MOGAN: Morphologic-structure-aware Generative Learning from a Single Image

Jinshu Chen, Qihui Xu, Qi Kang, *Senior Member, IEEE*, and MengChu Zhou, *Fellow, IEEE*

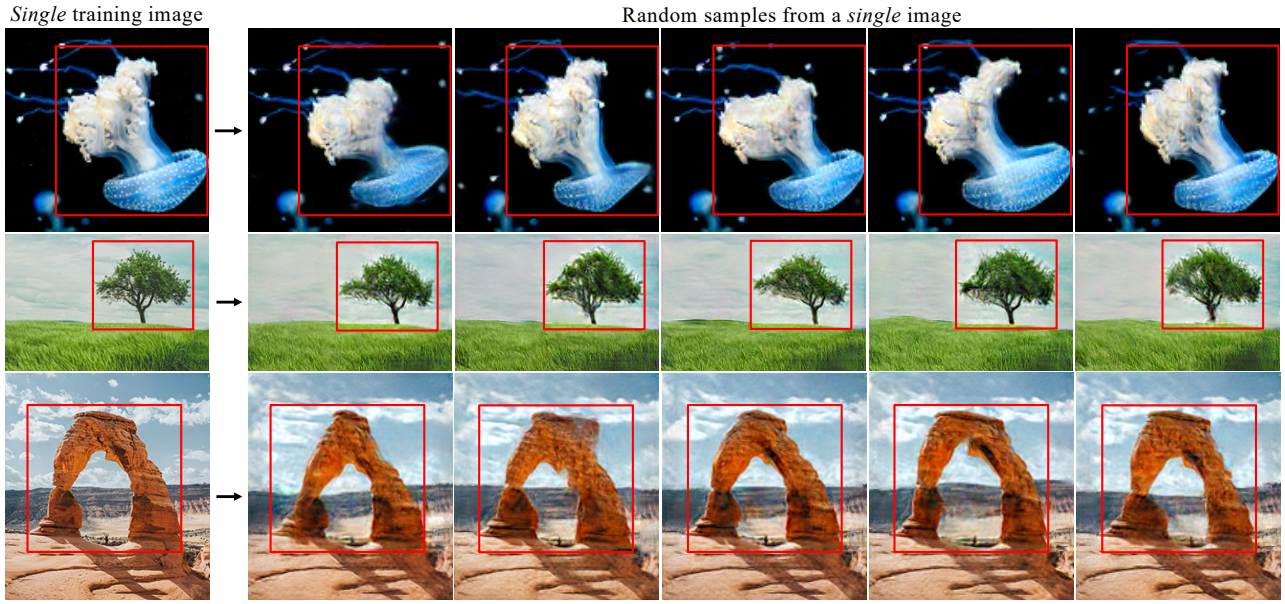

Fig. 1. Random generation learned from a single image. We introduce an unconditional generative model which is competent for interactive ROI-based image generation tasks while based on one image only. It can generate various samples of high quality which own both rational structures and diverse appearances.

***Abstract*—In most interactive image generation tasks, given regions of interest (ROI) by users, the generated results are expected to have adequate diversities in appearance while maintaining correct and reasonable structures in original images. Such tasks become more challenging if only limited data is available. Recently proposed generative models complete training based on only one image. They pay much attention to the monolithic feature of the sample while ignoring the actual semantic information of different objects inside the sample. As a result, for ROI-based generation tasks, they may produce inappropriate samples with excessive randomicity and without maintaining the related objects' correct structures. To address this issue, this work introduces a MOrphologic-structure-aware Generative Adversarial Network named MOGAN that produces random samples with diverse appearances and reliable structures based on only one image. For training for ROI, we propose to utilize the data coming from the original image being augmented and bring in a novel module to transform such augmented data into knowledge containing both structures and appearances, thus enhancing the model's comprehension of the sample. To learn the rest areas other than ROI, we employ binary masks to ensure the generation isolated from ROI. Finally, we set parallel and hierarchical branches of the mentioned learning process. Compared with other single image GAN schemes, our approach focuses on internal features including the maintenance of rational structures and variation on appearance. Experiments confirm a better capacity of our model on ROI-based image generation tasks than its competitive peers.**

***Index Terms*—Generative adversarial networks, single sample, morphologic awareness, ROI-based image generation tasks**

I. Introduction

In many interactive image generation tasks, users tend to be more interested in certain targets or objects in a given sample (i.e., regions of interest or ROI) while paying less attention to the rest areas (called background). As a kind of unsupervised model, Generative Adversarial Networks (GANs) [1] are capable of most generation tasks, which have greatly promoted the development of many fields such as image inpainting [36], [47], [48], image-to-image translation [37], [38], [45], [67] and image synthesis [39], [40], [46]. However, owing to their frail structures that are hard to converge, GANs heavily depend on large datasets or plenty of prior knowledge to complete their training, thus making it an obstacle for GANs to get widely utilized.

In many cases, it is hard to get access to sufficient data of high quality to meet a GAN's training needs. Under such circumstances, effectively learning robust features from a few samples (or a single sample at an extreme case [2]) has become a crucial challenge. Besides many classical tasks [3]–[6], recently proposed models [7]–[9] can accomplish image generation tasks efficiently by adopting hierarchical GAN

pyramid structures [18], [41]–[44], making it possible for unconditional GANs to generate various samples based on only one image. Nevertheless, such models treat different patches of an image equally regardless of their actual semantic information. Given ROI, the models mentioned tend to generate confusing results because they neither provide any interface for users to specify objects as ROI or background nor can they guarantee that the specific objects own proper structures in their generated results. Intuitively, these models produce multiform fake samples by choosing some random patches and applying "copy-shift-paste" operations, which are quite likely to destroy the rational structures of objects inside the sample.

To overcome such deficiency, we follow the same basic precondition (i.e., rely on only one natural image) and propose a novel MOrphologic-structure-aware GAN named **MOGAN**. Our target is to acquire samples that are abundant in appearance diversities while keeping the original structures of objects inside the sample correct. Besides the image, our model takes sets of coordinates specified by users as input to distinguish between ROI and background. Inside the model, we set up two parallel branches to generate ROI and background separately, which are both organised in a hierarchical way but own different characteristics. For the ROI branch, under the premise that no extra prior data is introduced, we propose a method that augment the original image into different forms and such augmented data can be used to learn ample knowledge of both correct structures and morphological patterns. We design a lightweight style extraction module to learn an affine transform from such data then act on the original dataflow, thus providing a guidance for the generation process. This module can be trained end-to-end along with the whole model. For the background branch, the sample for learning is the rest of the image excluding the given ROI. We apply a binary mask to the original image to shield pixels in the position of ROI and thereby switch the task of generating a complete image to generating an image with a mask, which reduces the difficulty of generation to a certain extent.

Finally, we analyze MOGAN's capabilities of managing different tasks including ROI-based random image generation, image editing and single image animation. We test and compare MOGAN with other models in terms of generated results' quality. By analyzing results qualitatively and quantitatively, this work shows that our proposed model achieves better performance than its peers. Moreover, we investigate the effects of different components on our method's performance by conducting several ablation experiments.

In summary, this work aims to make two contributions:

1. On the condition of a single image for training, we propose a novel method for generative models to gain morphologic diversities while maintaining correct structures. It utilizes the morphologic information coming from the augmented original image and we design a lightweight style injector to inject such knowledge to the model.

