Research Practice Final Report: Object-Centric GANs from In-the-wild Partial-Observable Images

Bingliang, Zhang
AI Class, IIIS, Tsinghua University

Introduction

In general, in order to train a GAN model, a training dataset should be built first. Each instance in the dataset is regarded as a sample from a hidden joint data distribution. The generator learns to generate instances lying in the same distribution as the training dataset. There are several limitations in this setting. First, such a training dataset should follow the i.i.d. assumption, which indicates an equal relationship between different instances. This requires content in different instances should be constrained in a certain scope. Second, each instance contributes the same for training the generator. No matter what the content is, the generator is trained to match each instance in the sense of distribution. As a result, the generator learns to imitate each instance. That's the reason why most success requires highly structurally aligned and selected dataset, such as human and animal faces which approximately obeys the above requirements. Training GANs directly from In-the-wild unprocessed images, for example from the Internet, is still a great challenge for current methods.

Our insight is can we find a way to focus directly on modeling certain target object and ignoring unrelated confusing contents? The contribution of each image for training the generator depends on the proportion of the target object contained in it. Then the entire dataset can be roughly divided into two parts, the full and partial images. Instead of mimicking the training dataset, the generator learns to generate clean images containing complete target object and simple backgrounds. The main benefit is that it reduces the requirement for the training dataset. Any images as long as containing some part of the target object can be added in, which makes training GANs from in-the-wild images possible.

There are several challenges that should be considered in this setting. First, we need to make a clear distinction between the main target object, occlusions, and background. Fortunately, thanks to the recent success of image segmentation techniques, this problem can be solved well. Second, the model should be aware of the proportion of the target object and corresponding region. Only in this way can it generate clean, full object images only. In addition, the modeling should not depend on the property of a certain category, it needs to generalize well to different classes.

To solve these challenges, we propose our object-centric GANs which allow us to train from in-the-wild images and generate clean, full-body object images. To distinguish between main object and occlusions, we use the off-the-shelf open vocabulary images segmentation model *Detic* (Zhou et al. 2022). Also, we modify the structure of the state-of-the-art StyleGAN3 to allow partial image generation, which is crucial to modeling partial and full object images. Specifically, a coordinate indicating the location and scale are added as input for the generator, which allows it to only generate contents in the corresponding region.

To test and verify our proposed method, we conduct several experiments on both synthetic dataset and real in-the-wild dataset. Current experimental results show a great potential of object-centric GAN to directly model the target object and generate full object images. The trained generator not only ignores the occlusion but also learns from partial observable images. One can refer to Fig 1. To summarize our contribution:

- propose a complete pipeline to train GAN from the in-the-wild dataset and give possible solutions to challenges including occlusion, partial observable issue, and any resolution source.
- demonstrate the potential to directly model the object through the modified StyleGAN3 architecture and patch-based training strategy.
- introduce a supervised network to predict coordinates of images to align them in canonical object space, which allows us to utilize both full and partial in-the-wild images.

Related Work

Image Synthesis

Recently unconditional image synthesis such as StyleGAN family (Karras et al. 2021), AnyresGAN (Chai et al. 2022) and CoordGAN(Mu et al. 2022) and conditional image synthesis including DALL·E 2(Ramesh et al. 2022) and Imagen(Saharia et al. 2022) have achieved great success. However, training of such models usually either requires a clean, aligned and manually processed dataset or requires an extremely huge dataset containing hundreds of millions of images from different categories. As a result, it often requires a lot of manpower and resources to process the data sets, which proposes a limitation in the flexibility of training the generative model. Our work tries to achieve a similar or even better generation quality of GANs with an in-the-wild image dataset.

