

SinGAN: Learning a Generative Model from a Single Natural Image

Tamar Rott Shaham
Technion

Tali Dekel
Google Research

Tomer Michaeli
Technion

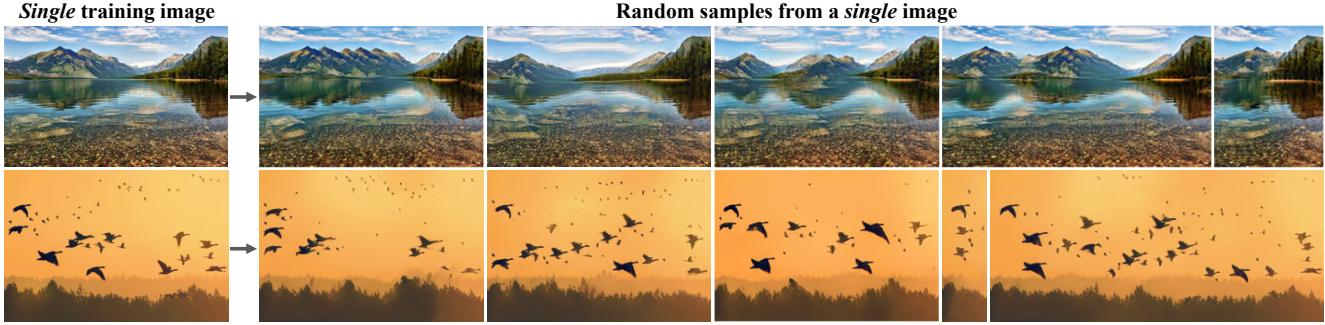


Figure 1: Image generation learned from a single training image. We propose *SinGAN*—a new unconditional generative model trained on a *single natural image*. Our model learns the image’s patch statistics across multiple scales, using a dedicated multi-scale adversarial training scheme; it can then be used to generate new realistic image samples that preserve the original patch distribution while creating new object configurations and structures.

Abstract

We introduce *SinGAN*, an unconditional generative model that can be learned from a single natural image. Our model is trained to capture the internal distribution of patches within the image, and is then able to generate high quality, diverse samples that carry the same visual content as the image. *SinGAN* contains a pyramid of fully convolutional GANs, each responsible for learning the patch distribution at a different scale of the image. This allows generating new samples of arbitrary size and aspect ratio, that have significant variability, yet maintain both the global structure and the fine textures of the training image. In contrast to previous single image GAN schemes, our approach is not limited to texture images, and is not conditional (i.e. it generates samples from noise). User studies confirm that the generated samples are commonly confused to be real images. We illustrate the utility of *SinGAN* in a wide range of image manipulation tasks.

1. Introduction

Generative Adversarial Nets (GANs) [19] have made a dramatic leap in modeling high dimensional distributions of visual data. In particular, unconditional GANs have shown remarkable success in generating realistic, high quality samples when trained on class specific datasets (e.g., faces [33], bedrooms[47]). However, capturing the distribution of highly diverse datasets with multiple object classes

(e.g. ImageNet [12]), is still considered a major challenge and often requires conditioning the generation on another input signal [6] or training the model for a specific task (e.g. super-resolution [30], inpainting [41], retargeting [45]).

Here, we take the use of GANs into a new realm – *unconditional* generation learned from a *single natural image*. Specifically, we show that the internal statistics of patches within a single natural image typically carry enough information for learning a powerful generative model. *SinGAN*, our new single image generative model, allows us to deal with general natural images that contain complex structures and textures, without the need to rely on the existence of a database of images from the same class. This is achieved by a pyramid of fully convolutional light-weight GANs, each is responsible for capturing the distribution of patches at a different scale. Once trained, *SinGAN* can produce diverse high quality image samples (of arbitrary dimensions), which semantically resemble the training image, yet contain new object configurations and structures¹ (Fig. 1).

Modeling the internal distribution of patches within a single natural image has been long recognized as a powerful prior in many computer vision tasks [64]. Classical examples include denoising [65], deblurring [39], super resolution [18], dehazing [2, 15], and image editing [37, 21, 9, 11, 50]. The most closely related work in this context is [48], where a bidirectional patch similarity measure is defined and optimized to guarantee that the patches of an image after manipulation are the same as the

¹Code available at: <https://github.com/tamarott/SinGAN>

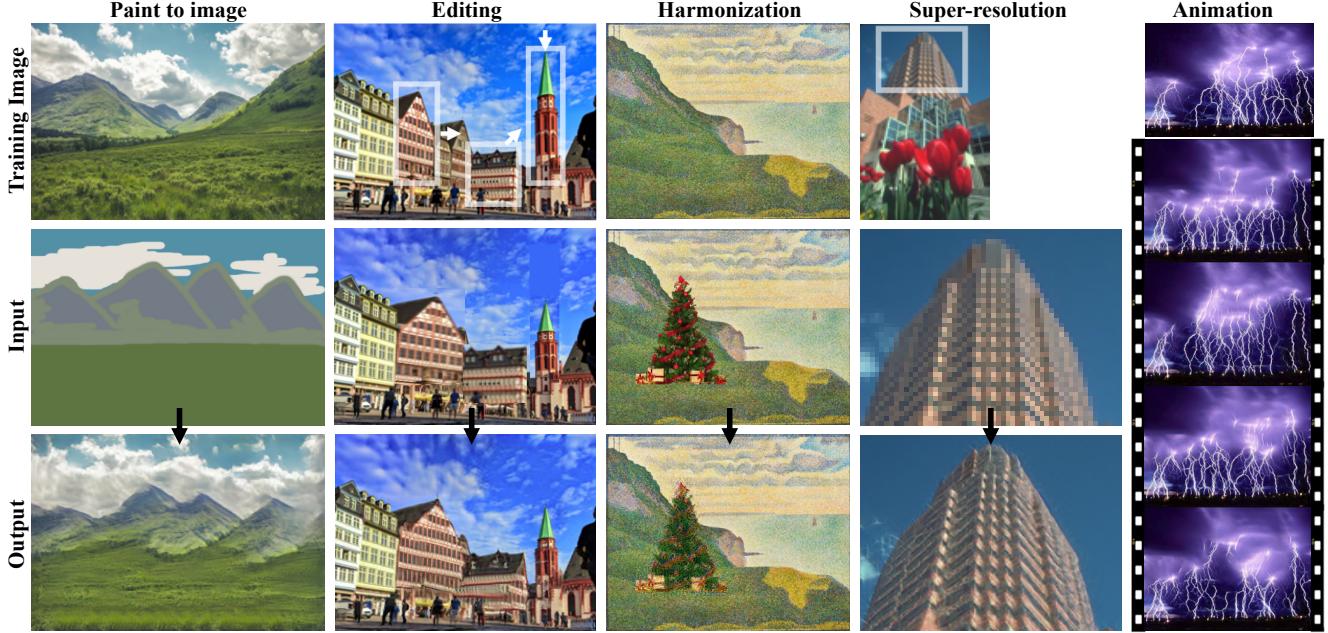


Figure 2: Image manipulation. SinGAN can be used in various image manipulation tasks, including: transforming a paint (clipart) into a realistic photo, rearranging and editing objects in the image, harmonizing a new object into an image, image super-resolution and creating an animation from a single input. In all these cases, our model observes only the training image (first row) and is trained in the same manner for all applications, with no architectural changes or further tuning (see Sec. 4).

original ones. Motivated by these works, here we show how SinGAN can be used within a simple unified learning framework to solve a variety of image manipulation tasks, including paint-to-image, editing, harmonization, super-resolution, and animation from a single image. In all these cases, our model produces high quality results that preserve the internal patch statistics of the training image (see Fig. 2 and our [project webpage](#)). All tasks are achieved with *the same* generative network, without any additional information or further training beyond the original training image.