2. To well accomplish ROI-based image generation tasks (generating images according to regions users are interested in), we introduce a novel model with parallel branches to handle the concurrent and separate generation of ROI and background. In this way we manage to generate various realistic images with only one image to learn from.

In addition, this work analyzes MOGAN's abilities of managing different interactive image generation tasks such as generating random samples based on a single image, image editing and animation. Experiments are performed to validate that MOGAN achieves better performance on the quality of generated samples than its peers.

II. Related Work

A. GANs for Image Processing

Deep learning methods have excellent performances in image generation and feature extraction. Goodfellow et al. proposed a Generative Adversarial Network (GAN) [1] based on the idea of a zero-sum game. GAN has become an important branch in the area of deep learning. It has a wide range of applications in the field of image processing [10]–[17], [49]. Image generation is the most common one. For image generation tasks, the characteristic of GANs that generating based on noise makes the GANs' generated results diverse. More and more techniques and tricks have also been proposed to enhance the stability of GANs' training process [59]–[62].

However, most GANs used for image generation tasks rely on specific datasets or pre-trained models [18], [19]. Getting GANs well-trained on specific datasets places restrictions on GANs' flexibility for different tasks. Instead of extracting the common features of different images, this work focuses on the information contained in a single natural image.

B. Single-image GANs

In recent years, researchers have used the information contained in a single image to build deep learning models, thereby solving the problem of inadequate training data in some cases. InGAN [20] represents the first such model that applies GANs to the processing of a single image. However, it is a conditional generative model. Its generation process requires specific images as input (i.e., mapping the image to an image), thus leading to its poor generalization and failure to generate images randomly. To overcome its drawbacks, Tamar et al. [7] proposed SinGAN. It realizes random generation based on a single image by using an unconditional generation model (i.e., mapping from random noise to an image). Hence, it is suitable for many different image processing tasks. Then ConSinGAN [8] was proposed with a series of improved techniques for training single-image unconditional GANs. HP-VAE-GAN [9] was designed for single-image video generation. But the last three models [7]–[9] share serious defects like excessive randomness and uncontrollability on ROI-based tasks. Compared with them, our proposed model can generate random samples with correct structures and variable appearances based on a single image and users' expectations. It broadens the application range of single image generation.

III. Method

We first give a brief introduction to SinGAN [7] along with a short discussion about its limitations. Then we introduce MOGAN in detail.

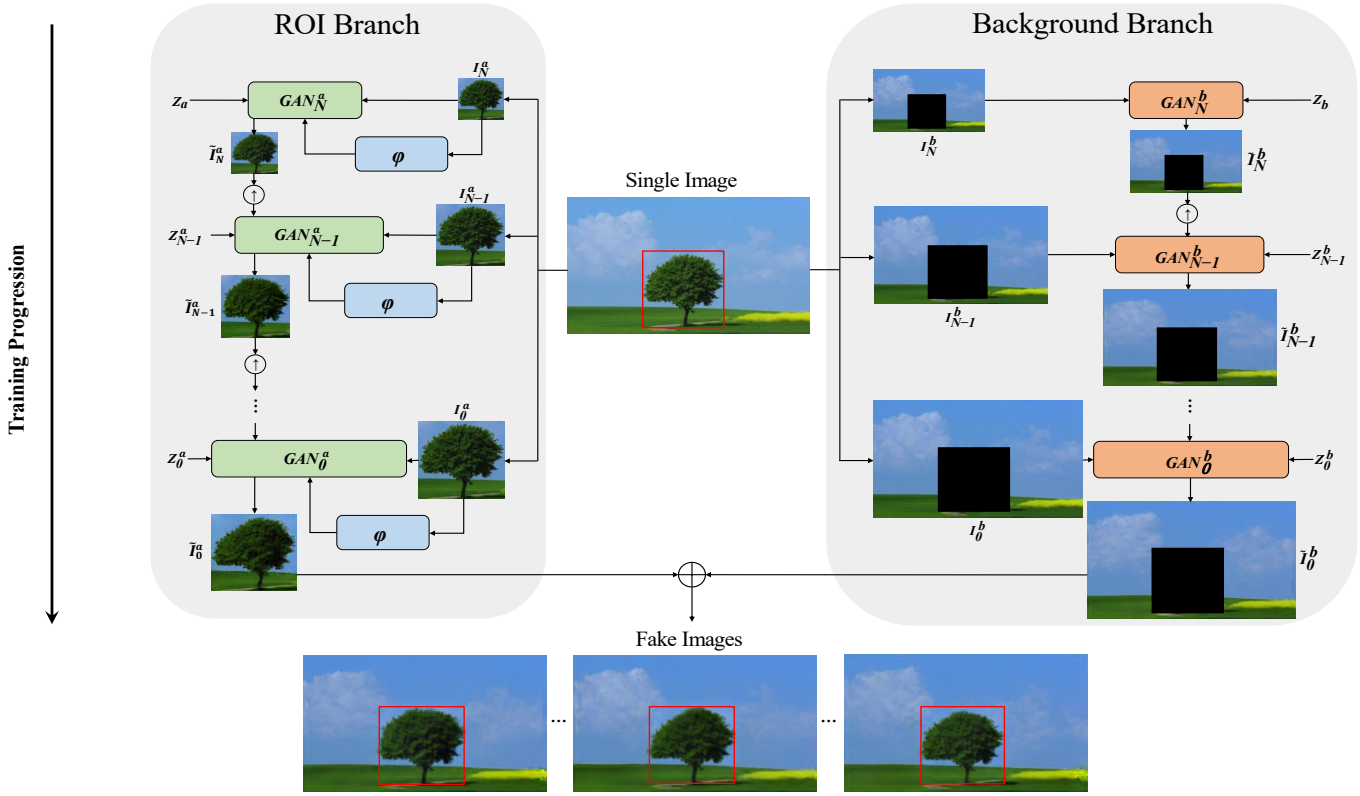

Fig. 2. MOGAN contains two parallel hierarchical branches responsible for the generation of ROI and background. The ROI branch takes ROI cut from the original image as the training target while the background branch takes the original image with a binary mask standing for regions of background. Finally, the generated results produced from two branches can be fused into complete images which are of high quality.

A. SinGAN

SinGAN is a kind of unconditional GAN (generating samples from latent vectors). It is able to generate diverse samples from randomly sampled noise based on only one single natural image. Obeying the design of a hierarchical structure, it stacks several sub-GANs into a pyramid structure. All sub-GANs share exactly the same structure but not parameters. It takes a latent vector sampled from a Gaussian distribution along with the generated result from the previous one as input (by noting that the input of the first sub-GAN is the latent vector only). The training target of each sub-GAN is the original natural image downsampled to different sizes. Such a design makes the learning goal of the overall model gradually shift from small-scale samples which are rich in global structural information to large-scale samples which contain plenty of texture details, deepening the model's comprehension of a given sample.

However, SinGAN's generative process fails to distinguish between different areas or instances inside a given image. Qualitatively, the embodiment of the generated samples' diversity seems like randomly choosing patches from the original image, copying, randomly shifting and then pasting. As a consequence, such a random "copy-shift-paste" type of generation is quite likely to break down an object's structure, thereby resulting in possible irrational outcomes. In many interactive generation tasks, there are some regions in an image in which users are more interested (i.e., ROI). It is thus required for a method to generate random samples with as many changes as possible but no changes in the original semantic structures. For example, if a tree is ROI that interest users, the shape of its crown or bending angle of its trunk may change, while neither the crown nor the trunk should disappear, and the semantic structure that "the crown is above the trunk" should not change either. Such ROI-based tasks are hard for SinGAN to complete.

