COMPOSITIONAL GAN:
LEARNING CONDITIONAL IMAGE COMPOSITION

Anonymous authors
Paper under double-blind review

ABSTRACT

Generative Adversarial Networks (GANs) can produce images of surprising complexity and realism, but are generally structured to sample from a single latent source ignoring the explicit spatial interaction between multiple entities that could be present in a scene. Capturing such complex interactions between different objects in the world, including their relative scaling, spatial layout, occlusion, or viewpoint transformation is a challenging problem. In this work, we propose to model object composition in a GAN framework as a self-consistent composition-decomposition network. Our model is conditioned on the object images from their marginal distributions and can generate a realistic image from their joint distribution. We evaluate our model through qualitative experiments and user evaluations in scenarios when either paired or unpaired examples for the individual object images and the joint scenes are given during training. Our results reveal that the learned model captures potential interactions between the two object domains given as input to output new instances of composed scene at test time in a reasonable fashion.

1 INTRODUCTION

Generative Adversarial Networks (GANs) have emerged as a powerful method for generating images conditioned on a given input. The input cue could be in the form of an image (Isola et al., 2017; Zhu et al., 2017; Liu et al., 2017; Azadi et al., 2017; Wang et al., 2017; Pathak et al., 2016), a text phrase (Zhang et al., 2017; Reed et al., 2016b;a; Johnson et al., 2018) or a class label layout (Mirza & Osindero, 2014; Odena et al., 2016; Antoniou et al., 2017). The goal in most of these GAN instantiations is to learn a mapping that *translates* a given sample from source distribution to generate a sample from the output distribution. This primarily involves transforming either a single object of interest (apples to oranges, horses to zebras, label to image etc.), or changing the style and texture of the input image (day to night etc.). However, these direct input-centric transformations do not directly capture the fact that a natural image is a 2D projection of a *composition* of multiple objects interacting in a 3D visual world. In this work, we explore the role of compositionality in learning a function that maps images of different objects sampled from their marginal distributions (e.g., chair and table) into a combined sample (table-chair) that captures their joint distribution.

Modeling compositionality in natural images is a challenging problem due to the complex interaction possible among different objects with respect to relative scaling, spatial layout, occlusion or viewpoint transformation. Recent work using spatial transformer networks (Jaderberg et al., 2015) within a GAN framework (Lin et al., 2018) decomposes this problem by operating in a geometric warp parameter space to find a geometric modification for a foreground object. However, this approach is only limited to a fixed background and does not consider more complex interactions in the real world. Another recent work on scene generation conditioned on text and a scene graph and explicitly provides reasoning about objects and their relations (Johnson et al., 2018).

We develop a novel approach to model object compositionality in images. We consider the task of composing two input object images into a joint image that captures their joint interaction in natural images. For instance, given an image of a chair and a table, our formulation should be able to generate an image containing the same chair-table pair interacting naturally. For a model to be able to capture the composition correctly, it needs to have the knowledge of occlusion ordering, i.e., a table comes in front of chair, and spatial layout, i.e., a chair slides inside table. To the best of our knowledge, we

are among the first to solve this problem in the image conditional space without any prior explicit information about the objects' layout.

Our key insight is to reformulate the problem of composition of two objects into first composing the given object images to generate the joint combined image which models the object interaction, and then decomposing the joint image back to obtain individual ones. This reformulation enforces a self-consistency constraint (Zhu et al., 2017) through a composition-decomposition network. However, in some scenarios, one does not have access to the paired examples of same object instances with their combined compositional image, for instance, to generate the joint image from the image of a given table and a chair, we might not have any example of that particular chair besides that particular table while we might have images of other chairs and other tables together. We add an inpainting network to our composition-decomposition layers to handle the unpaired case as well.

Through qualitative and quantitative experiments, we evaluate our proposed Compositional-GAN approach in two training scenarios: (a) paired: when we have access to paired examples of individual object images with their corresponding composed image, (b) unpaired: when we have a dataset from the joint distribution without being paired with any of the images from the marginal distributions.

2 RELATED WORK

Generative adversarial networks (GANs) have been used in a wide variety of settings including image generation (Denton et al., 2015; Yang et al., 2017b; Karras et al., 2017) and representation learning (Radford et al., 2016; Salimans et al., 2016; Liu & Tuzel, 2016; Chen et al., 2016). The loss function in GANs have been shown to be very effective in optimizing high quality images conditioned on available information. Conditional GANs (Mirza & Osindero, 2014) generate appealing images in a variety of applications including image to image translation both in the case of paired (Isola et al., 2017) and unpaired data (Zhu et al., 2017), inpainting missing image regions (Pathak et al., 2016; Yang et al., 2017a), generating photorealistic images from labels (Mirza & Osindero, 2014; Odena et al., 2016), and solving for photo super-resolution (Ledig et al., 2016; Lai et al., 2017).

Image composition is a challenging problem in computer graphics where objects from different images are to be overlayed in one single image. Appearance and geometric differences between these objects are the obstacles that can result in non-realistic composed images. Zhu et al. (2015) addressed the composition problem by training a discriminator network that could distinguish realistic composite images from synthetic ones. Tsai et al. (2017) developed an end-to-end deep CNN for image harmonization to automatically capture the context and semantic information of the composite image. This model outperformed its precedents (Sunkavalli et al., 2010; Xue et al., 2012) which transferred statistics of hand-crafted features to harmonize the foreground and the background in the composite image. Recently, Lin et al. (2018) used spatial transformer networks as a generator by performing geometric corrections to warp a masked object to adapt to a fixed background image. Moreover, Johnson et al. (2018) computed a scene layout from given scene graphs which revealed an explicit reasoning about relationships between objects and converted the layout to an output image. Despite the success all these approaches gained in improving perceptual realism, they lack the realistic complex problem statement where no explicit prior information about the scene layout is given. In the general case which we address, each object should be rotated, scaled, and translated in addition to occluding others and/or being occluded to generate a realistic composite image.

3 COMPOSITIONAL GAN ARCHITECTURE

We propose a generative network for composing two objects by learning how to handle their relative scaling, spatial layout, occlusion, and viewpoint transformation. Given a set of images from the marginal distribution of the first object, $X = \{x_1, \cdots, x_n\}$, and a set of images from the marginal distribution of the second object, $Y = \{y_1, \cdots, y_n\}$, in addition to a set of real images from their joint distribution containing both objects, $C = \{c_1, \cdots, c_n\}$, we generate realistic composite images containing objects *given* from the first two sets. We propose a conditional generative adversarial network for two scenarios: (1) paired inputs-output in the training set where each image in C is correlated with an image in X and one in Y, and (2) unpaired training data where images in C are not paired with images in X and Y. It is worth noting that our goal is not to learn a generative model of all possible compositions, but learn to output the mode of the distribution. The modular components of our proposed approach are critical in learning the mode of plausible compositions. For instance, our relative appearance flow network handles the viewpoint and the spatial transformer network

handles affine transformation eventually making the generator invariant to these transformations. In the following sections, we first summarize a conditional generative adversarial network, and then will discuss our network architecture and its components for the two circumstances.

