Structural-analogy from a Single Image Pair

Sagie Benaim*[1] Ron Mokady*[1] Amit Bermano[1]
Daniel Cohen-Or[1] Lior Wolf[1,2]

[1]Tel Aviv University [2]Facebook AI Research

Abstract. The task of unsupervised image-to-image translation has seen substantial advancements in recent years through the use of deep neural networks. Typically, the proposed solutions learn the characterizing distribution of two large, unpaired collections of images, and are able to alter the appearance of a given image, while keeping its geometry intact. In this paper, we explore the capabilities of neural networks to understand image *structure* given only a single pair of images, A and B. We seek to generate images that are *structurally aligned*: that is, to generate an image that keeps the appearance and style of B, but has a structural arrangement that corresponds to A. The key idea is to map between image patches at different scales. This enables controlling the granularity at which analogies are produced, which determines the conceptual distinction between style and content. In addition to *structural alignment*, our method can be used to generate high quality imagery in other conditional generation tasks utilizing images A and B only: guided image synthesis, style and texture transfer, text translation as well as video translation. Our code and additional results are available in `https://github.com/rmokady/structural-analogy/`.

1 Introduction

The task of image-to-image translation has seen tremendous growth in recent years [3, 4, 14, 16–18, 23, 24, 26, 33, 36–38, 41, 42]. In the typical setting, an image is generated such that it depicts the same scene as in an input source image, but displays the visual properties of a target one. At first, supervision was in the form of explicit matching image pairs in both domains, illustrating the exact same content [14]. These sort of examples are hard to obtain, and hence researchers searched for ways to relax this requirement, yielding a series of *unpaired* translation methods. In this satisfying setting, a large collection of images is still required for each domain, but they do not need to be matched [16, 23, 24, 36, 41].

Recent works, however, have demonstrated that a lot of information can already be extracted from just a single image, due the information residing within the internal statistics of patches comprising the image [6, 27, 28, 32, 40]. While very insightful and inspiring, these works are restricted to learning the said distribution of a single image, and are not appropriate to the image-to-image translation task (see Section 2 for a more comprehensive discussion).

The literature can also be organized by the the distinction between content and style, although these concepts are not necessarily disjointed. When translating a bundle of

* Equal contribution.

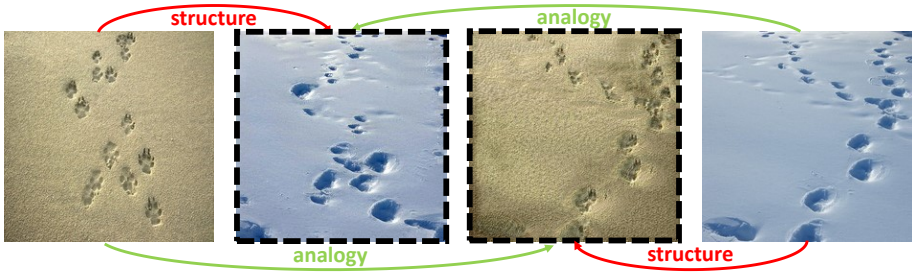


Fig. 1. Our method takes two images as input (left and right), and generates images that consist of features from one image, spatially structured analogically to the other.

balloons to a flock of birds, should the shape of each balloon be preserved? should the arrangement within the bundle? Should color stripes within a balloon be shaped like feathers? After successfully translating painting styles, textures [14, 41], and environment illumination [13, 23], recent work has experimented with changing shapes in the image, along with texture [1]. Other work consider the case of geometry altering transformations [20, 34].

In this paper, we present the first work to combine both of the discussed concepts, i.e. performs image-to-image translation from a single pair, and aims at learning structure along with appearance. Specifically, our work looks for *structural-analogies*, generating an image that is: (1) similarly *aligned*: i.e. it follows the spatial distribution, or *structure*, of features from the target image, and (2) *analogous*: i.e. it depicts features that correlate to the target structure, but are from the source image. This is in contrast, for example, to Deep Image Analogy [21], which looks for analogous image positions, effectively transferring appearance between the images while preserving shapes.

In our case, performing the analogy on only small scale features induces traditional appearance transfer, but the more scales we add, the larger the features that are being replaced during the translation, granting intuitive control over the style-vs.-content distinction. Such images can be seen in Figure 1, where the appearance is translated (e.g. sand becomes snow), but in addition larger features, such as the foot prints, are also translated as to their counterparts nicely and wholesomely. This is done with no supervision directing the translation to keep semantic structure intact, expect for a similarity loss on image patches.

At each scale, our method first generates a sample of the source domain, conditioned by the previous scale, and a random input, in a similar fashion to the previously proposed SinGAN [27]. Then, using a cycle consistency process [16, 36, 41], the sample is translated to the target domain, where it is expected to match the target image. The driving concept is that for a correctly structured (or *well aligned*) sample, a shape preserving style transfer would produce an image that is similar to the target one at every corresponding scale.

We demonstrate our method on the task of *structural alignment*, as well the conditional generation tasks of guided image synthesis, style and texture transfer and texture synthesis. We achieve comparable or favorable results to the state-of-the-art, demon-

strating that unsupervised image-to-image translation from only two images is indeed possible, thanks to the powerful tool of multi-scale structural analogy, and information residing in patch statistics of single images.

2 Related Work

In recent years, the emergence of Generative Adversarial Networks (GANs) [8] has changed the landscape of many fields, and in particular, led to an intense advancement in generative image modeling. GANs can learn how to generate images with a distribution that closely resembles the one of the training set. Nevertheless, unconditional GANs are still quite limited and require a huge amount of training data to achieve plausible visual results [2].

Conditional GANs on the other hand, generally allows higher fidelity and yield higher visual quality. The most notable advancement has been in the development of image-to-image translation methods, which dramatically revolutionized many applications. Pix2Pix [14] have presented a generic method that learns to translate between two domains. Their method is based on a conditional GAN that learns from (supervised) paired data. They utilize both a conditional GAN loss to generate realistic looking images and $L1$ loss to match the ground truth image. Pix2PixHD [33] uses two separate low-res and high-res networks to generate images of high resolution. SPADE [26] performs semantic image synthesis by introducing a new type of conditional normalization. BicycleGAN [42], as another example, injects random noise into the generator and attempts to recover the noise from the generated output. All of these works, however, require costly annotation of matching pairs of images, which is unfeasible for many applications.

Learning to translate from (unsupervised) unpaired data is more challenging. Taigman et al. [31] presented a method based on a conditional GANs to transfer between unpaired domains, using a perceptual loss that is based on a classifier. CycleGAN [41], DiscoGAN [16] and DualGAN [36] presented generic networks with cycle-consistency that learn to translate between two given unpaired sets without additional supervision. These works have lead the way to many unpaired image-to-image translation networks [13, 18, 22, 23]. Nevertheless, these techniques mainly transfer the appearance (or style) from one image to another, and struggle in transferring the shape or the structure, as required for creating an analogy that is truly faithful to the target domain.

A number of recent works have challenged this problem using various means. For example, Benaim et al. [1] use a disentangled latent space to map domain-specific content. Wu et al. [34] propose disentangling the space of images into a Cartesian product of appearance and geometry latent spaces to allow for a geometry-aware translation. Katzir et al. [15] propose applying the translation on deep feature maps of a VGG network. These approaches require a large collection of images, stratified in a way that exposes the structural information.

Recently, a number of works [27, 28, 40] presented GANs capable of learning and generating from the internal patch distribution of a single image. However, since the patch statistics are not trained in the context of another image, such single sample models cannot effectively map between two very different distributions. While technically, one

can force such models to perform such a mapping, our experiments show that this leads to poor outcomes.

Our work is also related to style transfer methods [5, 10]. The most notable work based on neural network is of Gatys et al. [7] and Huang and Belongie [12] which attempt to transfer the style or a source image to a target one. When applied at a very fine scale, our work compares with such methods, using a completely different technique. A specific work in the context of text stylization by example [35] maps a texture to a binary map of a letter or word. This is a much more limiting scenario than ours. Even though, as we demonstrate, our method is also able to produce pleasing images of stylized text.