B. Proposed MOGAN

Motivated by the problems mentioned above, we introduce our MOGAN with its structure shown in Fig. 2.

Parallel-branch architecture: In our problem setting, the usable information includes a single natural image I and a series of coordinates $\{(x_1^{min}, y_1^{min}, x_1^{max}, y_1^{max}), ...,(x_m^{min}, y_m^{min}, x_m^{max}, y_m^{max})\}$ provided by users to mark m regions to which they pay attention. Allowing the model to deal with ROI and background areas separately leads to the problem of disentanglement. Under the circumstance that the number of learnable samples is limited to one, many existing methods of disentanglement based on extra specific prior data [21]–[23] are not applicable. Therefore, we use two different latent vectors marked as Z_a and Z_b to be responsible for the generation of ROI and background in turn similarly to [24]–[26]. Since our requirements for the generation of ROI and background are often different, the structures in charge of generating ROI and background should be independent of

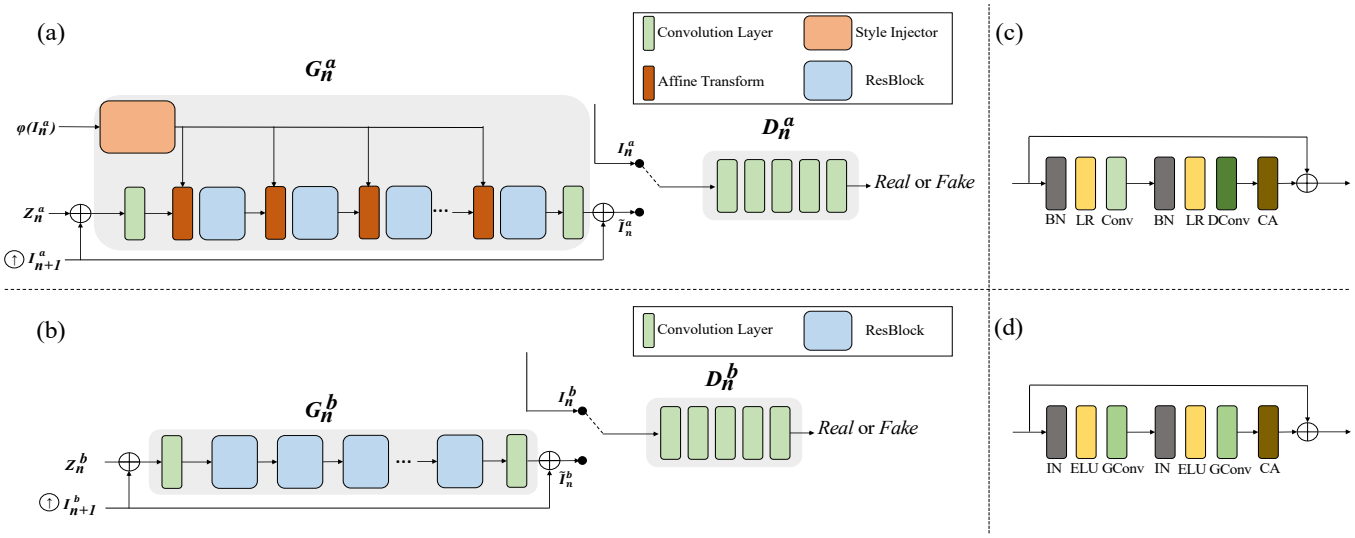

Fig. 3. Details of sub-GANs in two branches. (a) ROI branch. Generators are organised based on residual blocks mainly contain a convolution layer and a deformable convolution layer. A novel module named a style injector that transforms the augmented original image into knowledge of structures and appearances controls the style of generation through affine transforms. (b) Background branch. Generators are built based on residual blocks mainly containing two gated convolution layers. Discriminators of both branches are Markovian discriminators. Details of residual blocks in ROI branch and background branch are shown in (c) and (d) respectively.

each other such that special adjustments can be applied to them, respectively. Therefore, we set up two parallel branches that take the two latent vectors mentioned above as input respectively and conduct their generation processes separately. In most cases where ROI can be well separated from the background, the final generated results of the two branches can be fused together directly. For some complex samples, there may be the discontinuity on the boundary. To handle it, we may take simple interpolation methods as post-processing on the edges. Note that our parallel architecture can handle the two parts' generation synchronously while the methods in [25], [26] have to deal with a generation process in turn. Compared with [56], our MOGAN learns the style during the training while [56] uses pre-extracted style expression. Moreover, [56] regards style information as explicit targets to supervise the training while we only treat the style information as guidance.

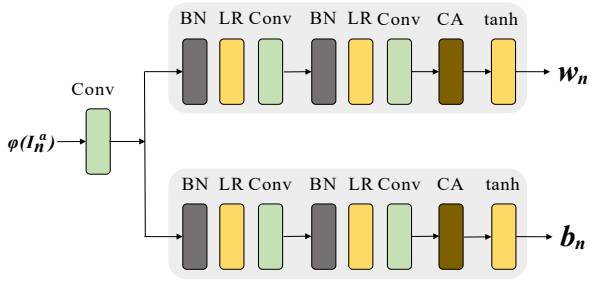

Fig. 4. Details of a style injector. It is a lightweight encoder essentially, which contains two bypasses producing a weight and a bias respectively. Taking the augmented original image as the input, it controls the changing direction of the generator's dataflow through affine transforms.

ROI Branch: For this branch, we expect the generated results to own sufficient morphological changes under the premise that the semantic structure remains correct. The learning target of the ROI branch is the ROI part cut from I according to the coordinates mentioned above, which is marked as I_a. The overall framework of the branch obeys a hierarchical design similar to SinGAN's [7]. To be more specific, we organise a number of sub-GANs $\{GAN^a_0,...,GAN^a_N\}$ and stack them into a pyramid. GAN^a_n consists of a generator G^a_n and a discriminator D^a_n which are trained adversarially. Meanwhile, we set up an image pyramid $\{I^a_0,...,I^a_N\}$ by downsampling I_a for N times based on the rescale factor r^N, where $r>1$. The training process starts from the smallest scale of the image pyramid, i.e., I^a_N along with GAN^a_N, and the level of training scales goes up when the previous scale has finished training. For each I^a_n with the corresponding GAN^a_N, G^a_n learns to map the sum of a noise Z^a_n randomly sampled from a Gaussian distribution and the upsampled output of G^a_{n+1} into a fake sample. To stabilize the training process, we add the output of the generator and the upsampled output of G^a_{n+1} together as the final generated result marked as $\tilde{I}^a_n$. D^a_n attempts to tell I^a_n and $\tilde{I}^a_n$ apart. Note that the input to G^a_N is Z_a only and each GAN^a_n is frozen after being trained.

Here comes the problem: how to increase the diversity of generated results without destroying the semantic structure of the ROI. Without introducing additional training data or prior knowledge such as pre-trained models, we notice that there contains plenty of diverse morphologic information along with rational semantic structures inside the augmented original images. As a guidance, such augmented data not only shows feasible changing directions about appearance but also emphasizes the structure information for the generator. Thus, during the training of each GAN^a_n, we transform I^a_n into different forms through regular data-augmentation methods including flipping vertically and horizontally, padding and randomly rescaling and so on, which are marked as φ in a general way.