1.1. Related Work

Single image deep models Several recent works proposed to “overfit” a deep model to a single training example [51, 60, 46, 7, 1]. However, these methods are designed for specific tasks (*e.g.*, super resolution [46], texture expansion [60]). Shocher *et al.* [44, 45] were the first to introduce an internal GAN based model for a single natural image, and illustrated it in the context of retargeting. However, their generation is conditioned on an input image (*i.e.*, mapping images to images) and is not used to draw random samples. In contrast, our framework is purely generative (*i.e.* maps noise to image samples), and thus suits many different image manipulation tasks. *Unconditional* single image GANs have been explored only in the context of texture generation [3, 27, 31]. These models do not generate meaningful samples when trained on non-texture images (Fig. 3). Our method, on the other hand, is not restricted to texture and can handle general natural images (*e.g.*, Fig. 1).

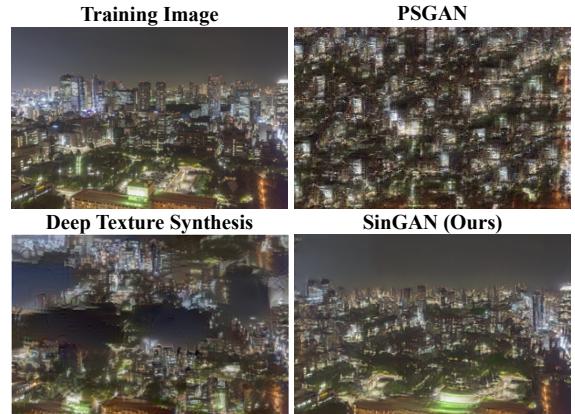


Figure 3: SinGAN vs. Single Image Texture Generation. Single image models for texture generation [3, 16] are not designed to deal with natural images. Our model can produce realistic image samples that consist of complex textures and non-repetitive global structures.

Generative models for image manipulation The power of adversarial learning has been demonstrated by recent GAN-based methods, in many different image manipulation tasks [61, 10, 62, 8, 53, 56, 42, 53]. Examples include interactive image editing [61, 10], sketch2image [8, 43], and other image-to-image translation tasks [62, 52, 54]. However, all these methods are trained on class specific datasets, and here too, often condition the generation on another input signal. We are not interested in capturing common features among images of the same class, but rather con-

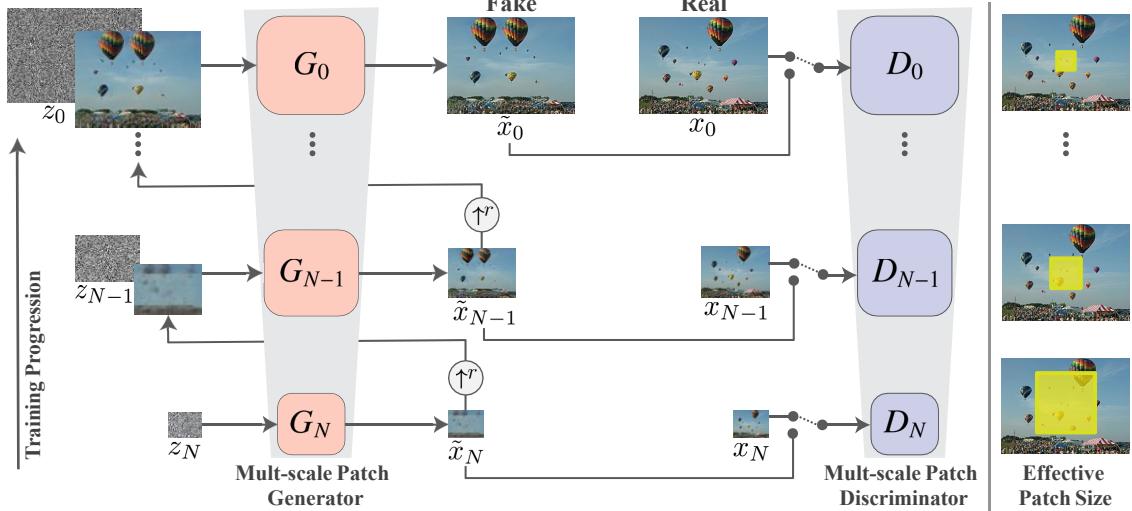


Figure 4: SinGAN’s multi-scale pipeline. Our model consists of a pyramid of GANs, where both training and inference are done in a coarse-to-fine fashion. At each scale, G_n learns to generate image samples in which all the overlapping patches cannot be distinguished from the patches in the down-sampled training image, x_n , by the discriminator D_n ; the effective patch size decreases as we go up the pyramid (marked in yellow on the original image for illustration). The input to G_n is a random noise image z_n , and the generated image from the previous scale \tilde{x}_n , upsampled to the current resolution (except for the coarsest level which is purely generative). The generation process at level n involves all generators $\{G_N \dots G_n\}$ and all noise maps $\{z_N, \dots, z_n\}$ up to this level. See more details at Sec. 2.

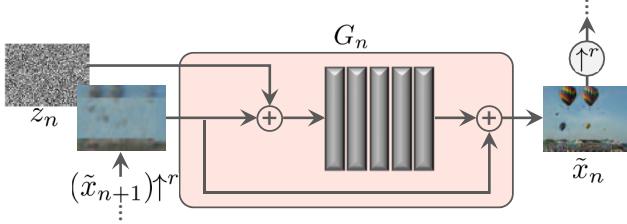


Figure 5: Single scale generation. At each scale n , the image from the previous scale, \tilde{x}_{n+1} , is upsampled and added to the input noise map, z_n . The result is fed into 5 conv layers, whose output is a residual image that is added back to $(\tilde{x}_{n+1}) \uparrow^r$. This is the output \tilde{x}_n of G_n .

sider a different source of training data – all the overlapping patches at multiple scales of a single natural image. We show that a powerful generative model can be learned from this data, and can be used in a number of image manipulation tasks.

2. Method

Our goal is to learn an *unconditional* generative model that captures the internal statistics of a *single* training image x . This task is conceptually similar to the conventional GAN setting, except that here the training samples are patches of a single image, rather than whole image samples from a database.

