

# DiffPop: Plausibility-Guided Object Placement Diffusion for Image Composition

Jiacheng Liu<sup>1</sup>, Hang Zhou<sup>2</sup>, Shida Wei<sup>1</sup>, Rui Ma<sup>†1</sup>

<sup>1</sup>Jilin University, China  
<sup>2</sup>Simon Fraser University, Canada



**Figure 1:** Given one or multiple foreground objects, our plausibility-guided diffusion model can generate plausible object placement with diverse scale and location, as well as structural coherence to the background image.

## Abstract

In this paper, we address the problem of plausible object placement for the challenging task of realistic image composition. We propose DiffPop, the first framework that utilizes plausibility-guided denoising diffusion probabilistic model to learn the scale and spatial relations among multiple objects and the corresponding scene image. First, we train an unguided diffusion model to directly learn the object placement parameters in a self-supervised manner. Then, we develop a human-in-the-loop pipeline which exploits human labeling on the diffusion-generated composite images to provide the weak supervision for training a structural plausibility classifier. The classifier is further used to guide the diffusion sampling process towards generating the plausible object placement. Experimental results verify the superiority of our method for producing plausible and diverse composite images on the new Cityscapes-OP dataset and the public OPA dataset, as well as demonstrate its potential in applications such as data augmentation and multi-object placement tasks. Our dataset and code will be released.

## CCS Concepts

• Computing methodologies → Image manipulation; Computer vision;

## 1. Introduction

Image composition involves creating realistic composite images by combining specific foreground objects with background images. It has wide applications in fields such as entertainment and creative industries. In this work, we focus on the object placement task, which is a subtask of image composition. Object placement refers to accurately pasting a foreground object onto a background image with a suitable scale and location, resulting in a realistic composite image. This technology finds utility in various scenarios. For ex-

ample, in product or advertisement design, object placement can provide suggestions for the optimal placement of logos. In entertainment content, such as popular mobile phone AR applications, virtual objects can be automatically inserted into scenes for visualization or interaction.

While several works [LLG\*18, LYW\*18, ZWM\*20, TCA\*19, ZMZ\*22, ZLNZ22, ZLC\*23] have been proposed to address the object placement task, there are still challenges to be addressed. Previous methods [LLG\*18, LYW\*18, ZWM\*20, ZLNZ22] adopt generative adversarial networks (GANs) [GPAM\*20] for adversarial training, which are prone to mode collapse and lack of diversity in generation. To mitigate such issue, more stable generators such

<sup>†</sup> Corresponding author

as VAE-GANs [LSLW16, ZMZ\*22] are adopted later, yet mode collapse issue still exists. Another challenge lies in the lack of training data for complex scenes. Taking Cityscapes dataset [COR\*16] as an example, the existing object scales and locations are not sufficient for training a functional placement network. The main reason is, training solely on real images leads to an abundance of positive samples but no negative samples, resulting in lower plausibility of the composite image. Meanwhile, the recent OPA dataset [LLZ\*21] provide positive and negative binary labels for rational or plausible object placement. With these labels, the OPA dataset is suitable for training classifiers for *assessing* the binary plausibility of the object placement result. However, how to efficiently utilize OPA’s positive and negative labels for *generating* plausible object placement still remains to be a problem.

In contrast to GANs which are prone to unstable training and limited diversity, diffusion models [HJA20, SME20, ND21, DN21, HS22] provide the advantage of stable training and enhanced diversity. By continuously adding noise to real samples and then denoising, diffusion models enable the generation of realistic samples and demonstrate state-of-the-art performance in various image generation tasks, including unconditional image generation, image inpainting, and image super-resolution. Compared to other generative models, diffusion models can capture and model the intricate dependencies in images and lead to visually impressive results and improved generation performance. Furthermore, the classifier-guided diffusion models [DN21] can utilize the classifier guidance during the sampling process to enhance the generation quality of specific class samples. It is a natural thought to explore the applicability of diffusion models for the object placement task.

In this paper, we propose DiffPop, the first plausibility-guided diffusion probabilistic model for object placement. Initially, we employ a self-supervised training scheme to train a object placement diffusion model directly on the object transformation parameters, without explicit conditioning on the foreground and background information. This enables the diffusion model to learn the distribution of scales and locations for foreground objects w.r.t. background scenes from real data. Then, we train a structural plausibility classifier which evaluates the generated placement at each time step and guides the diffusion sampling process towards the desired direction, i.e., plausible object placement. Specifically, when training such a plausibility classifier, we either take the existing positive/negative labels from the existing dataset (e.g., OPA) or adopt a human-in-the-loop strategy to address the issue of plausibility measurement. For the latter case, we manually label the composite images produced from our initial unguided diffusion model into distinct positive and negative classes, based on the criterion of the image-level plausibility and realism. Such easy-to-obtain human annotations can provide sufficient weak supervision for learning the binary plausibility classifier, which can be efficiently used in the guided object placement diffusion.

To verify the effectiveness of our method, we conduct extensive experiments on the OPA dataset and Cityscapes-OP, a new dataset created by manually labeling the composite images produced from the initial unguided diffusion model. In addition, we also show the applications of our method in creating composite images for data augmentation, as well as its extension for multi-object placement.

In summary, our contributions are as follows:

- We propose DiffPop, the first plausibility-guided diffusion framework that aims to generate plausible object placement for image composition. Specifically, we learn a structural plausibility classifier to provide guidance on the diffusion-based object placement generation process.
- We employ the human-in-the-loop strategy to obtain image-level weak supervision for training the plausibility classifier. And we create a new dataset Cityscapes-OP, which can be used for training plausibility-guided diffusion model for placing objects on scenes with more complex and structural backgrounds than those in OPA dataset.
- Experimental results demonstrate that our method achieves state-of-the-art object placement performance in terms of plausibility and diversity on both Cityscapes-OP and OPA datasets. Our approach also shows promising results in creating composite images for data augmentation and multi-object placement.

## 2. Related work

### 2.1. Object placement

Image composition involves creating a composite image that appears realistic by combining a specific foreground object with a background image. In the field of computer vision, Niu et al. [NCL\*21] categorized image composition into four branches: object placement, image blending, image harmonization, and shadow generation. These branches address various challenges encountered during the image composition process. In this paper, we focus on the object placement task which aims to determining the appropriate scale and location for the foreground object.

