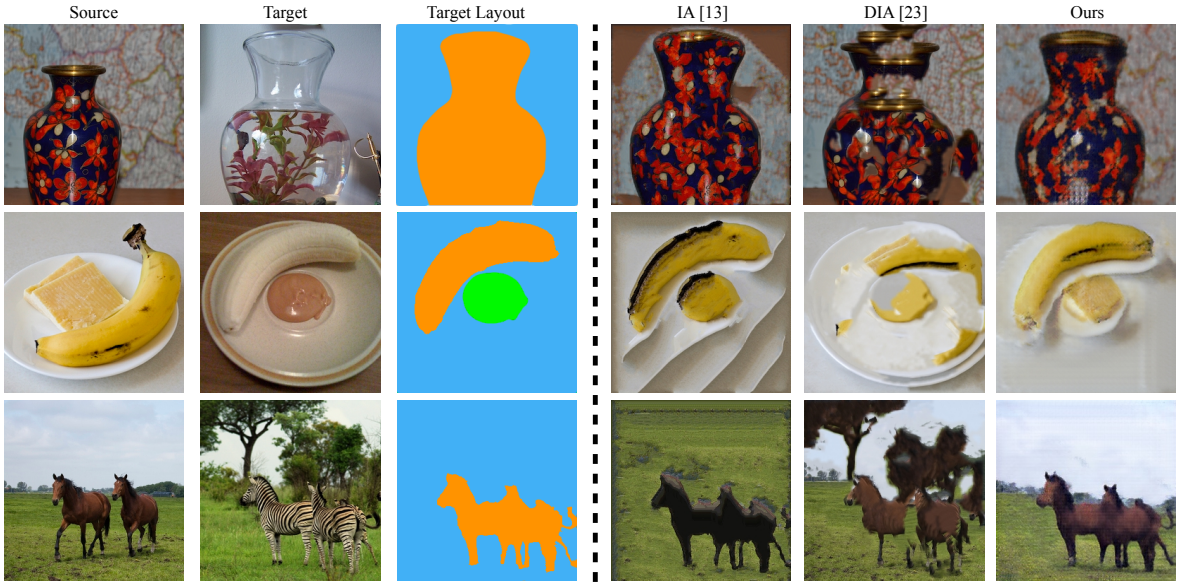

Figure 4: Visual quality comparison with IA [13] and DIA [23] using the segmentation map of anther image as target layout. Image sources: COCO-Stuff [6] dataset.

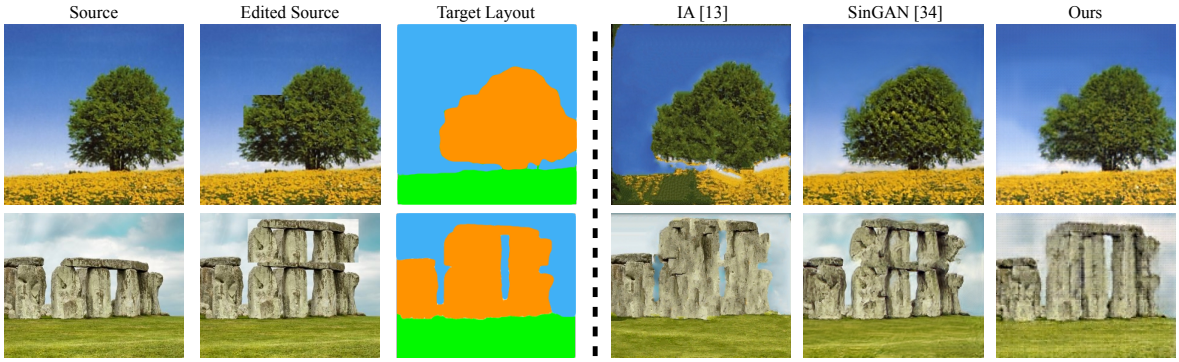

Figure 5: Visual quality comparison with IA [13] and SinGAN [34] when applying copy-and-paste operations on the segmentation maps. Image sources: the Web.

in comparison are evaluated in the following two aspects: 1) appearance similarity between the source image and the target image; and 2) semantic consistency of the target image with the target segmentation map.