In order to extract useful information from $\varphi(I_n^a)$, we build an extra lightweight module named a style injector which takes $\varphi(I_n^a)$ as input and outputs a weight w_n and a bias b_n. Marking the style injector for GAN_n^a as SI_n^a, the process mentioned can be described as:

$$[w_n, b_n] = SI_n^a(\varphi(I_n^a)) \tag{1}$$

Next, we apply the learned affine transform on G_n^a's original dataflow, guiding the generator towards an expected generation direction provided by $\varphi(I_n^a)$. The learned w_n will be multiplied on the G_n^a's original dataflow and the learned b_n will be added. As for the structure of the module, the two bypasses responsible for w_n and b_n have the same formation based on residual design but do not share parameters. The architecture of the module is shown in Fig. 4. Note that our module can be trained end-to-end together with the whole model. This is different from the methods in [18], [27], [28] that all need pre-training. The total generation process can be expressed as:

$$\tilde{I}_n^a = \begin{cases} G_n^a(Z_a, \varphi(I_n^a)), & n = N \\ G_n^a(Z_n^a, \varphi(I_n^a), (\tilde{I}_{n+1}^a \uparrow^{upsample})), & n < N \end{cases} \tag{2}$$

Now that we have brought in more information which may not only lead to better training results but confuse the generator as well. We then consider a more robust design for G_n^a . In order to handle all sizes of samples, we build G_n^a based on a fully convolutional architecture. We set convolution layers at the beginning and end of G_n^a, while we place several residual blocks [63] in the middle. Since interference caused by $\varphi(I_n^a)$ is hard to eliminate, we then treat it as a strong noise. Taking the vacant areas appearing at the four corners of the image after rotating for example, such interference can severely disturb the data distribution in certain specific areas, which is harmful and different from randomly sampled noise Z_n^a that uniformly acts on the dataflow. To cope with this issue, we change the last convolution layer inside each residual block to a deformable convolution layer [29], [30] and add a channel-wise attention layer [31] behind it, which makes G_n^a focus on valuable regions and overlook other disturbing information. Before each convolution layer and deformable convolution layer, we set a BatchNorm layer [64] and a LeakyReLU layer. Another advantage of our design is that we can easily arrange where a style injector plays a part. To be specific, we set a style injector right in front of every residual block, thus ensuring that both the learnable details and bad influence can be handled in a timely fashion. Details of GAN_n^a are shown in Fig. 3a and Fig. 3c.

As for other structure and training details of the ROI branch, D_n^a is a Markovian discriminator [32], [33] with a fully convolution structure. For the discriminators, besides the traditional adversarial loss marked as L_0, we use WGAN-GP loss [34] marked as $L_{WGAN-GP}$ to stabilize D_n^a's training process. WGAN-GP loss can be expressed as:

$$L_{WGAN-GP} = (\left\| \nabla_{\tilde{I}_n^a} D(\tilde{I}_n^a) \right\| - 1)^2 \tag{3}$$

The final loss function for the discriminators can be expressed as:

$$L_D = L_0(G_n^a, D_n^a) + \lambda L_{WGAN-GP}(D_n^a) \tag{4}$$

For G_n^a, besides traditional adversarial loss, we choose mean squared error (MSE) and cosine distance together as loss function. They measure the similarity between $\tilde{I}_n^a$ and I_n^a in different aspects: cosine distance marked as L_1 emphasizes the coherence of global direction and may tolerate local structure difference to some extent. It can be described as:

$$L_1 = 1 - cos(\tilde{I}_n^a, I_n^a) \tag{5}$$

MSE marked as L_2 requires pixel-level consistency to restrain G_n^a from generating bad texture futures. It can be described as:

$$L_2 = \left\| \tilde{I}_n^a - I_n^a \right\|^2 \tag{6}$$

The final loss function for the generators can be expressed as:

$$L_G = L_0(G_n^a, D_n^a) + \alpha L_1(G_n^a) + \beta L_2(G_n^a) \tag{7}$$

Background Branch: Actually, background regions of different samples may vary considerably in complexity. In other words, some samples may own extremely simple background even pure color while others may contain various disparate instances. Since users are less interested in the background, we expect background generation to maintain the global uniformity primarily while making changes in a few local patches, thus no need to introduce extra modules like a style injector into background generation. It doesn't mean it's unnecessary for the generated background to change. For some tasks like image-manipulating task, we believe that only changes in ROI are needed. However, ROI-based tasks also include certain tasks like the data augmentation task where changes of the background are expected: Samples with diversity in both ROI and background are more beneficial for the following CV tasks than those with diversity in only the ROI part. We aim to make our method more general for ROI-based tasks and we believe our "two-branch" idea gives a more proper way.

To isolate the influence of ROI, we apply a binary mask to I and mark the result as I_b. Similar to methods of ROI, we organise another GAN pyramid $\{GAN_0^b,...,GAN_N^b\}$ along with an image pyramid $\{I_0^b,...,I_N^b\}$, where I_n^b becomes the training target of GAN_n^b. Then the generation target can be approximated to a less-difficult image inpainting task for areas inside the mask which are eventually replaced by the generated results of the ROI branch. Similar to the ROI branch, the generation process can be described as:

$$\tilde{I}_n^b = \begin{cases} G_n^b(Z_b), & n = N \\ G_n^b(Z_n^b, (\tilde{I}_{n+1}^b \uparrow^{upsample})), & n < N \end{cases} \tag{8}$$

For the generator in GAN_n^b marked as G_n^b, we adopt similar residual-type of design as mentioned in the ROI's branch but replace all the convolution layers and deformable convolution layers with gated convolution layers [35]. This enables the model to learn the soft mask while learning the pixels outside

Fig. 5. Randomly generated samples. Our model can produce diverse images across different areas and topics for ROI-based image generation tasks. The generated results maintain the original structure of the objects while get plenty of changes on appearance.

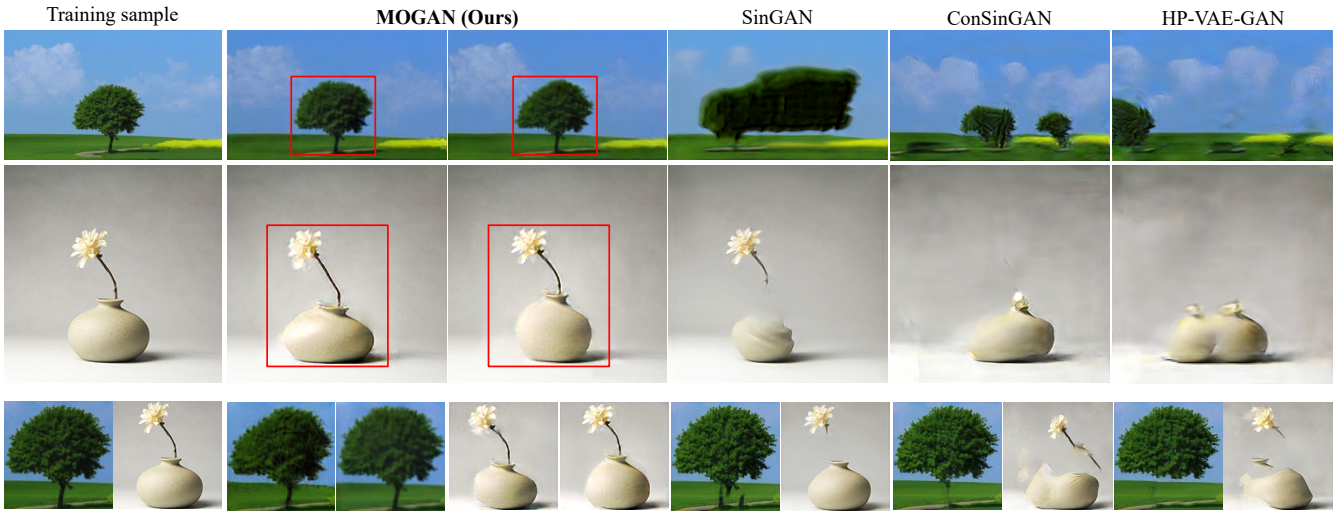

Fig. 6. Randomly generated samples from SinGAN, ConSinGAN, HP-VAE-GAN and ours. For the whole image, other models tend to generate blurry or confusing results. For the ROI part only, other results easily get stuck in overfitting or irrational outcomes while ours are changeable and stable.