Traditional object placement methods [RHB18, WWY\*19, FSW\*19, GMBK17, ZZL\*20] employ explicit rules to find suitable locations and scales for the foreground object. On the other hand, learning-based object placement methods [ZMZ\*22, LLG\*18, ZLNZ22, ZWM\*20, TCA\*19] typically predict or generate affine transformation matrices to determine the location and scale of the foreground object on the background image. Lin et al. [LYW\*18] introduced a novel GAN architecture that leverages a spatial transformer network (STN) as a generator to transform the foreground objects based on generated transformation parameters, resulting in the generation of realistic composite images. This deep learning-based approach has significantly advanced the field of object placement. Lee et al. [LLG\*18] proposed an end-to-end VAE-GAN that generates transformation matrices and shapes for objects through self-supervised and unsupervised training. This method mitigates the risk of mode collapse during GAN training and has been widely adopted in subsequent work. Tripathi et al. [TCA\*19] incorporated an additional discriminator network during GAN training to facilitate targeted data augmentation for downstream tasks. Zhang et al. [ZWM\*20] utilized self-supervised data pairs obtained through pre-trained instance segmentation and image inpainting methods to ensure diversity for object placement. Liu et al. [LLZ\*21] created a dedicated object placement dataset called OPA and introduced the SimOPA classifier to assess the object placement. Zhou et al. [ZLNZ22] transformed the object placement problem into a graph node completion task and employed binary classification loss to train the discriminator network, which

makes full use of labeled negative samples. SAC-GAN [ZMZ\*22] incorporated edge and semantic information from the object and background image to improve the structural coherence of the composite results. TopNet [ZLC\*23] proposed the use of Transformers to learn the relationship between object features and local background features, resulting in improved generation of object scale and location.

In contrast to the aforementioned methods, our approach is based on the diffusion model, which offers diversity and stable training. Our method effectively utilizes positive and negative samples to guide the diffusion model in generating more reasonable scales and locations, leading to the composition of realistic images. Additionally, our guided-diffusion framework can also be extended to simultaneously placement of multiple objects, which cannot be achieved by previous methods.

## 2.2. Diffusion models

Ho et al. [HJA20] introduced denoising diffusion probabilistic models (DDPM), a generative model that optimizes the network by continuously adding noise to real samples and using the network to denoise. This approach enables the network to generate realistic samples. Song et al. [SME20] made improvements to the diffusion model to enhance sampling speed. Nichol et al. [ND21] further enhanced the diffusion model’s ability to generate high-quality samples. For conditional image synthesis, Dharival et al. [DN21] enhanced sampling quality by incorporating classifier guidance during the diffusion model’s sampling process. They utilized the gradient of the classifier to balance the diversity and plausibility of the generated samples. Liu et al. [LPA\*23] introduced a unified semantic diffusion guidance framework that allows guidance through language, image, or both. Ho et al. [HS22] jointly trained conditional and unconditional diffusion models, combining the resulting conditional and unconditional score estimates to achieve a balance between sample quality and diversity. This approach frees the diffusion model from the limitations of classifier-guided sampling. Nichol et al. [NDR\*21] proposed GLIDE, a model capable of generating high-quality images conditioned on text. They demonstrated that unconditional guidance is superior to CLIP guidance for language-based conditioning. Ramesh et al. [RDN\*22] proposed unCLIP, which leverages the feature space of CLIP and the diffusion model to generate images from textual descriptions in a zero-shot manner. Saharia et al. [SCS\*22] introduced Imagen, a framework that combines large Transformer language models with diffusion models to empower the network with the ability to generate images from textual prompts. Robin et al. [RBL\*22] applied diffusion models directly to latent spaces, resulting in significant computational resource savings for text-to-image generation. Hachnochi et al. [HZO\*23] applied diffusion models to the field of image composition. They iteratively injected contextual information from background images into inserted foreground objects, allowing control over the degree of change in the foreground object.

In contrast to above methods, we focus on the plausibility-guided object placement, i.e., we first train a plausibility-classifier based on the weakly annotated images produced from an unguided diffusion model and then use the classifier to guide the diffusion sampling process for producing plausible results.

## 3. Method

Given a scene image as background and an object patch as foreground, we seek to learn scale and spatial distributions of object placement so that realistic image composition can be obtained. The training pipeline of our DiffPop framework contains two stages as shown in Fig. 2: in Stage-1, an unguided object placement denoising diffusion model is trained to learn the distributions of object scales and locations; in Stage-2, a structural plausibility classifier is trained with the manually labeled composite images generated by the previously trained model. At inference time, as shown in Fig. 3, given a background image and an object patch, we generate 2D transformation (scale and location) for plausible object placement using the classifier-guided diffusion and adopt the copy-paste scheme for image composition.

### 3.1. Unguided object placement denoising diffusion

As described before, we first train an unguided denoising diffusion model for learning the distributions of object scales and locations in the given dataset.

**Diffusion process.** The forward diffusion process is a pre-defined Markov chain that operates on object placement  $\mathbf{x} \in \mathbb{R}^D$ , where  $\mathbf{x} = [s, v, h]$ ,  $s$  is the relative scale of the object-image pair and  $(v, h)$  are the vertical and horizontal offsets of the object to the image, as shown in Fig. 2 (above). To initiate the diffusion process, we start with a clean object placement  $\mathbf{x}_0$  sampled from the underlying distribution  $q(\mathbf{x}_0)$ . Then, we gradually add Gaussian noise to  $\mathbf{x}_0$ , resulting in a series of intermediate object placement variables  $\mathbf{x}_{1:T}$  following a predetermined schedule of linearly increasing noise variances, denoted as  $\beta_{1:T}$ . The joint distribution  $q(\mathbf{x}_t | \mathbf{x}_{t-1})$  of the diffusion process is formulated as:

$$q(\mathbf{x}_{1:T} | \mathbf{x}_0) = \prod_{t=1}^T q(\mathbf{x}_t | \mathbf{x}_{t-1}), \quad (1)$$

where the diffusion step at time  $t$  is defined as:

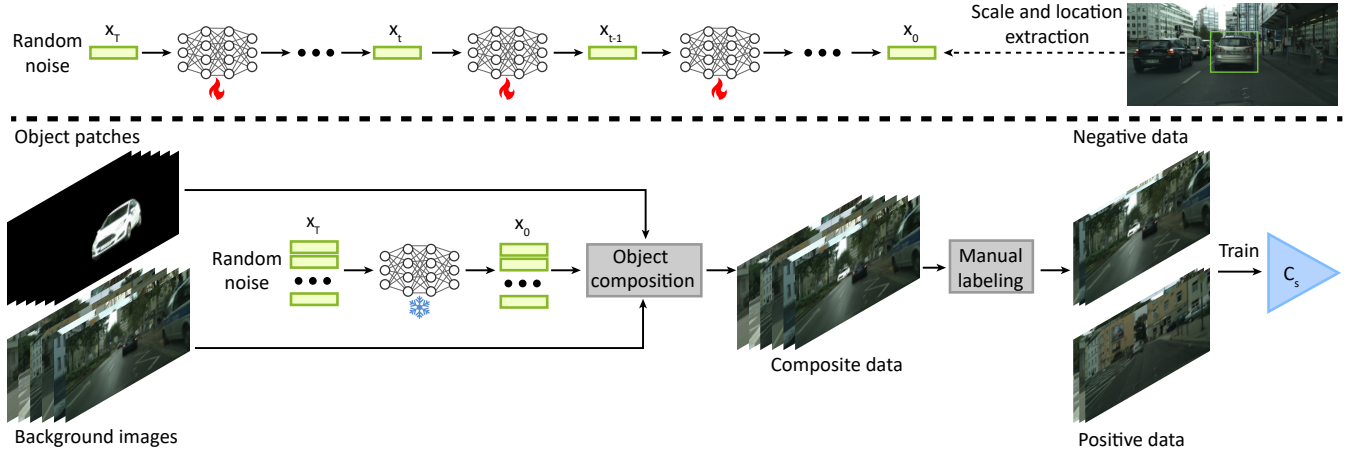
$$q(\mathbf{x}_t | \mathbf{x}_{t-1}) = \mathcal{N}(\mathbf{x}_t; \sqrt{1 - \beta_t} \mathbf{x}_{t-1}, \beta_t \mathbf{I}). \quad (2)$$

Thanks to the properties of Gaussian distribution, we can directly sample  $\mathbf{x}_t$  without the recursive formulation by:

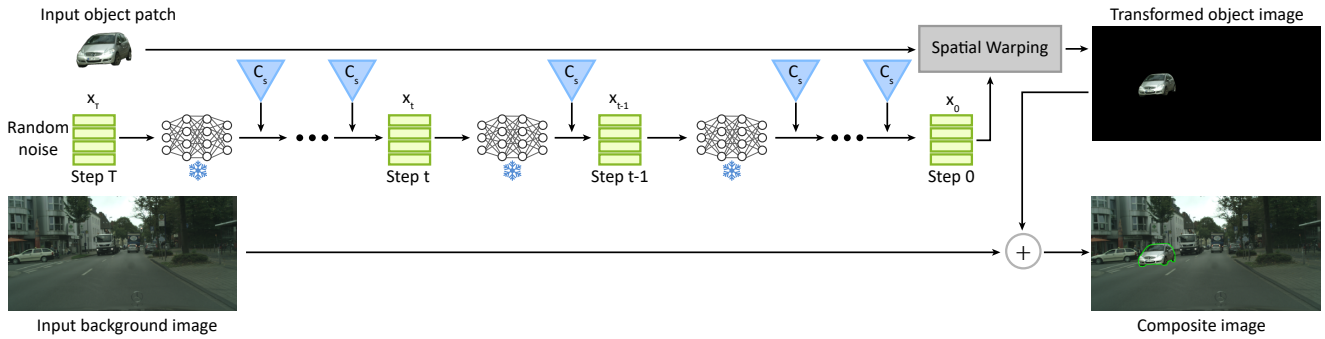
$$q(\mathbf{x}_t | \mathbf{x}_0) = \mathcal{N}(\mathbf{x}_t; \sqrt{\bar{\alpha}_t} \mathbf{x}_0, (1 - \bar{\alpha}_t) \mathbf{I}), \quad (3)$$

where  $\mathbf{x}_t = \sqrt{\bar{\alpha}_t} \mathbf{x}_0 + \sqrt{1 - \bar{\alpha}_t} \epsilon$ ,  $\alpha_t = 1 - \beta_t$ ,  $\bar{\alpha}_t = \prod_{r=1}^t \alpha_r$ ,  $\epsilon$  is the noise to corrupt  $\mathbf{x}_t$ .

**Denoising process.** The denoising process, also known as generative process, is parameterized as a Markov chain with learnable reverse Gaussian transitions. Given a noisy object placement sampled from a standard multivariate Gaussian distribution, denoted as  $\mathbf{x}_T \sim \mathcal{N}(\mathbf{0}, \mathbf{I})$ , serving as the initial state. The goal is to correct each state  $\mathbf{x}_t$  at every time step, producing a cleaner version  $\mathbf{x}_{t-1}$  using the learned Gaussian transition  $p_\theta(\mathbf{x}_{t-1} | \mathbf{x}_t)$ . This transition is determined by a learnable network denoted as  $\Theta$ . By iteratively applying this reverse process until the maximum number of steps  $T$  is reached, the final state  $\mathbf{x}_0$ , representing the desired clean object placement, is obtained. The joint distribution of the generative



**Figure 2:** The training pipeline of our DiffPop, consisting of two stages. In Stage-1 (above), we train an unguided object placement denoising diffusion model on object scales and locations. In Stage-2 (below), we first utilize the generated scales and locations from above pre-trained diffusion model to form the corresponding transformation matrix, followed by applying object composition to obtain the composite images; then, we adopt the human-in-the-loop strategy and manually label composite images into positive and negative classes based on plausibility and realism of composite images. These labeled data are then used for training the structural plausibility classifier  $C_s$ .



**Figure 3:** The inference pipeline of our DiffPop. We sample a random placement  $\mathbf{x}_T$  from the Gaussian noise at step  $T$  and iteratively denoise it till step 0 to obtain the desired placement  $\mathbf{x}_0$ . When sampling at step  $t$ , we take the gradient from the structural plausibility classifier  $C_s$  for guided generation towards scene-level structural coherence.

process, denoted as  $p_\theta(\mathbf{x}_{0:T})$ , is expressed as follows:

$$p_\theta(\mathbf{x}_{0:T}) = p(\mathbf{x}_T) \prod_{t=1}^T p_\theta(\mathbf{x}_{t-1} | \mathbf{x}_t), \quad (4)$$

$$p_\theta(\mathbf{x}_{t-1} | \mathbf{x}_t) = \mathcal{N}(\mathbf{x}_{t-1}; \mu_\theta(\mathbf{x}_t, t), \Sigma_\theta(\mathbf{x}_t, t)), \quad (5)$$

where the parameters  $\mu_\theta(\mathbf{x}_t, t)$  and  $\Sigma_\theta(\mathbf{x}_t, t)$  represent the predicted mean and covariance, respectively, of the Gaussian distribution for  $\mathbf{x}_{t-1}$ . These parameters are obtained by taking  $\mathbf{x}_t$  as input into the denoising network  $\Theta$ . For simplicity, we set predefined constants for  $\Sigma_\theta(\mathbf{x}_t, t)$  as in DDPM [HJA20]. Subsequently,  $\mu_\theta(\mathbf{x}_t, t)$  can be reparameterized by subtracting the predicted noise according to Bayes's theorem:

$$\mu_\theta(\mathbf{x}_t, t) = \frac{1}{\sqrt{\alpha_t}} \left( \mathbf{x}_t - \frac{\beta_t}{\sqrt{1-\alpha_t}} \epsilon_\theta(\mathbf{x}_t, t) \right). \quad (6)$$