3.1 CONDITIONAL GENERATIVE ADVERSARIAL NETWORKS

Starting from a random noise vector, z, GANs generate images c of a specific distribution by adversarially training a generator, G, versus a discriminator, D. While the generator tries to produce realistic images, the discriminator opposes the generator by learning to distinguish between real and fake images. In the conditional GAN models (cGANs), an auxiliary information, x, in the form of an image or a label is fed into the model alongside the noise vector ($\{x, z\} \rightarrow c$) Goodfellow (2016); Mirza & Osindero (2014). The objective of cGANs would be therefore an adversarial loss function formulated as:

$$\mathcal{L}_{cGAN}(G, D) = \mathbb{E}_{x,c\sim p_{\text{data}}(x,c)}[\log D(x, c)] + \mathbb{E}_{x\sim p_{\text{data}}(x),z\sim p_z(z)}[1 - \log D(x, G(x, z))]$$

where G and D minimize and maximize this loss function, respectively.

The convergence of the above GAN objective and consequently the quality of generated images would be improved if an L_1 loss penalizing deviation of generated images from their ground-truth is added. Thus, the generator's objective function would be summarized as:

$$G^* = \arg \min_G \max_D \mathcal{L}_{cGAN}(G, D) + \lambda \mathbb{E}_{x,c\sim p_{\text{data}}(x,c),z\sim p_z(z)}[\|c - G(x, z)\|_1]$$

In our proposed compositional GAN, model ($\{(x, y), z\} \rightarrow c$) is conditioned on *two* input images, (x, y), concatenated channel-wise in order to generate an image from the target distribution $p_{\text{data}}(c)$. We have access to real samples of these three distributions during training (in two paired and unpaired scenarios). Similar to (Isola et al., 2017), we ignore random noise as the input to the generator, and dropout is the only source of randomness in the network.

3.2 RELATIVE APPEARANCE FLOW NETWORK (RAFN)

In some specific domains, the relative view point of the objects should be changed accordingly to generate a natural composite image. Irrespective of the paired or unpaired inputs-output cases, we train a relative encoder-decoder appearance flow network (Zhou et al., 2016), G_{RAFN}, taking the two input images and synthesizing a new viewpoint of the first object, x_i^{RAFN}, given the viewpoint of the second one, y_i encoded in its binary mask. The relative appearance flow network is trained on a set of images in X with arbitrary azimuth angles $\alpha_i \in \{-180°, -170°, \cdots, 170°, 180°\}$ along with their target images in an arbitrary new viewpoint with azimuth angle $\theta_i \in \{-180°, -170°, \cdots, 170°, 180°\}$ and a set of foreground masks of images in Y in the target viewpoints. The network architecture for our relative appearance flow network is illustrated in the appendix and its loss function is formulated as:

$$
\begin{aligned}
\mathcal{L}(G_{\text{RAFN}}) &= \mathcal{L}_{L_1}(G_{\text{RAFN}}) + \lambda \mathcal{L}_{\text{BCE}}(G_{\text{RAFN}}^M) \\
&= \mathbb{E}_{(x_i,y_i)\sim p_{\text{data}}(x_i,y_i)}[\|x_i^r - G_{\text{RAFN}}(M_{y_i}^{\text{fg}}, x_i)\|_1] \\
&\quad + \lambda \mathbb{E}_{x_i\sim p_{\text{data}}(x_i)}[\hat{M}_{x_i}^{\text{fg}} \log M_{x_i}^{\text{fg}} + (1 - \hat{M}_{x_i}^{\text{fg}}) \log(1 - M_{x_i}^{\text{fg}})]
\end{aligned}
\tag{1}
$$

As mentioned above, G_{RAFN} is the encoder-decoder network predicting appearance flow vectors, which after a bilinear sampling generates the synthesized view. The encoder-decoder mask generating network, G_{RAFN}^M, shares weights in its encoder with G_{RAFN}, while its decoder is designed for predicting foreground mask of the synthesized image. Moreover, x_i^r is the ground-truth image for x_i in the new viewpoint, $\hat{M}_{x_i}^{\text{fg}}$ is its predicted foreground mask, and $M_{x_i}^{\text{fg}}$, $M_{y_i}^{\text{fg}}$ are the ground-truth foreground masks for x_i and y_i, respectively.

3.3 PAIRED TRAINING DATA

In this section, we propose a model, G, for composing two objects when there is a corresponding composite real image for each pair of input images in the training set. In addition to the relative AFN discussed in the previous section, to relatively translate the center-oriented input objects, we train our variant of the spatial transformer network (STN) (Jaderberg et al., 2015) which simultaneously takes the two RGB images, x_i^{RAFN} and y_i, concatenated channel-wise and translates them to x_i^T and
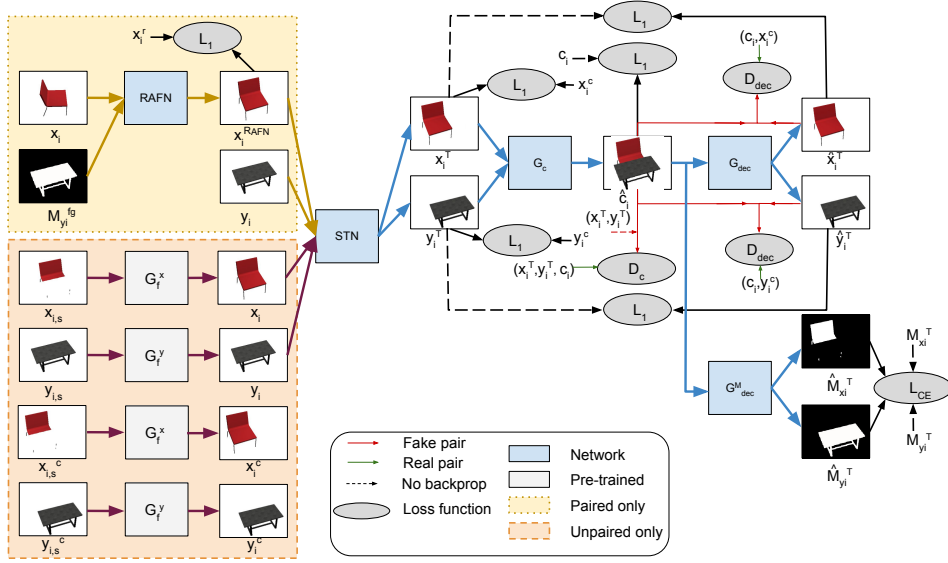
3

Figure 1: Compositional GAN training model both for paired and unpaired training data. The yellow box refers to the RAFN step for synthesizing a new viewpoint of the first object given the foreground mask of the second one, which will be applied only during training with *paired data*. The orange box represents the process of inpainting the input segmentations for training with *unpaired data*. Rest of the model would be similar for the paired and unpaired cases which includes the STN followed by the self-consistent composition-decomposition network.

y_i^T based on their spatial relation encoded in the training composite images. The architecture of this network is illustrated in Appendix B.

