COCO-GAN: Generation by Parts via Conditional Coordinating

Chieh Hubert Lin[1] Chia-Che Chang[1] Yu-Sheng Chen[2]
Da-Cheng Juan[3] Wei Wei[3] Hwann-Tzong Chen[1]

[1]National Tsing Hua University [2]National Taiwan University [3]Google AI

Abstract

Humans can only interact with part of the surrounding environment due to biological restrictions. Therefore, we learn to reason the spatial relationships across a series of observations to piece together the surrounding environment. Inspired by such behavior and the fact that machines also have computational constraints, we propose COnditional COordinate GAN (COCO-GAN) of which the generator generates images by parts based on their spatial coordinates as the condition. On the other hand, the discriminator learns to justify realism across multiple assembled patches by global coherence, local appearance, and edge-crossing continuity. Despite the full images are never generated during training, we show that COCO-GAN can produce ***state-of-the-art-quality*** *full images during inference. We further demonstrate a variety of novel applications enabled by teaching the network to be aware of coordinates. First, we perform extrapolation to the learned coordinate manifold and generate off-the-boundary patches. Combining with the originally generated full image, COCO-GAN can produce images that are larger than training samples, which we called "beyond-boundary generation". We then showcase panorama generation within a cylindrical coordinate system that inherently preserves horizontally cyclic topology. On the computation side, COCO-GAN has a built-in divide-and-conquer paradigm that reduces memory requisition during training and inference, provides high-parallelism, and can generate parts of images on-demand.*

1. Introduction

The human perception has only partial access to the surrounding environment due to biological restrictions (such as the limited acuity area of the fovea), and therefore humans infer the whole environment by "assembling" few local views obtained from their eyesight. This recognition can be done partially because humans are able to associate the spatial coordination of these local views with the environment (where they are situated in), then correctly assemble these local views, and recognize the whole environment. Currently, most of the computational vision models assume to have access to full images as inputs for down-streaming tasks, which sometimes may become a computational bottleneck of modern vision models when dealing with large field-of-view images. This limitation piques our interest and raises an intriguing question: "*is it possible to train generative models to be aware of coordinate system for generating local views* (i.e. *parts of the image) that can be assembled into a globally coherent image?*"

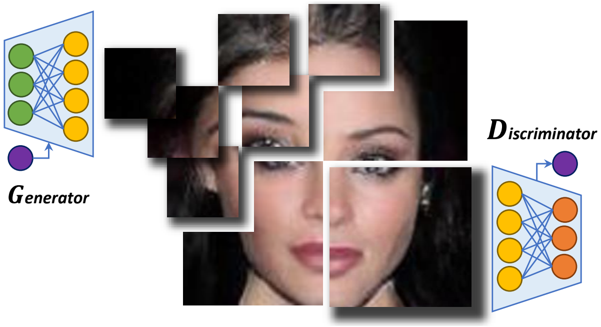


Figure 1: COCO-GAN generates and discriminates only parts of the full image via conditional coordinating. Despite the full images are never generated during training, the generator can still produce full images that are visually indistinguishable to standard GAN samples during inference.

Conventional GANs [9] target at learning a generator that models a mapping from a prior latent distribution (normally a unit Gaussian) to the real data distribution. To achieve generating high-quality images by parts, we introduce coordinate systems within an image and divide image generation into separated parallel sub-procedures. Our framework, named COnditional COordinate GAN (COCO-GAN), aims at learning a coordinate manifold that is orthogonal to the latent distribution manifold. After a latent vector is sampled, the generator conditions on each spatial coordinate and generate patches at each corresponding spatial position. On the other hand, the discriminator learns to judge whether adjacent patches are structurally sound, visually homogeneous, and continuous across the edges be-

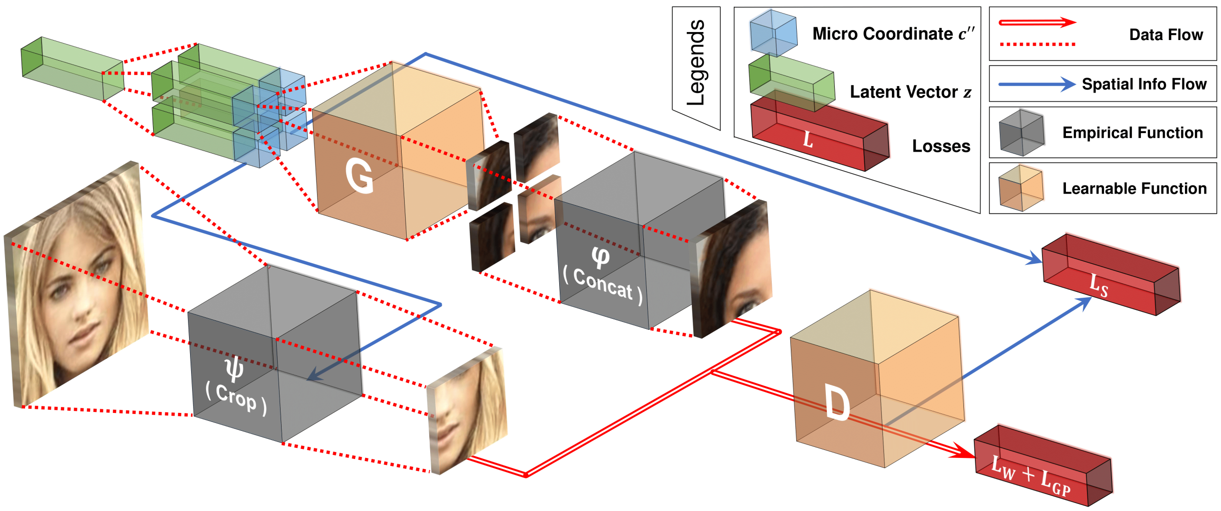


Figure 2: An overview of COCO-GAN training. The latent vectors are duplicated multiple times, concatenated with micro coordinates, and feed to the generator to generate micro patches. Then we concatenate multiple micro patches to form a larger macro patch. The discriminator learns to discriminate between real and fake macro patches and an auxiliary task predicting the coordinate of the macro patch. Note that the full images are only generated in the testing phase (Appendix A).

tween multiple patches. Figure 1 depicts the high-level idea.