Deep image analogies [21] is apparently the closest work to ours, since they use a single pair of images to transfer style. Unlike us, they assume that the two input images share similar semantics. By analyzing their VGG deep features, they mix content features and style features to synthesize novel images. Conceptually, their work is significantly different than ours, since they cannot change the structure of the image, moreover, their method is not a generative model, and no novel patches are generated. Lastly, we do not rely on a pre-trained classifier, which is a form of supervision.

The concept of visual analogy was explored in classical works, for example in the context of image re-targeting [29]. Two visual signals are defined to be visually similar if all patches of one (at multiple scales) are contained in the other (*completeness*), and vice versa (*coherence*). The same concepts are relevant to our setting: our key idea is to produce a mapping in which the patch distribution of a source image is mapped to its corresponding patch distribution of a target image and vice versa. When the multi-scale distributions match, in both directions, completeness and coherence are guaranteed.

3 Method

In unsupervised image to image translation, the learner is typically provided with images from two externally collected datasets, and considers the distributions governing these datasets to generate aligned mappings between the two distributions [3, 4, 16–18, 23, 24, 36–38, 41].

In our setting, we consider instead the internal patch distributions of two images only, A and B. For a given image x (either A or B), and for a particular scale $n = 0, 1, 2, \ldots, N$, let the distribution of images having the same patch distribution as x at this scale be P_x^n. In other words, since each image is composed of many patches at different scales, we learn from the single image x, a distribution of images P_x^n, that shares the same scale-dependent patch distributions. The downsampling factor of each scale n is given by r^{N-n} for some $r > 1$. Scale N, therefore, is the original image resolution for both A and B. Starting at the coarsest scale 0, and up to scale N, our method attempts to find a matching between each image $a_n \in P_A^n$ and an image $b_n \in P_B^n$.

For each scale n, our method employs two patch GANs [19, 41] for generating samples in P_A^n and P_B^n. Each patch-GAN consists of a fully convolutional generator G_A^n and discriminator D_A^n for A, and G_B^n and D_B^n for B. The architecture of G_A^n and D_A^n is outlined in Sec. 3.4. For simplicity, we describe the unconditional and conditional generation process from A to B. The generation process from B to A is entirely symmetric. An illustration of our method is provided in Fig 2.

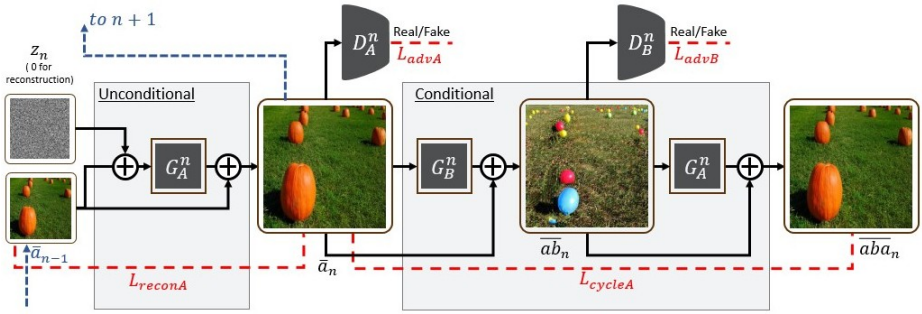


Fig. 2. An illustration of our method for the image of pumpkins, A, and an image of balls, B, in the direction of A to B for $0 < n < K$. G_A^n first unconditionally generates an image of pumpkins at scale n, by using as input the upscaled generated pumpkin image at the previous scale, $\uparrow \overline{a}_{n-1}$, with added random noise, z_n. The output of G_A^n is added to $\uparrow \overline{a}_{n-1}$ to generate $\overline{a}_n$. A patch-discriminator D_A^n is used to ensure $\overline{a}_n$ belongs to P_A^n (images of pumpkins at scale n). G_B^n then maps $\overline{a}_n$ to $\overline{ab}_n$ by adding the minimal amount of detail to $\overline{a}_n$. A patch-disciminator D_B^n ensures that $\overline{ab}_n$ belongs to P_B^n (images of balls at scale n). G_n^A is then used to map $\overline{ab}_n$ to $\overline{aba}_n$. To ensure that $\overline{a}_n$ and $\overline{ab}_n$ are aligned, a cycle loss is employed between to $\overline{a}_n$ and $\overline{aba}_n$.

3.1 Unconditional sample generation

For a scale n, to map between P_A^n and P_B^n, we first generate a sample $a \in P_A^n$ in an unconditional fashion. To do so, we follow, for the unconditional case, a similar procedure to the recently proposed SinGAN algorithm [27], for the first K scales. Let a_n be the sample A down-sampled by a factor of n, and z_n be Gaussian noise of the same shape and dimension as a_n. At the coarsest scale, $n = 0$, the output of G_A^0 is defined as (overline is used to denote generated images):

$$\overline{a}_0 = G_A^0(z_0) \tag{1}$$

Moving to finer scales $n > 0$, the generator accepts as input both z_n and an upscaled image of $\overline{a}_{n-1}$ to the dimension of a_n, denoted by $\uparrow \overline{a}_{n-1}$. For $n < K$, for some fixed K, we opt for using residual training:

$$\overline{a}_n = G_A^n(z_n + \uparrow \overline{a}_{n-1}) + \uparrow \overline{a}_{n-1} \tag{2}$$

This teaches the network to add missing details of $\uparrow \overline{a}_{n-1}$ specific to scale n. For $n \geq K$, we do not use residual training:

$$\overline{a}_n = G_A^n(z_n + \uparrow \overline{a}_{n-1}) \tag{3}$$

K is chosen as a hyperparameter which helps determine the level at which aligned mappings are produced (see Sec. 3.4 for details).

3.2 Conditional sample generation

Given the unconditional sample $a \in P_A^n$ generated at scale n, we now wish to map it to its corresponding sample $b \in P_B^n$. The same generator is used for both upscaling (Eq. 2

and Eq. 3) and for mapping between the domains. For $n < K$, we also use residual training:

$$\overline{ab}_n = G_B^n(\overline{a}_n) + \overline{a}_n \tag{4}$$

For $n > K$, no residual training is applied (see Sec. 3.4 for details):

$$\overline{ab}_n = G_B^n(\overline{a}_n) \tag{5}$$

Our key observation is that for each scale, a good alignment is one where each patch in the patch distribution of A is mapped to its corresponding patch in the patch distribution of B and vice versa. The goal of this mapping is, therefore, that: (1) at the coarsest scale $n = 0$, when mapping an image $a \in P_A^0$ to $b \in P_B^0$, G_B^0 should map global structure of a to global structure of b. For example, the ground at the bottom of a should be mapped to grass at the bottom of b, and daylight in a may be mapped to darker light in b. (2) at finer scales $n > 0$, G_B^n generates finer details in P_B^n, and so beyond the global, G_B^n should map finer details, such as style or texture in a to such details in b. This goal is achieved through the various loss terms employed by our method.

3.3 Loss Terms

Adversarial Loss. To ensure that $\overline{a}_n$ belongs to P_A^n and $\overline{ab}_n$ belongs to P_B^n, we employ adversarial patch-GAN loss using a markovian discriminator D_n^A [19, 41]. In particular, D_n^A is fully convolutional and uses the same architecture as G_n^A to produce a map of the same dimension of $\overline{a}_n$. D_n^A maps each (overlapping) patch of its input as real or fake. In particular, G_n^A attempts to fool D_n^A into classifying $\overline{a}_n$'s patches as real, while D_n^A attempts to classify $\overline{a}_n$'s patches as fake and a_n's patches as real. We use the following loss for G_n^A and D_n^A:

$$\mathcal{L}_{adv}^1(D_n^A, G_n^A) = l(D_n^A(\overline{a}_n), \mathbf{0}) + l(D_n^A(a_n), \mathbf{1})$$

Where $\mathbf{0}$ (resp. $\mathbf{1}$) is a matrix of the same shape as a_n that contains 0s (respectively 1s), and l is the WGAN-GP loss [9]. The fake image $\overline{a}_n$ is randomly generated by G_n^A, using equations 1, 2 and 3 by sampling z_n as a Gaussian noise. Note that the losses are defined for a batch size of one, as used in our experiments.