**Network training objective.** We extract the ground-truth ob-

ject placements from the images to train our placement denoising network in a self-supervised manner. The network  $\Theta$  is a simply 4-layer MLPs, with input and output sizes of  $N \times 3$ . We train the network following  $\epsilon$ -prediction from DDPM [HJA20] with  $\ell_2$  loss:

$$L_{\text{unguided}} = \mathbb{E}_{\mathbf{x}_0, \epsilon, t} \|\epsilon - \epsilon_\theta(\mathbf{x}_t, t)\|^2. \quad (7)$$

### 3.2. Plausibility-guided object placement diffusion

As the unguided diffusion model only learns the object placement distribution, it does not consider the scene-level structural coherence and may fail to generate plausible placement conditioned on given objects and scene images. Inspired by the classifier-guided conditional generation in [DN21], we train a classifier based on annotated structural plausibility, and use its gradient to guide the diffusion sampling process towards scene-level structural coherence. To train such a plausibility classifier, we either take the existing positive/negative labels from the existing dataset (e.g., OPA)



or adopt a human-in-the-loop strategy to address the annotation of plausibility measurement.

**Human-in-the-loop plausibility labeling.** Existing datasets like Cityscapes were not initially designed for the object placement tasks, resulting in lack of ground-truth annotations for training an object composition network. Although self-supervised training scheme [ZWM\*20,ZMZ\*22] can be used to learn the object placement distributions from positive examples, the negative examples which are essential for training a binary classifier for plausibility measurement are generally missing. To resolve this issue, we employ a human-in-the-loop strategy to manually assign positive and negative labels to the composite images produced by the unguided object placement diffusion model, based on the criterion of the plausibility and realism of the images, as shown in Fig. 2 (below). Simply, a positive label means it is structure-coherent between the inserted object and the background scene, and the overall image is plausible, and vice versa. These human annotations can provide essential weak supervision for learning the plausibility classifier which can measure the results from the unguided diffusion model and further be used to guide the diffusion sampling process.

**Structural plausibility classifier.** To guide the diffusion model towards generating plausible object placements, we train a structural plausibility classifier  $C_s$  and use its gradient to guide the diffusion sampling process. The  $C_s$  is simply defined as a ResNet-18 backbone binary classifier, targeting to judge the structural plausibility of the composite image at the scene-level. The classifier takes the semantic scene layout combined with the object mask as inputs and trained with manually annotated positive/negative labels in a supervised manner. For the semantic layout of input scene, we directly utilize the semantic map provided by the dataset and process it into binary masks, while each mask is corresponding to one category. To obtain the composite layout from the input object mask and processed binary scene layout, we transform the 2D object patch using spatial warping proposed by spatial transformer network (STN) [JSZ\*15] with the affine transformation matrix  $A_t$  generated by the unguided diffusion model, where

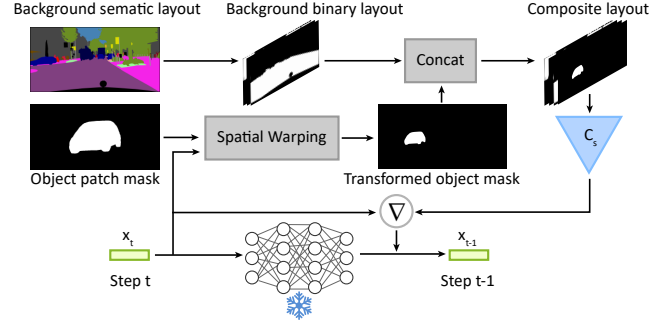
$$A_t = \begin{bmatrix} s_t & 0 & h_t \\ 0 & s_t & v_t \end{bmatrix}. \quad (8)$$

The classifier is independently trained using results from unguided diffusion and will be frozen during the guided diffusion process.

**Classifier-guided diffusion.** Once the classifier  $C_s$  is trained, we use its gradient to guide the sampling process of the object placement diffusion model (Fig. 4). Specifically, the sampling formula is defined as follows:

$$\mathbf{x}_{t-1} \leftarrow \mathcal{N}(\mu + \lambda \Sigma \nabla_{\mathbf{x}_t} \log p_{\phi}(C_s(\mathbf{x}_t)|\mathbf{x}_t), \Sigma), \quad (9)$$

where  $\lambda$  is the guidance scale factor which determines how much the gradient of the classifier  $C_s$  affect the sampling of the diffusion model,  $\phi$  is the parameters of  $C_s$ . Specifically, we generate a transformation based on the sampling result of time  $t$  and utilize it to transform a given object patch and combine it with the target background layout. Then, the composite layout (see Sec. 4.6 for details) is fed into  $C_s$  to obtain a plausibility score, which is the probability of the binary classification. We compute the gradient of the score w.r.t. the sampling result at time  $t$ , and the gradient is



**Figure 4:** The detailed illustration for biased sampling guided by the structural plausibility classifier  $C_s$ .  $C_s$  takes the composite layouts as the input and outputs the probability of structural plausibility, which is further utilized in the guided sampling of object placement diffusion model at step  $t - 1$ .

used to guide the sampling result at time  $t - 1$ . After  $T$  iterations, the final placement  $\mathbf{x}_0 = [s_0, h_0, v_0]$  can be obtained. Finally, we use the affine transformation matrix  $A_0$  formed by  $(s_0, h_0, v_0)$  (see Eq. 8) to transform the object, and paste the transformed object to the background image to obtain the composite image.

## 4. Results

We conduct a series of experiments to test how DiffPop performs on two datasets: Cityscapes-OP and OPA. Quantitative and qualitative comparisons with related methods show the superiority of our method in generating plausible and diverse results. In addition, we also explore the potential of our method in applications such as data augmentation and multi-object placement.

### 4.1. Dataset

**Cityscapes-OP dataset.** We build Cityscapes-OP based on composite images produced from the unguided diffusion model trained on the Cityscapes [COR\*16] dataset. Specifically, we collect 75 different foreground objects and use the unguided diffusion model to generate the placement of each object onto 600 different background scenes. The full dataset provides human-annotated plausibility labels for all composite images, including 12,000 (3,869 positive / 8,131 negative) composite images for training and 1,000 (395 positive / 605 negative) composite images for testing. More details on creating the dataset can be found in the supplementary material.

**OPA dataset.** OPA [LLZ\*21] dataset is specifically designed for assessing the object placement. Hence, each image is annotated with positive/negative rationality labels indicating whether the object placement is rational (or plausible). All the images are collected from COCO [LMB\*14] dataset and each composite image is generated with one background, one object, and one placement bounding box. The dataset includes 62,074 (21,376 positive/40,698 negative) composite images for training and 11,396 (3,588 positive/7,808 negative) composite images for testing. The composite images in OPA dataset contain 1,389 different background scenes and 4,137 different foreground objects in 47 different categories.

