

# ST-GAN: Spatial Transformer Generative Adversarial Networks for Image Compositing

Chen-Hsuan Lin<sup>1\*</sup> Ersin Yumer<sup>2,3\*</sup> Oliver Wang<sup>2</sup> Eli Shechtman<sup>2</sup> Simon Lucey<sup>1,3</sup>

<sup>1</sup>Carnegie Mellon University <sup>2</sup>Adobe Research <sup>3</sup>Argo AI

chlin@cmu.edu meyumer@gmail.com {owang, elishe}@adobe.com slucey@cs.cmu.edu

## Abstract

We address the problem of finding realistic geometric corrections to a foreground object such that it appears natural when composited into a background image. To achieve this, we propose a novel Generative Adversarial Network (GAN) architecture that utilizes Spatial Transformer Networks (STNs) as the generator, which we call Spatial Transformer GANs (ST-GANs). ST-GANs seek image realism by operating in the geometric warp parameter space. In particular, we exploit an iterative STN warping scheme and propose a sequential training strategy that achieves better results compared to naive training of a single generator. One of the key advantages of ST-GAN is its applicability to high-resolution images indirectly since the predicted warp parameters are transferable between reference frames. We demonstrate our approach in two applications: (1) visualizing how indoor furniture (e.g. from product images) might be perceived in a room, (2) hallucinating how accessories like glasses would look when matched with real portraits.

## 1. Introduction

Generative image modeling has progressed remarkably with the advent of convolutional neural networks (CNNs). Most approaches constrain the possible appearance variations within an image by learning a low-dimensional embedding as an encoding of the natural image subspace and making predictions from this at the pixel level. We refer to these approaches here as *direct image generation*. Generative Adversarial Networks (GANs) [7], in particular, have demonstrated to be an especially powerful tool for realistic image generation. They consist of a generator network ( $\mathcal{G}$ ) that produces images from codes, and a discriminator network ( $\mathcal{D}$ ) that distinguishes real images from fake ones. These two networks play a minimax game that results in  $\mathcal{G}$  generating realistic looking images and  $\mathcal{D}$  being unable to distinguish between the two when equilibrium is reached.

\*Work done during CHL's internship at Adobe Research.

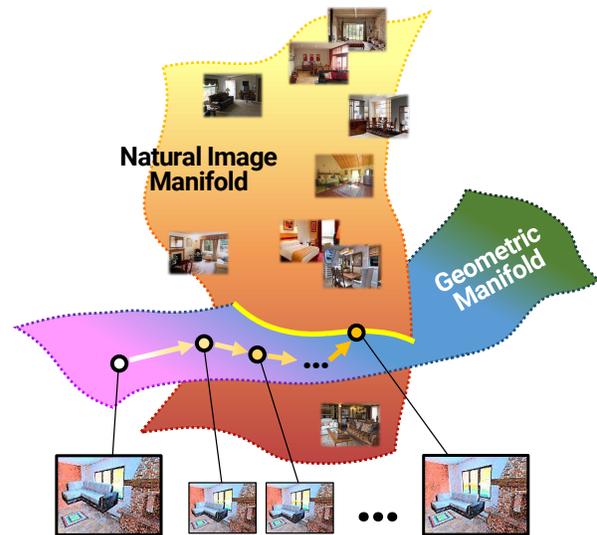


Figure 1: Composite images easily fall outside the natural image manifold due to appearance and geometric discrepancies. We seek to learn *geometric corrections* that sequentially warp composite images towards the intersection of the geometric and natural image manifolds.

Direct image generation, however, has its limitations. As the space of all images is very high-dimensional and image generation methods are limited by finite network capacity, direct image generation methods currently work well only on restricted domains (e.g. faces) or at low resolutions.

In this work, we leverage Spatial Transformer Networks (STNs) [11], a special type of CNNs capable of performing geometric transformations on images, to provide a simpler way to generate realistic looking images – by restricting the space of possible outputs to a well-defined *low-dimensional geometric transformation* of real images. We propose Spatial Transformer Generative Adversarial Networks (ST-GANs), which learn Spatial Transformer generators within a GAN framework. The adversarial loss enables us to learn geometric corrections resulting in a warped image that lies at the *intersection* of the natural image man-

ifold and the geometric manifold – the space of geometric manipulations specific to the target image (Fig. 1). To achieve this, we advocate a sequential adversarial training strategy to learn iterative spatial transformations that serve to break large transformations down into smaller ones.

We evaluate ST-GANs in the context image compositing, where a source foreground image and its mask are warped by the Spatial Transformer generator  $\mathcal{G}$ , and the resulting composite is assessed by the discriminator  $\mathcal{D}$ . In this setup,  $\mathcal{D}$  tries to distinguish warped composites from real images, while  $\mathcal{G}$  tries to fool  $\mathcal{D}$  by generating as realistic looking as possible composites. To the best of our knowledge, we are the first to address the problem of realistic image generation through geometric transformations in a GAN framework. We demonstrate this method on the application of compositing furniture into indoor scenes, which gives a preview of, for example, how purchased items would look in a house. To evaluate in this domain, we created a synthetic dataset of indoor scene images as the background with masked objects as the foreground. We also demonstrate ST-GANs in a fully unpaired setting for the task of compositing glasses on portrait images. A large-scale user study shows that our approach improves the realism of image composites.

Our main contributions are as follows:

- We integrate the STN and GAN frameworks and introduce ST-GAN, a novel GAN framework for finding realistic-looking geometric warps.
- We design a multi-stage architecture and training strategy that improves warping convergence of ST-GANs.
- We demonstrate compelling results in image compositing tasks in both paired and unpaired settings as well as its applicability to high-resolution images.

## 2. Related Work

**Image compositing** refers to the process of overlaying a masked foreground image on top of a background image. One of the main challenges of image compositing is that the foreground object usually comes from a different scene than the background, and therefore it is not likely to match the background scene in a number of ways that negatively effects the realism of the composite. These can be both appearance differences (due to lighting, white balance, and shading differences) and geometric differences (due to changes in camera viewpoint and object positioning).