The main backbone of our proposed model consists of a self-consistent composition-decomposition network both as conditional generative adversarial networks. The composition network, G_c, takes the two translated input RGB images, x_i^T and $y_i{}^T$, concatenated channel-wise in a batch, with size $N \times 6 \times H \times W$, and generates their corresponding output, $\hat{c}_i$, with size $N \times 3 \times H \times W$ composed of the two input images appropriately. This generated image will be then fed into the decomposition network, G_{dec}, to be decomposed back into its constituent objects, $\hat{x}_i^T$ and $\hat{y}_i^T$ in addition to G_{dec}^M that predicts probability segmentation masks of the composed image, $\hat{M}_{x_i}$ and $\hat{M}_{y_i}$. The two decomposition components G_{dec} and G_{dec}^M share their weights in their encoder network but are different in the decoder. We assume the ground-truth foreground masks of the inputs and the target composite image are available, thus we remove background from all images in the network for simplicity. A GAN loss with gradient penalty (Gulrajani et al., 2017) is applied on top of generated images $\hat{c}_i, \hat{x}_i^T, \hat{y}_i^T$ to make them look realistic in addition to multiple L_1 loss functions penalizing deviation of generated images from their ground-truth. An schematic of our full network, G, is represented in Figure 1 and the loss function is summarized as:

$$\mathcal{L}(G) = \lambda_1[\mathcal{L}_{L_1}(G_c) + \mathcal{L}_{L_1}(G_{\text{dec}})] + \lambda_2\mathcal{L}_{\text{CE}}(G_{\text{dec}}^M) + \lambda_3[\mathcal{L}_{\text{cGAN}}(G_c, D_c) + \mathcal{L}_{\text{cGAN}}(G_{\text{dec}}, D_{\text{dec}})] \quad (2)$$

$$\text{where } \mathcal{L}_{L_1}(G_c) = \mathbb{E}_{(x_i,y_i,c_i)\sim p_{\text{data}}(x_i,y_i,c_i)}[\|c_i - \hat{c}_i\|_1] \quad (3)$$

$$\mathcal{L}_{L_1}(G_{\text{dec}}) = \mathbb{E}_{(x_i,y_i)\sim p_{\text{data}}(x_i,y_i)}[\|(x_i^T, y_i^T) - G_{\text{dec}}(\hat{c}_i)\|_1]$$

$$\mathcal{L}_{\text{cGAN}}(G_c, D_c) = \mathbb{E}_{(x_i,y_i,c_i)\sim p_{\text{data}}(x_i,y_i,c_i)}[\log D_c(x_i^T, y_i^T, c_i)]$$

$$+ \mathbb{E}_{(x_i,y_i)\sim p_{\text{data}}(x_i,y_i)}[1 - \log D_c(x_i^T, y_i^T, \hat{c}_i)]$$

$$\mathcal{L}_{\text{cGAN}}(G_{\text{dec}}, D_{\text{dec}}) = \mathbb{E}_{(x_i,y_i)\sim p_{\text{data}}(x_i,y_i)}[\frac{1}{2}\log D_{\text{dec}}(\hat{c}_i, x_i^c) + \frac{1}{2}\log D_{\text{dec}}(\hat{c}_i, y_i^c)]$$

$$+ \mathbb{E}_{(x_i,y_i)\sim p_{\text{data}}(x_i,y_i)}[\frac{1}{2}(1 - \log D_{\text{dec}}(\hat{c}_i, \hat{x}_i^T)) + \frac{1}{2}(1 - \log D_{\text{dec}}(\hat{c}_i, \hat{y}_i^T))]$$

and $\hat{c}_i = G_c(x_i^T, y_i^T)$, $(x_i^T, y_i^T) = \text{STN}(x_i^{\text{RAFN}}, y_i)$, and x_i^c, y_i^c are the ground-truth transposed full object inputs corresponding to c_i. Moreover, $\mathcal{L}_{L_1}(G_{\text{dec}})$ is the self-consistency constraint penalizing

deviation of decomposed images from their corresponding transposed inputs and $\mathcal{L}_{\text{CE}}$ is the cross entropy loss applied on the predicted probability segmentation masks. We also added the gradient penalty introduced by Gulrajani et al. (2017) to improve convergence of the GAN loss functions. If viewpoint transformation is not needed for the objects' domain, one can replace x_i^{RAFN} with x_i in the above equations. The benefit of adding decomposition networks will be clarified in Section 3.5.

3.4 Unpaired Training Data

Here, we propose a variant of our model discussed in section 3.3 for broader object domains where paired inputs-outputs are not available or hard to collect. In this setting, there is not one-to-one mapping between images in sets X, Y and images in the composite domain, C. However, we still assume that foreground and segmentation masks are available for images in all three sets during training. Therefore, the background is again removed for simplicity.

Given the segmentation masks, M_{x_i}, M_{y_i}, of the joint ground-truth image, c_i, we first crop and resize object segments $x_{i,s}^c = c_i \odot M_{x_i}$ and $y_{i,s}^c = c_i \odot M_{y_i}$ to be at the center of the image, similar to the input center-oriented objects at test time, calling them as $x_{i,s}$ and $y_{i,s}$. For each object, we add a self-supervised inpainting network (Pathak et al., 2016), G_{f}, as a component of our compositional GAN model to generate full objects from the given segments of image c_i, reinforcing the network to learn object occlusions and spatial layouts more accurately. For this purpose, we apply a random mask on each $x_i \in X$ to zero out pixel values in the mask region and train a conditional GAN, G_{f}^x, to fill in the missing regions. To guide this masking process toward a similar task of generating full objects from segmentations, we can use the foreground mask of images in Y for zeroing out images in X. Another cGAN network, G_{f}^y, should be trained similarly to fill in the missing regions of masked images in Y. The loss function for each inpainting network would be as:

$$\mathcal{L}(G_{\text{f}}) = \mathcal{L}_{L_1}(G_{\text{f}}) + \lambda \mathcal{L}_{\text{cGAN}}(G_{\text{f}}, D_{\text{f}}) \tag{4}$$

Therefore, starting from two inpainting networks trained on sets X and Y, we generate a full object from each segment of image c_i both for the center oriented segments, $x_{i,s}$ and $y_{i,s}$, and the original segments, $x_{i,s}^c$ and $y_{i,s}^c$. This step is summarized in the bottom left block of Figure 1. Now, given (x_i^c, y_i^c, c_i), we can train a spatial transformer network similar to the model for paired data discussed in section 3.3 followed by the composition-decomposition networks to generate composite image and its probability segmentation masks. Since we start from segmentations of the joint image rather than an input x_i from a different viewpoint, we skip training the RAFN end-to-end in the compositional network, and use its pre-trained model discussed in section 3.2 at test time.

3.5 Inference Refinement Network

After training the network, we study performance of the model on new images, x, y, from the marginal distributions of sets X and Y along with their foreground masks to generate a natural-looking composite image containing the two objects. However, since generative models cannot generalize very well to a new example, we continue optimizing network parameters given the two input test instances to remove artifacts and generate sharper results (Azadi et al., 2017). Since the ground-truth for the composite image and the target spatial layout of the objects are not available at test time, the self-consistency cycle in our decomposition network provides the only supervision for penalizing deviation from the original objects through an L_1 loss.