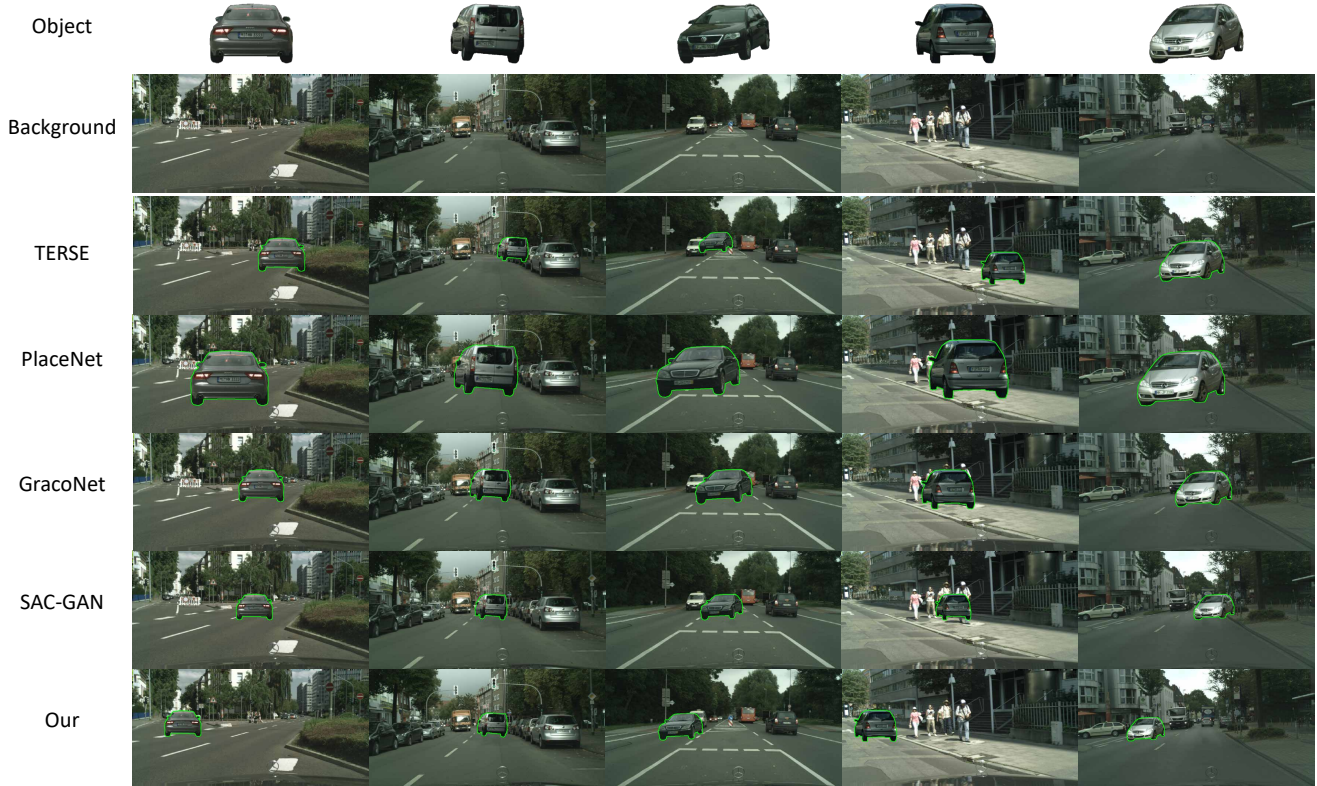


Figure 5: Qualitative results of single object placement on Cityscapes-OP. Best viewed with zoom-in.

## 4.2. Network training details

We trained two networks in our DiffPop framework, namely the diffusion network  $\Theta$  and classifier  $C_s$ . All models are implemented using PyTorch [PGM\*19] and trained on one RTX 3060 GPU. By default, we adopt Adam [KB14] optimizer with a learning rate of 0.0001. For training the diffusion network, we employ the following training hyperparameters: step  $T=100$ , batch size=128 and epochs=400. For training  $C_s$ , we use batch size=100. Note that for the OPA dataset, since it does not provide the semantic segmentation map of each image as the Cityscapes, the  $C_s$  for OPA is directly trained on the composite images instead of the composite structural layouts. From the comparison results and ablation study, such image-level classifier can also be used to boost the performance for diffusion-based object placement.

## 4.3. Evaluation metrics

In our evaluation, we employ multiple metrics to comprehensively assess the quality of the generated composite images in terms of plausibility and diversity.

**User study on plausibility.** To evaluate the plausibility of the composite images, we first conduct a user study in which human participants provide subjective assessments. Specifically, the user study was conducted by 20 computer science graduate students. Each questionnaire consisted of 30 image groups, with each group comprising a set of results generated from different methods for

placing one foreground object onto a background image. Each participant was asked to score each image in every group on a scale of 1-5, based on two criteria: 1) Plausibility of the size and location of the foreground object placed on the background image; 2) Overall structural coherence of the image, independent of factors such as color, shadow and resolution.

**Objective metrics on plausibility.** Additionally, we utilize the accuracy metric to quantitatively evaluate the plausibility of the composite images. Following the similar way of defining accuracy in GracoNet [ZLNZ22], our accuracy metric is defined as the percentage between the number of positive examples predicted by the plausibility classifier  $C_s$  and the total number of composite images. Moreover, we employ the FID (Fréchet Inception Distance) metric [HRU\*17], which is also used to measure the realism and plausibility by quantifying the similarity between the distribution of the composite results and that of positive samples in the test set.

**Metrics on diversity.** Furthermore, to assess the diversity and variation in the generated composite images, we adopt the LPIPS (Learned Perceptual Image Patch Similarity) metric [ZIE\*18]. This perceptual similarity metric captures the dissimilarity between images and serves as an indicator of diversity. Additionally, we measure the  $(\Delta s, \Delta h, \Delta v)$  values to evaluate the scaling, horizontal, and vertical variations, respectively. Higher values of  $(\Delta s, \Delta h, \Delta v)$  indicate larger range of object placement variations and higher diversity in the generated composite images.