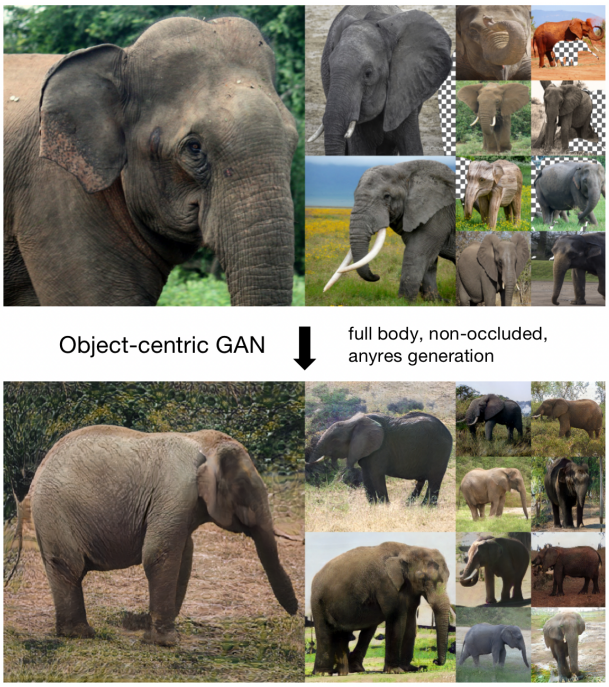


Figure 1: The main problem and visualization of results. The top images indicate any resolution in-the-wild dataset, which contains occlusion (checkerboard pattern), the partial observable object. The bottom images are generations of full body, non-occluded any resolution images from our proposed method object-centric GAN. The resolutions from left to right are 512, 256, 128.

Object-centric Learning

Discovering and learning to directly model object is the core of object-centric learning. Different from traditional frequency-based learning, object-centric learning requires the model to be aware of the concept of object, a continuous group of pixels that has a similar semantic meaning. Slot attention (Locatello et al. 2020) is a transformer-based architecture proposed to aggregate information only from pixels of a certain category. It shows the potential to discover different objects in a simple scene. GIRAFFE(Niemeyer and Geiger 2021), a 3D aware generative model, modeled a scene as the composition of different objects which allows us to manipulate a single object in a complex scene. However, different from learning to distinguish different objects in images, our method learns to model different parts and distinguish between the partial and full body of a given object.

Internet Image

The one of simplest ways to obtain a set of images for a certain category is by searching on the Internet. However, they often contain a lot of outliers and data biases, which may reduce the quality of the dataset and harm subsequent training. Self-distilled StyleGAN (Mokady et al. 2022) used an inversion score to filter out those unqualified images and trained the model with the self-distilled dataset. Yu et al. (Yu et al. 2015) tried to train a classifier to filter outliers with a small human-label dataset. In our method, we try to combine automatic filtering and human labeling together for a better utility of Internet images.

Formulation

In this section, we give the detailed and mathematical formulation of our problem. Given a certain category, say elephant, we assume there is a hidden distribution of all full body, non-occluded images $P_{data}(I)$, where I is a continuous 2D RGB image, or mathematically a function. For simplicity, we assume the 2D region is $[0,1] \times [0,1]$ and each point in the region corresponds to a continuous 3-dimensional RGB value in $[0,1]^3$, then I is a function such that:

$$I : [0,1]^2 \to [0,1]^3$$
$$I(x,y) = (r,g,b)$$

Let $D_{h,w}(I)$ be the restriction of function I on a uniform $h \times w$ grids of $[0,1]^2$, which corresponds to a general rectangle RGB image of height h and width w. Then a sample of any resolution image can be represented as:

$$D_{h,w}(I), \quad I \sim P_{data}$$

Partial observable images can be regarded as a *patch* in a full body image, whose location and scale are indicated by a 4-dimensional coordinates:

$$c = (x_1, y_1, x_2, y_2) \in [0,1]^4, \quad x_1 < x_2, y_1 < y_2$$

Let $C(I;c)$ be restriction of I in domain $[x_1, x_2] \times [y_1, y_2]$. Intuitively, $C(I,c)$ is the partially cropped patch of a larger full body image I. Thus we can model a partial-observable image with height h, width w and coordinate c from the in-the-wild dataset as below:

$$C(D_{h,w}(I);c), \quad I \sim P_{data} \tag{1}$$

So far, we have modeled any resolution and partial observable issues. One thing that remained is occlusion. Usually, occlusion means the parts obscured by other irrelevant objects. However, based on this definition, we need to analyze depth relationships in 2D images before determining occluded regions. So here we relax the definition of occlusion as parts that are occupied by other irrelevant confusing objects. For example, if the elephant is our main target to model, then a person in the image is regarded as an occlusion. Thus, we can represent occlusion as a binary mask of the same resolution. Let b_i to be binary occlusion mask of partial observable image $C(D_{h,w}(I);c)$ with height h and width w. For pixel located at (x,y), $b_i(x,y) = 1$ if and only if pixel is occluded.