We freeze the weights of the relative spatial transformer and appearance flow networks, and only refine the weights of the composition-decomposition layers where the GAN loss will be applied given the real samples from our training set. We again ignore background for simplicity given the foreground masks of the input instances. The ground-truth masks of the transposed full input images, $M_x^{\text{fg}}, M_y^{\text{fg}}$ can be also obtained by applying the pre-trained RAFN and STN on the input masks. We then use the Hadamard product to multiply the predicted masks $\hat{M}_x, \hat{M}_y$ with the objects foreground masks $M_x^{\text{fg}}, M_y^{\text{fg}}$, respectively to eliminate artifacts outside of the target region for each object. One should note that $M_x^{\text{fg}}, M_y^{\text{fg}}$ are the foreground masks of the *full* transposed objects while $\hat{M}_x$ and $\hat{M}_y$ are the predicted *segmentation* masks. Therefore, the loss function for this refinement would be:

$$\mathcal{L}(G) = \lambda(\|\hat{x}^T - x^T\|_1 + \|\hat{M}_x \odot \hat{c} - \hat{M}_x \odot x^T\|_1 \tag{5}$$
$$+ \|\hat{y}^T - y^T\|_1 + \|\hat{M}_y \odot \hat{c} - \hat{M}_y \odot y^T\|_1) + [\mathcal{L}_{\text{cGAN}}(G_c, D_c) + \mathcal{L}_{\text{cGAN}}(G_{\text{dec}}, D_{\text{dec}})]$$
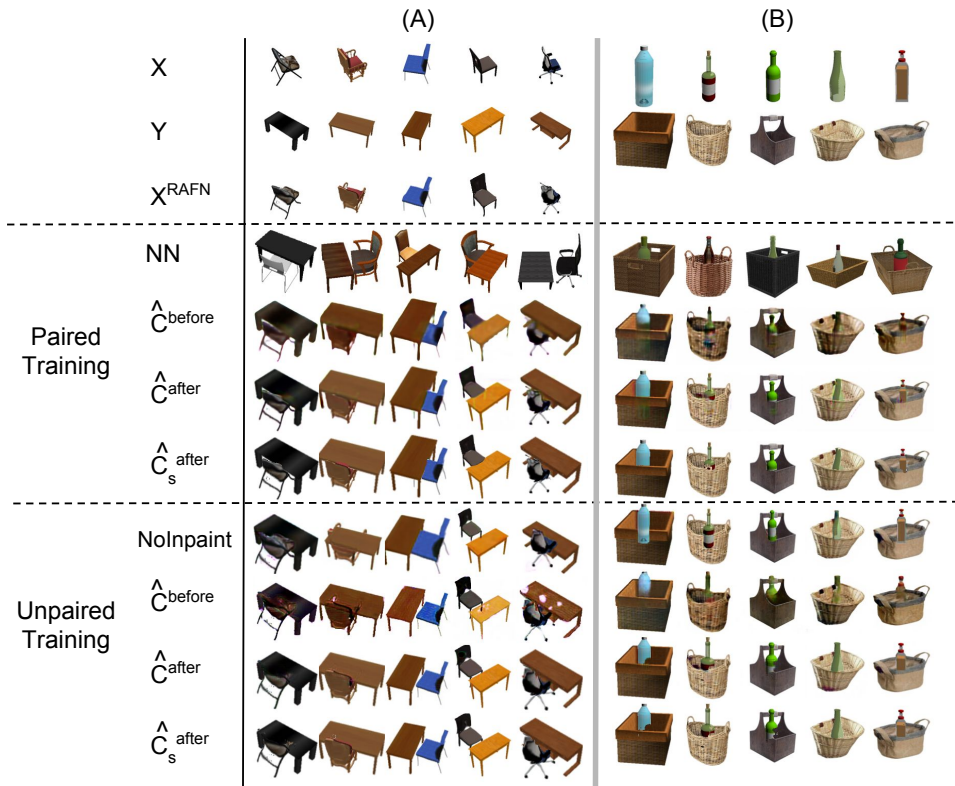
Figure 2: Test results on the chair-table (A) and basket-bottle (B) composition tasks trained with either paired or unpaired data. "NN" stands for the nearest neighbor image in the paired training set, and "NoInpaint" shows the results of the unpaired model without the inpainting network. In both paired and unpaired cases, $\hat{c}^{\text{before}}$ and $\hat{c}^{\text{after}}$ show outputs of the generator before and after the inference refinement network, respectively. Also, $\hat{c}_s^{\text{after}}$ represents summation of masked transposed inputs after the refinement step.

where $\hat{x}^T, \hat{y}^T$ are the generated decomposed images, and x^T and y^T are the transposed inputs.

In the experiments, we will present: (1) images generated directly from the composition network, $\hat{c}$, before and after this refinement step, (2) images generated directly based on the predicted segmentation masks as $\hat{c}_s = \hat{M}_x \odot x^T + \hat{M}_y \odot y^T$.

4 EXPERIMENTS

In this section, we study the performance of our compositional GAN model for both the paired and unpaired scenarios through multiple qualitative and quantitative experiments in different domains. First, we use the Shapenet dataset (Chang et al., 2015) as our main source of input objects and study two composition tasks: (1) a chair next to a table, (2) a bottle in a basket. Second, we show our model performing equally well when one object is fixed and the other one is relatively scaled and linearly transformed to generate a composed image. We present our results on the CelebA dataset (Liu et al., 2015) composed with sunglasses downloaded from the web. In all our experiments, the values for the training hyper-parameters are set to $\lambda_1 = 100$, $\lambda_2 = 50$, $\lambda_3 = 1$, and the inference $\lambda = 100$.

4.1 COMPOSING A CHAIR WITH A TABLE

Composition of a chair and a table is a challenging problem since viewpoints of the two objects should be similar and one object should be partially occluded *and/or* partially occlude the other one depending on their viewpoint. This problem cannot be resolved by considering each object as a separate individual layer of an image. By feeding in the two objects simultaneously to our proposed network, the model learns to relatively transform each object and composes them reasonably.

We manually made a collection of 1K composite images from Shapenet chairs and tables which can be used for both the paired and unpaired training models. In the paired scenario, we use the pairing information between each composite image and its constituent full chair and table besides their foreground masks. On the other hand, to show the performance of our model on the unpaired examples as well, we ignore the individual chairs and tables used in each composite image, and use a different subset of Shapenet chairs and tables as real examples of each individual set. We made sure that these two subsets do not overlap with the chairs and tables in composite images to avoid the occurrence of implicit pairing in our experiments.

Chairs and tables in the input-output sets can pose in a random azimuth angle in the range $[-180°, 180°]$ at steps of $10°$. As discussed in section 3.2, feeding in the foreground mask of an arbitrary table with a random azimuth angle in addition to the input chair to our trained relative appearance flow network synthesizes the chair in the viewpoint consistent with the table. Synthesized test chairs as X^{RAFN} are represented in the third row of Figure 2-A.