Existing photo-editing software features various image appearance adjustment operations for that allows users to create realistic composites. Prior work has attempted to automate appearance corrections (*e.g.* contrast, saturation) through Poisson blending [25] or more recent deep learning approaches [41, 29]. In this work, we focus on the second challenge: correcting for *geometric* inconsistencies between source and target images.

**Spatial Transformer Networks (STNs)** [11] are one way to incorporate learnable image warping within a deep learning framework. A Spatial Transformer module consists of a subnetwork predicting a set of warp parameters followed by a (differentiable) warp function.

STNs have been shown effective in resolving geometric variations for discriminative tasks as well as a wide range of extended applications such as robust filter learning [4, 13], image/view synthesis [40, 6, 23, 36], and 3D representation learning [14, 34, 39]. More recently, Inverse Compositional STNs (IC-STNs) [16] advocated an iterative alignment framework. In this work, we borrow the concept of iterative warping but do not enforce recurrence in the geometric prediction network; instead, we add different generators at each warping step with a sequential training scheme.

**Generative Adversarial Networks (GANs)** [7] are a class of generative models that are learned by playing a minimax optimization game between a generator network  $\mathcal{G}$  and a discriminator network  $\mathcal{D}$ . Through this adversarial process, GANs are shown to be capable of learning a generative distribution that matches the empirical distribution of a given data collection. One advantage of GANs is that the loss function is essentially learned by the discriminator network, which allows for training in cases where ground truth data with strong supervision is not available.

GANs are utilized for data generation in various domains, including images [26], videos [30], and 3D voxelized data [32]. For images in particular, it has been shown to generate compelling results in a vast variety of conditional image generation problems such as super-resolution [15], inpainting [24], image-to-image translation [10, 43, 18], and image editing/manipulation [42].

Recently, STNs were also sought to be adversarially trained for object detection [31], where adversarial examples with feature deformations are generated to robustify object detectors. LR-GAN [35] approached direct image generation problems with additional STNs onto the (directly) generated images to factorize shape variations. We explore the context of STNs with GANs in the space of *conditional* image generation from given inputs, which is a more direct integration of the two frameworks.

## 3. Approach

Our goal is *realistic geometric correction* for image compositing given a background image  $\mathcal{I}_{BG}$  and foreground object  $\mathcal{I}_{FG}$  with a corresponding mask  $\mathcal{M}_{FG}$ . We aim to correct the camera perspective, position and orientation of the foreground object such that the resulting composite looks natural. The compositing process can be expressed as:

$$\begin{aligned} \mathcal{I}_{comp} &= \mathcal{I}_{FG} \odot \mathcal{M}_{FG} + \mathcal{I}_{BG} \odot (1 - \mathcal{M}_{FG}) \\ &= \mathcal{I}_{FG} \oplus \mathcal{I}_{BG} . \end{aligned} \quad (1)$$

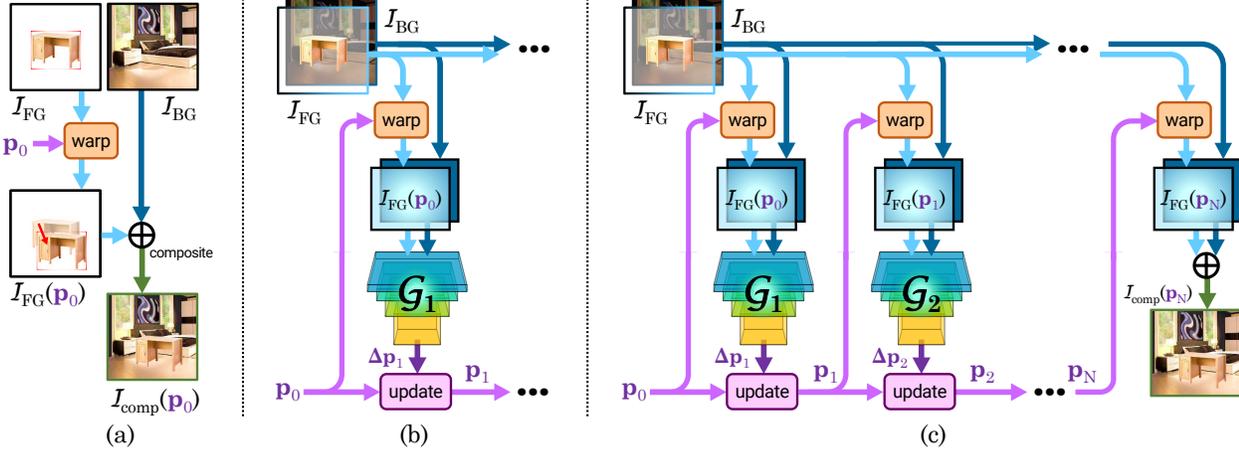


Figure 2: Background. (a) Given an initial composite transformation  $\mathbf{p}_0$ , the foreground image and mask is composited onto the background image using (1). (b) Using **Spatial Transformer Networks** (STNs), a geometric prediction network  $\mathcal{G}_1$  predicts an update  $\Delta \mathbf{p}_1$  conditioned on the foreground and background images, resulting in the new parameters  $\mathbf{p}_1$ . The update is performed with warp composition (3). (c) Our final form is an *iterative* STN to predict a series of accumulative warp updates on the foreground such that the resulting composite image falls closer to the natural image manifold.

For simplicity, we further introduce the notation  $\oplus$  to represent compositing (with  $\mathcal{M}_{FG}$  implied within  $\mathcal{I}_{FG}$ ). Given the composite parameters  $\mathbf{p}_0$  (defining an initial warp state) of  $\mathcal{I}_{FG}$ , we can rewrite (1) as

$$\mathcal{I}_{comp}(\mathbf{p}_0) = \mathcal{I}_{FG}(\mathbf{p}_0) \oplus \mathcal{I}_{BG}, \quad (2)$$

where images are written as functions of the warp parameters. This operator is shown in Fig. 2(a).

In this work, we restrict our geometric warp function to homography transformations, which can represent approximate 3D geometric rectifications for objects that are mostly planar or with small perturbations. As a result, we are making an assumption that the perspective of the foreground object is *close* to the correct perspective; this is often the case when people are choosing similar, but not identical, images from which to composite the foreground object.