Then the overall problem can be formulated as given a partial observable training dataset S of size N

$$S = \{C(D_{h_i,w_i}(I_i)); c_i)\}_{i=1}^{N}, \quad I_i \sim P_{data} \tag{2}$$

with unknown coordinates c_i and occlusion masks b_i, train a generator G such that the distribution of generated instances matches with full data distribution

$$P_G \approx P_{data}$$

In another word, G directly generates fully observable, any-resolution images of a certain object. In this definition, any images with $c = (0, 1, 0, 1)$ are full object images and others are partial images. However in experiments, images is treated as full object images if $c = (x_1, x_2, y_1, y_2)$ is very close to $(0, 1, 0, 1)$ (e.g. $x_1, y_1 < 0.05$, $x_2, y_2 > 0.95$). So that, the entire dataset can be roughly separated into the full and partial dataset.

Since in general, b_i and c_i are unknown, we propose some preprocessing steps to predict them. To summary, the problem can be formulated sequentially:

- Given an in-the-wild dataset S of size N, predicts b_i and c_i, $\forall i$.
- Train a generator G, so that the induced distribution P_G approximates real data distribution P_{data}.

Method

In this section, we give a detailed explanation of our proposed Object-centric GANs. The pipeline can be roughly divided into three stages–pre-processing, coordinate prediction, and any-resolution patch-based training. The illustration of the training pipeline is shown in Figure 2.

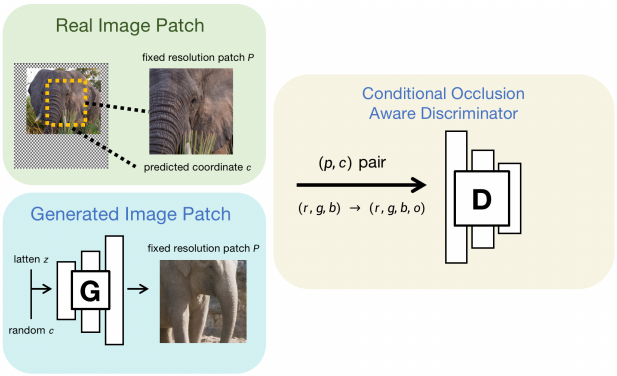


Figure 2: Patched-based training pipeline with conditional occlusion aware discriminator.

Pre-processing

For a given class, say elephant, one can crawl and get images through the Internet search. It can be images from google search or other image websites like Flickr. We refer to these images as *raw dataset* later. Due to the accuracy of the selection procedure, the raw dataset usually contains some outliers, which should be filtered out. In this step, we utilize an off-the-shelf any-resolution, open vocabulary segmentation model *Detic* to obtain the Segmentation mask of the target object and other objects. Here we directly adopt the segmentation mask of other objects, the union of all instance segmentation masks excluded from the target object, as *occlusion mask*. Then three following steps are performed:

- Filtering. Images are removed if the proportion of the target object is less than a threshold (in the experiment we use 10%). In this way, we make sure each remained image contains at least some useful information about the target object.
- Cropping. Remained images are then cropped by the dilated (in the experiment we dilate 5% of each side) bounding box of the target object. If more than one instance is contained in an image (for example, an image of 3 elephants), then each instance is processed individually while regarding other instances as occlusion.
- Obtaining occlusion mask. As we mentioned above, we directly use the union of other objects' segmentation masks as the occlusion mask. However, considering that there may be many objects in some pictures, we use a threshold to screen out some objects that can be ignored to ensure that as much content as possible remains.