Table 1: Quantitative comparisons of appearance similarity from user study and semantic consistency measured by pixel accuracy and mean IOU.

	IA [13]	DIA [23]	Ours
Avg. User Ranking ↓	2.26	2.005	1.735
Pixel-wise Accuracy ↑	43.9	41.9	54.0
Mean IOU ↑	44.0	38.7	45.7

4.2.1 User study. To evaluate the appearance similarity of the generated image to the source image, we conduct a user study in the following way. We randomly select 10 pairs of images with the same class labels from the COCO-Stuff [6] dataset. For each pair, with one image as source and the other for providing the target layout, we transfer the source image to the layout of the other image using our method, Image Analogies (IA) [13] and Deep Image Analogy (DIA) [23]. IA and DIA are the two most related works to ours. Note that DIA needs a pair of images as source and target, while our method and IA only need one source image and two segmentation maps. We display the results in random order and ask 20 users to rank the appearance similarity with the source image as reference. Then we calculate the average ranking of each method across all images and users. Table 1 shows the superiority of the proposed method against the two competitors.

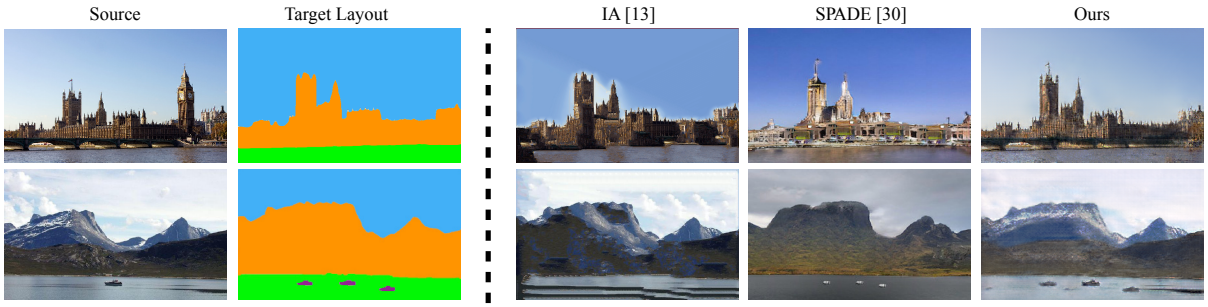

Figure 8: Visual quality comparison with IA [13] and SPADE [30]. The images are from the COCO-Stuff [6] dataset and the ADE20K [45] dataset.

Figure 9: Semantic manipulation results. We can manipulate the image by moving, resizing, or changing shapes of instances in the source segmentation map. The images are from the Web.

4.2.2 Pixel-wise accuracy and mean IOU. To evaluate the semantic consistency of the generated images with the target segmentation map, we use Detectron2's panoptic segmentation model [40] to predict the segmentation maps of generated images and then calculate pixel-wise accuracy and mean Intersection-over-Union (mIOU) with the target segmentation map [24, 30]. The images for evaluation are the same as those in the user study. As shown in Table 1, the proposed method achieves the highest accuracy.

4.3 Qualitative Results

4.3.1 Comparison to previous image analogies. Our task setting of *Semantic Image Analogy* is closet to that of IA [13], in which the target segmentation map can be arbitrary. The other closely related work, DIA [23], requires a pair of images as source and target. For a fair comparison with DIA, we use the segmentation map from this "paired image" as the target layout of our method and IA. As shown in Figure 4, our method produces both natural and semantic aligned results, while DIA produces unrealistic results when the source image and the target image are not semantically similar and IA tends to fill the changed instance with repeated textures.

4.3.2 Comparison to single-image GANs. We also compare our method with the versatile single-image generative model SinGAN [34], especially its editing application. To obtain a fair comparison with SinGAN, we only apply the "copy-and-paste" operation on the source segmentation map to produce the target segmentation map. Meanwhile, the same operation is applied to the source image to produce the edited source image. Following the settings in SinGAN [34], we first train a single-image generative model on the source image and then inject a downsampled version of the edited source image into the early coarse scales of the trained model. As shown in Figure 5, the SinGAN editing often changes the unedited area and produces undesired textures, or simply blurs the pasted objects, without consideration of the semantic structure, which leads to a very similar version of the edited result. In contrast, our method resembles the patches from the source image according to the guidance of the target semantic layout and generates semantically meaningful regions. As an accompanying comparison, IA [13] often produces averaging textures when the area with the same segmentation label is relatively large.