We opt to go beyond texture generation, and to deal with more general natural images. This requires capturing the statistics of complex image structures at many different scales. For example, we want to capture global properties

such as the arrangement and shape of large objects in the image (*e.g.* sky at the top, ground at the bottom), as well as fine details and texture information. To achieve that, our generative framework, illustrated in Fig. 4, consists of a hierarchy of patch-GANs (Markovian discriminator) [31, 26], where each is responsible for capturing the patch distribution at a different scale of x . The GANs have small receptive fields and limited capacity, preventing them from memorizing the single image. While similar multi-scale architectures have been explored in conventional GAN settings (*e.g.* [28, 52, 29, 52, 13, 24]), we are the first explore it for internal learning from a single image.

2.1. Multi-scale architecture

Our model consists of a pyramid of generators, $\{G_0, \dots, G_N\}$, trained against an image pyramid of $x: \{x_0, \dots, x_N\}$, where x_n is a downsampled version of x by a factor r^n , for some $r > 1$. Each generator G_n is responsible of producing realistic image samples w.r.t. the patch distribution in the corresponding image x_n . This is achieved through adversarial training, where G_n learns to fool an associated discriminator D_n , which attempts to distinguish patches in the generated samples from patches in x_n .

The generation of an image sample starts at the coarsest scale and sequentially passes through all generators up to the finest scale, with noise injected at every scale. All the generators and discriminators have the same receptive field and thus capture structures of decreasing size as we go up the generation process. At the coarsest scale, the generation is purely generative, *i.e.* G_N maps spatial white Gaussian noise z_N to an image sample \tilde{x}_N ,

$$\tilde{x}_N = G_N(z_N). \quad (1)$$

The effective receptive field at this level is typically $\sim 1/2$ of the image's height, hence G_N generates the general layout of the image and the objects' global structure. Each of the generators G_n at finer scales ($n < N$) adds details that were not generated by the previous scales. Thus, in addition to spatial noise z_n , each generator G_n accepts an upsampled version of the image from the coarser scale, *i.e.*,

$$\tilde{x}_n = G_n(z_n, (\tilde{x}_{n+1})^{\uparrow r}), \quad n < N. \quad (2)$$

All the generators have a similar architecture, as depicted in Fig. 5. Specifically, the noise z_n is added to the image $(\tilde{x}_{n+1})^{\uparrow r}$, prior to being fed into a sequence of convolutional layers. This ensures that the GAN does not disregard the noise, as often happens in conditional schemes involving randomness [62, 36, 63]. The role of the convolutional layers is to generate the missing details in $(\tilde{x}_{n+1})^{\uparrow r}$ (residual learning [22, 57]). Namely, G_n performs the operation

$$\tilde{x}_n = (\tilde{x}_{n+1})^{\uparrow r} + \psi_n(z_n + (\tilde{x}_{n+1})^{\uparrow r}), \quad (3)$$

where ψ_n is a fully convolutional net with 5 conv-blocks of the form Conv(3×3)-BatchNorm-LeakyReLU [25]. We start with 32 kernels per block at the coarsest scale and increase this number by a factor of 2 every 4 scales. Because the generators are fully convolutional, we can generate images of arbitrary size and aspect ratio at test time (by changing the dimensions of the noise maps).

2.2. Training

We train our multi-scale architecture sequentially, from the coarsest scale to the finest one. Once each GAN is trained, it is kept fixed. Our training loss for the n th GAN is comprised of an adversarial term and a reconstruction term,

$$\min_{G_n} \max_{D_n} \mathcal{L}_{\text{adv}}(G_n, D_n) + \alpha \mathcal{L}_{\text{rec}}(G_n). \quad (4)$$

The adversarial loss \mathcal{L}_{adv} penalizes for the distance between the distribution of patches in x_n and the distribution of patches in generated samples \tilde{x}_n . The reconstruction loss \mathcal{L}_{rec} insures the existence of a specific set of noise maps that can produce x_n , an important feature for image manipulation (Sec. 4). We next describe \mathcal{L}_{adv} , \mathcal{L}_{rec} in detail. See Supplementary Materials (SM) for optimization details.

Adversarial loss Each of the generators G_n is coupled with a Markovian discriminator D_n that classifies each of the overlapping patches of its input as real or fake [31, 26]. We use the WGAN-GP loss [20], which we found to increase training stability, where the final discrimination score is the average over the patch discrimination map. As opposed to single-image GANs for textures (*e.g.*, [31, 27, 3]), here we define the loss over the whole image rather than over random crops (a batch of size 1). This allows the net to learn boundary conditions (see SM), which is an important feature in our setting. The architecture of D_n is the same as the net ψ_n within G_n , so that its patch size (the net's receptive field) is 11×11 .

Reconstruction loss We want to ensure that there exists a specific set of input noise maps, which generates the original image x . We specifically choose $\{z_N^{\text{rec}}, z_{N-1}^{\text{rec}}, \dots, z_0^{\text{rec}}\} = \{z^*, 0, \dots, 0\}$, where z^* is some fixed noise map (drawn once and kept fixed during training). Denote by \tilde{x}_n^{rec} the generated image at the n th scale when using these noise maps. Then for $n < N$,

$$\mathcal{L}_{\text{rec}} = \|G_n(0, (\tilde{x}_{n+1})^{\uparrow r}) - x_n\|^2, \quad (5)$$

and for $n = N$, we use $\mathcal{L}_{\text{rec}} = \|G_N(z^*) - x_N\|^2$.

The reconstructed image \tilde{x}_n^{rec} has another role during training, which is to determine the standard deviation σ_n of the noise z_n in each scale. Specifically, we take σ_n to be proportional to the root mean squared error (RMSE) between $(\tilde{x}_{n+1})^{\uparrow r}$ and x_n , which gives an indication of the amount of details that need to be added at that scale.

3. Results

We tested our method both qualitatively and quantitatively on a variety of images spanning a large range of scenes including urban and nature scenery as well as artistic and texture images. The images that we used are taken from the Berkeley Segmentation Database (BSD) [35], Places [59] and the Web. We always set the minimal dimension at the coarsest scale to 25px, and choose the number of scales N s.t. the scaling factor r is as close as possible to 4/3. For all the results, (unless mentioned otherwise), we resized the training image to maximal dimension 250px.

Qualitative examples of our generated random image samples are shown in Fig. 1, Fig. 6, and many more examples are included in the SM. For each example, we show a number of random samples with the same aspect ratio as the original image, and with decreased and expanded dimensions in each axis. As can be seen, in all these cases, the generated samples depict new realistic structures and configuration of objects, while preserving the visual content of the training image. Our model successfully preserves global structure of objects, *e.g.* mountains (Fig. 1), air balloons or pyramids (Fig. 6), as well as fine texture information. Because the network has a limited receptive field (smaller than the entire image), it can generate new combinations of patches that do not exist in the training image. Furthermore, we observe that in many cases reflections and shadows are realistically synthesized, as can be seen in Fig. 6 and Fig. 1 (and the first example of Fig. 8). Note that SinGAN's architecture is resolution agnostic and can thus be used on high resolution images, as illustrated in Fig. 7 (see 4Mpix results in the SM). Here as well, structures at all scales are nicely generated, from the global arrangement of sky, clouds and mountains, to the fine textures of the snow.