We perform a series of experiments that set the generator to generate patches under different configurations. The results show that COCO-GAN can achieve *state-of-the-art* generation quality in multiple setups with "Frchet Inception Distance" (FID) [11] score measurement. Furthermore, to our surprise, even if the generated patch sizes are set to as small as 4×4 pixels, the full images that are composed by ***1024*** separately generated patches can still consistently form complete and plausible human faces. To further demonstrate the generator indeed learns the coordinate manifold, we perform an extrapolation experiment on the coordinate condition. Interestingly, the generator is able to generate novel contents that are never explicitly presented in the real data. We show that COCO-GAN can produce 384×384 images that are larger than the 256×256 real training samples. We call such a procedure "beyond-boundary generation"; all the samples created through this procedure are ***guaranteed*** to be novel samples, which is a powerful example of artificial creativity.

We then investigate another series of novel applications and merits brought about by teaching the network to be aware of the coordinates. The first is panorama generation. To preserve the native horizontally-cyclic topology of panoramic images, we apply cylindrical coordinate to COCO-GAN training process and show that the generated samples are indeed horizontally cyclic. Next, we demonstrate that the "image generation by parts" schema is highly parallelable and saves a significant amount of memory for both training and inference. Furthermore, as the generation procedures of patches are disjoint, COCO-GAN inherently supports generation on-demand, which particularly fits applications for computation-restricted environments, such as mobile and virtual reality. Last but not the least, we show that by adding an extra prediction branch that reconstructs latent vectors, COCO-GAN can generate an entire image with respect to a patch of real image as guidance, which we call "patch-guided generation".

COCO-GAN unveils the potential of generating high-quality images with conditional coordinating. This property enables a wide range of new applications, and can further be used by other tasks with encoding-decoding schema. With the "generation by parts" property, COCO-GAN is highly parallelable and intrinsically inherits the classic divide-and-conquer design paradigm, which facilitates future research toward large field-of-view data generation.

2. COCO-GAN

Overview. COCO-GAN consists of two networks (a generator G and a discriminator D), two coordinate systems (a finer-grained micro coordinate for G and a coarser-grained macro coordinate for D), and images of three sizes: full images (real: x, generated: s), macro patches (real: x', generated: s') and micro patches (generated: s'').

The generator of COCO-GAN is a conditional model that generates micro patches with $s'' = G(z, c'')$, where z is a latent vector and c'' is a micro coordinate condition designating the spatial location of s'' to be generated. The final goal of G is to generate realistic and *seamless* full images by assembling a set of s'' altogether with a merging function φ. In practice, we find that setting φ as a concatenation function without overlapping is sufficient for COCO-GAN

We list all the used symbols in Appendix B.

to synthesize high-quality images. Note that the size of the micro patches and φ also imply a cropping transformation ψ, cropping out a macro patch x' from a real image x, which is used to sample real macro patches for training D.

In the above setting, the seams between consecutive patches become the major obstacle of full image realism. To mitigate this issue, we train the discriminator with larger macro patches that are assembled with multiple micro patches. Such a design aims to introduce the continuity and coherence of multiple consecutive or nearby micro patches into the consideration of adversarial loss. In order to fool the discriminator, the generator has to close the gap at the boundaries between the generated patches.

COCO-GAN is trained with three loss terms: patch Wasserstein loss L_W, patch gradient penalty loss L_{GP}, and spatial consistency loss L_S. For L_W and L_{GP}, compared with conventional GANs that use full images x for both G and D training, COCO-GAN only cooperates with macro patches and micro patches. Meanwhile, the spatial consistency loss L_S is an ACGAN-like [20] loss function. Depending on the design of φ, we can calculate macro coordinate c' for the macro patches x'. L_S aims at minimizing the distance loss between the real macro coordinate c' and the discriminator-estimated macro coordinate $\hat{c}'$. The loss functions of COCO-GAN are

$$\begin{cases} L_W + \lambda L_{GP} + \alpha L_S, & \text{for the discriminator}\, D, \\ -L_W + \alpha L_S, & \text{for the generator}\, G. \end{cases} \tag{1}$$

Spatial coordinate system. We start with designing the two spatial coordinate systems, a *micro* coordinate system for the generator G and a *macro* coordinate system for the discriminator D. Depending on the design of the aforementioned merging function φ, each macro coordinate $c'_{(i,j)}$ is associated with a matrix of micro coordinates: $\boldsymbol{C''}_{(i,j)} = \left[c''_{(i:i+N,j:j+M)}\right]$, whose complete form is

$$\boldsymbol{C}''_{(i,j)} = \begin{bmatrix} c''_{(i,j)} & c''_{(i,j+1)} & \cdots & c''_{(i,j+M-1)} \\ c''_{(i+1,j)} & c''_{(i+1,j+1)} & \cdots & c''_{(i+1,j+M-1)} \\ \vdots & \vdots & \ddots & \vdots \\ c''_{(i+N-1,j)} & c''_{(i+N-1,j+1)} & \cdots & c''_{(i+N-1,j+M-1)} \end{bmatrix}.$$

During COCO-GAN training, we uniformly sample all combinations of $\boldsymbol{C}''_{(i,j)}$. The generator G conditions on each micro coordinate $c''_{(i,j)}$, and learns to accordingly produce micro patches $s''_{(i,j)}$ by $G(z, c''_{(i,j)})$. The matrix of generated micro patches $\boldsymbol{S}''_{(i,j)} = G(z, \boldsymbol{C}''_{(i,j)})$ are produced ***independently*** while sharing the same latent vector z across the micro coordinate matrix.