Coordinate Prediction

For each partial observable image, we want to predict the hidden unknown coordinates $c = (x_1, y_1, x_2, y_2)$ given only partial image itself. The accuracy of coordinates determines whether we can effectively use the information of the partial-observable images. Because it involves the understanding of the specific content of the image and the modeling of the object, it is hard to train the predictor in an unsupervised manner.

So in this step, we try to utilize some human effort. specifically, we first manually select a *small subset of full body images* from the entire dataset. Once we have this set of full body images, say $S' = \{D_{r_{h_i,w_i}}(I_i)\}_{i=1}^{M}$, we can simulate a small paired dataset by crop patches with uniform randomly selected coordinates c. The small paired dataset $\{X_i, Y_i\}$ is represented as below:

$$\{X_i = (C(D_{r_i}(I_i); c_i), Y_i = c_i\} \tag{3}$$

Then a coordinate predictor C_θ can be trained supervisedly to minimize predicted coordinates and ground truth coordinates:

$$\hat{\theta} = \arg\min_{\theta} \sum_{i=1}^{M} \mathcal{L}(Y_i, C_\theta(X_i)) \tag{4}$$

Inspired by the literature on object detection, we use the sum of ℓ_1 loss and GIoU (Rezatofighi et al. 2019) loss as the final loss

An interactive user interface is designed to speed up the selection process. As shown by the following experiment. For a certain class, usually, $100 \sim 300$ images are enough to achieve great prediction accuracy. It usually can be done by one person in less than 15 minutes.

In order to further improve the accuracy of the prediction, we also adopted a two-step bootstrapping technique. That is we train the first $C_{\theta,1}$ with the paired dataset from manually selected full object images. Based on its prediction, we can select all full object images from the entire dataset, which are usually hundreds of times large than manually selected ones. Then train the second $C_{\theta,2}$ on the much larger paired dataset. The prediction accuracy usually improves by a large margin.

Any Resolution Patch Based Training

Once coordinate predictor $C_{\hat{\theta}}$ is trained, it can be used to compute the coordinate of the entire dataset and select all

full body images. In another word, we can have the following paired training dataset:

$$\{X_i = C(D_{r_i}(I_i), c_i), Y_i = \hat{c}_i)\}, \quad Y_i = C_{\hat{\theta}}(X_i)$$

The goal of this step is to train a generator G that can simulate distribution P_{data}. In order to do this, we proposed the following components.

Modified StyleGAN3 Coordinate Based Generator Inspired by the AnyresGAN (Chai et al. 2022), we can model StyleGAN3 generator as a two input function $G(z, c)$, where z follows multivariate gaussian distribution and $c = (x_1, y_1, x_2, y_2)$ is a square coordinate with property $x_2 - x_1 = y_2 - y_1$. Then $G(z, c)$ is supposed to generate an image patch at location c at a *fixed* resolution. This modeling can be easily done by replacing the fixed coordinate grid with patch-dependent coordinates in the StyleGAN3 generator.

For simplicity, we define $c_0 := (0, 1, 0, 1)$. Then the induced full body generation distribution P_G can be represented as:

$$G(z, c_0), \quad z \sim \mathcal{N}$$

Conditioned Occlusion Aware Discriminator In general unconditional GAN training, the discriminator is optimized to distinguish between real and generated images. However, in our case, the generator only synthesizes a square patch at a given coordinate. The patches have various locations and scales. So instead of taking a full image as input, the discriminator takes in an image patch and conditions on coordinate and scale. Then the discriminator is optimized to classify real and generated patches of a given location and scale.