In addition, to study our network components, we visualize model outputs at different steps in Figure 2-A. To evaluate our network with paired training data on a new input chair and table represented as X and Y, respectively, we find its nearest neighbor composite example in the training set in terms of its constituent chair and table features extracted from a pre-trained VGG19 network (Simonyan & Zisserman, 2014). As shown in the fourth row of Figure 2-A, nearest neighbors are different enough to be certain that network is not memorizing its training data. We also illustrate output of the network before and after the inference refinement step discussed in section 3.5 in terms of the generator's prediction, $\hat{c}$, as well as the direct summation of masked transposed inputs, $\hat{c}_s$, for both paired and unpaired training models. The refinement step sharpens the synthesized image and removes artifacts generated by the model. Our results from the model trained on unpaired data shown in the figure is comparable with those from paired data. Moreover, we depict the performance of the model without our inpainting network in the eighth row, where occlusions are not correct in multiple examples. Figure 2-A emphasizes that our model has successfully resolved the challenges involved in this composition task, where in some regions such as the chair handle, table is occluding the chair while in some other regions such as table legs, chair is occluding the table. More exemplar images as well as some of the failure cases for both paired and unpaired scenarios are presented in Appendix C.1.

We have also conducted an Amazon Mechanical Turk evaluation (Zhang et al., 2016) to compare the performance of our algorithm in different scenarios including training with and without paired data and before and after the final inference refinement network. From a set of 90 test images of chairs and tables, we have asked 60 evaluators to select their preferred composite image generated by the model trained on paired data versus images generated by the model trained on unpaired data, both after the inference refinement step. As a result, 57% of the composite images generated by our model trained on paired inputs-outputs were preferred to the ones generated through the unpaired scenario. It shows that even without paired examples during training, our proposed model performs reasonably well. We have repeated the same study to compare the quality of images generated before and after the inference refinement step, where the latter was preferred 71.3% of the time to the non-refined images revealing the benefit of the the last refinement module in generating higher-quality images.

4.2 COMPOSING BOTTLE WITH A BASKET

In this experiment, we address the compositional task of putting a bottle in a basket. Similar to the chair-table problem, we manually composed Shapenet bottles with baskets to prepare a training set of 100 paired examples. We trained the model both with and without the paired data, similarly to section 4.1, and represent outputs of the network before and after the inference refinement in Figure 2-B. In addition, nearest neighbor examples in the paired training set are shown for each new input instance (fourth row) as well as the model's predictions in the unpaired case without the inpainting network (eighth column). As clear from the results, our inpainting network plays a critical role in the success of our unpaired model specially for handling occlusions. This problem statement is similarly interesting since the model should identify which pixels to be occluded. For instance in the first column of Figure 2-B, the region inside the basket is occluded by the blue bottle while the region outside is occluding the latter. More examples are shown in Appendix C.2.

Similarly, we evaluate the performance of our model on this task through an Amazon Mechanical Turk study with 60 evaluators and a set of 45 test images. In summary, outputs from our paired training were preferred to the unpaired case 57% of the time and the inference refinement step was
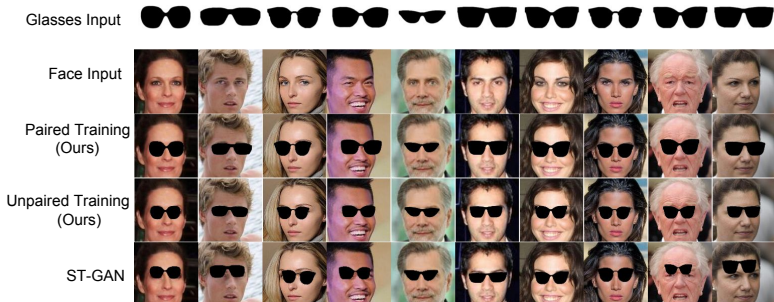
Figure 3: Test examples for the face-sunglasses composition task. Top two rows: input sunglasses and face images; 3rd and 4th rows: the output of our compositional GAN for the paired and unpaired models, respectively; Last row: images generated by the ST-GAN (Lin et al., 2018) model.

observed to be useful in 64% of examples. These results confirm the benefit of the refinement module and the comparable performance of training in the unpaired scenario with the paired training case.

Ablation Studies and Other Baselines: In Appendix C.3, we repeat the experiments with each component of the model removed at a time to study their effect on the final composite image. In addition, in Appendix C.4, we show the poor performance of two baseline models (CycleGAN (Zhu et al., 2017) and Pix2Pix (Isola et al., 2017)) in a challenging composition task.

4.3 COMPOSING FACES WITH GLASSES

In this section, we compose a pair of sunglasses with a face image, similar to (Lin et al., 2018), where the latter should be fixed while sunglasses should be rescaled and transformed relatively. We used the CelebA dataset (Liu et al., 2015), followed its training/test splits and cropped images to 128×128 pixels. We hand-crafted 180 composite images of celebrity faces from the training split with sunglasses downloaded from the web to prepare a paired training set. However, we could still use our manual composite set for the unpaired case with access to the segmentation masks separating sunglasses from faces. In the unpaired scenario, we used 6K images from the training split while not overlapping with our composite images to be used as the set of individual faces during training. In this case, since pair of glasses is always occluding the face, we report results based on summation of the masked transposed inputs, $\hat{c}_s$ for both the paired training data and the unpaired one. We also compare our results with the ST-GAN model (Lin et al., 2018) which assumes images of faces as a fixed background and warps the glasses in the geometric warp parameter space. Our results both in paired and unpaired cases, shown in Figure 3, look more realistic in terms of the scale, rotation angle, and location of the sunglasses with the cost of only 180 paired training images or 180 unpaired images with segmentation masks. More example images are illustrated in Appendix C.5.

To confirm this observation, we have studied the results by asking 60 evaluators to score our model predictions versus ST-GAN on a set of 75 test images. According to this study where we compare our model trained on paired data with ST-GAN, 84% of the users evaluated favorably our network predictions. Moreover, when comparing ST-GAN with our unpaired model, 73% of the evaluators selected the latter. These results confirm the ability of our model in generalizing to the new test examples and support our claim that both our paired and unpaired models significantly outperform the recent ST-GAN model in composing a face with a pair of sunglasses.

5 CONCLUSION AND FUTURE WORK

In this paper, we proposed a novel Compositional GAN model addressing the problem of object composition in conditional image generation. Our model captures the relative linear and viewpoint transformations needed to be applied on each input object (in addition to their spatial layout and occlusions) to generate a realistic joint image. To the best of our knowledge, we are among the first to solve the compositionality problem without having any explicit prior information about object's layout. We evaluated our compositional GAN through multiple qualitative experiments and user evaluations for two cases of paired versus unpaired training data. In the future, we plan to extend this work toward generating images composed of multiple (more than two) and/or non-rigid objects.

REFERENCES

Antreas Antoniou, Amos Storkey, and Harrison Edwards. Data augmentation generative adversarial networks. *arXiv preprint arXiv:1711.04340*, 2017.

Samaneh Azadi, Matthew Fisher, Vladimir Kim, Zhaowen Wang, Eli Shechtman, and Trevor Darrell. Multi-content gan for few-shot font style transfer. *arXiv preprint arXiv:1712.00516*, 2017.

Angel X. Chang, Thomas Funkhouser, Leonidas Guibas, Pat Hanrahan, Qixing Huang, Zimo Li, Silvio Savarese, Manolis Savva, Shuran Song, Hao Su, Jianxiong Xiao, Li Yi, and Fisher Yu. ShapeNet: An Information-Rich 3D Model Repository. Technical Report arXiv:1512.03012 [cs.GR], Stanford University — Princeton University — Toyota Technological Institute at Chicago, 2015.