The design principle of the $\boldsymbol{C}''_{(i,j)}$ construction is that, the accordingly generated micro patches $\boldsymbol{S}''_{(i,j)}$ should be spatially close to each other. Then the micro patches are merged by the merging function φ to form a complete macro patch $s'_{(i,j)} = \varphi(\boldsymbol{S}''_{(i,j)})$ as a coarser partial-view of the imagery full-scene. Meanwhile, we assign $s'_{(i,j)}$ with a new macro coordinate $c'_{(i,j)}$ under the macro coordinate system with respect to $\boldsymbol{C}''_{(i,j)}$. On the real data side, we directly sample macro coordinates $c'_{(i,j)}$, then produce real macro patches $x'_{(i,j)} = \psi(x, c'_{(i,j)})$ with the cropping function ψ. Note that the design choice of the micro coordinates $\boldsymbol{C}''_{(i,j)}$ is also correlated with the topological characteristic of the micro/macro coordinate systems (for instance, the cylindrical coordinate system for panoramas used in Section 3.4).

In Figure 2, we illustrate one of the most straightforward designs for the above heuristic functions that we have adopted throughout our experiments. The micro patches are always a neighbor of each other and can be directly combined into a square-shaped macro patch using φ. We observe that setting φ to be a concatenation function is sufficient for G to learn smoothly, and eventually to produce seamless and high-quality images.

During the testing phase, depending on the design of the micro coordinate system, we can infer a corresponding spatial coordinate matrix $\boldsymbol{C}''_{\textit{\textbf{full}}}$. Such a matrix is used to independently produce all the micro patches required for constituting the full image.

Loss functions. The patch Wasserstein loss L_W is a macro-patch-level Wasserstein distance loss similar to Wasserstein-GAN [1] loss. It forces the discriminator to distinguish between the real macro patches x' and fake macro patches s', and on the other hand, encourages the generator to confuse the discriminator with seemingly realistic micro patches s''. Its complete form is

$$L_W = \mathop{\mathbb{E}}_{x,c'} \left[D(\psi(x, c')) \right] - \mathop{\mathbb{E}}_{z,\boldsymbol{C}''} \left[D(\varphi(G(z, \boldsymbol{C}''))) \right] . \tag{2}$$

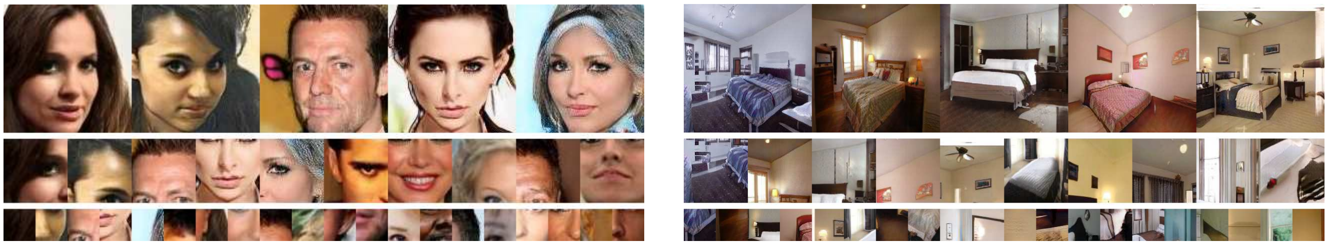
Again, note that $G(z, \boldsymbol{C}'')$ represents that the micro patches are generated through independent processes. We apply Gradient Penalty [10] to the macro patches discrimination:

$$L_{GP} = \mathop{\mathbb{E}}_{\hat{s}'} \left[(\|\nabla_{\hat{s}'} D(\hat{s}')\|_2 - 1)^2 \right] , \tag{3}$$

where $\hat{s}' = \epsilon\, s' + (1 - \epsilon)\, x'$ is calculated between randomly paired s' and x' with a random number $\epsilon \in \left[0, 1\right]$.

Finally, the spatial consistency loss L_S is similar to ACGAN loss [20]. The discriminator is equipped with an auxiliary prediction head A, which aims to estimate the macro coordinate of a given macro patch with $A(x')$. A slight difference is that both c'' and c' have relatively more continuous values than the discrete setting of ACGAN. As a result, we apply a distance measurement loss for L_S, which is an L_2-loss. It aims to train G to generate corresponding micro patches by $G(z, c'')$ with respect to the given spatial condition c''. The spatial consistency loss is

$$L_S = \mathop{\mathbb{E}}_{c'} \left[\|c' - A(x')\|_2 \right] . \tag{4}$$



(a) CelebA (N2,M2,S32) (full image: 128×128).

(b) LSUN bedroom (N2,M2,S64) (full image: 256×256).

Figure 3: COCO-GAN generates visually smooth and globally coherent full images without any post-processing. The three rows from top to bottom show: (a) the generated full images, (b) macro patches, and (c) micro patches. For the first five columns, each column uses the same latent vector, *e.g.*, the leftmost full image (first row), the leftmost micro patch (second row), and the leftmost micro patch (third row) share the same latent vector. Note that the columns are not aligned due to different sizes. More results can be found in the Appendix F.