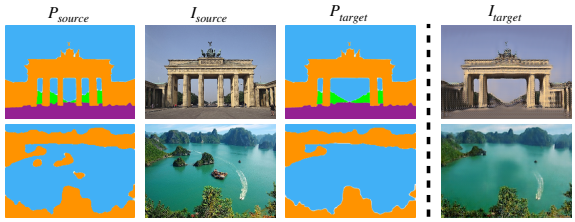

Figure 10: Object removal. The images are from the Web.

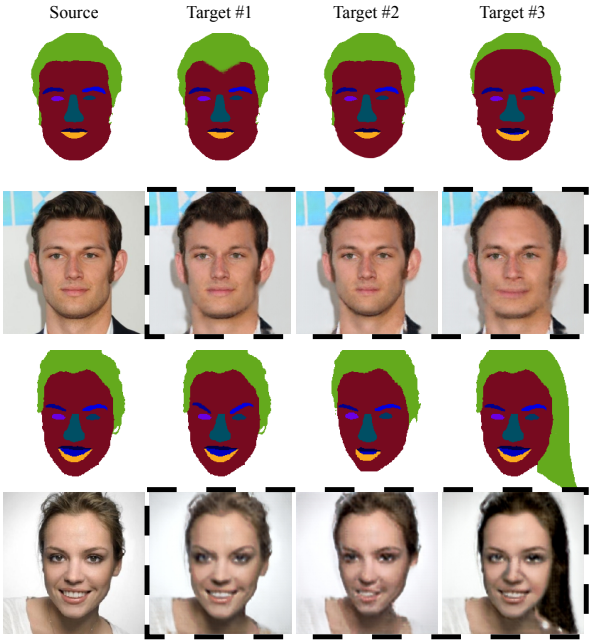

Figure 11: Face editing. The images are from the CelebAMask-HQ [22] dataset.

4.3.3 Comparison to conditional GANs. To compare with conditional image synthesis models which convert segmentation maps into images leveraging an external dataset, we inference the SPADE model [30], which is trained on landscape images from Flickr, with the target segmentation map as input. As shown in Figure 8, while the generated results of this conditional model are semantically consistent with the target segmentation, their content is limited to the training dataset and loses the appearance of the source image. Our method produces images that are faithful to the source images in terms of appearance and semantically aligned with the target segmentations. As an accompanying comparison, IA [13] fails to preserve the local appearance of changed instances.

4.3.4 Semantic manipulation results. Our method enables semantic manipulation of images through their segmentation maps. We can either move, resize, or remove instances in the source segmentation map to obtain the target layout. As shown in Figure 9, our method produces quality results with arbitrary semantic changes while the local appearance of the changed instance is well preserved.

Figure 12: Sketch-to-image synthesis. The images are from the Pix2pix work [16].

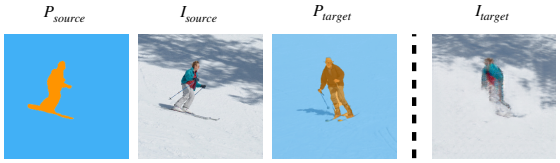

Figure 13: Failure case when a relatively small object has unique fine structures.

4.4 Applications

Our flexible task setting of *Semantic Image Analogy* enables various applications. Thanks to the dense conditional input, we can resemble the patches inside the image with pixel-level control. In Figure 10, 11 and 12, we show three applications of our method, including 1) object removal, where we can easily remove the undesired object by modifying the class labels in the segmentation map into the background class, 2) face editing, where we can edit facial images by changing the shapes of face components in the segmentation map, and 3) sketch-to-image synthesis, where we can use other spatial conditions like edge maps as conditional input.