Xi Chen, Yan Duan, Rein Houthooft, John Schulman, Ilya Sutskever, and Pieter Abbeel. Infogan: interpretable representation learning by information maximizing generative adversarial nets. In *NIPS*, 2016.

Emily L Denton, Soumith Chintala, Arthur Szlam, and Rob Fergus. Deep generative image models using a laplacian pyramid of adversarial networks. In *NIPS*, 2015.

Ian Goodfellow. NIPS 2016 tutorial: Generative adversarial networks. *arXiv preprint arXiv:1701.00160*, 2016.

Ishaan Gulrajani, Faruk Ahmed, Martin Arjovsky, Vincent Dumoulin, and Aaron C Courville. Improved training of wasserstein gans. In *NIPS*, 2017.

Sergey Ioffe and Christian Szegedy. Batch normalization: Accelerating deep network training by reducing internal covariate shift. *arXiv preprint arXiv:1502.03167*, 2015.

Phillip Isola, Jun-Yan Zhu, Tinghui Zhou, and Alexei A Efros. Image-to-image translation with conditional adversarial networks. In *CVPR*, 2017.

Max Jaderberg, Karen Simonyan, Andrew Zisserman, et al. Spatial transformer networks. In *NIPS*, 2015.

Justin Johnson, Agrim Gupta, and Li Fei-Fei. Image generation from scene graphs. *CVPR*, 2018.

Tero Karras, Timo Aila, Samuli Laine, and Jaakko Lehtinen. Progressive growing of gans for improved quality, stability, and variation. *arXiv preprint arXiv:1710.10196*, 2017.

Wei-Sheng Lai, Jia-Bin Huang, Narendra Ahuja, and Ming-Hsuan Yang. Deep laplacian pyramid networks for fast and accurate super-resolution. In *CVPR*, 2017.

Christian Ledig, Lucas Theis, Ferenc Huszár, Jose Caballero, Andrew Cunningham, Alejandro Acosta, Andrew Aitken, Alykhan Tejani, Johannes Totz, Zehan Wang, et al. Photo-realistic single image super-resolution using a generative adversarial network. *arXiv preprint*, 2016.

Chen-Hsuan Lin, Ersin Yumer, Oliver Wang, Eli Shechtman, and Simon Lucey. St-gan: Spatial transformer generative adversarial networks for image compositing. *arXiv preprint arXiv:1803.01837*, 2018.

Ming-Yu Liu and Oncel Tuzel. Coupled generative adversarial networks. In *Advances in neural information processing systems*, pp. 469–477, 2016.

Ming-Yu Liu, Thomas Breuel, and Jan Kautz. Unsupervised image-to-image translation networks. In *NIPS*, 2017.

Ziwei Liu, Ping Luo, Xiaogang Wang, and Xiaoou Tang. Deep learning face attributes in the wild. In *ICCV*, 2015.

Mehdi Mirza and Simon Osindero. Conditional generative adversarial nets. *arXiv preprint arXiv:1411.1784*, 2014.

Augustus Odena, Christopher Olah, and Jonathon Shlens. Conditional image synthesis with auxiliary classifier gans. *arXiv preprint arXiv:1610.09585*, 2016.

Deepak Pathak, Philipp Krähenbühl, Jeff Donahue, Trevor Darrell, and Alexei Efros. Context encoders: Feature learning by inpainting. In *CVPR*, 2016.

Alec Radford, Luke Metz, and Soumith Chintala. Unsupervised representation learning with deep convolutional generative adversarial networks. In *ICLR*, 2016.

Scott Reed, Zeynep Akata, Xinchen Yan, Lajanugen Logeswaran, Bernt Schiele, and Honglak Lee. Generative adversarial text-to-image synthesis. In *ICML*, 2016a.

Scott E Reed, Zeynep Akata, Santosh Mohan, Samuel Tenka, Bernt Schiele, and Honglak Lee. Learning what and where to draw. In *NIPS*, 2016b.

Tim Salimans, Ian Goodfellow, Wojciech Zaremba, Vicki Cheung, Alec Radford, and Xi Chen. Improved techniques for training gans. In *NIPS*, 2016.

Karen Simonyan and Andrew Zisserman. Very deep convolutional networks for large-scale image recognition. *arXiv preprint arXiv:1409.1556*, 2014.

Kalyan Sunkavalli, Micah K Johnson, Wojciech Matusik, and Hanspeter Pfister. Multi-scale image harmonization. In *ACM Transactions on Graphics (TOG)*. ACM, 2010.

Yi-Hsuan Tsai, Xiaohui Shen, Zhe Lin, Kalyan Sunkavalli, Xin Lu, and Ming-Hsuan Yang. Deep image harmonization. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2017.

Ting-Chun Wang, Ming-Yu Liu, Jun-Yan Zhu, Andrew Tao, Jan Kautz, and Bryan Catanzaro. High-resolution image synthesis and semantic manipulation with conditional gans. *arXiv preprint arXiv:1711.11585*, 2017.

Su Xue, Aseem Agarwala, Julie Dorsey, and Holly Rushmeier. Understanding and improving the realism of image composites. *ACM Transactions on Graphics (TOG)*, 2012.

Chao Yang, Xin Lu, Zhe Lin, Eli Shechtman, Oliver Wang, and Hao Li. High-resolution image inpainting using multi-scale neural patch synthesis. In *CVPR*, 2017a.

Jianwei Yang, Anitha Kannan, Dhruv Batra, and Devi Parikh. Lr-gan: Layered recursive generative adversarial networks for image generation. *arXiv preprint arXiv:1703.01560*, 2017b.

Han Zhang, Tao Xu, Hongsheng Li, Shaoting Zhang, Xiaolei Huang, Xiaogang Wang, and Dimitris Metaxas. Stackgan: Text to photo-realistic image synthesis with stacked generative adversarial networks. In *ICCV*, 2017.

Richard Zhang, Phillip Isola, and Alexei A Efros. Colorful image colorization. In *ECCV*, 2016.

Tinghui Zhou, Shubham Tulsiani, Weilun Sun, Jitendra Malik, and Alexei A Efros. View synthesis by appearance flow. In *ECCV*, 2016.

Jun-Yan Zhu, Philipp Krahenbuhl, Eli Shechtman, and Alexei A Efros. Learning a discriminative model for the perception of realism in composite images. In *ICCV*, 2015.

Jun-Yan Zhu, Taesung Park, Phillip Isola, and Alexei A Efros. Unpaired image-to-image translation using cycle-consistent adversarial networks. In *ICCV*, 2017.