TABLE I
SCORES OF AVERAGE SIFID AND AVERAGE DIVERSITY CALCULATED ON SAMPLES GENERATED BY SINGAN [7], CONSINGAN [8], HP-VAE-GAN [9] AND MOGAN. RESULTS SHOW THAT OUR MODEL ACHIEVE THE BEST PERFORMANCE THAN OTHERS.

Metrics	SinGAN [7]	ConSinGAN [8]	HP-VAE-GAN [9]	**MOGAN (Ours)**
SIFID (whole)	0.72	0.63	0.61	**0.22**
Diversity (whole)	0.42	0.49	**0.51**	0.20
GQI (whole)	0.58	0.78	0.84	**0.91**
SIFID (ROI-only)	0.19	0.59	0.56	**0.11**
Diversity (ROI-only)	0.21	**0.51**	0.50	0.39
GQI (ROI-only)	1.11	0.86	0.89	**3.55**

the mask. Besides, we replace the BatchNorm-LeakyReLU layers used in the ROI branch with InstanceNorm-ELU [65] layers. Other training details including loss functions and structures of discriminators are as same as the ROI branch's. Details of GAN_n^b are shown in Fig. 3b and Fig. 3d.

Training Details: For both the ROI branch and the background branch, we use the Adam optimizer [50] with $\beta_1 = 0$ and $\beta_2 = 0.99$. The learning rate of every generator and discriminator in both branches is set to 0.0003. For the loss function of all generators in the ROI branch, α is set to 50 but β varies according to different scales: For coarse scales (such as scale N and N-1), β is set to 10; For other scales, β is set to 5. As for generators in the background branch, α is set to 50 and β is set to 10 for all scales. In both branches, λ for all discriminators is set to 1. We stack three ResBlocks in all generators and five convolution layers in all discriminators. The augmenting methods we take include the random vertical flip, the random horizontal flip, the random rotation, the random affine transform, the random perspective transform and the random erasing [51].

IV. RESULTS

We first explain MOGAN's abilities of managing different ROI-based generation tasks along with revealing some of the results. Next, we compare the performance of our model with its peers' qualitatively and quantitatively. Finally, we validate the effectiveness of our model's components through an ablation study.

A. Applications

Generating Random Samples: *To generate random samples from noise through training against a single natural image* is one of the basic capacities of our model. Samples randomly generated by our model are listed in Fig. 1 and Fig. 5 which contain diverse kinds of images that our model has enough robustness against samples of different styles and topics.

Qualitatively, the generated outcome of ROI has kept a reasonable structure by comparing it with the original image's. In the meantime, ROI generated results get visible diversification on appearance such as shape and posture. As for backgrounds, the generated results exhibit smooth changes in local parts and retain the global layout similar to the original sample's. Note that the model shows the prominent effects on samples that have more freedom degrees to change on appearance. As for samples with a solid structure which is hard to change, the effects seem to reveal in the aspects of postures.

It is worth noting that, we use the raw augmented data (not well pre-trained models or any other specific prior knowledge like some existing methods [57], [58]) to enhance structure awareness. However, the raw data contains strong noise as described in sec. III-B. Injecting such noise into the model may disturb the training and cause blur results. As a result,

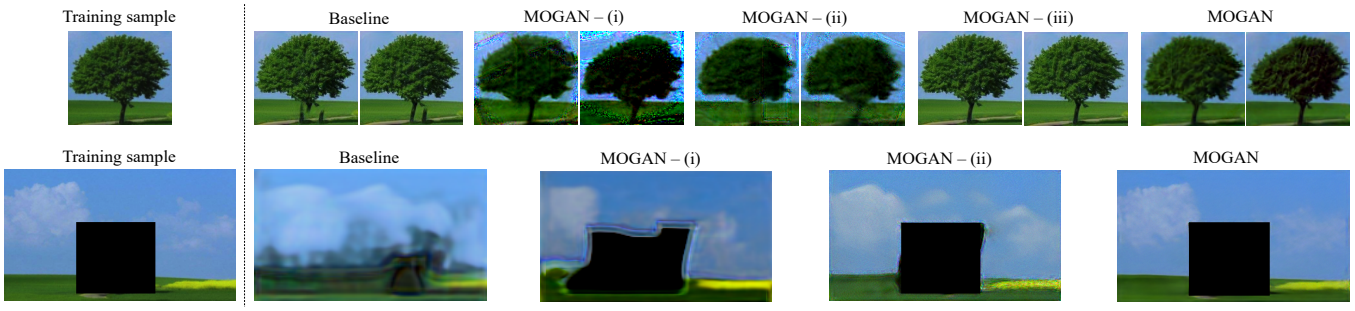

Fig. 7. Results of an ablation study. It is clear that a style injector plays an important part in generating diversely. Deformable convolution and channel attention both make the results refiner. Gated convolution enables the model to handle data with masks.

TABLE II
QUANTITATIVE RESULTS OF THE ABLATION STUDY. RESULTS INDICATE THE SAME CONCLUSION AS FIG. 7.

Settings	SIFID (ROI)	Diversity (ROI)	GQI (ROI)
Baseline	0.19	0.21	1.11
MOGAN - (i)	0.21	**0.42**	2.00
MOGAN - (ii)	0.16	0.29	1.81
MOGAN - (iii)	0.13	0.11	0.85
MOGAN	**0.11**	0.39	**3.55**
Settings	SIFID (background)	Diversity (background)	GQI (background)
Baseline	0.88	**0.84**	0.61
MOGAN - (i)	0.56	0.63	1.13
MOGAN - (ii)	0.40	0.48	1.20
MOGAN	**0.24**	0.31	**1.29**

intuitively, we may find slight shadow or artifacts in the generated results. It is unavoidable to an extent and a kind of trade-off in our opinions. To address this issue, we think that it would help if limiting the effects of the style injecting, e.g., multiplying a factor c $(0 < c < 1)$ on w and b produced by the style injector before they are applied onto the model; setting more ResBlocks between every two affine transforms.

Editing: *To select some of the patches and paste them onto the other location of the original image, then output a harmonious result.* We notice the target of image editing tasks exactly fits the capacity of our model. To perform editing, we take the image pasted with edited patches as the input of a style injector. In this way, the edited information works similarly to the augmented data. For more details, we first train a MOGAN against the image for editing. Then we freeze all trainable parameters of the model, input the image with edited patches to the style injector and start a forward process of the model.