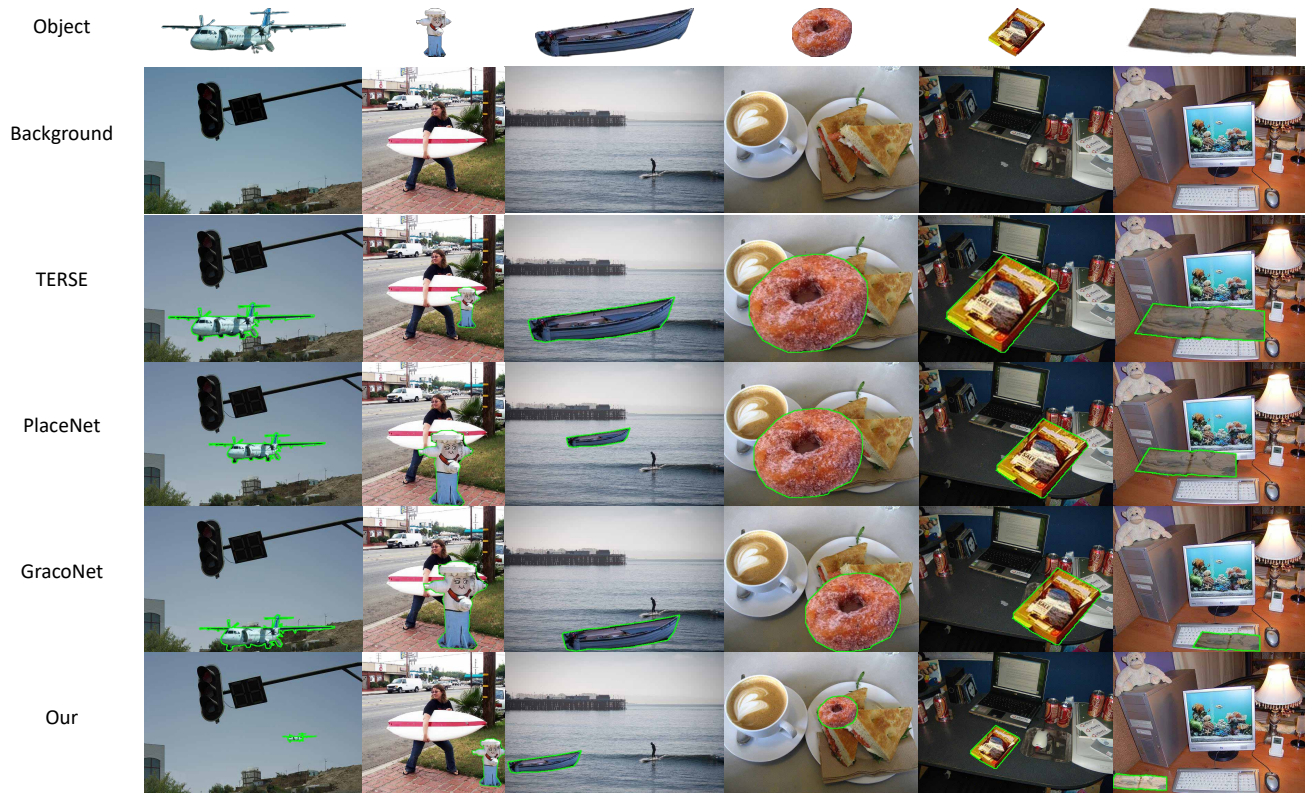


Figure 6: Qualitative results of single object placement on OPA dataset. Best viewed with zoom-in.

#### 4.4. Comparisons

For Cityscapes-OP dataset, we compare our DiffPop, against most related methods including TERSE [TCA\*19], PlaceNet [ZWM\*20], GracoNet [ZLNZ22], and SAC-GAN [ZMZ\*22]. However, for the OPA dataset, the SAC-GAN is not included in the comparison due to the unavailability of semantic segmentation labels. For fair comparison, we adopt the experimental setup of GracoNet and incorporate the binary classification loss [ZLNZ22] into the other methods, enabling their discriminator networks to utilize labeled positive and negative samples.

**Qualitative comparisons.** Fig. 5 and Fig. 6 show qualitative comparison results on Cityscapes-OP and OPA datasets. For Cityscapes-OP, we show different representative scene scenarios: empty road, crowded road and pedestrians occupied on the streets etc. As shown in Fig. 5, our method demonstrates superior performance compared to other methods in these cases. As our method explicitly consider the structural coherence in the structural plausibility classifier, our results show more plausible structural relations among the inserted objects and other objects in the scene. Also, comparing the GAN-based methods, our diffusion-based methods can generate object placement with larger diversity, e.g., the vehicle can be placed near the border of the image, instead of mainly appear near the image center. Similarly, for OPA dataset, our method outperforms others for different scenes and object categories, by

Table 1: Quantitative object placement results for different methods on Cityscapes-OP dataset. Best results are bolded and second best are underscored.

Method	Plausibility			Diversity			
	User study $\uparrow$	Acc. $\uparrow$	FID $\downarrow$	LPIPS $\uparrow$	$\Delta s\uparrow$	$\Delta h\uparrow$	$\Delta v\uparrow$
TERSE	—	0.782	33.85	0	0	0	0
PlaceNet	2.43	<u>0.888</u>	68.50	<b>0.117</b>	0.0392	<u>0.0420</u>	<b>0.0417</b>
GracoNet	2.80	0.867	34.78	0.027	0.0199	0.0152	0.0093
SAC-GAN	<u>2.97</u>	0.822	<u>29.98</u>	0.012	0.0018	0.0180	0.0009
Ours	<b>3.37</b>	<b>0.921</b>	<b>18.20</b>	<u>0.069</u>	<b>0.0755</b>	<b>0.2006</b>	<u>0.0399</u>

Table 2: Quantitative object placement results for different methods on OPA dataset. Best results are bolded and second best are underscored.

Method	Plausibility			Diversity			
	User study $\uparrow$	Acc. $\uparrow$	FID $\downarrow$	LPIPS $\uparrow$	$\Delta s\uparrow$	$\Delta h\uparrow$	$\Delta v\uparrow$
TERSE	—	0.629	46.64	0	0	0	0
PlaceNet	2.72	0.641	34.39	0.161	0.0666	0.0756	0.0807
GracoNet	2.89	<u>0.798</u>	<u>28.29</u>	<u>0.193</u>	<u>0.1247</u>	<u>0.1302</u>	0.0611
Ours	<b>3.59</b>	<b>0.954</b>	<b>19.76</b>	<b>0.202</b>	<b>0.1536</b>	<b>0.2081</b>	<b>0.1314</b>

generating object placements with larger variation in object scales and more plausible location that lead to higher scene coherence.

**Quantitative comparisons.** In Tables 1 and 2, we provide quantitative results for object placement and the results demonstrate that our method outperforms the compared methods on both datasets for plausibility measurements. For the diversity, since TERSE lacks

**Table 3:** Quantitative results of data augmentation with different methods on Cityscapes dataset. IoU ( $\uparrow$ ) for evaluating image segmentation performance for different categories are compared. Note the data augmentation is only conducted for the Car category.