The core module of our network design is an STN (Fig. 2(b)), where the geometric prediction network  $\mathcal{G}$  predicts a correcting update  $\Delta \mathbf{p}_1$ . We condition  $\mathcal{G}$  on both the background and foreground images, since knowing how an object should be transformed to fit a background scene requires knowledge of the complex interaction between the two. This includes geometry of the object and the background scene, the relative camera position, and semantic understanding of realistic object layouts (*e.g.* having a window in the middle of the room would not make sense).

### 3.1. Iterative Geometric Corrections

Predicting large displacement warp parameters from image pixels is extremely challenging, so most prior work on image alignment predict local geometric transformations in

an iterative fashion [9, 20, 2, 33, 17]. Similarly, we propose to use iterative STNs to predict a series of warp updates, shown in Fig. 2(c). At the  $i$ th iteration, given the input image  $\mathcal{I}$  and the previous warp state  $\mathbf{p}_{i-1}$ , the correcting warp update  $\Delta \mathbf{p}_i$  and the new warp state  $\mathbf{p}_i$  can be written as

$$\begin{aligned} \Delta \mathbf{p}_i &= \mathcal{G}_i(\mathcal{I}_{FG}(\mathbf{p}_{i-1}), \mathcal{I}_{BG}) \\ \mathbf{p}_i &= \mathbf{p}_{i-1} \circ \Delta \mathbf{p}_i, \end{aligned} \quad (3)$$

where  $\mathcal{G}_i(\cdot)$  is the geometric prediction network and  $\circ$  denotes composition of warp parameters. This family of iterative STNs preserves the original images from loss of information due to multiple warping operations [16].

### 3.2. Sequential Adversarial Training

In order for STNs to learn geometric warps that map images closer to the natural image manifold, we integrate them into a GAN framework, which we refer to as ST-GANs. The motivation for this is two-fold. First, learning a realistic geometric correction is a multi-modal problem (*e.g.* a bed can reasonably exist in multiple places in a room); second, supervision for these warp parameters are typically not available. The main difference of ST-GANs from conventional GANs is that (1)  $\mathcal{G}$  generates a set of *low-dimensional* warp parameter updates instead of images (the whole set of pixel values); and (2)  $\mathcal{D}$  gets as input the warped foreground image composite with the background.

To learn gradual geometric improvements toward the natural image manifold, we adopt a sequential adversarial training strategy for iterative STNs (Fig. 3), where the geometric predictor  $\mathcal{G}$  corresponds to the stack of generators  $\mathcal{G}_i$ . We start by training a single  $\mathcal{G}_1$ , and each subsequent new

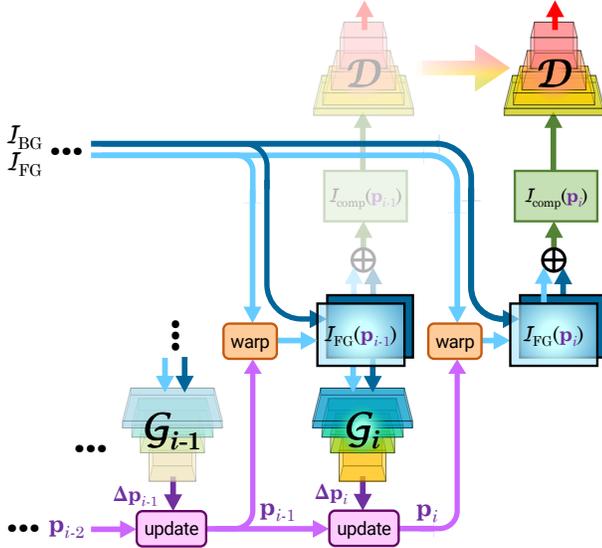


Figure 3: **Sequential adversarial training** of ST-GAN. When learning a new warp state  $\mathbf{p}_i$ , only the new generator  $\mathcal{G}_i$  is updated while the previous ones are kept fixed. A single discriminator (learned from all stages) is continuously improved during the sequential learning process.

generator  $\mathcal{G}_i$  is added and trained by fixing the weights of all previous generators  $\{\mathcal{G}_j\}_{j=1\dots i-1}$ . As a result, we train only  $\mathcal{G}_i$  and  $\mathcal{D}$  by feeding the resulting composite image at warp state  $\mathcal{I}_{\text{comp}}(\mathbf{p}_i)$  into the discriminator  $\mathcal{D}$  and matching it against the real data distribution. This learning philosophy shares commonalities with the Supervised Descent Method [33], where a series of linear regressors are solved greedily, and we found it makes the overall training faster and more robust. Finally, we fine-tune the entire network end-to-end to achieve our final result. Note that we use the same discriminator  $\mathcal{D}$  for all stages of the generator  $\mathcal{G}_i$ , as the fundamental measure of “geometric fakeness” does not change over iterations.

### 3.3. Adversarial Objective

We optimize the Wasserstein GAN (WGAN) [1] objective for our adversarial game. We note that ST-GAN is amenable to any other GAN variants [21, 38, 3], and that the choice of GAN architecture is orthogonal to this work.

The WGAN minimax objective at the  $i$ th stage is

$$\min_{\mathcal{G}_i} \max_{\mathcal{D} \in \mathbb{D}} \mathbb{E}_{x \sim \mathbb{P}_{\text{fake}}^{\mathbf{p}_i}} [\mathcal{D}(x(\mathbf{p}_i))] - \mathbb{E}_{y \sim \mathbb{P}_{\text{real}}} [\mathcal{D}(y)] , \quad (4)$$

$\mathbf{p}_i \sim \mathbb{P}_{\mathbf{p}_i | \mathbf{p}_{i-1}}$

where  $y = \mathcal{I}_{\text{real}}$  and  $x = \mathcal{I}_{\text{comp}}$  are drawn from the real data and fake composite distributions, and  $\mathbb{D}$  is the set of 1-Lipschitz functions enforced by adding a gradient penalty term  $\mathcal{L}_{\text{grad}}$  [8]. Here,  $\mathbf{p}_i$  (where  $\mathcal{G}_i$  is implied, defined in (3)) is drawn from the posterior distribution conditioned on  $\mathbf{p}_{i-1}$

(recursively implied). When  $i = 1$ , the initial warp  $\mathbf{p}_0$  is drawn from  $\mathbb{P}_{\text{pert}}$ , a predefined distribution for geometric data augmentation.