Also, in order to stimulate object-centric modeling, occlusion should not be used as the criterion for discriminator classification. The simplest way is to multiply image patches with occlusion masks to blackout the occluded region: $X_i \times b_i$. However, simply blackout may lead to unstable training and collapse. We find appending the occlusion mask as an additional channel will solve this issue and stabilize training. For real images, we directly use the occlusion mask obtained in pre-process. For generated images, to maintain the same distribution, we randomly select an occlusion mask from the real images dataset. This prevents the generator from modeling unrelated occlusion. In another word, the RGB value is extended to a *RGBO* value. For original non-occluded region, $(r, g, b) \rightarrow (r, g, b, 0)$ while occluded region $(r, g, b) \rightarrow (0, 0, 0, 1)$. One can refer to Fig 3 for understanding blackout and append operations.

Balanced Sampling and Cut-mix Regularization Our ultimate goal is to generate full body images, that is $G(z, c_0)$. However during training, coordinate c is randomly selected from the training set $\{\hat{c}_i\}_{i=1}^N$, which leads to a training and inference gap since in general full body images are less than partial observable images. To reduce this gap, we oversample full body images so that 50% of the time $c \approx c_0$.

In addition, we introduce a cut-mix regularization to reinforce global consistency. Specifically, given the same latent z, if we replace the content of location c in $G(z, c_0)$ with $G(z, c)$, the resulting image should also be able to cheat the discriminator as $G(z, c_0)$ does.

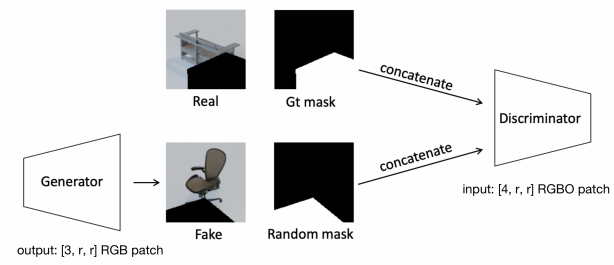


Figure 3: Occlusion-aware Input for the discriminator. For the real image patch, we adopt the predicted occlusion mask. For fake-generated image patch, we utilize a randomly selected occlusion mask from all predicted ones. This mitigates the instability of training and ignores the occluded region.

By combining these two techniques, we reduce the training and inference gap. The quality of generated full body images is improved by a large margin.

Any-res Generation

We notice that $G(z, c_0)$ only gives a full body image at a fixed resolution. To get any-resolution images, we can divide the entire image space into several fixed-resolution patches with coordinates. Each patch can be generated separately with the same latent code z and finally can be combined into a higher resolution image. Our training procedure guarantees the consistency of different patches with the same latent z.

Experimental results

To test and verify our proposed method, we conducted experiments with both the synthetic dataset and the in-the-wild dataset for each part of our pipeline.

Dataset

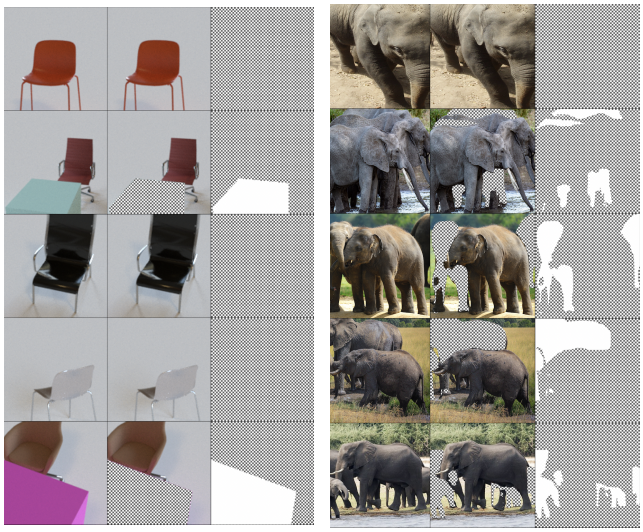
Visualization of dataset is shown in Figure 4. We use a black-white checkerboard to indicate non-observable parts.

Synthetic chair dataset We synthesize a partial observable chair dataset with occlusion from 3D chair models. In this dataset, we have the ground truth *occlusion masks* and *coordinates* of each image. Some random cubes are added as occlusion. All images are at a resolution of 128. The size of the dataset is $20K$.