4.5 Limitations

Our method suffers in scenarios where a relatively small object has unique fine structures. The SFT module may not be able to capture the unique appearance within the limited segmentation label. In Figure 13, we show such a case where the skier is not well reconstructed after pose change, especially the ski poles and the ski boards with fine structures.

5 CONCLUSION

In this paper, we define the *Semantic Image Analogy* task and propose a self-supervised framework that learns the semantically meaningful dense correspondences between an image and its segmentation map. As demonstrated by extensive experiments and applications, our model generates quality results with dense control of the spatial condition in the context provided by a single image, which can be hardly achieved with existing models.

ACKNOWLEDGMENTS

We acknowledge funding from National Key R&D Program of China under Grant 2017YFA0700800 and National Natural Science Foundation of China under Grants 61671419, U19B2038 and 61620106009.

REFERENCES

[1] Badour Albahar and Jia-Bin Huang. 2019. Guided Image-to-Image Translation With Bi-Directional Feature Transformation. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*. 9015–9024.

[2] Connelly Barnes, Eli Shechtman, Adam Finkelstein, and Dan B. Goldman. 2009. PatchMatch: a randomized correspondence algorithm for structural image editing. *ACM Trans. Graph.* 28, 3 (2009), 24.

[3] David Bau, Hendrik Strobelt, William S. Peebles, Jonas Wulff, Bolei Zhou, Jun-Yan Zhu, and Antonio Torralba. 2019. Semantic photo manipulation with a generative image prior. *ACM Trans. Graph.* 38, 4 (2019), 59:1–59:11.

[4] Sefi Bell-Kligler, Assaf Shocher, and Michal Irani. 2019. Blind Super-Resolution Kernel Estimation using an Internal-GAN. In *Advances in Neural Information Processing Systems 32: Annual Conference on Neural Information Processing Systems (NeurIPS)*. 284–293.

[5] Andrew Brock, Jeff Donahue, and Karen Simonyan. 2019. Large Scale GAN Training for High Fidelity Natural Image Synthesis. In *Proceedings of the 7th International Conference on Learning Representations (ICLR)*.

[6] Holger Caesar, Jasper R. R. Uijlings, and Vittorio Ferrari. 2018. COCO-Stuff: Thing and Stuff Classes in Context. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*. 1209–1218.

[7] Liang-Chieh Chen, Yukun Zhu, George Papandreou, Florian Schroff, and Hartwig Adam. 2018. Encoder-Decoder with Atrous Separable Convolution for Semantic Image Segmentation. In *Proceedings of the European Conference on Computer Vision (ECCV)*, Vol. 11211. 833–851.

[8] Yang Chen, Yu-Kun Lai, and Yong-Jin Liu. 2018. CartoonGAN: Generative Adversarial Networks for Photo Cartoonization. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*. 9465–9474.

[9] Li Cheng, S. V. N. Vishwanathan, and Xinhua Zhang. 2008. Consistent image analogies using semi-supervised learning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*.

[10] Haoye Dong, Xiaodan Liang, Yixuan Zhang, Xujie Zhang, Zhenyu Xie, Bowen Wu, Ziqi Zhang, Xiaohui Shen, and Jian Yin. 2020. Fashion Editing With Adversarial Parsing Learning. *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 8120–8128.

[11] Arnab Ghosh, Richard Zhang, Puneet K. Dokania, Oliver Wang, Alexei A. Efros, Philip H. S. Torr, and Eli Shechtman. 2019. Interactive Sketch & Fill: Multiclass Sketch-to-Image Translation. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*. 1171–1180.

[12] Ian J. Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron C. Courville, and Yoshua Bengio. 2014. Generative Adversarial Nets. In *Advances in Neural Information Processing Systems 27: Annual Conference on Neural Information Processing Systems (NeurIPS)*. 2672–2680.

[13] Aaron Hertzmann, Charles E. Jacobs, Nuria Oliver, Brian Curless, and David Salesin. 2001. Image analogies. In *Proceedings of the 28th Annual Conference on Computer Graphics and Interactive Techniques*. 327–340.