Effect of scales at test time Our multi-scale architecture allows control over the amount of variability between samples, by choosing the scale from which to start the generation at test time. To start at scale n , we fix the noise maps up

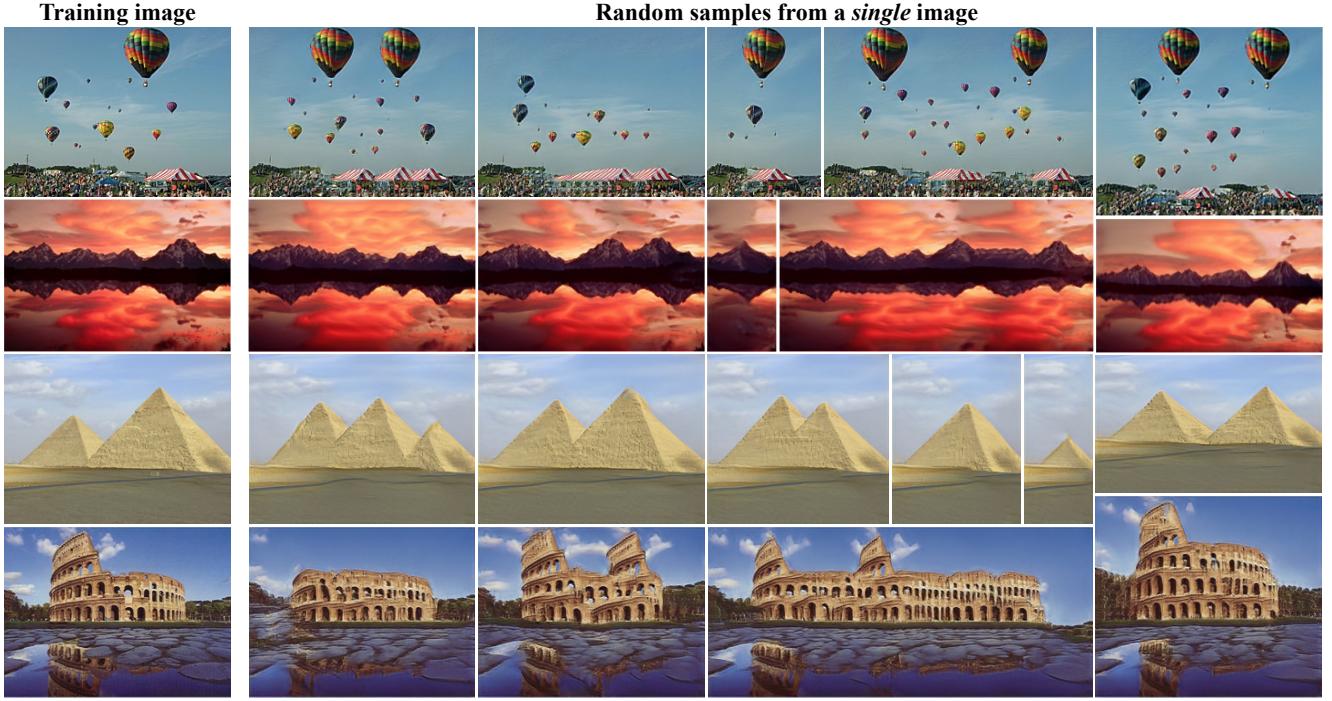


Figure 6: **Random image samples.** After training SinGAN on a single image, our model can generate realistic random image samples that depict new structures and object configurations, yet preserve the patch distribution of the training image. Because our model is fully convolutional, the generated images may have arbitrary sizes and aspect ratios. Note that our goal is not image retargeting – our image samples are random and optimized to maintain the patch statistics, rather than preserving salient objects. See SM for more results and qualitative comparison to image retargeting methods.



Figure 7: **High resolution image generation.** A random sample produced by our model, trained on the 243×1024 image (upper right corner); new global structures as well as fine details are realistically generated. See 4Mpix examples in SM.

to this scale to be $\{z_N^{\text{rec}}, \dots, z_{n+1}^{\text{rec}}\}$, and use random draws only for $\{z_n, \dots, z_0\}$. The effect is illustrated in Fig. 8. As can be seen, starting the generation at the coarsest scale ($n = N$), results in large variability in the global structure. In certain cases with a large salient object, like the Zebra image, this may lead to unrealistic samples. However, starting the generation from finer scales, enables to keep the global structure intact, while altering only finer image features (*e.g.* the Zebra’s stripes). See SM for more examples.

Effect of scales during training Figure 9 shows the effect of training with fewer scales. With a small number of scales, the effective receptive field at the coarsest level is

smaller, allowing to capture only fine textures. As the number of scales increases, structures of larger support emerge, and the global object arrangement is better preserved.

3.1. Quantitative Evaluation

To quantify the realism of our generated images and how well they capture the internal statistics of the training image, we use two metrics: (i) Amazon Mechanical Turk (AMT) “Real/Fake” user study, and (ii) a new single-image version of the Fréchet Inception Distance [23].

AMT perceptual study We followed the protocol of [26, 58] and performed perceptual experiments in 2 settings.

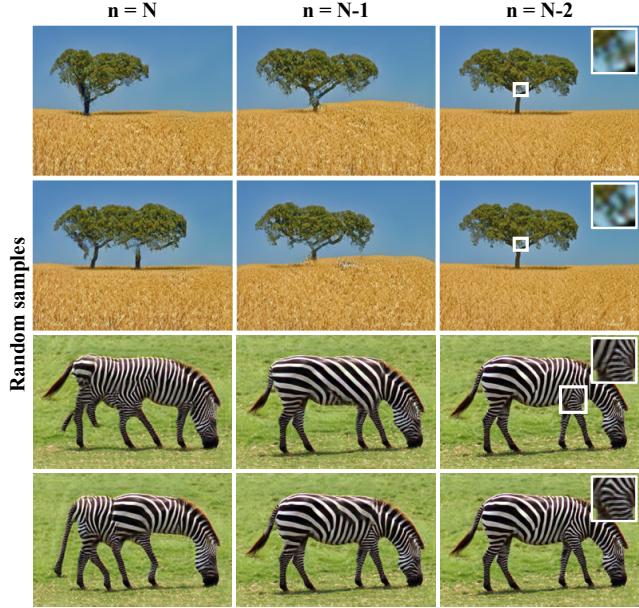


Figure 8: Generation from different scales (at inference). We show the effect of starting our hierarchical generation from a given level n . For our full generation scheme ($n = N$), the input at the coarsest level is random noise. For generation from a finer scale n , we plug in the downsampled original image, x_n , as input to that scale. This allows us to control the scale of the generated structures, *e.g.*, we can preserve the shape and pose of the Zebra and only change its stripe texture by starting the generation from $n = N - 1$.

(i) Paired (real vs. fake): Workers were presented with a sequence of 50 trials, in each of which a fake image (generated by SinGAN) was presented against its real training image for 1 second. Workers were asked to pick the fake image.
(ii) Unpaired (either real or fake): Workers were presented with a *single* image for 1 second, and were asked if it was fake. In total, 50 real images and a disjoint set of 50 fake images were presented in random order to each worker.