In-the-wild elephant dataset We crawl a real in-the-wild elephant dataset from Flickr with any resolution. Also, elephants in images are unaligned and usually partially observable. The human and the tree can be potential occlusion objects. The size of the dataset is $100K$, containing $25K$ full body elephants (given by pre-trained bounding box predictor).

Full object, non-occluded image generation

We trained object-centric GAN for both the chair and the elephant dataset. The results show the potential of generating full object without occlusion. For both datasets, we select $5000k$ image checkpoint for evaluation. Following we show both the qualitative and quantitative results.



Examples of chair dataset Example of elephant dataset

Figure 4: Visualization of image and occlusion mask.

Generating full object, non-occluded images

If generating images from trained G in base resolution, we can notice the generated images contain full object and no occlusion. The results are shown in Figure 5.

Generating any resolution images

Here we focus on any resolution generation. If we fix latent code z and generate images with different resolutions, we can notice that generator will preserve the main content and fill in more detailed textures when resolution increases. The results are shown in Figure **??**. Any resolution generation results are shown in Figure 7

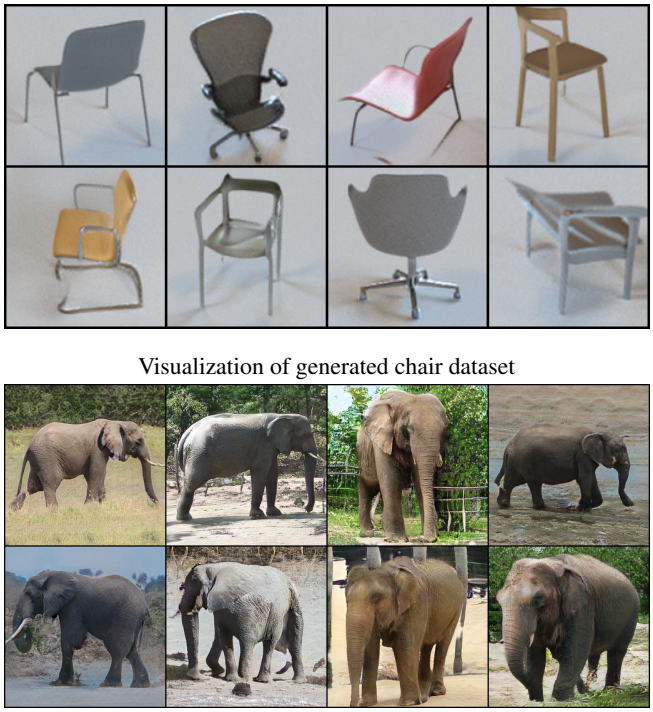
Quantitative evaluation with patch-FID

For quantitative evaluation, we utilize the modified patch-FID proposed in AnyresGAN. Specifically, we use the full body images selected by the bounding box predictor as the reference dataset. Then FID score is computed with uniform randomly selected patches from the reference dataset and generated patches from the generator. The coordinates distribution is the same for both real and generated images.

Here we show the patch-FID of the elephant dataset. In order for comparison, we adopt the model trained with $25k$ full object images as the baseline. The object-centric model indicates the one trained with entire $100k$ images ($25k$ full + $75k$ partial) together with balanced sampling and cut-mix regularization. The results are shown in Table 1.

Model	Size	Patch-FID
baseline	100k	7.76
object-centric GAN	100k	6.58

Table 1: The patch-FID comparison of elephant dataset. It shows that our proposed object-centric GAN can get improvement by incorporating partial-observable images. The size indicates the number of patches used in reference dataset.



Visualization of generated chair dataset

Visualization of generated elephant dataset

Figure 5: Generated images at base resolution with coordinates $c_0 = (0, 1, 0, 1)$. The images are almost full object, non-occluded images

Ablation: bootstrapping technique of coordinate prediction

To evaluate the effectiveness of bootstrapping technique, we compute the IoU score between predicted coordinates and ground truth ones on a validation dataset, whose images are excluded from the training dataset. Here we conduct experiments on the elephant dataset and the size of the validation dataset is $5k$. The results are shown in Table 2.