Data	PSPNet				DeepLabv3			
	Car	Truck	Bus	mIoU	Car	Truck	Bus	mIoU
Original	93.12	61.66	77.07	70.73	93.99	73.22	78.28	73.50
+TERSE	93.28	57.32	72.38	69.77	94.30	<b>76.41</b>	82.31	73.95
+PlaceNet	93.15	61.61	69.06	68.14	93.72	65.91	73.81	71.82
+GracoNet	93.38	63.98	73.19	69.50	93.84	69.84	77.16	72.87
+SAC-GAN	93.17	56.73	73.08	70.29	94.25	74.33	81.29	73.84
+Ours	<b>93.42</b>	<b>64.56</b>	<b>77.97</b>	<b>71.11</b>	<b>94.34</b>	75.28	<b>82.70</b>	<b>74.75</b>

random noise input, its results exhibit limited diversity on both datasets. For Cityscapes-OP dataset, since PlaceNet tends to generate larger object scales during for the object placement (see Fig. 5), it may result in significant pixel-level changes in the composite images. As a result, PlaceNet exhibits a higher LPIPS value, which may not accurately reflect the diversity of the object placement. However, when considering all the spatial variation metrics ( $\Delta s$ ,  $\Delta h$ ,  $\Delta v$ ) that reflect the diversity of generated object scales and locations, our method outperforms PlaceNet. Therefore, developing a more comprehensive evaluation metric of diversity, e.g., taking into account both LPIPS and ( $\Delta s$ ,  $\Delta h$ ,  $\Delta v$ ) metrics, will be worthy to investigate in the future.

#### 4.5. Applications

**Data augmentation.** We also investigate the potential of our approach for data augmentation, specifically for semantic segmentation tasks. First, we randomly select 100 car objects from the total object library of the Cityscapes-OP dataset. Then, for each image in the training set of the Cityscapes dataset, we randomly select a car object as the foreground and place it onto the background. For each compared object placement method, we generate 2,975 composite images for data augmentation. Subsequently, we train two semantic segmentation models PSPNet [ZSQ\*17] and DeepLabv3 [CPSA17], on the augmented data generated by different methods. To evaluate the performance of each method in terms of data augmentation, we employ the Intersection-over-Union (IoU) metrics to quantize the segmentation performance on individual object classes and compare the results of models trained with augmented images. The results in Table 3 demonstrate that our method outperforms baselines in most cases. It is important to note that in this experiment, the data augmentation is only performed for the Car class, and the expected outcome is to increase the IoU for Car, without compromising the IoU for other categories. Meanwhile, it is interesting to see that the performance Truck and Bus can also be consistently increased, which may be due to the less confusion among the three categories. Such results can verify the plausibility and diversity of our generated composite images to a certain degree.

**Multi-object placement.** Our DiffPop framework can also be extended for multi-object placement. Similar to the structural plausibility classifier, a *relational* plausibility classifier  $C_r$  can be trained to guide the diffusion sampling so that the generated results contain plausible spatial relationships between multiple objects. The relational plausibility classifier  $C_r$  is trained to discriminate whether the relation between two independently generated ob-

**Table 4:** Quantitative results of ablation study on  $C_s$  and guidance scale  $\lambda$  on Cityscapes-OP dataset.

$\lambda$	Plausibility		Diversity			
	Acc. $\uparrow$	FID $\downarrow$	LPIPS $\uparrow$	$\Delta s\uparrow$	$\Delta h\uparrow$	$\Delta v\uparrow$
0	0.558	20.42	0.059	0.0757	0.2119	0.0425
0.001	0.756	19.23	0.062	0.0752	0.2085	0.0404
0.002	0.816	18.91	0.063	0.0756	0.2049	0.0394
0.005	0.907	18.48	0.065	0.0739	0.2021	0.0380
0.01	0.921	18.20	0.069	0.0755	0.2006	0.0399
0.02	0.912	18.66	0.075	0.0794	0.2046	0.0473
0.05	0.809	20.53	0.087	0.1017	0.2050	0.0801
0.1	0.736	22.54	0.101	0.1223	0.2100	0.1154
0.2	0.759	24.36	0.111	0.1297	0.1982	0.1268
0.5	0.852	25.64	0.115	0.1239	0.1876	0.1008

**Table 5:** Quantitative results of ablation study on  $C_s$  and guidance scale  $\lambda$  on OPA dataset.

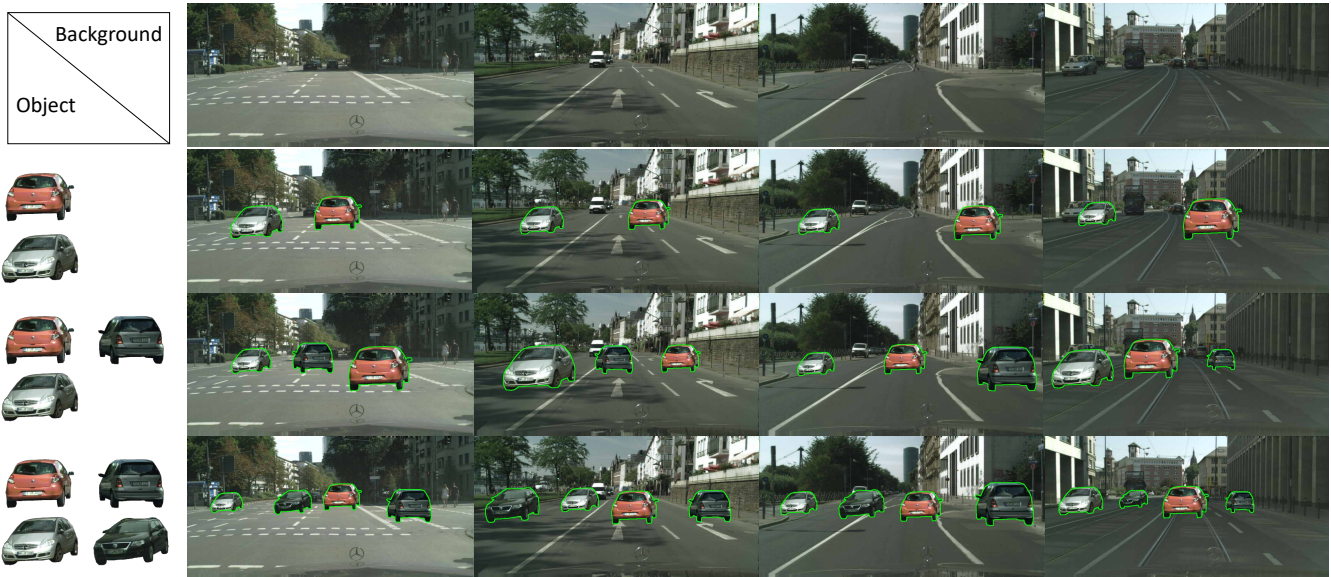
$\lambda$	Plausibility		Diversity			
	Acc. $\uparrow$	FID $\downarrow$	LPIPS $\uparrow$	$\Delta s\uparrow$	$\Delta h\uparrow$	$\Delta v\uparrow$
0	0.531	22.04	0.175	0.1829	0.1964	0.1809
0.01	0.823	19.82	0.177	0.1783	0.2292	0.1876
0.02	0.886	19.46	0.183	0.1709	0.2269	0.1714
0.05	0.933	19.71	0.196	0.1609	0.2159	0.1466
0.1	0.947	20.11	0.202	0.1565	0.2082	0.1362
0.2	0.954	19.76	0.202	0.1536	0.2081	0.1314
0.5	0.947	19.19	0.199	0.1545	0.2146	0.1350
1	0.925	18.71	0.197	0.1660	0.2281	0.1522
2	0.841	18.39	0.185	0.2076	0.2645	0.2171
5	0.610	25.04	0.130	0.2540	0.2881	0.3012