Similarly, G_n^B attempts to fool D_n^B into classifying $\overline{ab}_n$'s patches as real, while D_n^B attempts to classify $\overline{ab}_n$'s patches as fake and b_n's patches as real:

$$\mathcal{L}_{adv}^2(D_n^B, G_n^B) = l(D_n^B(\overline{ab}_n), \mathbf{0}) + l(D_n^B(b_n), \mathbf{1})$$

We also consider the analogous losses in the direction of B to A:

$$\mathcal{L}_{adv}^1(D_n^B, G_n^B) = l(D_n^B(\overline{b}_n), \mathbf{0}) + l(D_n^B(b_n), \mathbf{1})$$
$$\mathcal{L}_{adv}^2(D_n^A, G_n^A) = l(D_n^A(\overline{ba}_n), \mathbf{0}) + l(D_n^A(a_n), \mathbf{1})$$

We sum these four loss terms to define:

$$\mathcal{L}_{adv_n} = \mathcal{L}_{adv}^1(D_n^A, G_n^A) + \mathcal{L}_{adv}^1(D_n^B, G_n^B) + \mathcal{L}_{adv}^2(D_n^A, G_n^A) + \mathcal{L}_{adv}^2(D_n^B, G_n^B) \tag{6}$$

Reconstruction Loss. For $n > 0$, when no noise is used, we would like the generator G_A^n to reconstruct a_n and G_B^n to reconstruct b_n. Specifically:

$$\mathcal{L}^A_{recon_n} = ||G^n_A(\uparrow \overline{a}_{n-1}) - a_n||_2 \qquad \mathcal{L}^B_{recon_n} = ||G^n_B(\uparrow \overline{b}_{n-1}) - b_n||_2$$

When $n = 0$, we employ a fixed random noise $z_0 = z_a^*$ or $z_0 = z_b^*$ and require $\mathcal{L}^A_{recon_0} = ||G^0_A(z_a^*) - a_0||_2$, $\mathcal{L}^B_{recon_0} = ||G^0_B(z_b^*) - b_0||_2$. We define:

$$\mathcal{L}_{recon_n} = \mathcal{L}^A_{recon_n} + \mathcal{L}^B_{recon_n} \tag{7}$$

The reconstruction loss is used to control σ_n, the standard deviation of the Gaussian noise z_n, which indicates the level of detail required at each scale. In particular, σ_n is taken to be $|| \uparrow \overline{a}_{n-1} - a_n||_2$. In addition, it acts as the identity loss used in CycleGAN [41]. Without it, G_n^A and G_n^B can tint input images, as shown in the ablation study of Sec. 4.3.

Cycle Loss. To encourage the mapping between the two patch distribution to be aligned, we employ a cycle loss [16, 36, 41] for $n = 0, 1, .., K-1$. For $n \geq K$, it is not applied. The mapping back from $\overline{ab}_n$ to P_A^n and from $\overline{ba}_n$ to P_B^n is given as $\overline{bab}_n = G_B^n(\overline{ba}_n)$ and $\overline{bab}_n = G_B^n(\overline{ba}_n)$. The cycle loss is given by:

$$\mathcal{L}_{cycle_n} = ||\overline{a}_n - \overline{aba}_n||_2 + ||\overline{b}_n - \overline{bab}_n||_2 \tag{8}$$

The overall loss at scale n, for two hyper-parameters $\lambda_{cycle}, \lambda_{recon} > 0$, is:

$$\mathcal{L}_n = \min_{G_n^A, G_n^B} \max_{D_n^A, D_n^B} \mathcal{L}_{adv_n} + \lambda_{recon}\mathcal{L}_{recon_n} + \lambda_{cycle}\mathcal{L}_{cycle_n} \tag{9}$$

3.4 Training

Our algorithm trains the networks of each scale one at a time, keeping the networks of the previous scales fixed. In particular, after a set number of epochs (typically $10,000$), we stop training the networks at scale $n-1$ and move to scale n. When moving to scale n, the weights of the networks at this scale are initialized with the weights of the networks at the previous scale $n-1$. As shown in Sec. 4.3, such initialization is required to generate aligned mappings.

Our method enables a control over the type of alignment produced. This is in part through the choice of K. For $n < K$, we use residual training for both unconditional and conditional generation (Eq. 2 and Eq. 4). For the unconditional generation, this teaches the network to add missing details of $\uparrow \overline{a}_{n-1}$ specific to scale n. For the conditional generation setting, G_B^n learns to add the minimal amount of detail required for each patch of a_n, so as to generate an image belonging to P_B^n and to satisfy the cycle loss. We find that using this type of residual training is essential for generating aligned solutions. In particular, we observe that conditioning G_B^n on both $\overline{a}_n$ and $\uparrow \overline{ab}_{n-1}$ (the mapping of $\overline{a}_{n-1}$ at the previous scale), the algorithm learns to 'cheat' by embedding details in $\overline{a}_n$ in $\overline{ab}_n$, which still allow it to satisfy the cycle loss, while not producing aligned solutions

(see ablation Sec. 4.3). On the other hand, G_A^n already learns to add details specific to scale n through the unconditional pipeline.

From scale $n \geq K$ onward, the generator of the target domain does not need to maintain the exact details of the input image, which allows it the freedom to be faithful to the details of the target domain, thus improving the generation quality. This allows for a level of control in the alignment produced. For $n \geq K$, we, therefore, opt for a non-residual architecture, where G_A^n simply generates an image $\overline{ab}_n$, conditioned on $\overline{a}_n$. Additional architectural and training details are provided in Sec. B of the appendix.

3.5 Inference and Refinement

At inference time, we use $A\ (= a_N)$ itself or, in some experiments, begin by generating a sample $\overline{a}_N \in P_A^N$ as in Eq. 3. We then map a_N or $\overline{a}_N$ to $\overline{ab}_N$ using Eq. 5. No mapping is performed for $n < N$. For simplicity of notation, we write P_A^N as P_A and the mapping from $A\ (= a_N)$ to P_B^N as ab. We write $\overline{a}_N$ as $\overline{a}$ and $\overline{ab}_N$ as $\overline{ab}$. Similar annotation is used for B.

To improve the quality of $\overline{ab}_N$, we found it useful to use an additional refinement procedure in some cases. We train a separate 'refinement' network, which contains only the unconditional generation modules (Sec. 3.1) and losses (Eq. 6 and Eq. 7) for image b. We then insert $\overline{ab}_N$ to this network at a fine scale. This adds fine details to the image, while not affecting the overall alignment.

4 Results

We evaluate our method qualitatively and quantitatively on a variety of image pairs collected from the Berkeley Segmentation Database [25], Places [39] and the web. We consider pairs of images from different natural scenes, such as a mountain with snow and that of a pyramid on sand. We also considered objects with different local shape and structure, such as an image of hot air balloons and that of birds, to evaluate the ability of our method to adapt to the local structure of objects. While the main application of our method is that of structural alignment, we also evaluate our method on the conditional generation tasks of guided image synthesis, style and texture transfer, text translation and video translation. Lastly, an ablation analysis is performed to illustrate the effectiveness of the different components of our method.