3. Experiments

3.1. Quality of Generation by Parts

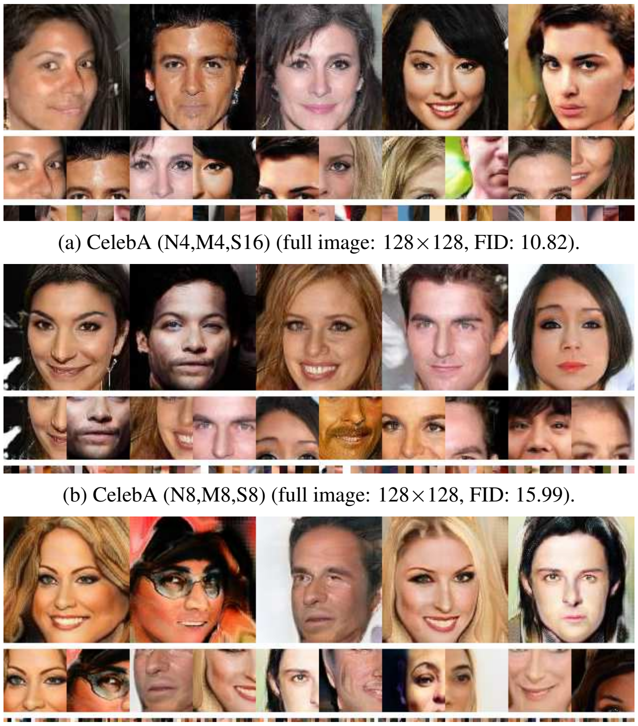
We start with validating COCO-GAN on CelebA [16] and LSUN [30] (bedroom). To verify that COCO-GAN can learn to generate the full image without the access to the full image, we first conduct a basic setting for both datasets in which the macro patch edge length (CelebA: 64 × 64, LSUN: 128 × 128) is 1/2 of the full image and the micro patch edge length (CelebA: 32 × 32, LSUN: 64 × 64) is 1/2 of the macro patch. We denote the above cases as CelebA (N2,M2,S32) and LSUN (N2,M2,S32), where N2 and M2 represent that a macro patch is composed of 2 × 2 micro patches, and S32 means each of the micro patches is 32 × 32 pixels. Our results in Figure 3 show that COCO-GAN generates high-quality images in the settings that the micro patch size is 1/16 of the full image.

To further show that COCO-GAN can learn more fine-grained and tiny micro patches under the same macro patch size setting, we sweep through the resolution of micro patch from 32×32, 16×16, 8×8, 4×4, labelled as (N2,M2,S32), (N4,M4,S16), (N8,M8,S8) and (N16,M16,S4), respectively. The results shown in Figure 4 suggest that COCO-GAN can learn coordinate information and generate images by parts even with extremely tiny 4 × 4 pixels micro patch.

We report Frchet Inception Distance (FID) [11] in Table 1 comparing with state-of-the-art GANs. Without additional hyper-parameter tuning, the quantitative results show that COCO-GAN is competitive with other state-of-the-art GANs. In Appendix L, we also provide Wasserstein distance and FID score through time as training indicators. The curves suggest that COCO-GAN is stable during training.

3.2. Latent Space Continuity

To demonstrate the space continuity more precisely, we perform the interpolation experiment in two directions: “full-images interpolation” and “coordinates interpolation”.



(a) CelebA (N4,M4,S16) (full image: 128×128, FID: 10.82).

(b) CelebA (N8,M8,S8) (full image: 128×128, FID: 15.99).

(c) CelebA (N16,M16,S4) (full image: 128×128, FID: 23.90).

Figure 4: Various sizes of micro patches (from 16 × 16 to 4×4, even smaller than any human face organs) consistently generate visually smooth and globally coherent full images. Each sub-figure consists of three rows, from top to bottom: full images, macro patches, and micro patches. For the first five columns, each column uses the same latent vector (similar with Figure 3). Better to view in high-resolution since the micro patches are very small. More generation results are available in the Appendix F.

Full-Images Interpolation. Intuitively, the inter-full-image interpolation is challenging for COCO-GAN, since all micro patches generated with different spatial coordinates must all change synchronously to make the full-image interpolation smooth. Nonetheless, as shown in Figure 5,

We describe the model details and hyper-parameters in Appendix C.

Dataset	CelebA 64×64	CelebA 128×128	LSUN Bedroom 64×64	LSUN Bedroom 256×256	CelebA-HQ 1024×1024
DCGAN [22] + TTUR [11]	12.5	-	57.5	-	
WGAN-GP [10] + TTUR [11]	-	-	9.5	-	-
IntroVAE [12]	-	-	-	8.84	-
PGGAN [13]	-	7.30	-	8.34	**7.48**
Proj. D [19] (our backbone)	-	19.55	-	-	-
Ours (N2,M2,S32)	**4.00**	**5.74**	**5.20**	**5.99***	9.49*

Table 1: The FID score suggests that COCO-GAN is competitive with other state-of-the-art generative models. FID scores are measured between 50,000 real and generated samples based on the original implementation provided at https://github.com/bioinf-jku/TTUR. Note that all the FID scores (except proj. D) are officially reported numbers. The real samples for evaluation are held-out from training.

we empirically find COCO-GAN can interpolate smoothly and synchronously without producing unnatural artifacts. We randomly sample two latent vectors z_1 and z_2. With any given interpolation point z' in the slerp-path [27] between z_1 and z_2, the generator uses the full spatial coordinate sequence $\boldsymbol{C''_{full}}$ to generate all corresponding patches. Then we assemble all the generated micro patches together and form a generated full image s.

Coordinates Interpolation. Another dimension of the interpolation experiment is inter-class (*e.g.* between spatial coordinate condition) interpolation with a fixed latent vector. We linearly-interpolate spatial coordinates between $[-1, 1]$ with a fixed latent vector z. The results in Figure 6 show that, although we only uniformly sample spatial coordinates within a discrete spatial coordinate set, the spatial coordinates interpolation is still overall continuous.

An interesting observation is about the interpolation at the position between the eyebrows. In Figure 6, COCO-GAN does not know the existence of the glabella between two eyes due to the discrete and sparse spatial coordinates sampling strategy. Instead, it learns to directly deform the shape of the eye to switch from one eye to another. This phenomenon raises an interesting discussion, even though the model learns to produce high-quality face images, it still may learn wrong relationships of objects behind the scene.

3.3. Beyond-Boundary Generation

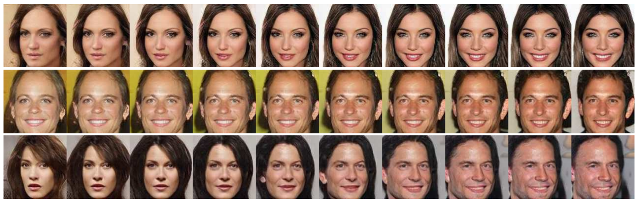


Figure 5: The results of full-images interpolation between two latent vectors show that all micro patches are changed synchronously in response to the change of the latent vector. More interpolation results are available in Appendix G.