Number of images	IoU (training set)	IoU (validation set)
300(base)	0.914	0.834
12500(bs;half)	0.897	0.862
25000(bs;all)	0.886	0.876

Table 2: The evaluation IoU score of elephant dataset. "base" indicates model trained with manually selected images and "bs", bootstrapping, indicates model trained with dataset selected by "base". An IoU score improvement on the validation dataset is observable.

Also, we can visualize the prediction results of the bounding box predictor trained with all full object dataset. Figure 8 shows the difference between predicted coordinates and ground truth ones. Figure 9 shows the warped partial images by the predicted coordinates. One can verify their correctness by intuition.

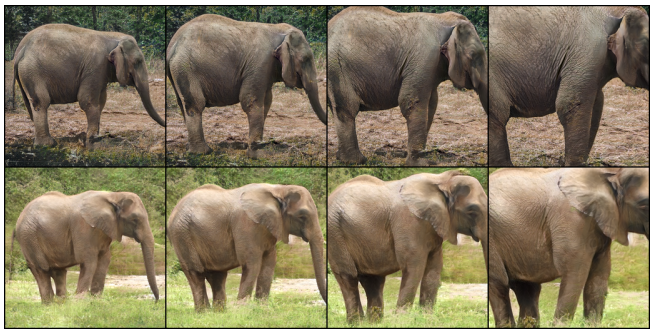


Figure 6: Visualization of generated patches of gradually increased resolution. From left to right, the resolutions are changed from 256, 320, 384, 448.

Figure 7: Visualization of any-res generation of chair dataset. The resolution from left to right is 512, 256, 128.

Discussion

In this section, we give the overall discussion of the overall proposed pipeline and object-centric GAN method.

Conclusion

As shown in the experiments, the object-centric GAN has a great potential to directly model the full-body object from the in-the-wild dataset. This gives us a possible solution to train a high-quality generator without complex alignment and selection. Instead, we can build data sets of arbitrary objects with a small amount of manpower and simple preprocessing, which makes it convenient for training a wider range of generators with a variety of categories.

Also, as proven by most any-resolution methods, patch-based training is a relatively useful technique to utilize information in various resolutions. With increasing resolution, the generator may fill in with some detailed texture and pattern.

Limitation & Possible Further Improvement

Although our method solves the proposed problem to a certain extent, it still has the following limitations:

Sequential Pipeline As mentioned in the method, we need to predict b_i and c_i before training the generator. Then the accuracy of prediction will restrict the quality of the generator to reach optimal since we fix the predicted b_i and c_i unchanged then.

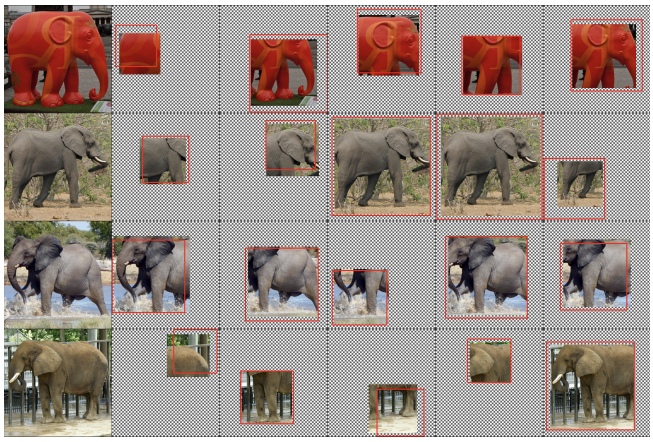


Figure 8: Visualization of evaluation of coordinate predictor on the validation dataset. The red bounding box indicates the ground truth coordinate and the warped image indicates the predicted coordinate.