4.1 Structural Alignment

Qualitative evaluation. In Fig. 3 and Fig. 4, we illustrate the structural alignment produced by our method for a variety of image pairs. Additional results are provided in Sec. A of the appendix. We evaluate our method against four baselines. First, we consider Deep Image Analogy (DIA) [21], since it is the closest work to ours. A second baseline is that of SinGAN [27]. Since SinGAN is aimed at unconditional image generation from a single image, we first train a SinGAN model for image B. We then insert a downsampled version of image A at a coarse scale of B's pretrained model (chosen to be 2, see Sec. B of the appendix) and use the image output from the last scale. Thirdly,

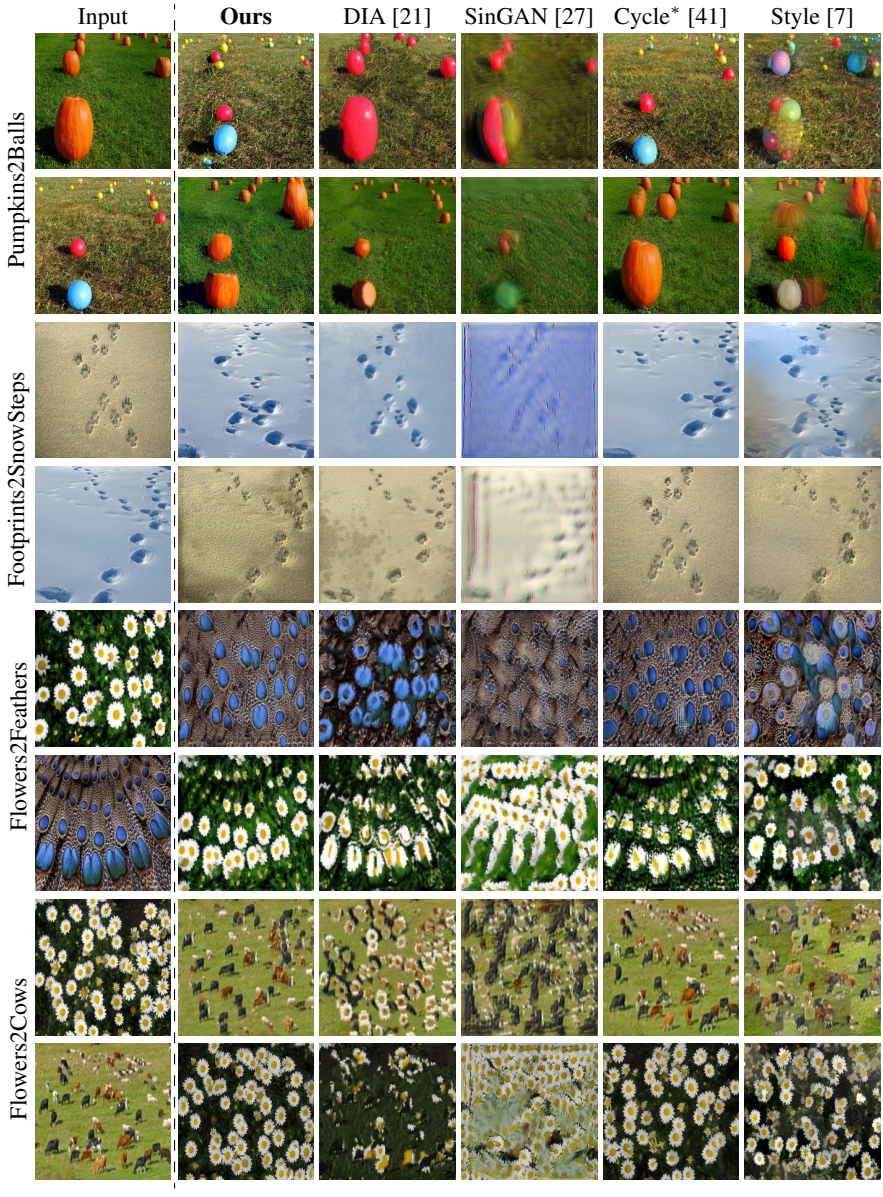


Fig. 3. Structural alignment for a collection of images containing objects of different local structure. We consider the mapping of our method and baseline ones. For every two rows, the first row is the translation from A to ab and the second is from B to ba. Cycle* stands for CycleGAN applied on randomly generated images using pretrained SinGAN models for A and B.

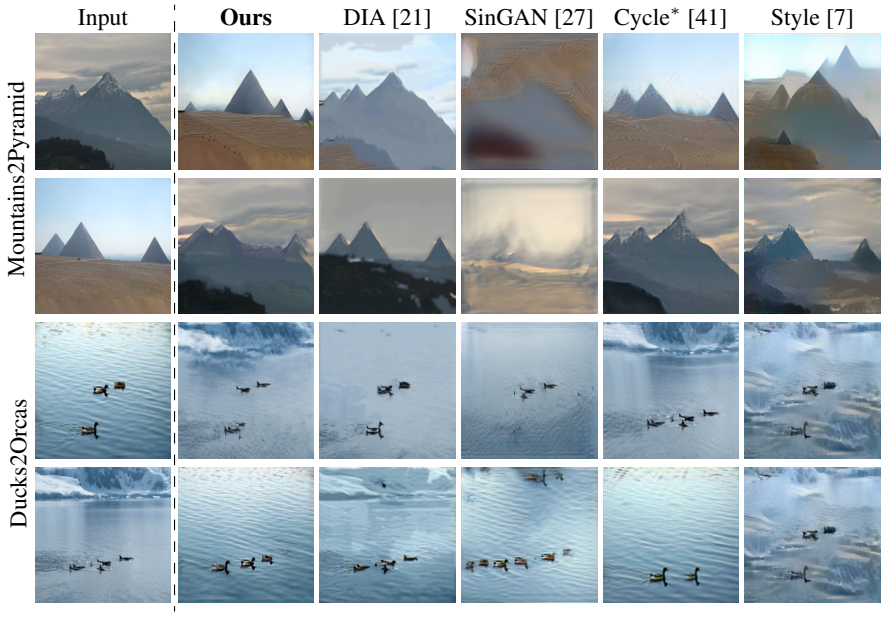


Fig. 4. Structural alignment for a collection of images depicting natural scenes. We consider the same setting as in Fig .3.

we train two SinGAN models for A and B and randomly generate a 1000 images from the internal distribution of each image. We then train a CycleGAN model [41] on these images. Lastly, we consider style transfer from A to B using the method of Gatys et al. [7]. Note that unlike our method, DIA [21] and Gatys et al. [7] use an additional level of supervision in the form of deep features of a pretrained VGG [30] network trained on Imagenet. Additional training details are provided in Sec. B of the appendix

In Fig. 3, we consider a variety of image pairs involving local objects of different shapes and structure. Unlike the baseline methods, our method is able to adapt to the structure and shape of objects in the target image. For example, when translating from an image of pumpkins to an image of balls of different sizes, our method is able to replace the large pumpkin (bottom left) with two balls in the correct position and shape, since there is no ball as large as the pumpkin. DIA, on the other hand, is unable to replace the shape of the pumpkin, producing an unrealistic image. SinGAN is unable to create an image of good quality, since it was trained on an image with a different patch distribution. Our third baseline (CycleGAN trained on images generated using two SinGANs) produces a realistic looking image of balls, but is unable to produce aligned solutions and seems to suffer from mode collapse, returning an image that is rather similar to the original one.

When translating from footprints of a dog in the sand to those of a human in the snow, our method is able to create realistically looking steps in the correct position, while preserving the perspective warp the target snow image displays. DIA and Gatys et

(a) (b)

Fig. 5. (a) Left: Input image A (hot air balloons). Right: Randomly generated samples $\overline{a}$ (top) and their translation $\overline{ab}$ (bottom). (b) As in (a) but for image B (birds).

al. [7] are unable to change the shape of the footprints, while CycleGAN produces an unaligned solution.

Similarly, our method is also able to align many small objects of different geometries, as depicted in the last two row pairs of the figure.

Fig. 4 illustrates the translation between different natural scenes. Here, again, we see our method succeed in finding plausible analogies (e.g. mountains to mountains, or animals to animals) and to place them correctly, while the baselines exhibit failure modes similar to those discussed beforehand.

In Fig. 5, we consider images of hot air balloons (A) and of birds (B). We generated random samples $\overline{a} \in P_A$ and their analogous solutions $\overline{ab}$, as described in Sec. 3.5 (and similarly for B). While small balloons are mapped to small birds, balloons of large size are mapped to a collection of birds, since no bird of matching size is present in P_B. Similarly, a flock of birds is mapped to a large balloon in the reverse direction.