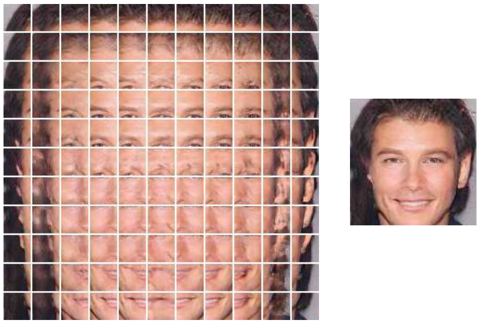


Figure 6: An example of spatial coordinates interpolation showing the spatial continuity of the micro patches. The spatial coordinates are interpolated between range $[-1, 1]$ of the micro coordinate with a fixed latent vector. More examples are shown in Appendix I.

COCO-GAN enables a new type of image generation that has never been achieved by GANs before: generate full images that are larger than ***any*** training sample ***from scratch***. In this context, all the generated images are ***guaranteed*** to be novel and original, since these generated images do not even exist in the training distribution. A supportive evidence is that the generated images have higher resolution than any sample in the training data. In comparison, existing GANs mostly have their output shape fixed after its creation and prove the generator can produce novel samples instead of memorizing real data via interpolating between generated samples.

A shared and interesting behavior of learned manifold of GANs is that, in most cases, the generator can still produce plausible samples with latent vectors slightly out of the training distribution, which we called extrapolation. We empirically observe that with a fixed z, extrapolation can be done on the coordinate condition *beyond* the training coordinates distribution. However, as the continuity among patches at these positions is not considered during training, the generated images might show a slight discontinuity at the border. As a solution, we apply a straightforward post-training process (described in Appendix E) for improving the continuity among patches.

In Figure 7, we perform the post-training process on

* The model is not fully converged due to computational resource constraints. One can obtain even lower FID with more GPU-days.

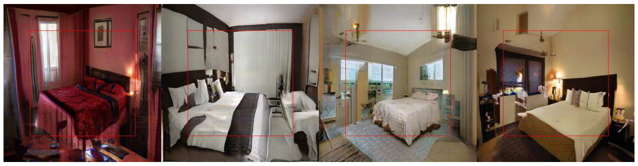


Figure 7: "Beyond-Boundary Generation" generates additional contents by extrapolating the learned coordinate manifold. Note that the generated samples are 384×384 pixels, whereas ***all*** of the training samples are of a smaller 256×256 resolution. The red box annotates the 256×256 region for regular generation without extrapolation. More generation samples are shown in Appendix E.

checkpoint of (N4,M4,S64) variant of COCO-GAN that trained on LSUN dataset. Then, we show that COCO-GAN generates high-quality 384×384 images: the original size is 256, with each direction being extended by one micro patch (64 pixels), resulting a size of 384×384. Note that the model is in fact trained on 256×256 images.

3.4. Panorama Generation & Partial Generation

Generating panoramas using GANs is an interesting problem but has never been carefully investigated. Different from normal image generation, panoramas are expected to be cylindrical and cyclic in the horizontal direction. However, normal GANs do not have built-in ability to handle such cyclic characteristic if without special types of padding mechanism support [4]. In contrast, COCO-GAN is a coordinate-system-aware learning framework. We can easily adopt a cylindrical coordinate system, and generate panoramas that are having "cyclic topology" in the horizontal direction as shown in Figure 8.

To train COCO-GAN with a panorama dataset under a cylindrical coordinate system, the spatial coordinate sampling strategy needs to be slightly modified. In the horizontal direction, the sampled value within the normalized range $[-1, 1]$ is treated as an angular value θ, and then is projected with $\cos(\theta)$ and $\sin(\theta)$ individually to form a unit-circle on a 2D surface. Along with the original sampling strategy on the vertical axis, a cylindrical coordinate system is formed.

We conduct our experiment on Matterport3D [2] dataset. We first take the sky-box format of the dataset, which consists of six faces of a 3D cube. We preprocess and project the sky-box to a cylinder using Mercator projection, then resize to 768×512 resolution. Since the Mercator projection creates extreme sparsity near the northern and southern poles, which lacks information, we directly remove the upper and lower $1/4$ areas. Eventually, the size of panorama we use for training is 768×256 pixels.

We also find COCO-GAN has an interesting connection with virtual reality (VR). VR is known to have a tight computational budget due to high frame-rate requirement and high-resolution demand. It is hard to generate full-scene for VR in real time using standard generative models. Some recent VR studies on omnidirectional view rendering and streaming [6, 21, 5] focus on reducing computational cost or network bandwidth by adapting to the user's viewport. COCO-GAN, with the generation-by-parts feature, can easily inherit the same strategy and achieve computation on-demand with respect to the user's viewpoint. Such a strategy can largely reduce unnecessary computational cost outside the region of interest, thus making image generation in VR more applicable.

3.5. Patch-Guided Image Generation

We further explore an interesting application of COCO-GAN named "Patch-Guided Image Generation". By training an extra auxiliary network Q within D that predicts the latent vector of each generated macro patch s', the discriminator is able to find a latent vector $z_{est} = Q(x')$ that generates a macro patch similar to a provided real macro patch x'. Moreover, the estimated latent vector z_{est} can be applied to the full-image generation process, and eventually generates an image that is partially similar to the original real macro patch, while globally coherent.

This application shares similar context to some bijection methods [8, 7, 3], despite COCO-GAN estimates the latent vector with a single macro patch instead of the full image. In addition, the application is also similar to image restoration [14, 28, 29] or image out-painting [23]. However, these related applications heavily rely on the information from the surrounding environment, which is not fully accessible from a single macro patch. In Figure 9, we show that our method is robust to extremely damaged images. More samples and analyses are described in Appendix K.

3.6. Computation-Friendly Generation

Recent studies in high-resolution image generation [13, 17, 12] have gained lots of success; however, a shared conundrum among these existing approaches is the computation being memory hungry. Therefore, these approaches make some compromises to reduce memory usage [13, 17]. Moreover, this memory bottleneck cannot be easily resolved without specific hardware support, which makes the generation of *over* 1024×1024 resolution images difficult to achieve. These types of high-resolution images are commonly seen in panoramas, street views, and medical images.