[14] Satoshi Iizuka, Edgar Simo-Serra, and Hiroshi Ishikawa. 2017. Globally and locally consistent image completion. *ACM Trans. Graph.* 36, 4 (2017), 107:1–107:14.

[15] Sergey Ioffe and Christian Szegedy. 2015. Batch Normalization: Accelerating Deep Network Training by Reducing Internal Covariate Shift. In *Proceedings of the 32nd International Conference on Machine Learning (ICML)*, Vol. 37. 448–456.

[16] Phillip Isola, Jun-Yan Zhu, Tinghui Zhou, and Alexei A. Efros. 2017. Image-to-Image Translation with Conditional Adversarial Networks. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*. 5967–5976.

[17] Justin Johnson, Alexandre Alahi, and Li Fei-Fei. 2016. Perceptual Losses for Real-Time Style Transfer and Super-Resolution. In *Proceedings of the European Conference on Computer Vision (ECCV)*, Vol. 9906. 694–711.

[18] Tero Karras, Timo Aila, Samuli Laine, and Jaakko Lehtinen. 2018. Progressive Growing of GANs for Improved Quality, Stability, and Variation. In *Proceedings of the 6th International Conference on Learning Representations (ICLR)*.

[19] Tero Karras, Samuli Laine, and Timo Aila. 2019. A Style-Based Generator Architecture for Generative Adversarial Networks. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*. 4401–4410.

[20] Diederik P. Kingma and Jimmy Ba. 2015. Adam: A Method for Stochastic Optimization. In *Proceedings of the 3rd International Conference on Learning Representations (ICLR)*.

[21] Diederik P. Kingma and Max Welling. 2014. Auto-Encoding Variational Bayes. In *Proceedings of the 2nd International Conference on Learning Representations (ICLR)*.

[22] Cheng-Han Lee, Ziwei Liu, Lingyun Wu, and Ping Luo. 2020. MaskGAN: Towards Diverse and Interactive Facial Image Manipulation. *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 5549–5558.

[23] Jing Liao, Yuan Yao, Lu Yuan, Gang Hua, and Sing Bing Kang. 2017. Visual attribute transfer through deep image analogy. *ACM Trans. Graph.* 36, 4 (2017), 120:1–120:15.

[24] Xihui Liu, Guojun Yin, Jing Shao, Xiaogang Wang, and Hongsheng Li. 2019. Learning to Predict Layout-to-image Conditional Convolutions for Semantic Image Synthesis. In *Advances in Neural Information Processing Systems 32: Annual Conference on Neural Information Processing Systems (NeurIPS)*. 568–578.

[25] Liqian Ma, Xu Jia, Stamatios Georgoulis, Tinne Tuytelaars, and Luc Van Gool. 2019. Exemplar Guided Unsupervised Image-to-Image Translation with Semantic Consistency. In *Proceedings of the 7th International Conference on Learning Representations (ICLR)*.

[26] Xudong Mao, Qing Li, Haoran Xie, Raymond Y. K. Lau, Zhen Wang, and Stephen Paul Smolley. 2017. Least Squares Generative Adversarial Networks. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*. 2813–2821.

[27] Mehdi Mirza and Simon Osindero. 2014. Conditional Generative Adversarial Nets. *CoRR* abs/1411.1784 (2014). arXiv:1411.1784

[28] Takeru Miyato, Toshiki Kataoka, Masanori Koyama, and Yuichi Yoshida. 2018. Spectral Normalization for Generative Adversarial Networks. In *Proceedings of the 6th International Conference on Learning Representations (ICLR)*.

[29] Chan Yong Park, Han-Gyu Kim, Dongkeon Lee, Zhun Li, Seung-Ho Han, and Ho-Jin Choi. 2018. Image Analogy with Gaussian Process. In *Proceedings of the 5th IEEE International Conference on Big Data and Smart Computing (BigComp)*. 522–525.

[30] Taesung Park, Ming-Yu Liu, Ting-Chun Wang, and Jun-Yan Zhu. 2019. Semantic Image Synthesis With Spatially-Adaptive Normalization. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*. 2337–2346.