Quantitative evaluation. To evaluate the structural alignment produced by our method, we measure the following: (C1) The realism of generated samples ab under the distribution of P_B. (C2) Structural alignment of image ab to A.

For (C1) we use the SIFID measure introduced by SinGAN [27]. SIFID is an extension of the popular FID metric [11] for a single image, used to evaluate how well the generated samples capture the internal statistics of the single image.

To further evaluate (C1), we have also conducted a user study. Our study is comprised of 50 users and 20 image pairs. For each image pair A and B, the real image A is shown first, followed by our and the baseline methods' translations ab at random. The user is asked to rank how real each generated image looks from a scale of 1 to 5. In Tab. 1, we report both the average SIFID and a mean opinion score for 50 samples ab (mapped from A), as described in Sec. 3.5.

To evaluate (C2), we have also conducted a user study. For each each image pair, we train a SinGAN model on A and randomly generate three other samples from P_A: $\overline{a}_1$,$\overline{a}_2$ and $\overline{a}_3$. For our method and each of the baseline methods at random, the user is shown the image ab and is asked to select which of $\overline{a}_1$, $\overline{a}_2$ and $\overline{a}_3$ and A (shown at random), is the correct source image. The better the alignment, the easier this task is for the user. The percentage of correct answers is reported in Tab. 1.

For the task of structural alignment, our method is superior to all baselines in realism, and fall slight short compared to DIA in alignment. As shown visually in Fig. 3 and

Table 1. Measuring the realism and structural alignment of generated samples ab for our method and for baseline methods. Row (1): average SIFID values for 50 generated images ab (lower value is better). Row (2): User study for (C1: realism), the realism of generated samples ab, measured as a mean opinion score on a scale of 1 to 5 (higher is better). Row (3): User study for (C2: alignment), structural alignment of ab to A, measured as the percentage of correct answers (higher is better). Rows (4-6): As for rows (1-3), but for the guided image synthesis task.

Measure	Task	**Ours**	DIA [21]	SinGAN [27]	Cycle* [41]	Style [7]
SIFID ↓	Structural alignment	0.097	0.723	1.455	0.099	0.103
Realism ↑	Structural alignment	3.72	2.00	1.57	3.52	1.62
Alignment ↑	Structural alignment	83.4%	85.0%	55.2%	43.3%	78.3%
SIFID ↓	Guided synthesis	0.191	0.716	1.573	0.235	0.208
Realism ↑	Guided synthesis	4.09	1.83	1.47	4.16	1.83
Alignment ↑	Guided synthesis	96.3%	100.0%	77.8%	37.0%	74.1%

Fig. 4, DIA is unable to change the shape of translated images and so, while the user can easily identify the correct mapping, its realism scores are significantly worse then ours.

4.2 Additional applications

Guided image synthesis. We consider the mapping of a sketch drawn in black and white, A, to a natural image, B, in Fig. 6. SinGAN [27] can perform a similar application, but requires A to be similar to the distribution of patches at a low scale of B. We do not make such assumption. For evaluation, we conduct a user study for realism (C1) and alignment (C2) as described in Sec. 4.1. As can be seen qualitatively in Fig. 6, DIA [21] does not change the image structure, hence it gets a perfect alignment score, but its images are not realistic. CycleGAN has similar realism score to ours, but exhibits mode collapse and so its alignment score is significantly lower.

Style and texture Transfer. Our method can also adapt the style or texture of a source image to a target one. This is illustrated in Fig. 7 and compared against baseline methods in Sec. A.1 of the appendix.

Text translation. In the task of text translation, we are given two images of text, and are asked to transfer the style characteristics of the first image to the second one. This is not a traditional style transfer task, since the style is embedded in the geometry of the image, rather than its appearance. Fig. 8 illustrates these mappings, and indeed our method correctly transfers the style characteristics of the input text including fine details, while DIA and CycleGAN fail in all cases. Style transfer of Gatys et al. [7] and SM-GAN [35] only partially transfer the style characteristics. SinGAN [27] performs most closely to us, but the generated texture does not resemble the input texture as well.

Video translation. In the task of video translation, we are given an input source video and a target image. We are asked to generate a video, in which every frame is structurally aligned to the corresponding frame in the source video, but belongs to the patch distribution of the target image. Consider a source video consisting of frames $f_1, \ldots, f_k$. We would like to incorporate these frames in training, to further improve the

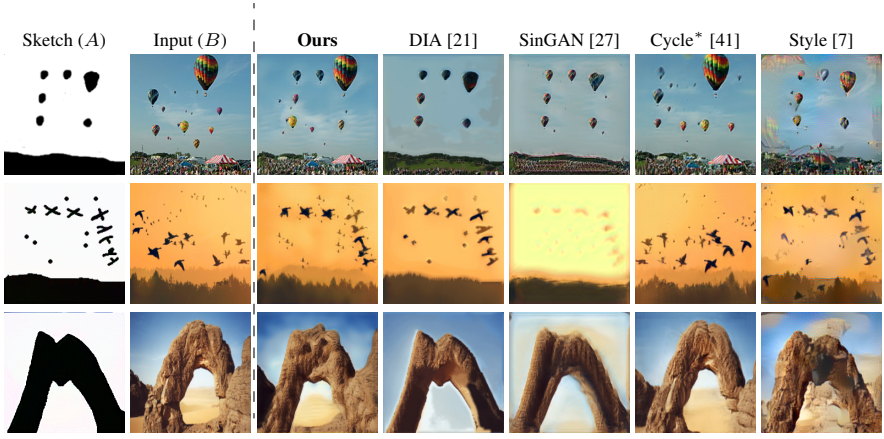


Fig. 6. From sketch A, and image B, we generate a structurally aligned image.

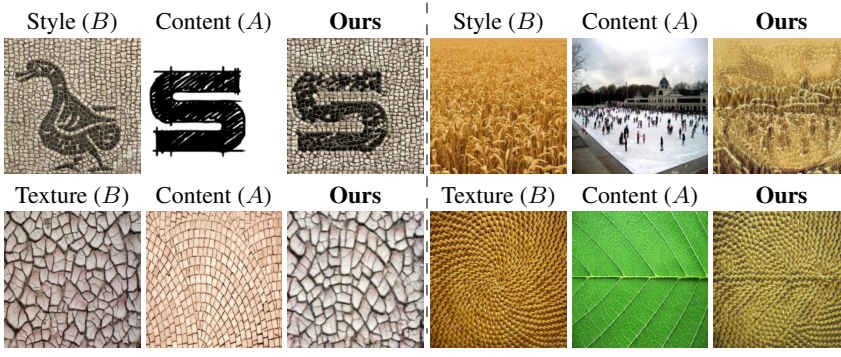


Fig. 7. An illustration of our method for the task of style and texture transfer.

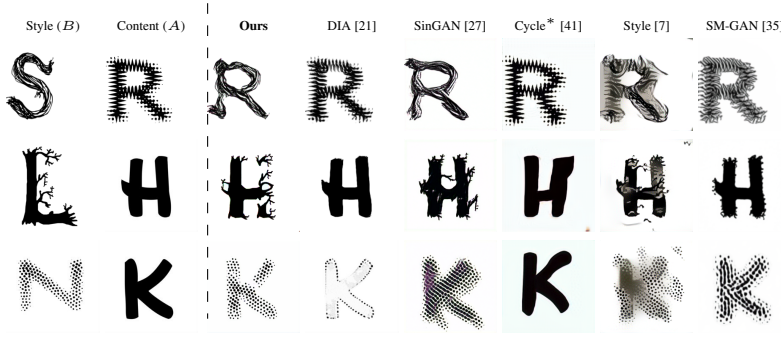


Fig. 8. Text translation. We also compare to the text transfer method of SM-GAN [35].