We repeated these two protocols for two types of generation processes: Starting the generation from the coarsest (N th) scale, and from scale $N - 1$ (as in Fig. 8). This way, we assess the realism of our results in two different variability levels. To quantify the diversity of the generated images, for each training example we calculated the standard deviation (std) of the intensity values of each pixel over 100 generated images, averaged it over all pixels, and normalized by the std of the intensity values of the training image.

The real images were randomly picked from the “places” database [59] from the subcategories Mountains, Hills, Desert, Sky. In each of the 4 tests, we had 50 different participants. In all tests, the first 10 trials were a tutorial including a feedback. The results are reported in Table 1.

As expected, the confusion rates are consistently larger in the unpaired case, where there is no reference for comparison. In addition, it is clear that the confusion rate decreases

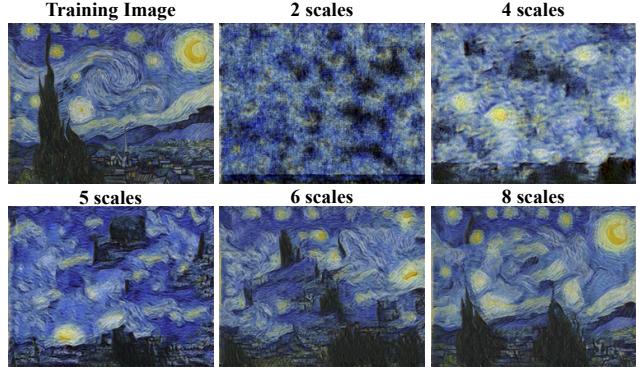


Figure 9: The effect of training with a different number of scales. The number of scales in SinGAN’s architecture strongly influences the results. A model with a small number of scales only captures textures. As the number of scales increases, SinGAN manages to capture larger structures as well as the global arrangement of objects in the scene.

1st Scale	Diversity	Survey	Confusion
N	0.5	paired unpaired	$21.45\% \pm 1.5\%$ $42.9\% \pm 0.9\%$
	0.35	paired unpaired	$30.45\% \pm 1.5\%$ $47.04\% \pm 0.8\%$
$N - 1$	0.35	paired unpaired	$30.45\% \pm 1.5\%$ $47.04\% \pm 0.8\%$

Table 1: “Real/Fake” AMT test. We report confusion rates for two generation processes: Starting from the coarsest scale N (producing samples with large diversity), and starting from the second coarsest scale $N - 1$ (preserving the global structure of the original image). In each case, we performed both a paired study (real-vs.-fake image pairs are shown), and an unpaired one (either fake or real image is shown). The variance was estimated by bootstrap [14].

with the diversity of the generated images. However, even when large structures are changed, our generated images were hard to distinguish from the real images (a score of 50% would mean perfect confusion between real and fake). The full set of test images are included in the SM.

Single Image Fréchet Inception Distance We next quantify how well SinGAN captures the internal statistics of x . A common metric for GAN evaluation is the Fréchet Inception Distance (FID) [23], which measures the deviation between the distribution of deep features of generated images and that of real images. In our setting, however, we only have a single real image, and are rather interested in its *internal* patch statistics. We thus propose the Single Image FID (SIFID) metric. Instead of using the activation vector after the last pooling layer in the Inception Network [49] (a single vector per image), we use the internal distribution of deep features at the output of the convolutional layer just before the second pooling layer (one vector per location in the map). Our SIFID is the FID between the statistics of those features in the real image and in the generated sample.

As can be seen in Table 2, the average SIFID is lower for

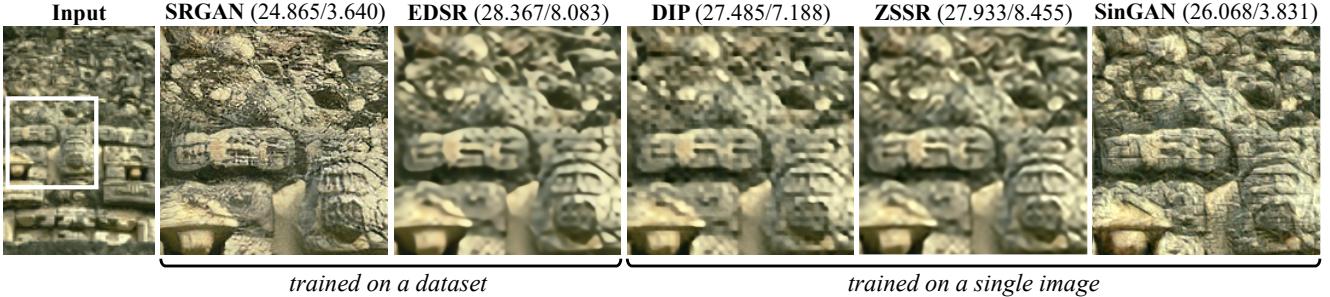


Figure 10: **Super-Resolution.** When SinGAN is trained on a low resolution image, we are able to super resolve. This is done by iteratively upsampling the image and feeding it to SinGAN’s finest scale generator. As can be seen, SinGAN’s visual quality is better than the SOTA internal methods ZSSR [46] and DIP [51]. It is also better than EDSR [32] and comparable to SRGAN [30], external methods trained on large collections. Corresponding PSNR and NIQE [40] are shown in parentheses.

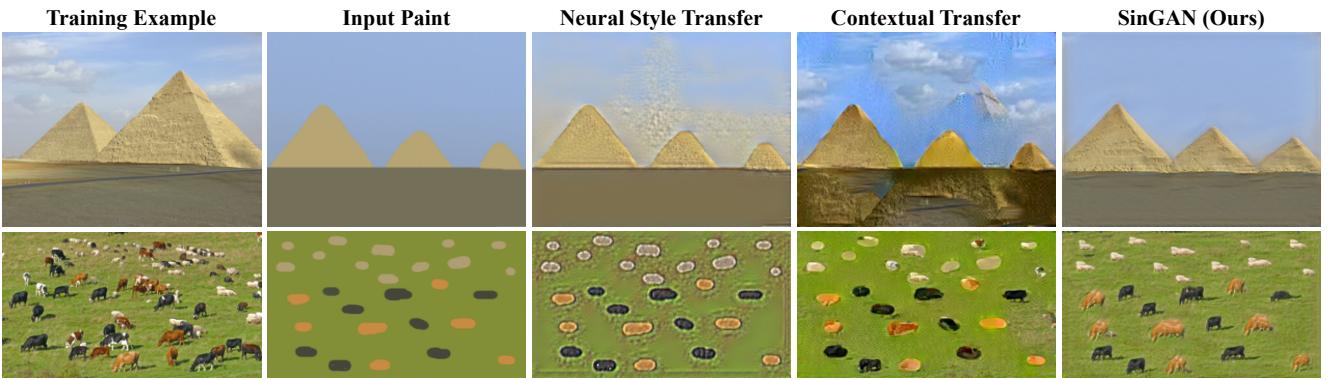


Figure 11: **Paint-to-Image.** We train SinGAN on a target image and inject a downsampled version of the paint into one of the coarse levels at test time. Our generated images preserve the layout and general structure of the clipart while generating realistic texture and fine details that match the training image. Well-known style transfer methods [17, 38] fail in this task.