A   RELATIVE APPEARANCE FLOW NETWORK (RAFN)

Architecture of our relative appearance flow network is illustrated in Figure 4 which is composed of an encoder-decoder set of convolutional layers for predicting the appearance flow vectors, which after a bilinear sampling generates the synthesized view. The second decoder (last row of layers in Figure 4) is for generating foreground mask of the synthesized image following a shared encoder network (Zhou et al., 2016). All convolutional layers are followed by batch normalization (Ioffe & Szegedy, 2015) and a ReLU activation layer except for the last convolutional layer in each decoder. In the flow decoder, output is fed into a Tanh layer while in the mask prediction decoder, the last convolutional layer is followed by a Sigmoid layer to be in the range $[0, 1]$.
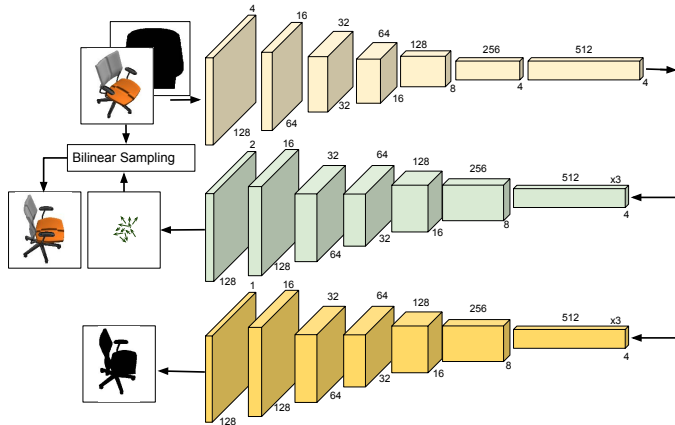

Figure 4: Relative Appearance Flow Network: Input is an image of a chair with 3 RGB channels concatenated channel-wise with the table foreground mask. Output is the appearance flow for synthesizing a new viewpoint of the chair. All layers are convolutional.

B   RELATIVE SPATIAL TRANSFORMER NETWORK

Diagram of our relative spatial transformer network is represented in Figure 5. The two input images (e.g., chair and table) are concatenated channel-wise and fed into the localization network to generate two set of parameters, θ_1, θ_2 for the affine transformations. This single network is simultaneously trained on the two images learning their relative transformations required to getting close to the given target images. In this figure, orange feature maps are the output of a conv2d layer (represented along with their corresponding number of channels and dimensions) and yellow maps are the output of max-pool2d followed by ReLU. The blue layers also represent fully connected layers.
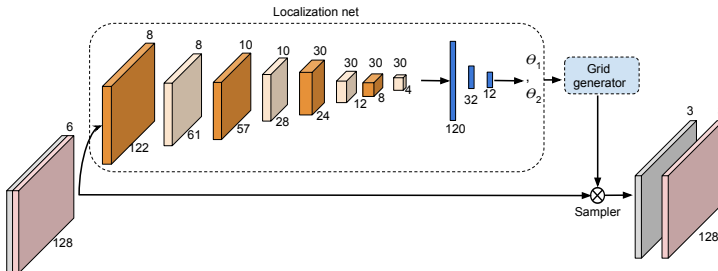

Figure 5: Relative Spatial Transformer Network: Input is an image of a chair with 3 RGB channels concatenated channel-wise with the table RGB image. Output is two transformed images each with 3 RGB channels.

C   ADDITIONAL RESULTS:

In this section, we provide a few extra test examples in different domains:

C.1 Composing a Chair with a Table

On the challenging problem of composing a chair with a table, we illustrate more test examples in Figure 6 and a few failure test examples in Figure 7 for both paired and unpaired training models. Here, viewpoint and linear transformations in addition to occluding object regions should be performed properly to generate a realistic image.
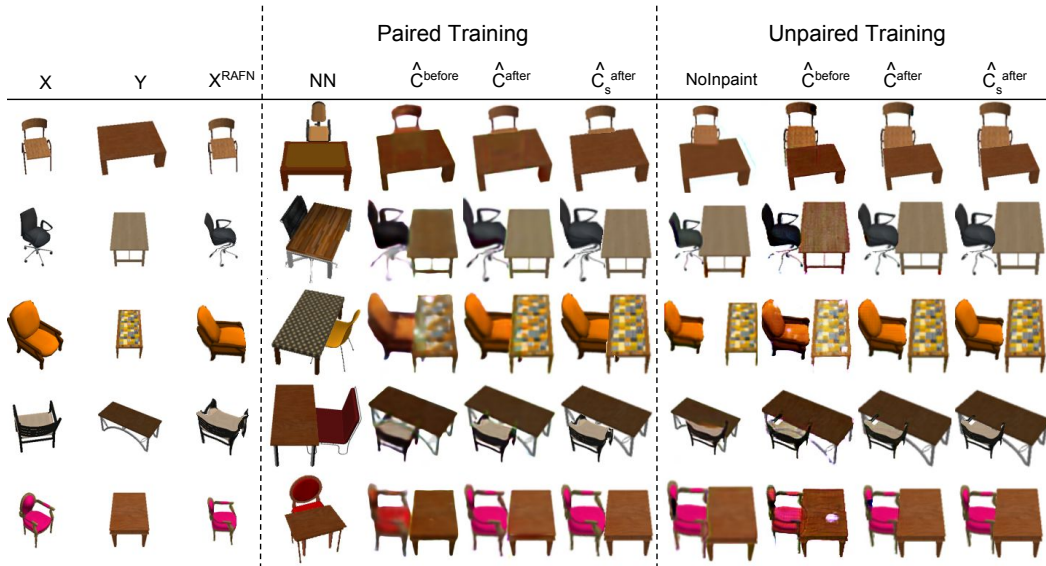

Figure 6: Test results on the chair-table composition task trained with either paired or unpaired data. "NN" stands for the nearest neighbor image in the paired training set, and "NoInpaint" shows the results of the unpaired model without the inpainting network. In both paired and unpaired cases, $\hat{c}^{\text{before}}$ and $\hat{c}^{\text{after}}$ show outputs of the generator before and after the inference refinement network, respectively. Also, $\hat{c}_s^{\text{after}}$ represents summation of masked transposed inputs after the refinement step.

C.2 Composing a Bottle with a Basket

In the bottle-basket composition, the main challenging problem is the relative scale of the objects besides their partial occlusions. In Figure 8, we visualize more test examples and study the performance of our model before and after the inference refinement step for both paired and unpaired scenarios. The third column of this figure represents the nearest neighbor training example found for each new input pair, (X, Y), through their features extracted from the last layer of the pre-trained VGG19 network (Simonyan & Zisserman, 2014). Moreover, the seventh column shows outputs of the trained unpaired network without including the inpainting component during training revealing the necessity of the inpainting network while training with unpaired data.

C.3 Ablation Study:

We repeat the experiments on composing a bottle with a basket, with each component of the model removed at a time, to study their effect on the final composite image. Qualitative results are illustrated in Figure 9. First and second columns show bottle and basket images which are concatenated channel-wise as the input to the network. Following columns are:
- *3rd column*: no reconstruction loss on the composite image results in wrong color and faulty occlusion,
- *4th column*: no cross-entropy mask loss in training results in faded bottles,
- *5th column*: no GAN loss in training and inference generates outputs with a different color and lower quality than the input image,
- *6th column*: no decomposition generator (G_{dec}) and self-consistent cycle results in partially missed bottles,
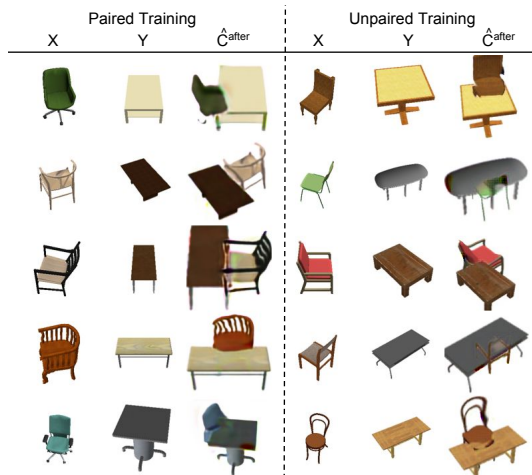- *7, 8th columns* represent full model in paired and unpaired scenarios.