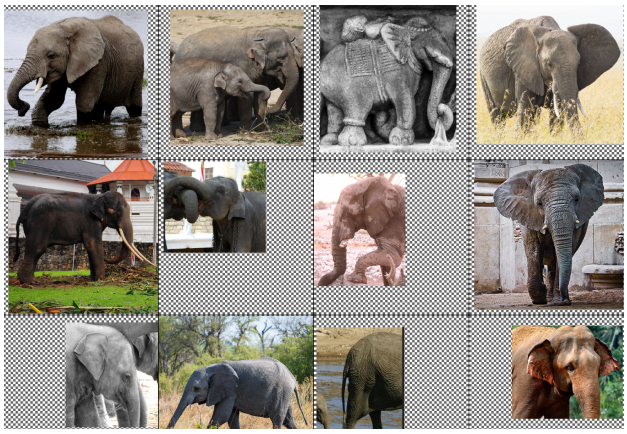


Figure 9: Visualization of warped image based on predicted coordinates. qualitatively, the prediction is accurate enough

One possible solution is that we try to joint training bounding box predictor and generator. However simply alternately training two networks collapse very frequently. A more effective joint training needs to be developed.

Human Effort To train the bounding box predictor, we still require a small subset of manually selected full object images (usually $100 \sim 300$). Although this is the trade-off between the versatility and quality of the generation, it still prevents our method from efficiently modeling objects from the in-the-wild dataset.

One possible solution is that instead of using a manually selected dataset, we can utilize some pre-trained models to automatically achieve that. The difficulty here is that it is very hard to find an effective pre-trained model appropriate for various objects.

References

Chai, L.; Gharbi, M.; Shechtman, E.; Isola, P.; and Zhang, R. 2022. Any-resolution Training for High-resolution Image Synthesis. *arXiv preprint arXiv:2204.07156* .

Karras, T.; Aittala, M.; Laine, S.; Härkönen, E.; Hellsten, J.; Lehtinen, J.; and Aila, T. 2021. Alias-free generative adversarial networks. *Advances in Neural Information Processing Systems* 34: 852–863.

Locatello, F.; Weissenborn, D.; Unterthiner, T.; Mahendran, A.; Heigold, G.; Uszkoreit, J.; Dosovitskiy, A.; and Kipf, T. 2020. Object-centric learning with slot attention. *Advances in Neural Information Processing Systems* 33: 11525–11538.

Mokady, R.; Tov, O.; Yarom, M.; Lang, O.; Mosseri, I.; Dekel, T.; Cohen-Or, D.; and Irani, M. 2022. Self-Distilled StyleGAN: Towards Generation from Internet Photos. In *Special Interest Group on Computer Graphics and Interactive Techniques Conference Proceedings*, 1–9.

Mu, J.; De Mello, S.; Yu, Z.; Vasconcelos, N.; Wang, X.; Kautz, J.; and Liu, S. 2022. CoordGAN: Self-Supervised Dense Correspondences Emerge from GANs. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 10011–10020.

Niemeyer, M.; and Geiger, A. 2021. Giraffe: Representing scenes as compositional generative neural feature fields. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 11453–11464.

Ramesh, A.; Dhariwal, P.; Nichol, A.; Chu, C.; and Chen, M. 2022. Hierarchical text-conditional image generation with clip latents. *arXiv preprint arXiv:2204.06125* .

Rezatofighi, H.; Tsoi, N.; Gwak, J.; Sadeghian, A.; Reid, I.; and Savarese, S. 2019. Generalized Intersection over Union .

Saharia, C.; Chan, W.; Saxena, S.; Li, L.; Whang, J.; Denton, E.; Ghasemipour, S. K. S.; Ayan, B. K.; Mahdavi, S. S.; Lopes, R. G.; et al. 2022. Photorealistic Text-to-Image Diffusion Models with Deep Language Understanding. *arXiv preprint arXiv:2205.11487* .

Yu, F.; Seff, A.; Zhang, Y.; Song, S.; Funkhouser, T.; and Xiao, J. 2015. Lsun: Construction of a large-scale image dataset using deep learning with humans in the loop. *arXiv preprint arXiv:1506.03365* .

Zhou, X.; Girdhar, R.; Joulin, A.; Krähenbühl, P.; and Misra, I. 2022. Detecting twenty-thousand classes using image-level supervision. *arXiv preprint arXiv:2201.02605* .