1st Scale	SIFID	Survey	SIFID/AMT Correlation
N	0.09	paired	-0.55
		unpaired	-0.22
$N - 1$	0.05	paired	-0.56
		unpaired	-0.34

Table 2: **Single Image FID (SIFID).** We adapt the FID metric to a single image and report the average score for 50 images, for full generation (first row), and starting from the second coarsest scale (second row). Correlation with AMT results shows SIFID highly agrees with human ranking.

generation from scale $N - 1$ than for generation from scale N , which aligns with the user study results. We also report the correlation between the SIFID scores and the confusion rates for the fake images. Note that there is a significant (anti) correlation between the two, implying that a small SIFID is typically a good indicator for a large confusion rate. The correlation is stronger for the paired tests, since SIFID is a paired measure (it operates on the pair x_n, \tilde{x}_n).

4. Applications

We explore the use of SinGAN for a number of image manipulation tasks. To do so, we use our model *after train-*

ing, with no architectural changes or further tuning and follow the same approach for all applications. The idea is to utilize the fact that at inference, SinGAN can only produce images with the same patch distribution as the training image. Thus, manipulation can be done by injecting (a possibly downsampled version of) an image into the generation pyramid at some scale $n < N$, and feed forwarding it through the generators so as to match its patch distribution to that of the training image. Different injection scales lead to different effects. We consider the following applications (see SM for more results and the injection scale effect).

Super-Resolution *Increase the resolution of an input image by a factor s .* We train our model on the low-resolution (LR) image, with a reconstruction loss weight of $\alpha = 100$ and a pyramid scale factor of $r = \sqrt[k]{s}$ for some $k \in \mathbb{N}$. Since small structures tend to recur across scales of natural scenes [18], at test time we upsample the LR image by a factor of r and inject it (together with noise) to the last generator, G_0 . We repeat this k times to obtain the final high-res output. An example result is shown in Fig. 10. As can be seen, the visual quality of our reconstruction exceeds that of state-of-the-art *internal* methods [51, 46] as well as *external* methods that aim for PSNR maximization [32].

	External methods		Internal methods		
	SRGAN	EDSR	DIP	ZSSR	SinGAN
RMSE	16.34	12.29	13.82	13.08	16.22
NIQE	3.41	6.50	6.35	7.13	3.71

Table 3: **Super-Resolution evaluation.** Following [5], we report distortion (RMSE) and perceptual quality (NIQE [40], lower is better) on BSD100 [35]. As can be seen, SinGAN’s performance is similar to that of SRGAN [30].

Interestingly, it is comparable to the externally trained SRGAN method [30], despite having been exposed to only a single image. Following [4], we compare these 5 methods in Table 3 on the BSD100 dataset [35] in terms of distortion (RMSE) and perceptual quality (NIQE [40]), which are two fundamentally conflicting requirements [5]. As can be seen, SinGAN excels in perceptual quality; its NIQE score is only slightly inferior to SRGAN, and its RMSE is slightly better.

Paint-to-Image *Transfer a clipart into a photo-realistic image.* This is done by downsampling the clipart image and feeding it into one of the coarse scales (*e.g.* $N - 1$ or $N - 2$). As can be seen in Figs. 2 and 11, the global structure of the painting is preserved, while texture and high frequency information matching the original image are realistically generated. Our method outperforms style transfer methods [38, 17] in terms of visual quality (Fig. 11).

Harmonization *Realistically blend a pasted object with a background image.* We train SinGAN on the background image, and inject a downsampled version of the naively pasted composite at test time. Here we combine the generated image with the original background. As can be seen in Fig. 2 and Fig. 13, our model tailors the pasted object’s texture to match the background, and often preserves its structure better than [34]. Scales 2,3,4 typically lead to good balance between preserving the object’s structure and transferring the background’s texture.

Editing *Produce a seamless composite in which image regions have been copied and pasted in other locations.* Here, again, we inject a downsampled version of the composite into one of the coarse scales. We then combine SinGAN’s output at the edited regions, with the original image. As shown in Fig. 2 and Fig. 12, SinGAN re-generates fine textures and seamlessly stitches the pasted parts, producing nicer results than Photoshop’s Content-Aware-Move.

Single Image Animation *Create a short video clip with realistic object motion, from a single input image.* Natural images often contain repetitions, which reveal different “snapshots” in time of the same dynamic object [55] (*e.g.* an image of a flock of birds reveals all wing postures of a single bird). Using SinGAN, we can travel along the manifold of all appearances of the object in the image, thus synthesizing motion from a single image. We found that for many types of images, a realistic effect is achieved by a random walk in z -space, starting with z^{rec} for the first

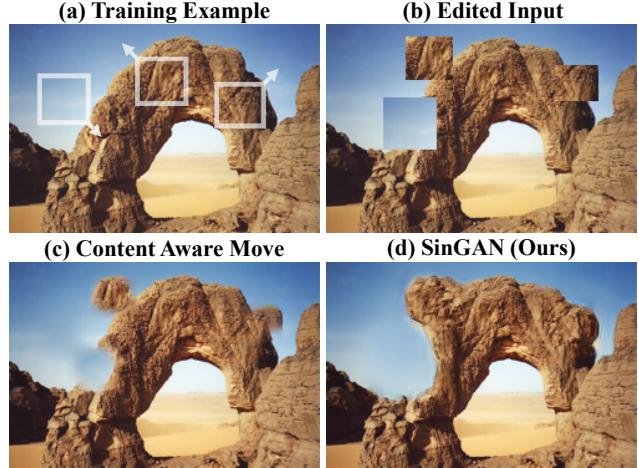


Figure 12: **Editing.** We copy and paste a few patches from the original image (a), and input a downsampled version of the edited image (b) to an intermediate level of our model (pretrained on (a)). In the generated image (d), these local edits are translated into coherent and photo-realistic structures. (c) comparison to Photoshop content aware move.



Figure 13: **Harmonization.** Our model is able to preserve the structure of the pasted object, while adjusting its appearance and texture. The dedicated harmonization method [34] overly blends the object with the background.

frame at all generation scales. Results are available on <https://youtu.be/xk8bWLZk4DU>.

5. Conclusion

We introduced SinGAN, a new unconditional generative scheme that is learned from a single natural image. We demonstrated its ability to go beyond textures and to generate diverse realistic samples for natural complex images. Internal learning is inherently limited in terms of *semantic* diversity compared to externally trained generation methods. For example, if the training image contains a single dog, our model will not generate samples of different dog breeds. Nevertheless, as demonstrated by our experiments, SinGAN can provide a very powerful tool for a wide range of image manipulation tasks.

Acknowledgements Thanks to Idan Kligvasser for valuable insights. This research was supported by the Israel Science Foundation (grant 852/17) and the Ollendorff foundation.