Single Image Animation: *To generate a short video based on a single image,* which is an extension of the ability of random image generation. For MOGAN, after being well-trained on a certain sample, we fix the type of data-augmented methods and adjust the level of the methods gradually. For example, we enlarge the rotation angles of augmented samples by degrees during a series of image generating processes. In this way, we can obtain a number of generated results that change smoothly and organise them into the form of video. Fig. 9 show a sample result.

B. Comparison

We make a comparison among the state-of-the-art models based on a single image. Fig. 6 and Fig. 8 show the comparison results of randomly generating tasks and editing tasks respectively.

Qualitatively, for the randomly generating task, SinGAN and other related models have the similarity in treating samples as a whole. They tend to equally deal with all objects inside as analogical patches regardless of their different importance and semantic information, which results in blurry and ambiguous results. But our model manages the problem very well by setting two parallel branches and generating ROI and the background separately. Next, we take samples that only contain ROI as the training targets and conduct experiments again. From the results, we can find that outcomes of the other models always get stuck in two situations: overfitting (excessively similar to the training target) or meaningless (structure of objects being destroyed). The basic reason of such results is that such models lack a proper guidance for added noise. Restrained by MSE, the noise in the end either affect the texture slightly, which makes the result seem like overfitting, or affect the structure which deforms the related objects too much. On the contrary, our model preserves the original structure to a large extent while making various changes emerging on the object owing to the style injector. Qualitative experiments in our paper are mostly conducted on the Unsplash Dataset due to its high quality. Note that we have not compare the generated results of background because other models cannot deal with data with masks, which will be explained at Sec. IV-C. For the editing task, intuitively, the results of ours are more vivid on the edges of the edited patch

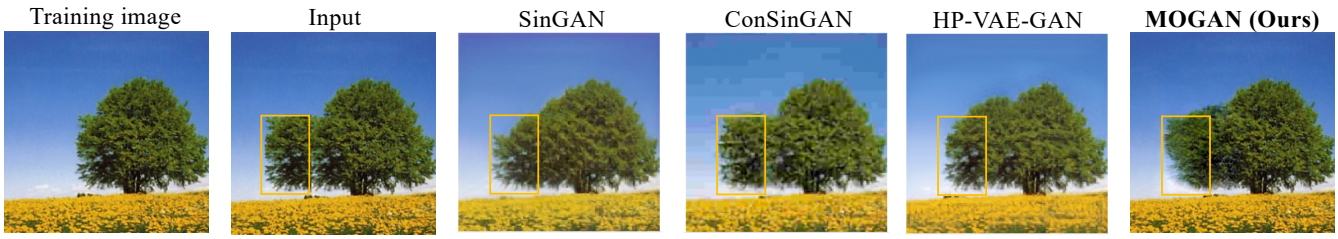

Fig. 8. Results of the single image editing task. Boxes in yellow stand for the original position of the edited patch in the input image. Intuitively, the results of MOGAN are more vivid on the edges of the edited patch than others. The texture of MOGAN's generated results is the finest than its competitive peers'.

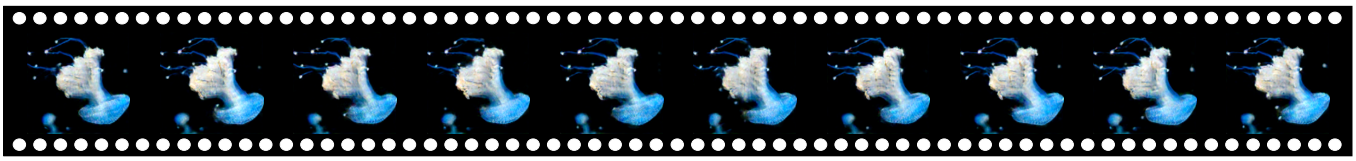

Fig. 9. One sample result of single image animation. We obtain such samples changing smoothly by adjusting the level of augmenting gradually.

than others'. The texture of our generated results is the finest than its competitive peers'.

Quantitatively, we take Single Image Frechet Inception Distance (SIFID) [7], [52], [66] as a metric. We take 50 different samples for models to learn from and then carry out comparison tests on 100 generated results for every sample. Moreover, we calculate the diversity following the method of the coefficient of variation (CV), which is calculated via dividing the standard deviation value by the mean of the samples. Specifically, over the 100 generated samples, we first calculate the CV of the intensity values of each pixel. We average all the CVs over all pixels as the score of the diversity of all generated results upon one training sample. Finally, considering that neither low SIFID score (perhaps caused by overfitting) nor high diversity (perhaps caused by chaos) alone can present high quality of generated samples, we define a metric calculated via dividing the diversity score by SIFID, named generation quality index (i.e., GQI). Samples used for training, inference and quantitative comparison experiments come from the ImageNet Dataset [53] and the Unsplash Dataset which are both standard and open-source datasets. We think they are more suitable for ROI-based tasks com-paring to the Places365 Dataset [54] and Berkeley Segmentation Dataset [55] which are claimed and used in the SinGAN paper, because each sample in both datasets has a clear topic and is easy to assign ROIs. Scores of different models are recorded in Table I. Coinciding with the qualitative analysis, our model has achieved better performance than its peers given the same samples. When trained against the whole image, the other three models get higher diversity scores than ours while their SIFID and GQI are lower due to their chaotic generated results. When trained against ROI only, SinGAN always produces overfitting results which increases its SIFID and GQI while reducing the diversity to a large extent. ConSinGAN and HP-VAE-GAN continue to generate meaningless images with high diversity but low quality. Our MOGAN's results are both diverse and realistic. Besides, GQI goes up when training on ROI only because of the lower difficulty. For our model, the diversity is lower for the whole image owing to the globally similar backgrounds.

C. Ablation Study

To analyze our design's impact on the generation process, we take SinGAN as the baseline and conduct ablation experiments on two branches separately. The qualitative results are described in Fig. 7 and the quantitative results are in Table II.

For the ROI branch, improvements we make onto a structure include (i) deformable convolution layer, (ii) channel attention layer, and (iii) style injector. The effects on the generated results after removing certain design methods can be seen in Fig. 7. Note that "-" means that the method is disabled and "removing deformable convolution layer" means replacing it with a full convolution layer. Obviously, the style injector plays an important role in generating diversely as models without it induce overfitting. Deformable convolution and channel attention layers both prevent the results from strong noise and stripes to a large extent, while the former plays a more effective role.

For the background branch, we introduced (i) gated convolution layers, and (ii) channel attention layers. We take away modules or methods above in turn and record the effects on the generated results in Fig. 7. Similarly, "removing gated convolution layer" means replacing it with a full convolution layer. The results suggest that without gated convolution layers, the model will treat the mask as an object of certain semantic information. Thus, gated convolution layers enable the model to handle masks which isolates background from ROI. Channel attention layers make the training more stable.

V. Conclusion

We have introduced MOGAN, an unconditional generative model to generate random samples based on only one natural

image. The generation results from our model can maintain correct structures and exhibit plenty of diversity in appearance, which is the main improvements over other recent models. As demonstrated by our experiments, MOGAN can produce samples of high quality over different kinds of images. Our future work intends to handle more detailed information of a single sample including color and texture, and to guide the generated results of ROI and background to cohere with each other.

REFERENCES

[1] I. J. Goodfellow, J. Pouget-Abadie, M. Mirza, B. Xu, D. Warde-Farley, S. Ozair, A. C. Courville, and Y. Bengio, "Generative adversarial nets," in *NIPS*, 2014.