We also constrain the warp update  $\Delta \mathbf{p}_i$  to lie within a trust region by introducing an additional penalty  $\mathcal{L}_{\text{update}} = \|\Delta \mathbf{p}_i\|_2^2$ . This is essential since ST-GAN may learn trivial solutions to remove the foreground (e.g. by translating it outside the image or shrinking it into nothing), leaving behind only the background image and in turn making the composite image realistic already.

When training ST-GAN sequentially, we update  $\mathcal{D}$  and  $\mathcal{G}_i$  alternating the respective loss functions:

$$\mathcal{L}_{\mathcal{D}} = \mathbb{E}_{x, \mathbf{p}_i} [\mathcal{D}(x(\mathbf{p}_i))] - \mathbb{E}_y [\mathcal{D}(y)] + \lambda_{\text{grad}} \cdot \mathcal{L}_{\text{grad}} \quad (5)$$

$$\mathcal{L}_{\mathcal{G}_i} = -\mathbb{E}_{x, \mathbf{p}_i} [\mathcal{D}(x(\mathbf{p}_i))] + \lambda_{\text{update}} \cdot \mathcal{L}_{\text{update}} , \quad (6)$$

where  $\lambda_{\text{grad}}$  and  $\lambda_{\text{update}}$  are the penalty weights for the  $\mathcal{D}$  gradient and the warp update  $\Delta \mathbf{p}_i$  respectively, and  $\mathcal{G}_i$  and  $\Delta \mathbf{p}_i$  are again implied through (3). When fine-tuning ST-GAN with  $N$  learned updates end-to-end, the generator objective is the sum of that from each  $\mathcal{G}_i$ , i.e.  $\mathcal{L}_{\mathcal{G}} = \sum_{i=1}^N \mathcal{L}_{\mathcal{G}_i}$ .

## 4. Experiments

We begin by describing the basic experimental settings.

**Warp parameterizations.** We parameterize a homography with the  $\mathfrak{sl}(3)$  Lie algebra [22], i.e. the warp parameters  $\mathbf{p} \in \mathfrak{sl}(3)$  and homography matrices  $\mathbf{H} \in \mathbb{S}\mathbb{L}(3)$  are related through the exponential map. Under this parameterization, warp composition can be expressed as the addition of parameters, i.e.  $\mathbf{p}_a \circ \mathbf{p}_b \equiv \mathbf{p}_a + \mathbf{p}_b \quad \forall \mathbf{p}_a, \mathbf{p}_b \in \mathfrak{sl}(3)$ .

**Model architecture.** We denote the following:  $\mathbf{C}(k)$  is a 2D convolutional layer with  $k$  filters of size  $4 \times 4$  and stride 2 (halving the feature map resolution) and  $\mathbf{L}(k)$  is a fully-connected layer with  $k$  output nodes. The input of the generators  $\mathcal{G}_i$  has 7 channels: RGBA for foreground and RGB for background, and the input to the discriminator  $\mathcal{D}$  is the composite image with 3 channels (RGB). All images are rescaled to  $120 \times 160$ , but we note that the parameterized warp can be applied to full-resolution images at test time.

The architecture of  $\mathcal{G}$  is  $\mathbf{C}(32)\text{-}\mathbf{C}(64)\text{-}\mathbf{C}(128)\text{-}\mathbf{C}(256)\text{-}\mathbf{C}(512)\text{-}\mathbf{L}(256)\text{-}\mathbf{L}(8)$ , where the output is the 8-dimensional (in the case of a homography) warp parameter update  $\Delta \mathbf{p}$ . For each convolutional layer in  $\mathcal{G}$ , we concatenate a down-sampled version of the original image (using average pooling) with the input feature map. For  $\mathcal{D}$ , we use a PatchGAN architecture [10], with layout  $\mathbf{C}(32)\text{-}\mathbf{C}(64)\text{-}\mathbf{C}(128)\text{-}\mathbf{C}(256)\text{-}\mathbf{C}(512)\text{-}\mathbf{C}(1)$ . Nonlinearity activations are inserted between all layers, where they are ReLU for  $\mathcal{G}$  and LeakyReLU with slope 0.2 for  $\mathcal{D}$ . We omit all normalization layers as we found them to deteriorate training performance.

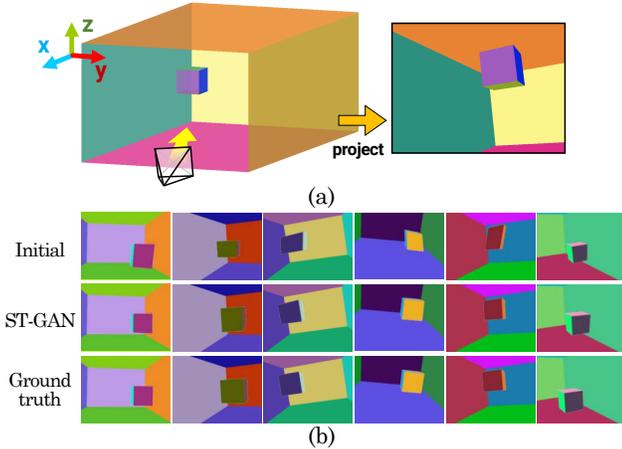


Figure 4: (a) We create a synthetic dataset of 3D cube renderings and validate the efficacy of ST-GAN by attempting to correct randomly generated geometric perturbations. (b) ST-GAN is able to correct the cubes to a right perspective, albeit a possible translational offset from the ground truth.

#### 4.1. 3D Cubes

To begin with, we validate whether ST-GANs can make geometric corrections in a simple, artificial setting. We create a synthetic dataset consisting of a 3D rectangular room, an axis-aligned cube inside the room, and a perspective camera (Fig. 4(a)). We apply random 3-DoF translations to the cube and 6-DoF perturbations to the camera, and render the cube/room pair separately as the foreground/background (of resolution  $120 \times 160$ ). We color all sides of the cube and the room randomly.