[31] Adam Paszke, Sam Gross, Francisco Massa, Adam Lerer, James Bradbury, Gregory Chanan, Trevor Killeen, Zeming Lin, Natalia Gimelshein, Luca Antiga, Alban Desmaison, Andreas Köpf, Edward Yang, Zachary DeVito, Martin Raison, Alykhan Tejani, Sasank Chilamkurthy, Benoit Steiner, Lu Fang, Junjie Bai, and Soumith Chintala. 2019. PyTorch: An Imperative Style, High-Performance Deep Learning Library. In *Advances in Neural Information Processing Systems 32: Annual Conference on Neural Information Processing Systems (NeurIPS)*. 8024–8035.

[32] Ethan Perez, Florian Strub, Harm de Vries, Vincent Dumoulin, and Aaron C. Courville. 2018. FiLM: Visual Reasoning with a General Conditioning Layer. In *Proceedings of the 32nd AAAI Conference on Artificial Intelligence (AAAI)*. 3942–3951.

[33] Scott E. Reed, Yi Zhang, Yuting Zhang, and Honglak Lee. 2015. Deep Visual Analogy-Making. In *Advances in Neural Information Processing Systems 28: Annual Conference on Neural Information Processing Systems (NeurIPS)*. 1252–1260.

[34] Tamar Rott Shaham, Tali Dekel, and Tomer Michaeli. 2019. SinGAN: Learning a Generative Model From a Single Natural Image. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*. 4569–4579.

[35] Assaf Shocher, Shai Bagon, Phillip Isola, and Michal Irani. 2019. InGAN: Capturing and Retargeting the "DNA" of a Natural Image. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*. 4491–4500.

[36] Denis Simakov, Yaron Caspi, Eli Shechtman, and Michal Irani. 2008. Summarizing visual data using bidirectional similarity. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*.

[37] Karen Simonyan and Andrew Zisserman. 2015. Very Deep Convolutional Networks for Large-Scale Image Recognition. In *Proceedings of the 3rd International Conference on Learning Representations (ICLR)*.

[38] Miao Wang, Guo-Ye Yang, Ruilong Li, Runze Liang, Song-Hai Zhang, Peter M. Hall, and Shi-Min Hu. 2019. Example-Guided Style-Consistent Image Synthesis From Semantic Labeling. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*. 1495–1504.

[39] Ting-Chun Wang, Ming-Yu Liu, Jun-Yan Zhu, Andrew Tao, Jan Kautz, and Bryan Catanzaro. 2018. High-Resolution Image Synthesis and Semantic Manipulation With Conditional GANs. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*. 8798–8807.

[40] Yuxin Wu, Alexander Kirillov, Francisco Massa, Wan-Yen Lo, and Ross Girshick. 2019. Detectron2. https://github.com/facebookresearch/detectron2.

[41] Jiahui Yu, Zhe Lin, Jimei Yang, Xiaohui Shen, Xin Lu, and Thomas S. Huang. 2018. Generative Image Inpainting With Contextual Attention. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*. 5505–5514.

[42] Tao Yu, Zongyu Guo, Xin Jin, Shilin Wu, Zhibo Chen, Weiping Li, Zhizheng Zhang, and Sen Liu. 2020. Region Normalization for Image Inpainting. In *Proceedings of the 34th AAAI Conference on Artificial Intelligence (AAAI)*. 12733–12740.

[43] Hang Zhang and Kristin J. Dana. 2018. Multi-style Generative Network for Real-Time Transfer. In *Proceedings of the European Conference on Computer Vision (ECCV) Workshops*, Vol. 11132. 349–365.

[44] Han Zhang, Tao Xu, and Hongsheng Li. 2017. StackGAN: Text to Photo-Realistic Image Synthesis with Stacked Generative Adversarial Networks. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*. 5908–5916.

[45] Bolei Zhou, Hang Zhao, Xavier Puig, Tete Xiao, Sanja Fidler, Adela Barriuso, and Antonio Torralba. 2019. Semantic Understanding of Scenes Through the ADE20K Dataset. *Int. J. Comput. Vis.* 127, 3 (2019), 302–321.

[46] Maria Zontak and Michal Irani. 2011. Internal statistics of a single natural image. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*. 977–984.