In contrast, COCO-GAN only requires partial views of the full image for both training and inference. Note that the memory consumption for training (and making inference) GANs grows approximately linearly with respect to the image size. Due to using only partial views, COCO-GAN changes the growth in memory consumption to be associated with the size of a macro patch, not the full image. For instance, on the CelebA 128×128 dataset, the (N2,M2,S16) setup of COCO-GAN reduces memory re-

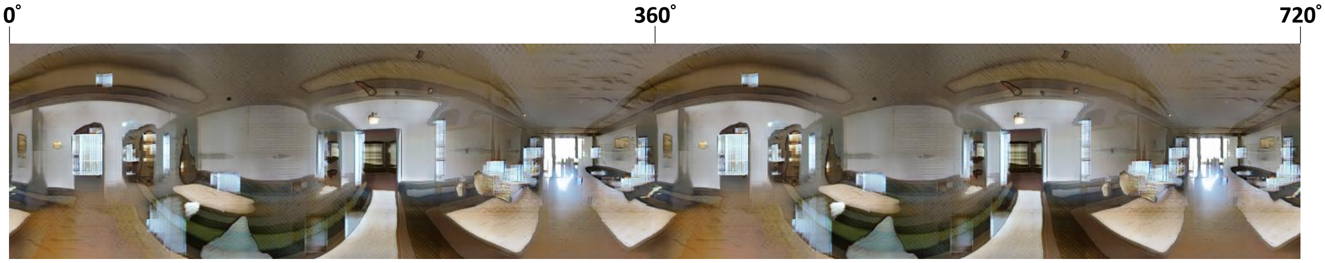


Figure 8: The generated panorama is cyclic in the horizontal direction since COCO-GAN is trained with a cylindrical coordinate system. Here, we paste the same generated panorama twice (from 360° to 720°) to better illustrate the cyclic property of the generated panorama. More generation results are provided in Appendix H.

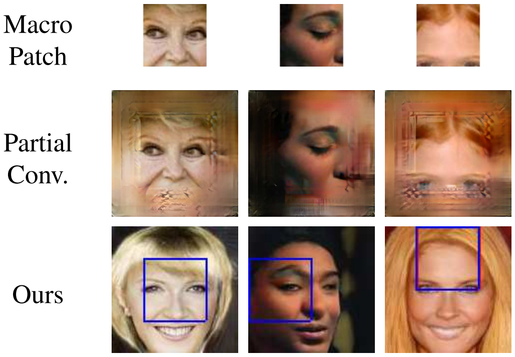


Figure 9: Patch-guided image generation loosely retains the local structures from the original image and make the full image still globally coherent. The quality outperforms the partial convolution [14]. The blue boxes visualize the predicted spatial coordinates $A(x')$, while the red boxes indicate the ground truth coordinates c'. Note that the generated images are **not** expected to be identical to the original real images. More examples are provided in Appendix K.

quirement from 17,184 MB (our projection discriminator backbone) to 8,992 MB (*i.e.*, 47.7% reduction), with a batch size 128. However, if the size of a macro patch is too small, COCO-GAN will be misled to learn incorrect spatial relation; in Figure 10, we show an experiment with a macro patch of size 32 × 32 and a micro patch size of 16 × 16. Notice the low quality (*i.e.*, duplicated faces). Empirically, the minimum requirement of macro patch size varies for different datasets; for instance, COCO-GAN does not show similar poor quality in panorama generation in Section 3.4, where the macro patch size is 1/48 of the full panorama. Future research on a) how to mitigate such effects (for instance, increase the receptive field of D without harming performance) and b) how to evaluate a proper macro patch size, may further advance the generation-by-parts property particularly in generating large field-of-view data.

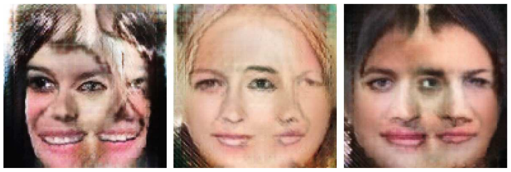


Figure 10: Examples to show that with macro patches smaller than 1/16 of the full image causes COCO-GAN to learn incorrect spatial relation. Note that this value may vary due to the nature (local structure, texture, etc) of each dataset being different.

Model	**best FID (150 epochs)**
COCO-GAN (cont. sampling)	6.13
COCO-GAN + optimal D	4.05
COCO-GAN + optimal G	6.12
Multiple G	7.26
COCO-GAN (N2,M2,S16)	4.87

Table 2: The ablation study shows that COCO-GAN (N2,M2,S16) can converge well with little trade-off in convergence speed on CelebA 64 × 64 dataset.

3.7. Ablation Study

In Table 2, the ablation study aims to analyze the **trade-offs** of each component of COCO-GAN. We perform experiments in CelebA 64 × 64 with four ablation configurations: "continuous sampling" demonstrates that using continuous uniform sampling strategy for spatial coordinates during training will result in moderate generation quality drop; "optimal D" lets the discriminator directly discriminate the full image while the generator still generates micro patches; "optimal G" lets the generator directly generate the full image while the discriminator still discriminates macro patches; "multiple G" trains an individual generator for each spatial coordinate.

We observe that, surprisingly, despite the convergence speed is different, "optimal discriminator", COCO-GAN, and "optimal generator" (ordered by convergence speed from fast to slow) can all achieve similar FID scores if with sufficient training time. The difference in convergence speed is expected since "optimal discriminator" provides the generator with more accurate and global adversarial loss. In contrast, the "optimal generator" has relatively more parameters and layers to optimize, which causes the convergence speed slower than COCO-GAN. Lastly, the "multiple generators" setting cannot converge well. Although it can also concatenate micro patches without obvious seams as COCO-GAN does, the full-image results often cannot agree and are not coherent. More experimental details and generated samples are shown in Appendix J.