We perturb the rendered foreground cubes with random homography transformations as the initial warp  $p_0$  and train ST-GAN by pairing the original cube as the ground-truth counterpart for  $\mathcal{D}$ . As shown in Fig. 4(b), ST-GAN is able to correct the perturbed cubes scale and perspective distortion w.r.t. the underlying scene geometry. In addition, ST-GAN is sometimes able to discover other realistic solutions (e.g. not necessarily aligning back to the ground-truth location), indicating ST-GAN’s ability to learn the multi-modal distribution of correct cube placements in this dataset.

#### 4.2. Indoor Objects

Next, we show how ST-GANs can be applied to practical image compositing domains. We choose the application of compositing furniture in indoor scenes and demonstrate its efficacy on both simulated and real-world images. To collect training data, we create a synthetic dataset consisting of rendered background scenes and foreground objects with masks. We evaluate on the synthetic test set as well as high-resolution real world photographs to validate whether ST-GAN also generalizes to real images.

Category	Training set		Test set	
	# 3D inst.	# pert.	# 3D inst.	# pert.
Bed	3924	11829	414	1281
Bookshelf	508	1280	58	137
Cabinet	9335	31174	1067	3518
Chair	196	609	22	60
Desk	64	1674	73	214
Dresser	285	808	31	84
Refrigerator	3802	15407	415	1692
Sofa	3604	11165	397	1144
<b>Total</b>	<b>22303</b>	<b>73946</b>	<b>2477</b>	<b>8130</b>

Table 1: Dataset statistics for the indoor object experiment, reporting the number of object instances chosen for perturbation, and the final number of rendered perturbed samples.

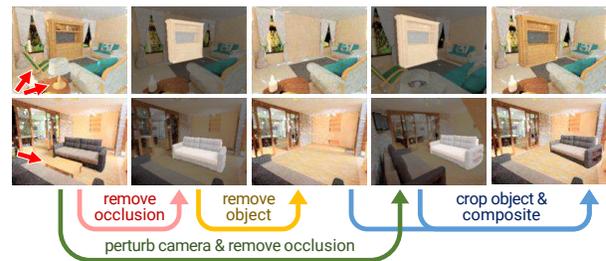


Figure 5: **Rendering pipeline.** Given an indoor scene and a candidate object, we remove occluding objects to create an occlusion-free scenario, which we do the same at another perturbed camera pose. We further remove the object to create a training sample pair with mismatched perspectives.

**Data preparation.** We render synthetic indoor scene images from the SUNCG dataset [28], consisting of 45,622 indoor scenes with over 5M 3D object instances from 37 categories [27]. We use the selected 41,499 scene models and the 568,749 camera viewpoints from Zhang *et al.* [37] and utilize Mitsuba [12] to render photo-realistic images with global illumination. We keep a list of candidate 3D objects consisting of all instances visible from the camera viewpoints and belonging to the categories listed in Table 1.

The rendering pipeline is shown in Fig. 5. During the process, we randomly sample an object from the candidate list, with an associated camera viewpoint. To emulate an occlusion-free compositing scenario, occlusions are automatically removed by detecting overlapping object masks. We render one image with the candidate object present (as the “real” sample) and one with it removed (as the background image). In addition, we perturb the 6-DoF camera pose and render the object with its mask (as the foreground image) for compositing. We thus obtain a rendered object as viewed from a *different camera perspective*; this simulates the image compositing task where the foreground and background perspectives mismatch. We note that a homography correction can only approximate these 3D perturbations, so there is *no planar ground-truth warp* to use for supervision.

Category	Initial warp	SDM [33]	HomographyNet [5]	ST-GAN (non-seq.)	ST-GAN (warp 1)	ST-GAN (warp 2)	ST-GAN (warp 4)	ST-GAN (end-to-end)	Ground truth
Bed	35.5 %	30.5 %	30.2 %	32.8 %	32.8 %	<b>46.8 %</b>	32.8 %	32.2 %	75.0 %
Bookshelf	21.1 %	33.9 %	35.1 %	16.7 %	26.4 %	26.2 %	39.5 %	<b>42.6 %</b>	68.9 %
Cabinet	20.9 %	19.8 %	35.0 %	36.6 %	14.3 %	31.2 %	44.4 %	<b>50.0 %</b>	74.3 %
Chair	32.8 %	36.8 %	47.6 %	50.9 %	<b>62.3 %</b>	42.7 %	50.0 %	58.6 %	68.7 %
Desk	18.9 %	13.1 %	36.1 %	35.4 %	29.2 %	29.0 %	39.4 %	<b>40.7 %</b>	65.1 %
Dresser	14.9 %	18.6 %	20.7 %	16.7 %	24.6 %	27.4 %	29.7 %	<b>48.4 %</b>	66.1 %
Refrigerator	37.1 %	21.4 %	50.0 %	37.7 %	28.6 %	47.1 %	39.7 %	<b>51.7 %</b>	81.6 %
Sofa	15.9 %	31.0 %	42.4 %	28.9 %	37.0 %	54.9 %	<b>56.1 %</b>	51.8 %	78.2 %
<b>Average</b>	24.6 %	25.6 %	37.1 %	31.9 %	31.9 %	38.2 %	41.5 %	<b>47.0 %</b>	72.6 %

Table 2: **AMT User studies** for the indoor objects experiment. Percentages represent the how often the images in each category were classified as “real” by Turkers. We can see that our final model, ST-GAN (end-to-end), substantially improves over geometric realism when averaged across all classes. Our realism performance improves with the number of warps trained as well as after the end-to-end fine-tuning. The ground truth numbers serve as a theoretical upper bound for all methods.

We report the statistics of our rendered dataset in Table 1. All images are rendered at  $120 \times 160$  resolution.

**Settings.** Similar to the prior work by Lin & Lucey [16], we train ST-GAN for  $N = 4$  sequential warps. During adversarial training, we rescale the foreground object randomly from  $\text{Unif}(0.9, 1.1)$  and augment the initial warp  $\mathbf{p}_0$  with a translation sampled from  $\mathcal{N}(0, 0.05)$  scaled by the image dimensions. We set  $\lambda_{\text{update}} = 0.3$  for all methods.