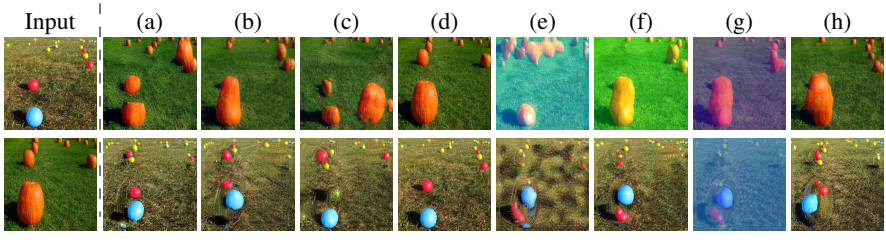
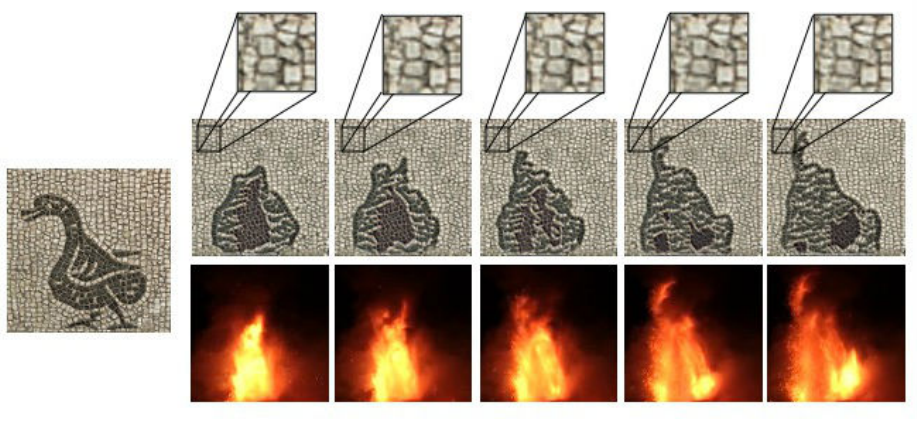


Fig. 9. An ablation analysis for the translation of balls to pumpkins.



(a) (b)

Fig. 10. Video translation. We translate between a source video of a volcano and mosaic target image. (a) Target image (b) Bottom row shows the input frames f_i. Middle row shows their translation v_i. Upper row shows a zoomed in part of the frame, where identical for all translated frames since there is no motion in this part. Frames $65, 85, 105, 135, 155$ are chosen for this illustration.

translation quality. Therefore, in each iteration, A is randomly chosen to be one of the frames, f_i. B is chosen to be the target image. At inference time, we translate one frame at a time to construct an analogous video. For frame f_i we first generate a refined frame f_i^r as follows:

$$f_i^r = G_A^N(z_N + \uparrow G_A^{N-1}(z_{N-1} + f_{i,N-1})) \tag{10}$$

Where $f_{i,N-1}$ is frame f_i resized to scale $N-1$. The translated frame v_i is given by:

$$v_i = G_B^N(f_i^r) \tag{11}$$

We find that using f_i^r results in a superior quality then using f_i in Eq. 11. In addition, z_n and z_{n-1} are fixed Gaussian noise which is used for all frames f_i. We find that using a fixed noise helps generate temporally consistent frames. For example, as shown in Fig. 10, some part of the frame remains fixed, while the rest changes in a manner which is consistent with the movement of the source volcano video. Furthermore, in some cases, we found that quantization of the colors of the frames prior to the translation improves the quality of the result. Full videos are provided in the project webpage: `https://sagiebenaim.github.io/structural-analogy/`.

4.3 Ablation Analysis

An ablation analysis is presented in Fig. 9 for the case of translating from pumpkins (A) to balls (B): (a) is the original translation as in Fig 3. In (b), the cycle loss (Eq. 8) is omitted for all but scale $K-1$ and in (c), no cycle loss is employed at all, both resulting in less alignment. In (b), for example, the two balls are mapped to a single large

pumpkin, while in (c) the generated pumpkin on the bottom right is unaligned. In (d), G_B^n in Eq.4 is changed to accept $\overline{a}_n$ as well as $\uparrow \overline{ab}_{n-1}$ — the translation from A to B at scale $n-1$. This results in mode collapse, in which the source image A is generated. In (e), no reconstruction loss (Eq. 7) is used, which results in no alignment and worse generation quality. Generated images also look tinted. In (f) a residual generator (see Sec. 3.4) is used for all scales while in (g) no residual training is used at all. In both cases alignment is reduced and images look tinted. In (h), we do not initialize the networks of scale n with those of scale $n-1$ (see Sec. 3.4), which results in a worse alignment and unrealistic images.

5 Conclusions

Up until recently, the problem of unsupervised image-to-image translation, without additional images from the same domains, was not considered possible. Our method is the first to consider only a single pair of images and to successfully generate structural analogies, even if the structure, or semantics, of the objects in the images differ significantly (e.g., bird and balloons). By considering structural analogies, our method goes beyond style transfer, and can change the structure of local objects or parts by matching the internal patch statistics of the source and target images. Our method can also be applied in other conditional generation tasks such as guided image synthesis, style and texture transfer and even text and video translation. Going forward, we hope to use our method in other conditional generation tasks, which traditionally require costly supervision, for example, for one-shot semantic segmentation from a single image.

References

1. Benaim, S., Khaitov, M., Galanti, T., Wolf, L.: Domain intersection and domain difference. In: ICCV (2019)
2. Brock, A., Donahue, J., Simonyan, K.: Large scale GAN training for high fidelity natural image synthesis. In: International Conference on Learning Representations (2019), https://openreview.net/forum?id=B1xsqj09Fm
3. Choi, Y., Choi, M., Kim, M., Ha, J.W., Kim, S., Choo, J.: Stargan: Unified generative adversarial networks for multi-domain image-to-image translation. arXiv preprint arXiv:1711.09020 (2017)
4. Conneau, A., Lample, G., Ranzato, M., Denoyer, L., Jégou, H.: Word translation without parallel data. arXiv preprint arXiv:1710.04087 (2017)
5. Efros, A., Leung, T.: Texture synthesis by non-parametric sampling. In: In International Conference on Computer Vision. pp. 1033–1038 (1999)
6. Gandelsman, Y., Shocher, A., Irani, M.: Double-dip": Unsupervised image decomposition via coupled deep-image-priors. In: The IEEE Conference on Computer Vision and Pattern Recognition (CVPR). vol. 6, p. 2 (2018)
7. Gatys, L.A., Ecker, A.S., Bethge, M.: Image style transfer using convolutional neural networks. In: The IEEE Conference on Computer Vision and Pattern Recognition (CVPR) (June 2016)
8. Goodfellow, I., Pouget-Abadie, J., Mirza, M., Xu, B., Warde-Farley, D., Ozair, S., Courville, A., Bengio, Y.: Generative adversarial nets. In: NIPS (2014)
9. Gulrajani, I., Ahmed, F., Arjovsky, M., Dumoulin, V., Courville, A.: Improved training of wasserstein gans (2017)
10. Hertzmann, A., Jacobs, C.E., Oliver, N., Curless, B., Salesin, D.H.: Image analogies. In: Proceedings of the 28th annual conference on Computer graphics and interactive techniques. pp. 327–340 (2001)
11. Heusel, M., Ramsauer, H., Unterthiner, T., Nessler, B., Hochreiter, S.: Gans trained by a two time-scale update rule converge to a local nash equilibrium. In: Advances in Neural Information Processing Systems. pp. 6626–6637 (2017)
12. Huang, X., Belongie, S.: Arbitrary style transfer in real-time with adaptive instance normalization. 2017 IEEE International Conference on Computer Vision (ICCV) (Oct 2017). https://doi.org/10.1109/iccv.2017.167, http://dx.doi.org/10.1109/ICCV.2017.167
13. Huang, X., Liu, M.Y., Belongie, S., Kautz, J.: Multimodal unsupervised image-to-image translation. In: ECCV (2018)
14. Isola, P., Zhu, J.Y., Zhou, T., Efros, A.A.: Image-to-image translation with conditional adversarial networks. In: CVPR (2017)
15. Katzir, O., Lischinski, D., Cohen-Or, D.: Cross-domain cascaded deep feature translation (2019)
16. Kim, T., Cha, M., Kim, H., Lee, J., Kim, J.: Learning to discover cross-domain relations with generative adversarial networks. arXiv preprint arXiv:1703.05192 (2017)
17. Lample, G., Conneau, A., Denoyer, L., Ranzato, M.: Unsupervised machine translation using monolingual corpora only. In: ICLR (2018)
18. Lee, H.Y., Tseng, H.Y., Mao, Q., Huang, J.B., Lu, Y.D., Singh, M.K., Yang, M.H.: Drit++: Diverse image-to-image translation viadisentangled representations. arXiv preprint arXiv:1905.01270 (2019)
19. Li, C., Wand, M.: Precomputed real-time texture synthesis with markovian generative adversarial networks. In: European conference on computer vision. pp. 702–716. Springer (2016)