[2] M. Zontak and M. Irani, "Internal statistics of a single natural image," *CVPR 2011*, pp. 977–984, 2011.

[3] M. Zontak, I. Mosseri, and M. Irani, "Separating signal from noise using patch recurrence across scales," *2013 IEEE Conference on Computer Vision and Pattern Recognition*, pp. 1195–1202, 2013.

[4] T. Michaeli and M. Irani, "Blind deblurring using internal patch recurrence," in *ECCV*, 2014.

[5] Y. Bahat and M. Irani, "Blind dehazing using internal patch recurrence," *2016 IEEE International Conference on Computational Photography (ICCP)*, pp. 1–9, 2016.

[6] G. Freedman and R. Fattal, "Image and video upscaling from local self-examples," *ACM Trans. Graph.*, vol. 30, pp. 12:1–12:11, 2011.

[7] T. R. Shaham, T. Dekel, and T. Michaeli, "Singan: Learning a generative model from a single natural image," *2019 IEEE/CVF International Conference on Computer Vision (ICCV)*, pp. 4569–4579, 2019.

[8] T. Hinz, M. Fisher, O. Wang, and S. Wermter, "Improved techniques for training single-image gans," in *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*, 2021, pp. 1300–1309.

[9] S. Gur, S. Benaim, and L. Wolf, "Hierarchical patch vae-gan: Generating diverse videos from a single sample," *ArXiv*, vol. abs/2006.12226, 2020.

[10] J.-Y. Zhu, P. Krähenbühl, E. Shechtman, and A. A. Efros, "Generative visual manipulation on the natural image manifold," in *ECCV*, 2016.

[11] T. Dekel, C. Gan, D. Krishnan, C. Liu, and W. Freeman, "Sparse, smart contours to represent and edit images," *2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 3511–3520, 2018.

[12] J.-Y. Zhu, T. Park, P. Isola, and A. A. Efros, "Unpaired image-to-image translation using cycle-consistent adversarial networks," in *Proceedings of the IEEE international conference on computer vision*, 2017, pp. 2223–2232.

[13] W. Chen and J. Hays, "Sketchygan: Towards diverse and realistic sketch to image synthesis," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2018, pp. 9416–9425.

[14] X. Wang and A. Gupta, "Generative image modeling using style and structure adversarial networks," in *European conference on computer vision*. Springer, 2016, pp. 318–335.

[15] J. Yu, Z. Lin, J. Yang, X. Shen, X. Lu, and T. S. Huang, "Generative image inpainting with contextual attention," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2018, pp. 5505–5514.

[16] G. Perarnau, J. Van De Weijer, B. Raducanu, and J. M. Álvarez, "Invertible conditional gans for image editing," *arXiv preprint arXiv:1611.06355*, 2016.

[17] J. Gu, Y. Shen, and B. Zhou, "Image processing using multi-code gan prior," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2020, pp. 3012–3021.

[18] T. Karras, S. Laine, and T. Aila, "A style-based generator architecture for generative adversarial networks," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2019, pp. 4401–4410.

[19] A. Brock, J. Donahue, and K. Simonyan, "Large scale gan training for high fidelity natural image synthesis," in *International Conference on Learning Representations*, 2018.

[20] A. Shocher, S. Bagon, P. Isola, and M. Irani, "Ingan: Capturing and remapping the" dna" of a natural image," *arXiv preprint arXiv:1812.00231*, 2018.

[21] L. Ma, Q. Sun, S. Georgoulis, L. Van Gool, B. Schiele, and M. Fritz, "Disentangled person image generation," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2018, pp. 99–108.

[22] T. Nguyen-Phuoc, C. Li, L. Theis, C. Richardt, and Y.-L. Yang, "Hologan: Unsupervised learning of 3d representations from natural images," in *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 2019, pp. 7588–7597.

[23] A. Pumarola, A. Agudo, A. Sanfeliu, and F. Moreno-Noguer, "Unsupervised person image synthesis in arbitrary poses," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2018, pp. 8620–8628.

[24] H. Kwak and B.-T. Zhang, "Generating images part by part with composite generative adversarial networks," *arXiv preprint arXiv:1607.05387*, 2016.

[25] K. K. Singh, U. Ojha, and Y. J. Lee, "Finegan: Unsupervised hierarchical disentanglement for fine-grained object generation and discovery," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2019, pp. 6490–6499.

[26] Y. Li, K. K. Singh, U. Ojha, and Y. J. Lee, "Mixnmatch: multifactor disentanglement and encoding for conditional image generation," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2020, pp. 8039–8048.

[27] X. Huang and S. Belongie, "Arbitrary style transfer in real-time with adaptive instance normalization," in *Proceedings of the IEEE International Conference on Computer Vision*, 2017, pp. 1501–1510.

[28] T. Karras, S. Laine, M. Aittala, J. Hellsten, J. Lehtinen, and T. Aila, "Analyzing and improving the image quality of stylegan," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2020, pp. 8110–8119.

[29] J. Dai, H. Qi, Y. Xiong, Y. Li, G. Zhang, H. Hu, and Y. Wei, "Deformable convolutional networks," in *Proceedings of the IEEE international conference on computer vision*, 2017, pp. 764–773.

[30] X. Zhu, H. Hu, S. Lin, and J. Dai, "Deformable convnets v2: More deformable, better results," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2019, pp. 9308–9316.

[31] W. Qilong, W. Banggu, Z. Pengfei, L. Peihua, Z. Wangmeng, and H. Qinghua, "Eca-net: Efficient channel attention for deep convolutional neural networks." 2020.

[32] C. Li and M. Wand, "Precomputed real-time texture synthesis with markovian generative adversarial networks," in *European conference on computer vision*. Springer, 2016, pp. 702–716.

[33] P. Isola, J.-Y. Zhu, T. Zhou, and A. A. Efros, "Image-to-image translation with conditional adversarial networks," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2017, pp. 1125–1134.

[34] I. Gulrajani, F. Ahmed, M. Arjovsky, V. Dumoulin, and A. C. Courville, "Improved training of wasserstein gans," in *NIPS*, 2017.

[35] J. Yu, Z. Lin, J. Yang, X. Shen, X. Lu, and T. S. Huang, "Free-form image inpainting with gated convolution," in *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 2019, pp. 4471–4480.

[36] C. Zheng, T.-J. Cham, and J. Cai, "Pluralistic image completion," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2019, pp. 1438–1447.

[37] I. Anokhin, P. Solovev, D. Korzhenkov, A. Kharlamov, T. Khakhulin, A. Silvestrov, S. Nikolenko, V. Lempitsky, and G. Sterkin, "High-resolution daytime translation without domain labels," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2020, pp. 7488–7497.

[38] P. Zhang, B. Zhang, D. Chen, L. Yuan, and F. Wen, "Cross-domain correspondence learning for exemplar-based image translation," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2020, pp. 5143–5153.

[39] A. Shocher, Y. Gandelsman, I. Mosseri, M. Yarom, M. Irani, W. T. Freeman, and T. Dekel, "Semantic pyramid for image generation," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2020, pp. 7457–7466.

[40] Y. Choi, Y. Uh, J. Yoo, and J.-W. Ha, "Stargan v2: Diverse image synthesis for multiple domains," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2020, pp. 8188–8197.

[41] T. Karras, T. Aila, S. Laine, and J. Lehtinen, "Progressive growing of gans for improved quality, stability, and variation," in *International Conference on Learning Representations*, 2018.