**Baselines.** One major advantage of ST-GAN is that it can learn from “realism” comparisons *without ground-truth warp parameters* for supervision. However, prior approaches require supervision directly on the warp parameters. Therefore, we compare against self-supervised approaches trained with random homography perturbations on foreground objects as input, yielding warp parameters as self-supervision. We reemphasize that such direct supervision is insufficient in this application as we aim to find the closest point on a manifold of realistic looking composites rather than fitting a specific paired model. Our baselines are (1) HomographyNet [5], a CNN-based approach that learns direct regression on the warp parameters, and (2) Supervised Descent Method (SDM) [33], which greedily learns the parameters through cascaded linear regression. We train the SDM baseline for 4 sequential warps as well.

**Quantitative evaluation.** As with most image generation tasks where the goal is realism, there is no natural quantitative evaluation possible. Therefore, we carry out a perceptual study on Amazon Mechanical Turk (AMT) to assess geometric realism of the warped composites. We randomly chose 50 test images from each category and gather data from 225 participants. Each participant was shown a composite image from a randomly selected algorithm (Table 2), and was asked whether they saw any objects whose shape does not look natural in the presented image.

We report the AMT assessment results in Table 2. On average, ST-GAN shows a large improvement of geometric realism, and quality improves over the sequential warps. When considering that the warp is restricted to homography transformations, these results are promising, as we are not correcting for more complicated view synthesis effects for out-of-plane rotations such as occlusions. Additionally, ST-GAN, which does not require ground truth warp parameters during training, greatly outperforms other baselines, while SDM yields no improvement and HomographyNet increases realism, but to a lesser degree.

**Ablation studies.** We found that learning iterative warps is advantageous: compared with a non-iterative version with the same training iterations (non-seq. in Table 2), ST-GAN (with multiple generators) approaches geometric realism more effectively with iterative warp updates. In addition, we trained an iterative HomographyNet [5] using the same sequential training strategy as ST-GAN but found little visual improvement over the non-iterative version; we thus focus our comparison against the original [5].

**Qualitative evaluation.** We present qualitative results in Fig. 6. ST-GAN visually outperforms both baselines trained with direct homography parameter supervision, which is also reflected in the AMT assessment results. Fig. 7 shows how ST-GAN updates the homography warp with each of its generators; we see that it learns gradual updates that makes a realism improvement at each step. In addition, we illustrate in Fig. 8 the effects ST-GAN learns, including gradual changes of the object perspective at different composite locations inside the room, as well as a “snapping” effect that predicts a most likely composite location given a neighborhood of initial locations. These features are automatically learned from the data, and they can be useful when implemented in interactive settings.



Figure 6: **Qualitative evaluation** on the indoor rendering test set. Compared to the baselines trained with direct homography supervision, ST-GAN creates more realistic composites. We find that ST-GAN is able to learn common object-room relationships in the dataset, such as beds being against walls. Note that ST-GANs corrects the perspectives but not necessarily scale, as objects often exist at multiple scales in the real data. We observe that ST-GAN occasionally performs worse for unusual objects (*e.g.* with peculiar colors, last column).

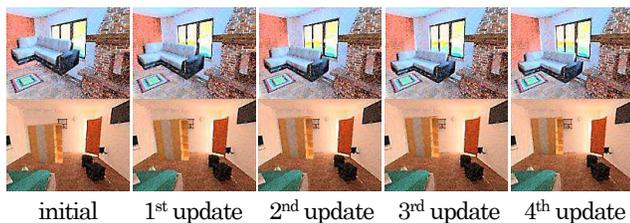


Figure 7: Visualization of iterative updates in ST-GAN, where objects make gradual improvements that reaches closer to realism in an incremental fashion.

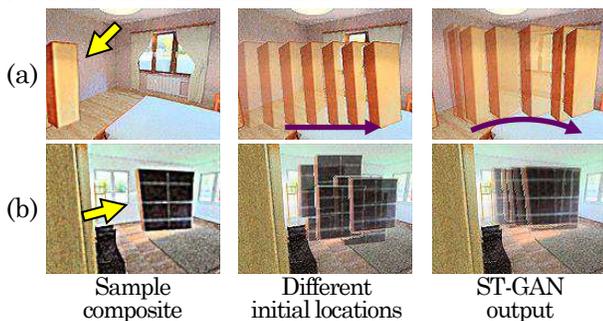


Figure 8: **Dragging and snapping.** (a) When an object is dragged across the scene, the perspective changes with the composite location to match that of the camera's. (b) ST-GAN “snaps” objects to where it would be frequently composited (*e.g.* a bookshelf is usually laid against the wall).



Figure 9: **Real world high-resolution test results.** Here we show our method applied to real images. The inputs are scaled down and fed to the network and then the warp parameters are applied at full resolution.

Finally, to test whether ST-GAN extends to real images, we provide a qualitative evaluation on photographic, high-resolution test images gathered from the Internet and manually masked (Fig 9). This is feasible since the warp parameters predicted from the low-resolution network input are transferable to high-resolution images. As a consequence, ST-GAN is indirectly applicable to various image resolutions and not strictly limited as with conventional GAN frameworks. Our results demonstrates the utilization of ST-GAN for high-quality image generation and editing.



Figure 10: The split of CelebA for the background and the real images, as well as the crafted glasses as the foreground.

### 4.3. Glasses

Finally, we demonstrate results in an entirely unpaired setting where we learn warping corrections for compositing glasses on human faces. The lack of paired data means that we do not necessarily have pictures of the same people both with and without glasses (ground truth).

**Data preparation.** We use the CelebA dataset [19] and follow the provided training/test split. We then use the “eyeglasses” annotation to separate the training set into two groups. The first group of people with glasses serve as the real data to be matched against in our adversarial settings, and the group of people without glasses serves as the background. This results in 152249 training and 18673 test images without glasses, and 10521 training images with glasses. We hand-crafted 10 pairs of frontal-facing glasses as the foreground source (Fig. 10). We note that there are no annotations about where or how the faces are placed, and we do not have any information where the different parts of the glasses are in the foreground images.