3.8. Non-Aligned Dataset

It is easy to get confused that the coordinate system would restrain COCO-GAN from learning on less aligned datasets. In fact, this is completely not true. For instance, the bedroom category of LSUN, the location, size and orientation of the bed are very dynamic and non-aligned. On the other hand, the Matterport3D panoramas are completely non-aligned in the horizontal direction.

To further resolve all the potential concerns, we propose *CelebA-syn*, which applies a random displacement on the raw data (different from data augmentation, this preprocessing directly affects the dataset) to mess up the face alignment. We first trim the raw images to 128×128. The position of the upper-left corner is sampled by $(x, y) = (25 + dx, 50 + dy)$, where $dx \sim \mathcal{U}(-25, 25)$ and $dy \sim \mathcal{U}(-25, 25)$. Then we resize the trimmed images to 64×64 for training. As shown in Figure 11, COCO-GAN can stably create reasonable samples of high diversity (also notice the high diversity at the eye positions).

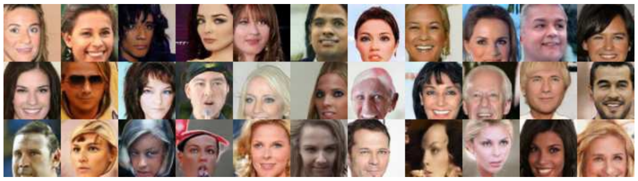


Figure 11: COCO-GAN can learn and synthesis samples with diverse position on the non-aligned *Celeba-syn*.

4. Related Work

Generative Adversarial Network (GAN) [9] and its conditional variant [18] have shown their potential and flexibility to many different tasks. Recent studies on GANs are focusing on generating high-resolution and high-quality synthetic images in different settings. For instance, generating images with 1024×1024 resolution [13, 17], generating images with low-quality synthetic images as condition [24], and by applying segmentation maps as conditions [26]. However, these prior works share similar assumptions: the model must process and generate the full image in a single shot. This assumption consumes an unavoidable and significant amount of memory when the size of the targeting image is relatively large, and therefore makes it difficult to satisfy memory requirements for both training and inference. Searching for a solution to this problem is one of the initial motivations of this work.

COCO-GAN shares some similarities to Pixel-RNN [25], which is a pixel-level generation framework while COCO-GAN is a patch-level generation framework. Pixel-RNN transforms the image generation task into a sequence generation task and maximizes the log-likelihood directly. In contrast, COCO-GAN aims at decomposing the computation dependencies between micro patches across the spatial dimensions, and then uses the adversarial loss to ensure smoothness between adjacent micro patches.

CoordConv [15] is another similar method but with fundamental differences. CoordConv provides spatial positioning information directly to the convolutional kernels in order to solve the coordinate transform problem and shows multiple improvements in different tasks. In contrast, COCO-GAN uses spatial coordinates as an auxiliary task for the GANs training, which enforces both the generator and the discriminator to learn coordinating and correlations between the generated micro patches. We have also considered incorporating CoordConv into COCO-GAN. However, empirical results show little visual improvement.

5. Conclusion and Discussion

In this paper, we propose COCO-GAN, a novel GAN incorporating the conditional coordination mechanism. COCO-GAN enables "generation by parts" and demonstrates the generation quality being competitive to state-of-the-arts. COCO-GAN also enables several new applications such as "Beyond-Boundary Generation" and "Panorama Generation", which serve as intriguing directions for future research on leveraging the learned coordinate manifold for (a) tackling with large field-of-view generation and (b) reducing computational requisition.

Particularly, given a random latent vector, Beyond-Boundary Generation generates images larger than *any* training sample by extrapolating the learned coordinate manifold, which is enabled exclusively by COCO-GAN. Future research on extending this property to other tasks or applications may further take advantage of such an out-of-distribution generation paradigm.

Although COCO-GAN has achieved a high generation quality comparable to state-of-the-art GANs, for several generated samples we still observe that the local structures may be discontinued or mottled. This suggests further studies on additional refinements or blending approaches that could be applied on COCO-GAN for generating more stable and reliable samples.

Acknowledgements

We sincerely thank David Berthelot and Mong-li Shih for the insightful suggestions, and National Center for High-performance Computing for computer time and facilities. Hwann-Tzong Chen was supported in part by MOST grants 107-2634-F-001-002 and 107-2218-E-007-047.

References

[1] Martín Arjovsky, Soumith Chintala, and Léon Bottou. Wasserstein generative adversarial networks. In *Proceedings of the 34th International Conference on Machine Learning, ICML 2017, Sydney, NSW, Australia, 6-11 August 2017*, pages 214–223, 2017.

[2] Angel X. Chang, Angela Dai, Thomas A. Funkhouser, Maciej Halber, Matthias Nießner, Manolis Savva, Shuran Song, Andy Zeng, and Yinda Zhang. Matterport3d: Learning from RGB-D data in indoor environments. In *2017 International Conference on 3D Vision, 3DV 2017, Qingdao, China, October 10-12, 2017*, pages 667–676, 2017.

[3] Chia-Che Chang, Chieh Hubert Lin, Che-Rung Lee, Da-Cheng Juan, Wei Wei, and Hwann-Tzong Chen. Escaping from collapsing modes in a constrained space. In *The European Conference on Computer Vision (ECCV)*, September 2018.

[4] Hsien-Tzu Cheng, Chun-Hung Chao, Jin-Dong Dong, Hao-Kai Wen, Tyng-Luh Liu, and Min Sun. Cube padding for weakly-supervised saliency prediction in 360 videos. In *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2018.

[5] Xavier Corbillon, Alisa Devlic, Gwendal Simon, and Jacob Chakareski. Optimal set of 360-degree videos for viewport-adaptive streaming. In *Proceedings of the 2017 ACM on Multimedia Conference, MM 2017, Mountain View, CA, USA, October 23-27, 2017*, pages 943–951, 2017.

[6] Xavier Corbillon, Gwendal Simon, Alisa Devlic, and Jacob Chakareski. Viewport-adaptive navigable 360-degree video delivery. In *IEEE International Conference on Communications, ICC 2017, Paris, France, May 21-25, 2017*, pages 1–7, 2017.