References

- [1] Yuki M Asano, Christian Rupprecht, and Andrea Vedaldi. Surprising effectiveness of few-image unsupervised feature learning. *arXiv preprint arXiv:1904.13132*, 2019. [2](#)
- [2] Yuval Bahat and Michal Irani. Blind dehazing using internal patch recurrence. In *2016 IEEE International Conference on Computational Photography (ICCP)*, pages 1–9. IEEE, 2016. [1](#)
- [3] Urs Bergmann, Nikolay Jetchev, and Roland Vollgraf. Learning texture manifolds with the periodic spatial GAN. *arXiv preprint arXiv:1705.06566*, 2017. [2, 4](#)
- [4] Yochai Blau, Roey Mechrez, Radu Timofte, Tomer Michaeli, and Lihi Zelnik-Manor. The 2018 pirm challenge on perceptual image super-resolution. In *European Conference on Computer Vision Workshops*, pages 334–355. Springer, 2018. [8](#)
- [5] Yochai Blau and Tomer Michaeli. The perception-distortion tradeoff. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 6228–6237, 2018. [8](#)
- [6] Andrew Brock, Jeff Donahue, and Karen Simonyan. Large scale GAN training for high fidelity natural image synthesis. *arXiv preprint arXiv:1809.11096*, 2018. [1](#)
- [7] Caroline Chan, Shiry Ginosar, Tinghui Zhou, and Alexei A Efros. Everybody dance now. *arXiv preprint arXiv:1808.07371*, 2018. [2](#)
- [8] Wengling Chen and James Hays. Sketchygan: towards diverse and realistic sketch to image synthesis. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 9416–9425, 2018. [2](#)
- [9] Taeg Sang Cho, Moshe Butman, Shai Avidan, and William T Freeman. The patch transform and its applications to image editing. In *2008 IEEE Conference on Computer Vision and Pattern Recognition*, pages 1–8. IEEE, 2008. [1](#)
- [10] Tali Dekel, Chuang Gan, Dilip Krishnan, Ce Liu, and William T Freeman. Sparse, smart contours to represent and edit images. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 3511–3520, 2018. [2](#)
- [11] Tali Dekel, Tomer Michaeli, Michal Irani, and William T Freeman. Revealing and modifying non-local variations in a single image. *ACM Transactions on Graphics (TOG)*, 34(6):227, 2015. [1](#)
- [12] J. Deng, W. Dong, R. Socher, L.-J. Li, K. Li, and L. Fei-Fei. ImageNet: A Large-Scale Hierarchical Image Database. In *CVPR09*, 2009. [1](#)
- [13] Emily L Denton, Soumith Chintala, Rob Fergus, et al. Deep generative image models using a laplacian pyramid of adversarial networks. In *Advances in neural information processing systems*, pages 1486–1494, 2015. [3](#)
- [14] Bradley Efron. Bootstrap methods: another look at the jackknife. In *Breakthroughs in statistics*, pages 569–593. Springer, 1992. [6](#)
- [15] Gilad Freedman and Raanan Fattal. Image and video upscaling from local self-examples. *ACM Transactions on Graphics (TOG)*, 30(2):12, 2011. [1](#)
- [16] Leon Gatys, Alexander S Ecker, and Matthias Bethge. Texture synthesis using convolutional neural networks. In *Advances in neural information processing systems*, pages 262–270, 2015. [2](#)
- [17] Leon A Gatys, Alexander S Ecker, and Matthias Bethge. Image style transfer using convolutional neural networks. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 2414–2423, 2016. [7, 8](#)
- [18] Daniel Glasner, Shai Bagon, and Michal Irani. Super-resolution from a single image. In *2009 IEEE 12th International Conference on Computer Vision (ICCV)*, pages 349–356. IEEE, 2009. [1, 7](#)
- [19] Ian Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, and Yoshua Bengio. Generative adversarial nets. In *Advances in Neural Information Processing Systems*, pages 2672–2680, 2014. [1](#)
- [20] Ishaan Gulrajani, Faruk Ahmed, Martin Arjovsky, Vincent Dumoulin, and Aaron C Courville. Improved training of wasserstein GANs. In *Advances in Neural Information Processing Systems*, pages 5767–5777, 2017. [4](#)
- [21] Kaiming He and Jian Sun. Statistics of patch offsets for image completion. In *European Conference on Computer Vision*, pages 16–29. Springer, 2012. [1](#)
- [22] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 770–778, 2016. [4](#)
- [23] Martin Heusel, Hubert Ramsauer, Thomas Unterthiner, Bernhard Nessler, and Sepp Hochreiter. GANs trained by a two time-scale update rule converge to a local nash equilibrium. In *Advances in Neural Information Processing Systems*, pages 6626–6637, 2017. [5, 6](#)
- [24] Xun Huang, Yixuan Li, Omid Poursaeed, John Hopcroft, and Serge Belongie. Stacked generative adversarial networks. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 5077–5086, 2017. [3](#)
- [25] Sergey Ioffe and Christian Szegedy. Batch normalization: Accelerating deep network training by reducing internal covariate shift. *arXiv preprint arXiv:1502.03167*, 2015. [4](#)
- [26] Phillip Isola, Jun-Yan Zhu, Tinghui Zhou, and Alexei A Efros. Image-to-image translation with conditional adversarial networks. *arXiv preprint*, 2017. [3, 4, 5](#)
- [27] Nikolay Jetchev, Urs Bergmann, and Roland Vollgraf. Texture synthesis with spatial generative adversarial networks. *Workshop on Adversarial Training, NIPS*, 2016. [2, 4](#)
- [28] Tero Karras, Timo Aila, Samuli Laine, and Jaakkko Lehtinen. Progressive growing of GANs for improved quality, stability, and variation. *arXiv preprint arXiv:1710.10196*, 2017. [3](#)
- [29] Tero Karras, Samuli Laine, and Timo Aila. A style-based generator architecture for generative adversarial networks. *arXiv preprint arXiv:1812.04948*, 2018. [3](#)
- [30] Christian Ledig, Lucas Theis, Ferenc Huszár, Jose Caballero, Andrew Cunningham, Alejandro Acosta, Andrew Aitken, Alykhan Tejani, Johannes Totz, Zehan Wang, et al. Photo-realistic single image super-resolution using a generative adversarial network. In *Proceedings of the IEEE conference on*