In this experiment, we train ST-GAN with  $N = 5$  sequential warps. We crop the aligned faces into  $144 \times 144$  images and resize the glasses to widths of 120 pixels initialized at the center. During training, we add geometric data augmentation by randomly perturbing the faces with random similarity transformations and the glasses with random homographies.

**Results.** The results are shown in Fig. 11. As with the previous experiments, ST-GAN learns to warp the foreground glasses in a gradual fashion that improves upon realism at each step. We find that our method can correctly align glasses onto the people’s faces, even with a certain amount of in-plane rotations. However, ST-GAN does a poorer job on faces with too much out-of-plane rotation.

While such an effect is possible to achieve by taking advantage of facial landmarks, our results are encouraging as no information was given about the structure of either domain, and we only had access to unpaired images of people with and without glasses. Nonetheless, ST-GAN was able to learn a realism manifold that drove the Spatial Transformer generators. We believe this demonstrates great potential to extend ST-GANs to other image alignment tasks where acquiring paired data is very challenging.

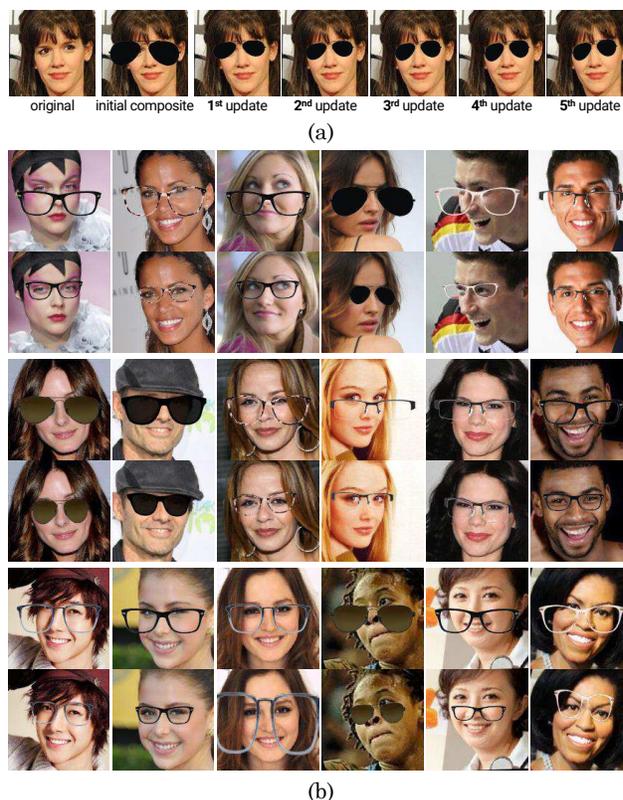


Figure 11: **Glasses compositing results.** (a) The glasses progressively moves into a more realistic position. (b) ST-GAN learns to warp various kinds of glasses such that the resulting positions are usually realistic. The top rows indicates the initial composite, and the bottom rows indicates the ST-GAN output. The last 4 examples shows failure cases, where glasses fail to converge onto the faces.

## 5. Conclusion

We have introduced ST-GANs as a class of methods to model *geometric realism*. We have demonstrated the potential of ST-GANs on the task of image compositing, showing improved realism in a large-scale rendered dataset, and results on fully unpaired real-world image data. It is our hope that this work will open up new revenues to the research community to continue to explore in this direction.

Despite the encouraging results ST-GAN achieves, there are still some limitations. We find that ST-GAN suffers more when presented imbalanced data, particularly rare examples (*e.g.* white, thick-framed glasses in the glasses experiment). In addition, we also find convergence of ST-GAN to fail with more extreme translation or in-plane rotation of objects. We believe a future analysis of the convergence properties of classical image alignment methods with GAN frameworks is worthy of investigation in improving the robustness of ST-GANs.