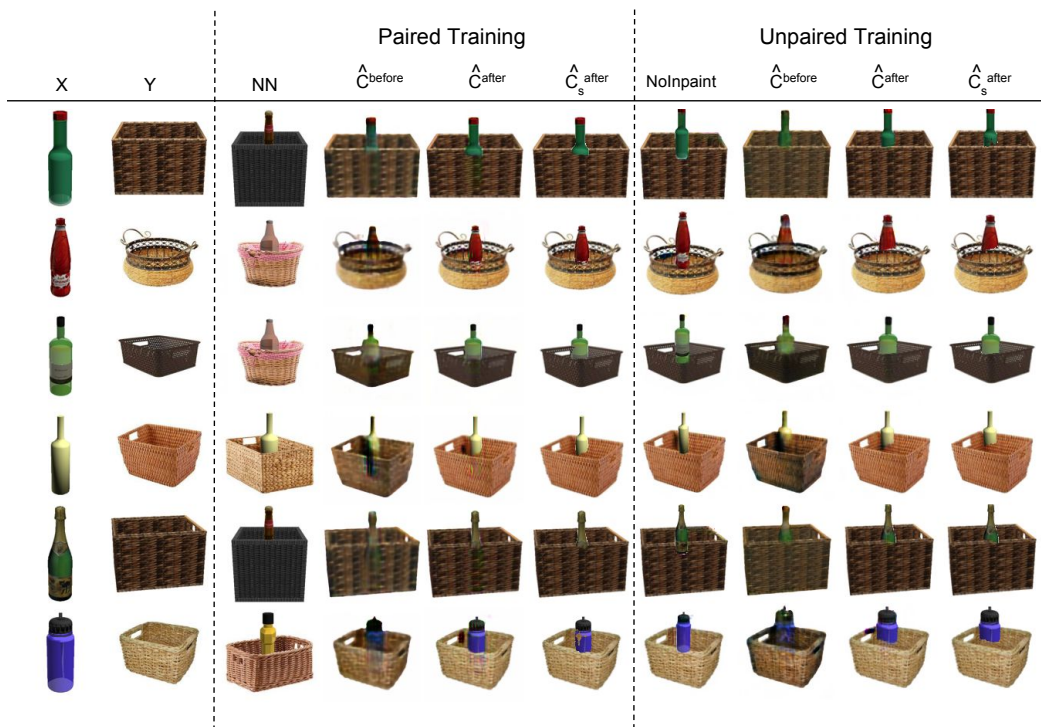Figure 7: Failure test cases for both the paired and unpaired models on the chair-table composition task.

Figure 8: More test results on the basket-bottle composition task trained with either paired or unpaired data. "NN" stands for the nearest neighbor image in the paired training set, and "NoInpaint" shows the results of the unpaired model without the inpainting network. In both paired and unpaired cases, $\hat{c}^{\text{before}}$ and $\hat{c}^{\text{after}}$ show outputs of the generator before and after the inference refinement network, respectively. Also, $\hat{c}_s^{\text{after}}$ represents summation of masked transposed inputs after the refinement step.

C.4 OTHER BASELINES:

The purpose of our model is to capture object interactions in the 3D space projected onto a 2D image by handling their spatial layout, relative scaling, occlusion, and viewpoint for generating a realistic image. These factors distinguish our model from CycleGAN (Zhu et al., 2017) and Pix2Pix (Isola et al., 2017) models whose goal is only changing the appearance of the given image. In this section, we compare to these two models. To be able to compare, we use the mean scaling and translating parameters of our training set to place each input bottle and basket together and have an input with 3
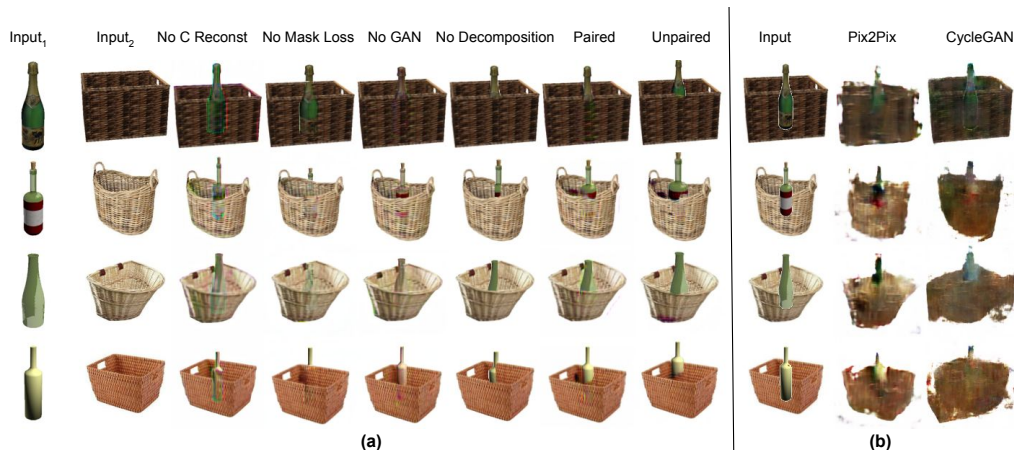
13

Figure 9: (a) Ablation Study: Output of our model without the component specified on top of each column. Input is the channel-wise concatenation of the bottle and basket shown in the first two columns, (b) Baselines: As the input (9th column), each bottle is added to the basket after being scaled and translated with constant parameters. Pix2Pix and CycleGAN outputs are shown on the right.

RGB channels (*9th column* in Figure 9). We then train a ResNet generator on our paired training data with an adversarial loss added with a L_1 regularizer. Since the structure of the input image might be different from its corresponding ground-truth image (due to different object scalings and layouts), ResNet model works better than a U-Net but still generating unrealistic images (*10th column* in Figure 9). We follow the same approach for the unpaired data through the CycleGAN model (*11th column* in Figure 9). As apparent from the qualitative results, it is not easy for either Pix2Pix or CycleGAN networks to learn the transformation between samples from the input distribution and that of the occluded outputs.

C.5 COMPOSING FACES WITH GLASSES

Adding a pair of sunglasses to an arbitrary face image requires a proper linear transformation of the sunglasses to align well with the face. We illustrate test examples of this composition problem in Figure 10 including results of both the paired and unpaired training scenarios in the third and fourth columns. In addition, the last column of each composition example case represents the outputs of the ST-GAN model (Lin et al., 2018).

Figure 10: Test examples for the face-sunglasses composition task. First two columns show input sunglasses and face images, 3rd and 4th columns show the output of our compositional GAN for the paired and unpaired models, respectively. Last column shows images generated by the ST-GAN (Lin et al., 2018) model.