[42] T.-C. Wang, M.-Y. Liu, J.-Y. Zhu, A. Tao, J. Kautz, and B. Catanzaro, "High-resolution image synthesis and semantic manipulation with conditional gans," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2018, pp. 8798–8807.

[43] E. L. Denton, S. Chintala, A. Szlam, and R. Fergus, "Deep generative image models using a laplacian pyramid of adversarial networks," in *NIPS*, 2015.
[44] H. Zhang, T. Xu, H. Li, S. Zhang, X. Wang, X. Huang, and D. N. Metaxas, "Stackgan: Text to photo-realistic image synthesis with stacked generative adversarial networks," in *Proceedings of the IEEE international conference on computer vision*, 2017, pp. 5907–5915.
[45] D. Bhattacharjee, S. Kim, G. Vizier, and M. Salzmann, "Dunit: Detection-based unsupervised image-to-image translation," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2020, pp. 4787–4796.
[46] C.-H. Lee, Z. Liu, L. Wu, and P. Luo, "Maskgan: Towards diverse and interactive facial image manipulation," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2020, pp. 5549–5558.
[47] L. Zhao, Q. Mo, S. Lin, Z. Wang, Z. Zuo, H. Chen, W. Xing, and D. Lu, "Uctgan: Diverse image inpainting based on unsupervised cross-space translation," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2020, pp. 5741–5750.
[48] Z. Yi, Q. Tang, S. Azizi, D. Jang, and Z. Xu, "Contextual residual aggregation for ultra high-resolution image inpainting," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2020, pp. 7508–7517.
[49] P. Xiang, L. Wang, F. Wu, J. Cheng, and M. Zhou, "Single-image deraining with feature-supervised generative adversarial network," *IEEE Signal Processing Letters*, vol. 26, no. 5, pp. 650–654, 2019.
[50] D. P. Kingma and J. Ba, "Adam: A method for stochastic optimization," *CoRR*, vol. abs/1412.6980, 2015.
[51] Z. Zhong, L. Zheng, G. Kang, S. Li, and Y. Yang, "Random erasing data augmentation," in *AAAI*, 2020.
[52] M. Heusel, H. Ramsauer, T. Unterthiner, B. Nessler, and S. Hochreiter, "Gans trained by a two time-scale update rule converge to a local nash equilibrium," in *NIPS*, 2017.
[53] J. Deng, W. Dong, R. Socher, L. Li, K. Li, and L. Fei-Fei, "Imagenet: A large-scale hierarchical image database," in *CVPR*, 2009.
[54] B. Zhou, À. Lapedriza, A. Khosla, A. Oliva, and A. Torralba, "Places: A 10 million image database for scene recognition," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 40, pp. 1452–1464, 2018.
[55] D. Martin, C. Fowlkes, D. Tal, and J. Malik, "A database of human segmented natural images and its application to evaluating segmentation algorithms and measuring ecological statistics," in *Proc. 8th Int'l Conf. Computer Vision*, vol. 2, July 2001, pp. 416–423.
[56] R. Wu, G. Zhang, S. Lu, and T. Chen, "Cascade ef-gan: Progressive facial expression editing with local focuses," *2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 5020–5029, 2020.
[57] S. Gu, J. min Bao, D. Chen, and F. Wen, "Priorgan: Real data prior for generative adversarial nets," *ArXiv*, vol. abs/2006.16990, 2020.
[58] A. Shocher, Y. Gandelsman, I. Mosseri, M. Yarom, M. Irani, W. Freeman, and T. Dekel, "Semantic pyramid for image generation," *2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 7455–7464, 2020.
[59] H. Lee, R. B. Grosse, R. Ranganath, and A. Ng, "Convolutional deep belief networks for scalable unsupervised learning of hierarchical representations," in *ICML '09*, 2009.
[60] M. Arjovsky, S. Chintala, and L. Bottou, "Wasserstein generative adversarial networks," in *ICML*, 2017.
[61] T. White, "Sampling generative networks: Notes on a few effective techniques," *ArXiv*, vol. abs/1609.04468, 2016.
[62] X. Mao, Q. Li, H. Xie, R. Y. K. Lau, Z. Wang, and S. P. Smolley, "Least squares generative adversarial networks," *2017 IEEE International Conference on Computer Vision (ICCV)*, pp. 2813–2821, 2017.
[63] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 770–778, 2016.
[64] S. Ioffe and C. Szegedy, "Batch normalization: Accelerating deep network training by reducing internal covariate shift," *ArXiv*, vol. abs/1502.03167, 2015.
[65] D. Ulyanov, A. Vedaldi, and V. Lempitsky, "Instance normalization: The missing ingredient for fast stylization," *ArXiv*, vol. abs/1607.08022, 2016.
[66] H. Zhang, I. J. Goodfellow, D. N. Metaxas, and A. Odena, "Self-attention generative adversarial networks," in *ICML*, 2019.
[67] P. Isola, J.-Y. Zhu, T. Zhou, and A. A. Efros, "Image-to-image translation with conditional adversarial networks," *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 5967–5976, 2017.

Jinshu Chen received the B.S. degree in automation from Tongji University, Shanghai, China, in 2019. He is currently pursuing the M.S. degree in control science and engineering from Tongji University, Shanghai. His research interests include computer vision, deep learning and image processing.

Qihui Xu recevied the B.S.degree in automatic control from Tongji University, Shanghai, China, in 2020. She is currently pursuing the M.S. degree in control science and engineering from Tongji University, Shanghai. Her research focuses on the use of deep learning technology for few-shot learning and image generation.

Qi Kang (Senior Member, IEEE) received the B.S. degree in automatic control, the M.S. degree in control theory and control engineering, and the Ph.D. degree in control theory and control engineering from Tongji University, Shanghai, China, in 2002, 2005, and 2009, respectively. From 2007 to 2008, he was a Research Associate with the University of Illinois, Chicago, IL, USA. From 2014 to 2015, he was a Visiting Scholar with the New Jersey Institute of Technology, Newark, NJ, USA. He is currently a Professor with the Department of Control Science and Engineering and the Shanghai Institute of Intelligent Science and Technology, Tongji University, Shanghai, China. His interests are in swarm intelligence, evolutionary computation, machine learning, and intelligent control and optimization in transportation, energy, and water systems.

MengChu Zhou (Fellow, IEEE) joined the New Jersey Institute of Technology (NJIT), Newark, NJ, USA, in 1990, where he is currently a Distinguished Professor. He has over 900 publications, including 12 books, over 600 journal articles (over 500 in IEEE TRANSACTIONS), 27 patents, and 29 book chapters. His interests are in Petri nets, intelligent automation, the Internet of Things, and big data. Prof. Zhou is fellow of International Federation of Automatic Control, American Association for the Advancement of Science and Chinese Association of Automation. He was a recipient of the Humboldt Research Award for U.S. Senior Scientists from the Alexander von Humboldt Foundation, the Franklin V. Taylor Memorial Award and the Norbert Wiener Award from the IEEE Systems, Man and Cybernetics Society, the Excellence in Research Prize and Medal from NJIT, and the Edison Patent Award from the Research and Development Council of New Jersey. He is also the Founding Editor of the IEEE Press Book Series on Systems Science and Engineering and the Editor-in-Chief of IEEE/CAA JOURNAL OF AUTOMATICA SINICA.