20. Li, S., Günel, S., Ostrek, M., Ramdya, P., Fua, P., Rhodin, H.: Deformation-aware unpaired image translation for pose estimation on laboratory animals. ArXiv **abs/2001.08601** (2020)
21. Liao, J., Yao, Y., Yuan, L., Hua, G., Kang, S.B.: Visual attribute transfer through deep image analogy. ACM Transactions on Graphics **36**(4), 1–15 (Jul 2017). https://doi.org/10.1145/3072959.3073683, http://dx.doi.org/10.1145/3072959.3073683
22. Liu, M., Ding, Y., Xia, M., Liu, X., Ding, E., Zuo, W., Wen, S.: Stgan: A unified selective transfer network for arbitrary image attribute editing. In: IEEE Conference on Computer Vision and Pattern Recognition (CVPR) (2019)
23. Liu, M.Y., Breuel, T., Kautz, J.: Unsupervised image-to-image translation networks. In: NIPS (2017)
24. Liu, M.Y., Tuzel, O.: Coupled generative adversarial networks. In: NIPS. pp. 469–477 (2016)
25. Martin, D., Fowlkes, C., Tal, D., Malik, J.: 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, pp. 416–423 (July 2001)
26. Park, T., Liu, M.Y., Wang, T.C., Zhu, J.Y.: Semantic image synthesis with spatially-adaptive normalization. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (2019)
27. Rott Shaham, T., Dekel, T., Michaeli, T.: Singan: Learning a generative model from a single natural image. In: Computer Vision (ICCV), IEEE International Conference on (2019)
28. Shocher, A., Bagon, S., Isola, P., Irani, M.: Ingan: Capturing and retargeting the "dna" of a natural image. In: The IEEE International Conference on Computer Vision (ICCV) (2019)
29. Simakov, D., Caspi, Y., Shechtman, E., Irani, M.: Summarizing visual data using bidirectional similarity. In: Proceedings of IEEE Conference on Computer Vision and Pattern Recognition. pp. 1 – 8 (07 2008). https://doi.org/10.1109/CVPR.2008.4587842
30. Simonyan, K., Zisserman, A.: Very deep convolutional networks for large-scale image recognition. CoRR **abs/1409.1556** (2014), http://arxiv.org/abs/1409.1556
31. Taigman, Y., Polyak, A., Wolf, L.: Unsupervised cross-domain image generation. In: International Conference on Learning Representations (ICLR) (2017)
32. Ulyanov, D., Vedaldi, A., Lempitsky, V.: Deep image prior. arXiv:1711.10925 (2017)
33. Wang, T.C., Liu, M.Y., Zhu, J.Y., Tao, A., Kautz, J., Catanzaro, B.: High-resolution image synthesis and semantic manipulation with conditional gans. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (2018)
34. Wu, W., Cao, K., Li, C., Qian, C., Loy, C.C.: Transgaga: Geometry-aware unsupervised image-to-image translation. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. pp. 8012–8021 (2019)
35. Yang, S., Wang, Z., Wang, Z., Xu, N., Liu, J., Guo, Z.: Controllable artistic text style transfer via shape-matching gan. 2019 IEEE/CVF International Conference on Computer Vision (ICCV) (Oct 2019). https://doi.org/10.1109/iccv.2019.00454, http://dx.doi.org/10.1109/ICCV.2019.00454
36. Yi, Z., Zhang, H., Tan, P., Gong, M.: DualGAN: Unsupervised dual learning for image-to-image translation. arXiv preprint arXiv:1704.02510 (2017)
37. Zhang, M., Liu, Y., Luan, H., Sun, M.: Adversarial training for unsupervised bilingual lexicon induction. In: Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers). vol. 1, pp. 1959–1970 (2017)
38. Zhang, M., Liu, Y., Luan, H., Sun, M.: Earth mover's distance minimization for unsupervised bilingual lexicon induction. In: Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing. pp. 1934–1945 (2017)
39. Zhou, B., Lapedriza, A., Xiao, J., Torralba, A., Oliva, A.: Learning deep features for scene recognition using places database. In: Ghahramani, Z., Welling, M., Cortes, C., Lawrence,

N.D., Weinberger, K.Q. (eds.) Advances in Neural Information Processing Systems 27, pp. 487–495. Curran Associates, Inc. (2014), http://papers.nips.cc/paper/5349-learning-deep-features-for-scene-recognition-using-places-data pdf
40. Zhou, Y., Zhu, Z., Bai, X., Lischinski, D., Cohen-Or, D., Huang, H.: Non-stationary texture synthesis by adversarial expansion. ACM Trans. Graph. **37**(4), 49:1–49:13 (Jul 2018). https://doi.org/10.1145/3197517.3201285, http://doi.acm.org/10.1145/3197517.3201285
41. Zhu, J.Y., Park, T., Isola, P., Efros, A.A.: Unpaired image-to-image translation using cycle-consistent adversarial networks. In: Proceedings of the IEEE international conference on computer vision. pp. 2223–2232 (2017)
42. Zhu, J.Y., Zhang, R., Pathak, D., Darrell, T., Efros, A.A., Wang, O., Shechtman, E.: Toward multimodal image-to-image translation. In: Advances in Neural Information Processing Systems (2017)

A Additional Qualitative Results

Fig. 12 and Fig. 11 present additional qualitative results for structural alignment as presented in Fig. 3 of the main text. Fig. 13 presents additional qualitative results for the task of guided image synthesis as shown in Fig. 6 of the main text.

A.1 Comparison to Style transfer methods

Fig. 7 of the main text presents style and texture transfer results of our model. In Fig. 14 we show a comparison to other style transfer methods of Gatys et al. [7] (Style) and Huang et al. [12] (AdaIn). As can be seen, our method is competitive with these methods.

B Additional Architecture and Training Details

B.1 Architecture and Training

The generators G_A^n and G_B^n, as well as the discriminators D_n^A and D_n^B, each consist of five convolutional blocks. Each block consists of a 3×3 convolutional layer and maintains the spatial resolution of the input using padding. After the convolutional layer, batch norm and a LeakyReLU (with slope of 0.2) activation are used. This results in a fixed effective receptive field of 11×11 for all generators and discriminators at each scale. As the input a_n increases in size for each scale, our algorithm starts by matching large patches between P_A^n and P_B^n (global structure and alignment of objects) and finishes at matching patches of smaller size. For the last convolutional block no batch norm is used and LeakyReLU is replaced with $tanh$. An adam optimizer is used with learning rate of 0.0005 and parameters $\beta_1 = 0.5$ and $\beta_2 = 0.999$. In Eq. 9, $\lambda_{recon} = 1.0$ and $\lambda_{cycle} = 10.0$ for all experiments.

We note that K, r and N (see Sec. 3) are chosen as hyperparameters. r is chosen to be 0.75 in all experiments. N is chosen such that the maximal image size at the finest scale is $220px$ and the minimal image size at the coarsest scale is $18px$. We used $K = N - 1$ for style and texture transfer experiments (see Sec. 4.1) and $K = N - 2$ for the other experiments.