- computer vision and pattern recognition*, pages 4681–4690, 2017. 1, 7, 8
- [31] Chuan Li and Michael Wand. Precomputed real-time texture synthesis with markovian generative adversarial networks. In *European Conference on Computer Vision*, pages 702–716. Springer, 2016. 2, 3, 4
- [32] Bee Lim, Sanghyun Son, Heewon Kim, Seungjun Nah, and Kyoung Mu Lee. Enhanced deep residual networks for single image super-resolution. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops*, pages 136–144, 2017. 7
- [33] Ziwei Liu, Ping Luo, Xiaogang Wang, and Xiaoou Tang. Deep learning face attributes in the wild. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 3730–3738, 2015. 1
- [34] Fujun Luan, Sylvain Paris, Eli Shechtman, and Kavita Bala. Deep painterly harmonization. *arXiv preprint arXiv:1804.03189*, 2018. 8
- [35] David Martin, Charless Fowlkes, Doron Tal, and Jitendra Malik. A database of human segmented natural images and its application to evaluating segmentation algorithms and measuring ecological statistics. In *null*, page 416. IEEE, 2001. 4, 8
- [36] Michael Mathieu, Camille Couprie, and Yann LeCun. Deep multi-scale video prediction beyond mean square error. *arXiv preprint arXiv:1511.05440*, 2015. 4
- [37] Roey Mechrez, Eli Shechtman, and Lihi Zelnik-Manor. Saliency driven image manipulation. In *2018 IEEE Winter Conference on Applications of Computer Vision (WACV)*, pages 1368–1376. IEEE, 2018. 1
- [38] Roey Mechrez, Itamar Talmi, and Lihi Zelnik-Manor. The contextual loss for image transformation with non-aligned data. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pages 768–783, 2018. 7, 8
- [39] Tomer Michaeli and Michal Irani. Blind deblurring using internal patch recurrence. In *European Conference on Computer Vision*, pages 783–798. Springer, 2014. 1
- [40] Anish Mittal, Rajiv Soundararajan, and Alan C Bovik. Making a completely blind image quality analyzer. *IEEE Signal Processing Letters*, 20(3):209–212, 2013. 7, 8
- [41] Deepak Pathak, Philipp Krahenbuhl, Jeff Donahue, Trevor Darrell, and Alexei A Efros. Context encoders: Feature learning by inpainting. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 2536–2544, 2016. 1
- [42] Guim Perarnau, Joost van de Weijer, Bogdan Raducanu, and Jose M Álvarez. Invertible conditional GANs for image editing. *arXiv preprint arXiv:1611.06355*, 2016. 2
- [43] Patsorn Sangkloy, Jingwan Lu, Chen Fang, Fisher Yu, and James Hays. Scribbler: Controlling deep image synthesis with sketch and color. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 5400–5409, 2017. 2
- [44] Assaf Shocher, Shai Bagon, Phillip Isola, and Michal Irani. Ingan: Capturing and remapping the “DNA” of a natural image. *arXiv preprint arXiv: arXiv:1812.00231*, 2018. 2
- [45] Assaf Shocher, Shai Bagon, Phillip Isola, and Michal Irani. InGAN: Capturing and Remapping the “DNA” of a Natural Image. *International Conference on Computer Vision (ICCV)*, 2019. 1, 2
- [46] Assaf Shocher, Nadav Cohen, and Michal Irani. Zero-Shot Super-Resolution using Deep Internal Learning. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 3118–3126, 2018. 2, 7
- [47] Nathan Silberman, Derek Hoiem, Pushmeet Kohli, and Rob Fergus. Indoor segmentation and support inference from rgbd images. In *European Conference on Computer Vision*, pages 746–760. Springer, 2012. 1
- [48] Denis Simakov, Yaron Caspi, Eli Shechtman, and Michal Irani. Summarizing visual data using bidirectional similarity. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 1–8. IEEE, 2008. 1
- [49] Christian Szegedy, Wei Liu, Yangqing Jia, Pierre Sermanet, Scott Reed, Dragomir Anguelov, Dumitru Erhan, Vincent Vanhoucke, and Andrew Rabinovich. Going deeper with convolutions. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 1–9, 2015. 6
- [50] Tal Tlusty, Tomer Michaeli, Tali Dekel, and Lihi Zelnik-Manor. Modifying non-local variations across multiple views. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 6276–6285, 2018. 1
- [51] Dmitry Ulyanov, Andrea Vedaldi, and Victor Lempitsky. Deep image prior. *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2018. 2, 7
- [52] Ting-Chun Wang, Ming-Yu Liu, Jun-Yan Zhu, Andrew Tao, Jan Kautz, and Bryan Catanzaro. High-resolution image synthesis and semantic manipulation with conditional GANs. *arXiv preprint arXiv:1711.11585*, 2017. 2, 3
- [53] Xiaolong Wang and Abhinav Gupta. Generative image modeling using style and structure adversarial networks. 2016. 2
- [54] Wenqi Xian, Patsorn Sangkloy, Varun Agrawal, Amit Raj, Jingwan Lu, Chen Fang, Fisher Yu, and James Hays. Texturegan: Controlling deep image synthesis with texture patches. In *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2018. 2
- [55] Xuemiao Xu, Liang Wan, Xiaopei Liu, Tien-Tsin Wong, Liansheng Wang, and Chi-Sing Leung. Animating animal motion from still. *ACM Transactions on Graphics (TOG)*, 27(5):117, 2008. 8
- [56] Jiahui Yu, Zhe Lin, Jimei Yang, Xiaohui Shen, Xin Lu, and Thomas S Huang. Generative image inpainting with contextual attention. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 5505–5514, 2018. 2
- [57] Kai Zhang, Wangmeng Zuo, Yunjin Chen, Deyu Meng, and Lei Zhang. Beyond a gaussian denoiser: Residual learning of deep cnn for image denoising. *IEEE Transactions on Image Processing*, 26(7):3142–3155, 2017. 4
- [58] Richard Zhang, Phillip Isola, and Alexei A Efros. Colorful image colorization. In *European conference on computer vision*, pages 649–666. Springer, 2016. 5
- [59] Bolei Zhou, Agata Lapedriza, Jianxiong Xiao, Antonio Torralba, and Aude Oliva. Learning deep features for scene recognition using places database. In *Advances in neural information processing systems*, pages 487–495, 2014. 4, 6
- [60] Yang Zhou, Zhen Zhu, Xiang Bai, Dani Lischinski, Daniel Cohen-Or, and Hui Huang. Non-stationary texture synthesis by adversarial expansion. *arXiv preprint arXiv:1805.04487*, 2018. 2

- [61] Jun-Yan Zhu, Philipp Krähenbühl, Eli Shechtman, and Alexei A Efros. Generative visual manipulation on the natural image manifold. In *European Conference on Computer Vision (ECCV)*, pages 597–613. Springer, 2016. [2](#)
- [62] Jun-Yan Zhu, Taesung Park, Phillip Isola, and Alexei A Efros. Unpaired image-to-image translation using cycle-consistent adversarial networks. In *IEEE International Conference on Computer Vision*, 2017. [2](#), [4](#)
- [63] Jun-Yan Zhu, Richard Zhang, Deepak Pathak, Trevor Darrell, Alexei A Efros, Oliver Wang, and Eli Shechtman. Toward multimodal image-to-image translation. In *Advances in Neural Information Processing Systems*, pages 465–476, 2017. [4](#)
- [64] Maria Zontak and Michal Irani. Internal statistics of a single natural image. In *CVPR 2011*, pages 977–984. IEEE, 2011. [1](#)
- [65] Maria Zontak, Inbar Mosseri, and Michal Irani. Separating signal from noise using patch recurrence across scales. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 1195–1202, 2013. [1](#)