[7] Jeff Donahue, Philipp Krhenbhl, and Trevor Darrell. Adversarial feature learning. In *International Conference on Learning Representations*, 2017.

[8] Vincent Dumoulin, Ishmael Belghazi, Ben Poole, Alex Lamb, Martin Arjovsky, Olivier Mastropietro, and Aaron Courville. Adversarially learned inference. In *International Conference on Learning Representations*, 2017.

[9] Ian J. Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron C. Courville, and Yoshua Bengio. Generative adversarial nets. In *Advances in Neural Information Processing Systems 27: Annual Conference on Neural Information Processing Systems 2014, December 8-13 2014, Montreal, Quebec, Canada*, pages 2672–2680, 2014.

[10] Ishaan Gulrajani, Faruk Ahmed, Martín Arjovsky, Vincent Dumoulin, and Aaron C. Courville. Improved training of wasserstein gans. In *Advances in Neural Information Processing Systems 30: Annual Conference on Neural Information Processing Systems 2017, 4-9 December 2017, Long Beach, CA, USA*, pages 5769–5779, 2017.

[11] 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 30: Annual Conference on Neural Information Processing Systems 2017, 4-9 December 2017, Long Beach, CA, USA*, pages 6629–6640, 2017.

[12] Huaibo Huang, Zhihang Li, Ran He, Zhenan Sun, and Tieniu Tan. Introvae: Introspective variational autoencoders for photographic image synthesis. In *Advances in Neural Information Processing Systems 31: Annual Conference on Neural Information Processing Systems 2018, NeurIPS 2018, 3-8 December 2018, Montréal, Canada.*, 2018.

[13] Tero Karras, Timo Aila, Samuli Laine, and Jaakko Lehtinen. Progressive growing of gans for improved quality, stability, and variation. *CoRR*, abs/1710.10196, 2017.

[14] Guilin Liu, Fitsum A. Reda, Kevin J. Shih, Ting-Chun Wang, Andrew Tao, and Bryan Catanzaro. Image inpainting for irregular holes using partial convolutions. *CoRR*, abs/1804.07723, 2018.

[15] Rosanne Liu, Joel Lehman, Piero Molino, Felipe Petroski Such, Eric Frank, Alex Sergeev, and Jason Yosinski. An intriguing failing of convolutional neural networks and the coordconv solution. In *Advances in Neural Information Processing Systems 31: Annual Conference on Neural Information Processing Systems 2018, NeurIPS 2018, 3-8 December 2018, Montréal, Canada.*, pages 9628–9639, 2018.

[16] Ziwei Liu, Ping Luo, Xiaogang Wang, and Xiaoou Tang. Deep learning face attributes in the wild. In *Proceedings of International Conference on Computer Vision (ICCV)*, December 2015.

[17] Lars M. Mescheder, Andreas Geiger, and Sebastian Nowozin. Which training methods for gans do actually converge? In *Proceedings of the 35th International Conference on Machine Learning, ICML 2018, Stockholmsmässan, Stockholm, Sweden, July 10-15, 2018*, pages 3478–3487, 2018.

[18] Mehdi Mirza and Simon Osindero. Conditional generative adversarial nets. *CoRR*, abs/1411.1784, 2014.

[19] Takeru Miyato and Masanori Koyama. cgans with projection discriminator. *CoRR*, abs/1802.05637, 2018.

[20] Augustus Odena, Christopher Olah, and Jonathon Shlens. Conditional image synthesis with auxiliary classifier gans. In *Proceedings of the 34th International Conference on Machine Learning, ICML 2017, Sydney, NSW, Australia, 6-11 August 2017*, pages 2642–2651, 2017.

[21] Cagri Ozcinar, Ana De Abreu, and Aljosa Smolic. Viewport-aware adaptive 360° video streaming using tiles for virtual reality. In *2017 IEEE International Conference on Image Processing, ICIP 2017, Beijing, China, September 17-20, 2017*, pages 2174–2178, 2017.

[22] Alec Radford, Luke Metz, and Soumith Chintala. Unsupervised representation learning with deep convolutional generative adversarial networks. *CoRR*, abs/1511.06434, 2015.

[23] Mark Sabini and Gili Rusak. Painting outside the box: Image outpainting with gans. *arXiv preprint arXiv:1808.08483*, 2018.

[24] Ashish Shrivastava, Tomas Pfister, Oncel Tuzel, Joshua Susskind, Wenda Wang, and Russell Webb. Learning from simulated and unsupervised images through adversarial training. In *2017 IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2017, Honolulu, HI, USA, July 21-26, 2017*, pages 2242–2251, 2017.

[25] Aäron van den Oord, Nal Kalchbrenner, and Koray Kavukcuoglu. Pixel recurrent neural networks. In *Proceedings of the 33nd International Conference on Machine Learning, ICML 2016, New York City, NY, USA, June 19-24, 2016*, pages 1747–1756, 2016.

[26] 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. *CoRR*, abs/1711.11585, 2017.

[27] Tom White. Sampling generative networks: Notes on a few effective techniques. *CoRR*, abs/1609.04468, 2016.

[28] Chao Yang, Xin Lu, Zhe Lin, Eli Shechtman, Oliver Wang, and Hao Li. High-resolution image inpainting using multi-scale neural patch synthesis. In *2017 IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2017, Honolulu, HI, USA, July 21-26, 2017*, pages 4076–4084, 2017.

[29] Raymond A. Yeh, Chen Chen, Teck-Yian Lim, Alexander G. Schwing, Mark Hasegawa-Johnson, and Minh N. Do. Semantic image inpainting with deep generative models. In *2017 IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2017, Honolulu, HI, USA, July 21-26, 2017*, pages 6882–6890, 2017.

[30] Fisher Yu, Yinda Zhang, Shuran Song, Ari Seff, and Jianxiong Xiao. LSUN: construction of a large-scale image dataset using deep learning with humans in the loop. *CoRR*, abs/1506.03365, 2015.