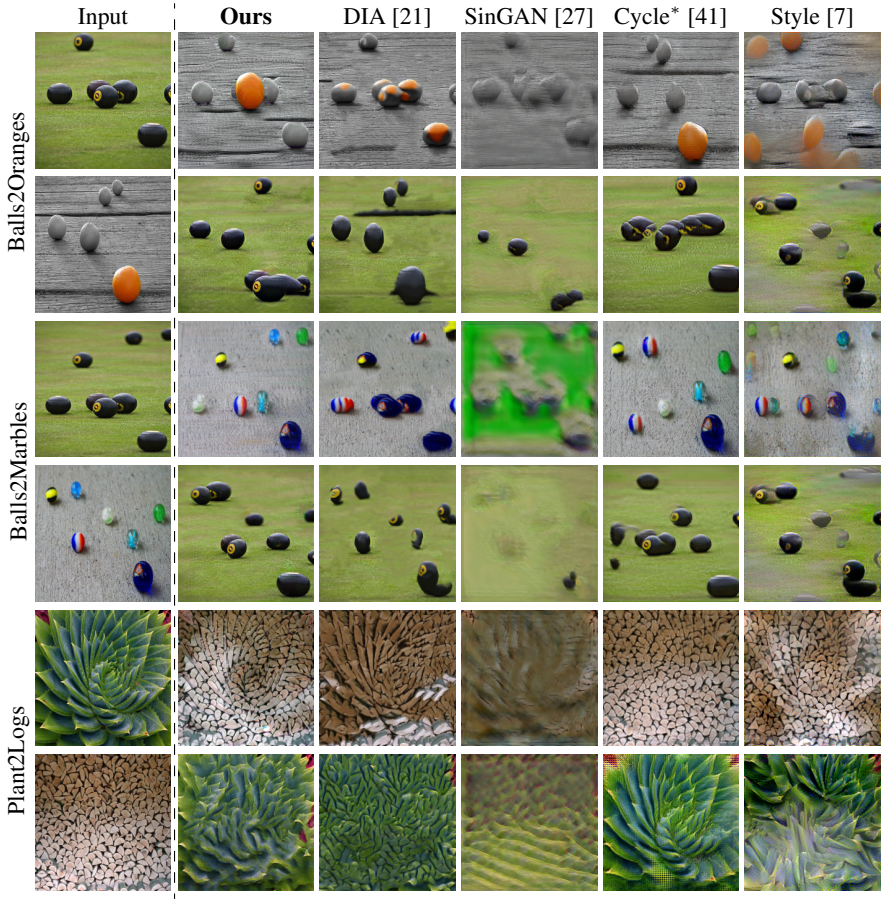


Fig. 11. Additional structural alignment results.

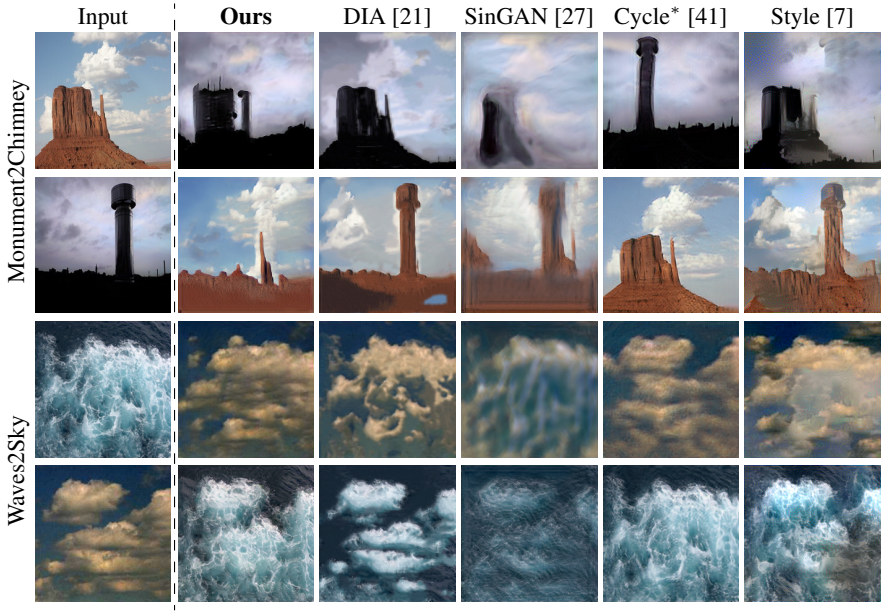


Fig. 12. Additional structural alignment results.

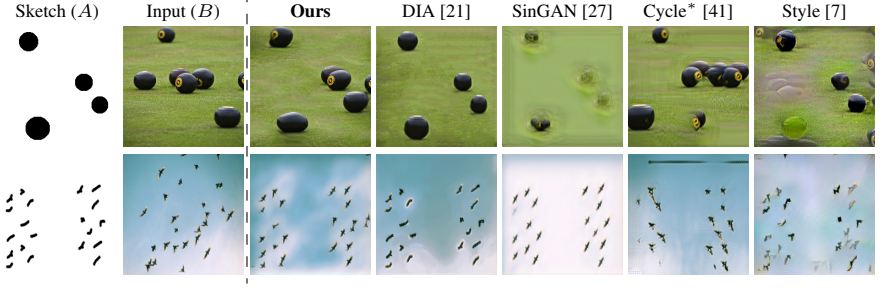


Fig. 13. Additional guided image synthesis results.

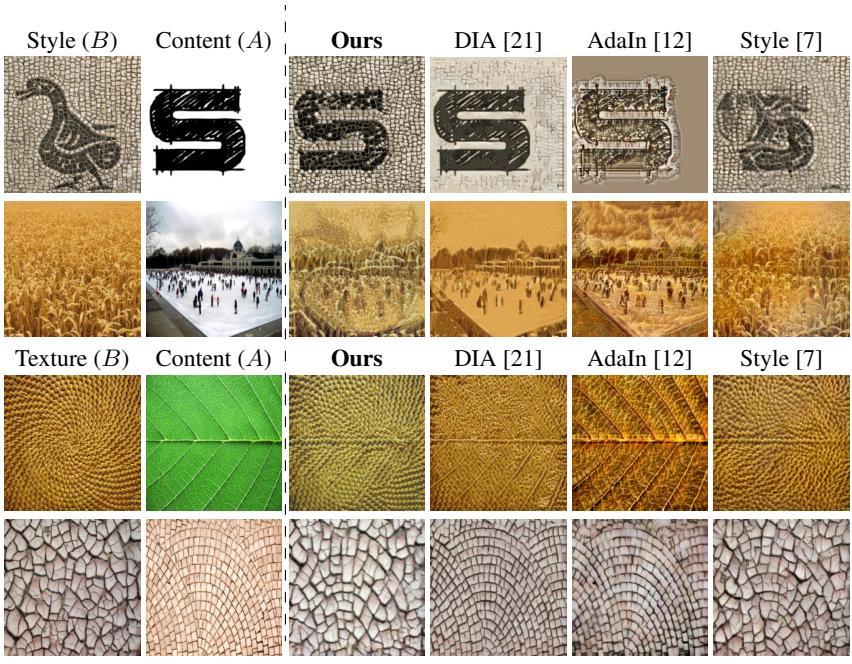


Fig. 14. Style and Texture transfer qualitative comparison.

B.2 Baselines

Baseline methods were trained in their original configuration using official public code. For SinGAN baseline used in Sec. 4.1, we insert a downsampled version of image A at a coarse scale of B's pretrained model. This coarse scale is chosen to be 2. We found that using a value of 1 resulted in no local structure being preserved, while for a scale larger then 2, generation quality was worse. For the CycleGAN model (CycleGAN trained on images generated using two SinGANs), we use the original configuration of SinGAN for both A and B (such as minimum image size of $25px$ and maximum image size of $250px$) and generated random images by applying random noise at all scales, to generated maximum variability. In AdaIn [12], a model is trained with many style and content images. As the original implementation uses only art images for style, we added images which are not artistic, as used in our method.