## References

- [1] M. Arjovsky, S. Chintala, and L. Bottou. Wasserstein gan. *arXiv preprint arXiv:1701.07875*, 2017. 4
- [2] S. Baker and I. Matthews. Lucas-kanade 20 years on: A unifying framework. *International journal of computer vision*, 56(3):221–255, 2004. 3
- [3] D. Berthelot, T. Schumm, and L. Metz. Began: Boundary equilibrium generative adversarial networks. *arXiv preprint arXiv:1703.10717*, 2017. 4
- [4] J. Dai, H. Qi, Y. Xiong, Y. Li, G. Zhang, H. Hu, and Y. Wei. Deformable convolutional networks. *arXiv preprint arXiv:1703.06211*, 2017. 2
- [5] D. DeTone, T. Malisiewicz, and A. Rabinovich. Deep image homography estimation. *arXiv preprint arXiv:1606.03798*, 2016. 6
- [6] Y. Ganin, D. Kononenko, D. Sungatullina, and V. Lempitsky. Deepwarp: Photorealistic image resynthesis for gaze manipulation. In *European Conference on Computer Vision*, pages 311–326. Springer, 2016. 2
- [7] I. Goodfellow, J. Pouget-Abadie, M. Mirza, B. Xu, D. Warde-Farley, S. Ozair, A. Courville, and Y. Bengio. Generative adversarial nets. In *Advances in neural information processing systems*, pages 2672–2680, 2014. 1, 2
- [8] I. Gulrajani, F. Ahmed, M. Arjovsky, V. Dumoulin, and A. Courville. Improved training of wasserstein gans. *arXiv preprint arXiv:1704.00028*, 2017. 4
- [9] B. K. Horn and B. G. Schunck. Determining optical flow. *Artificial intelligence*, 17(1-3):185–203, 1981. 3
- [10] P. Isola, J.-Y. Zhu, T. Zhou, and A. A. Efros. Image-to-image translation with conditional adversarial networks. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2017. 2, 4
- [11] M. Jaderberg, K. Simonyan, A. Zisserman, et al. Spatial transformer networks. In *Advances in Neural Information Processing Systems*, pages 2017–2025, 2015. 1, 2
- [12] W. Jakob. Mitsuba renderer, 2010. <http://www.mitsuba-renderer.org>. 5
- [13] X. Jia, B. De Brabandere, T. Tuytelaars, and L. V. Gool. Dynamic filter networks. In *Advances in Neural Information Processing Systems*, pages 667–675, 2016. 2
- [14] A. Kanazawa, D. W. Jacobs, and M. Chandraker. Warpnet: Weakly supervised matching for single-view reconstruction. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 3253–3261, 2016. 2
- [15] W.-S. Lai, J.-B. Huang, N. Ahuja, and M.-H. Yang. Deep laplacian pyramid networks for fast and accurate super-resolution. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2017. 2
- [16] C.-H. Lin and S. Lucey. Inverse compositional spatial transformer networks. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2017. 2, 3, 6
- [17] C.-H. Lin, R. Zhu, and S. Lucey. The conditional lucas & kanade algorithm. In *European Conference on Computer Vision*, pages 793–808. Springer, 2016. 3
- [18] M.-Y. Liu, T. Breuel, and J. Kautz. Unsupervised image-to-image translation networks. *arXiv preprint arXiv:1703.00848*, 2017. 2
- [19] Z. Liu, P. Luo, X. Wang, and X. Tang. Deep learning face attributes in the wild. In *Proceedings of International Conference on Computer Vision (ICCV)*, 2015. 8
- [20] B. D. Lucas and T. Kanade. An iterative image registration technique with an application to stereo vision. In *Proceedings of the 7th International Joint Conference on Artificial Intelligence - Volume 2, IJCAI’81*, pages 674–679, 1981. 3
- [21] X. Mao, Q. Li, H. Xie, R. Y. Lau, Z. Wang, and S. P. Smolley. Least squares generative adversarial networks. *arXiv preprint ArXiv:1611.04076*, 2016. 4
- [22] C. Mei, S. Benhimane, E. Malis, and P. Rives. Homography-based tracking for central catadioptric cameras. In *Intelligent Robots and Systems, 2006 IEEE/RSJ International Conference on*, pages 669–674. IEEE, 2006. 4
- [23] E. Park, J. Yang, E. Yumer, D. Ceylan, and A. C. Berg. Transformation-grounded image generation network for novel 3d view synthesis. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2017. 2
- [24] D. Pathak, P. Krahenbuhl, J. Donahue, T. Darrell, and A. A. Efros. Context encoders: Feature learning by inpainting. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 2536–2544, 2016. 2
- [25] P. Pérez, M. Gangnet, and A. Blake. Poisson image editing. In *ACM Transactions on graphics (TOG)*, volume 22, pages 313–318. ACM, 2003. 2
- [26] A. Radford, L. Metz, and S. Chintala. Unsupervised representation learning with deep convolutional generative adversarial networks. *arXiv preprint arXiv:1511.06434*, 2015. 2
- [27] N. Silberman, D. Hoiem, P. Kohli, and R. Fergus. Indoor segmentation and support inference from rgb-d images. *Computer Vision—ECCV 2012*, pages 746–760, 2012. 5
- [28] S. Song, F. Yu, A. Zeng, A. X. Chang, M. Savva, and T. Funkhouser. Semantic scene completion from a single depth image. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2017. 5
- [29] Y.-H. Tsai, X. Shen, Z. Lin, K. Sunkavalli, X. Lu, and M.-H. Yang. Deep image harmonization. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2017. 2
- [30] C. Vondrick, H. Pirsivash, and A. Torralba. Generating videos with scene dynamics. In *Advances In Neural Information Processing Systems*, pages 613–621, 2016. 2
- [31] X. Wang, A. Shrivastava, and A. Gupta. A-fast-rnn: Hard positive generation via adversary for object detection. *arXiv preprint arXiv:1704.03414*, 2017. 2
- [32] J. Wu, C. Zhang, T. Xue, B. Freeman, and J. Tenenbaum. Learning a probabilistic latent space of object shapes via 3d generative-adversarial modeling. In *Advances in Neural Information Processing Systems*, pages 82–90, 2016. 2
- [33] X. Xiong and F. De la Torre. Supervised descent method and its applications to face alignment. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 532–539, 2013. 3, 4, 6
- [34] X. Yan, J. Yang, E. Yumer, Y. Guo, and H. Lee. Perspective transformer nets: Learning single-view 3d object reconstruction without 3d supervision. In *Advances in Neural Information Processing Systems*, pages 1696–1704, 2016. 2

- [35] J. Yang, A. Kannan, D. Batra, and D. Parikh. Lr-gan: Layered recursive generative adversarial networks for image generation. *ICLR*, 2017. 2
- [36] R. Yeh, Z. Liu, D. B. Goldman, and A. Agarwala. Semantic facial expression editing using autoencoded flow. *arXiv preprint arXiv:1611.09961*, 2016. 2
- [37] Y. Zhang, S. Song, E. Yumer, M. Savva, J.-Y. Lee, H. Jin, and T. Funkhouser. Physically-based rendering for indoor scene understanding using convolutional neural networks. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2017. 5
- [38] J. Zhao, M. Mathieu, and Y. LeCun. Energy-based generative adversarial network. *arXiv preprint arXiv:1609.03126*, 2016. 4
- [39] T. Zhou, M. Brown, N. Snavely, and D. G. Lowe. Unsupervised learning of depth and ego-motion from video. *arXiv preprint arXiv:1704.07813*, 2017. 2
- [40] T. Zhou, S. Tulsiani, W. Sun, J. Malik, and A. A. Efros. View synthesis by appearance flow. In *European Conference on Computer Vision*, pages 286–301. Springer, 2016. 2
- [41] J.-Y. Zhu, P. Krahenbuhl, E. Shechtman, and A. A. Efros. Learning a discriminative model for the perception of realism in composite images. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 3943–3951, 2015. 2
- [42] J.-Y. Zhu, P. Krähenbühl, E. Shechtman, and A. A. Efros. Generative visual manipulation on the natural image manifold. In *Proceedings of European Conference on Computer Vision (ECCV)*, 2016. 2
- [43] J.-Y. Zhu, T. Park, P. Isola, and A. A. Efros. Unpaired image-to-image translation using cycle-consistent adversarial networks. *arXiv preprint arXiv:1703.10593*, 2017. 2