ject placements are plausible or not. More details about how to train  $C_r$  and how to use it together with the structural plausibility classifier  $C_s$  in the guided diffusion process are provided in the supplementary material. To evaluate the performance of our method for multi-object placement, we conduct preliminary qualitative experiments on Cityscapes-OP, as shown in Figure 7. Specifically, we selected 4 objects from the object library and 4 backgrounds from the background library. Then, we separately place 2, 3, and 4 objects on each of the four backgrounds, respectively. The results demonstrate that our method can be effectively extended for multi-object placement. More exploration and evaluation of our diffusion-based framework for multi-object placement will also be a promising future direction.

#### 4.6. Ablation study

**Guidance scale  $\lambda$ .** We conducted experiments to evaluate the effectiveness of  $C_s$  and the impact of different guidance scale factors  $\lambda$  on the performance of our method. The results are presented in Table 4 and Table 5, where a guidance scale of 0 indicates that  $C_s$  was not used for guidance. It can be seen that the guidance provided by  $C_s$  has a significant impact on all the evaluated metrics. When an appropriate guidance scale factor is used, notable improvements in accuracy, FID, and LPIPS can be obtained. However, if the guidance scale  $\lambda$  is too large, the accuracy and FID metrics tend to decrease, while the ( $\Delta s$ ,  $\Delta h$ ,  $\Delta v$ ) metrics increase. This is due to excessive guidance applied to the sampling of the diffusion model, which may lead to the generation of scales and locations





**Figure 7:** Qualitative results of multi-object placement on Cityscapes-OP. Best viewed with zoom-in.

**Table 6:** Quantitative results of ablation study on influence of different input type of  $C_s$  on Cityscapes-OP dataset.

Input Type	F1 $\uparrow$	TPR $\uparrow$	TNR $\uparrow$	Balanced Acc $\uparrow$
Mask + composite image	0.709	0.734	0.780	0.757
Mask + semantic label	0.797	0.838	0.826	0.832
Mask + binarized layout	0.864	0.906	0.874	0.890

that deviate from the real distribution. Notably, the selection of appropriate  $\lambda$  differs for different datasets. The OPA dataset contains multiple foreground and background classes, resulting in a wider distribution of real scales and locations. As a result, OPA exhibits a stronger tolerance to larger guidance scale factors. In contrast, the real object scales and locations in Cityscapes-OP dataset are relatively limited comparing to OPA. Hence, the  $\lambda$  values are generally smaller than those of OPA’s, which means weaker guidance is tolerated for Cityscapes-OP. Based on this experiment, we use  $\lambda = 0.01$  for Cityscapes-OP and  $\lambda = 0.2$  for all other experiments in this paper.

**Input of the structural plausibility classifier  $C_s$ .** We conducted experiments on Cityscapes-OP to assess the effect of training  $C_s$  on different types of inputs: RGB image with object mask, semantic label with object mask, and binarized semantic layout with object mask. Here, the binarized semantic layout is created by converting each semantic label to a separate binary mask and then concatenating them into a binarized semantic layout with 19 channels. From the results in Table 6, it can be observed the binarized semantic layout with object mask as the input for  $C_s$  yields the best performance in terms of F1-score, true positive rate (TPR), true negative rate (TNR) and balanced accuracy for evaluating the plausibility (defined in a similar way to the accuracy in Sec. 4.3). The reason may be the binarized semantic layout can capture more disen-

tangled scene structure and richer spatial features compared to the original images and semantic labels.



**Figure 8:** Failure cases on specific scenarios. Best viewed with zoom-in.

#### 4.7. Limitations

Though our method achieves promising results for plausible object placement, it still has certain limitations when dealing with specific scenarios, including cases with complex backgrounds, mismatched foreground and background, and adherence to traffic rules. Firstly, the complexity of backgrounds can pose challenges for generating realistic composite images, as observed in the first sample of Fig. 8. Secondly, our method struggles to handle situations where the foreground and background do not match, as illustrated in the second example. For example, if the foreground object depicts a car facing left and right while the background represents a road facing up and down, our method encounters difficulties in determining the compatibility between them, resulting in the generation of unrealistic composite images. Lastly, our approach currently lacks the ability to enforce adherence to traffic rules, as shown in the third example. Without the knowledge of different types of roads and the more detailed semantic information, it is hard to ensure proper placement of vehicles in accordance with specific traffic regulations.

## 5. Conclusion

We present DiffPop, a novel framework to learn object placement via plausibility-guided diffusion model. In contrast to previous works, our approach achieves superior performance in generating more plausible and diverse object placement, as well as more robust training comparing to the GAN-based methods. Specifically, our approach leverages a structural plausibility classifier to guide the diffusion model in sampling more reasonable scales and locations for the given object. To train the plausibility classifier, we introduce the Cityscapes-OP, a new dataset annotated with positive and negative plausibility labels, which can facilitate future research endeavors on plausible object placement learning. Besides, our DiffPop framework is also extensible to multi-object placement, showcasing its potential and efficacy in more complex image composition tasks. In the future, one interesting direction is to enhancing the object placement learning so that it can generalize to unseen categories, without intensive training. In addition, it is also worthy to explore how to integrate the object placement with the image harmonization into the same diffusion-based framework to create realistic image composition.

## References

- [COR\*16] CORDTS M., OMRAN M., RAMOS S., REHFELD T., ENZWEILER M., BENENSON R., FRANKE U., ROTH S., SCHIELE B.: The Cityscapes dataset for semantic urban scene understanding. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition* (2016), pp. 3213–3223. 2, 5, 11
- [CPSA17] CHEN L.-C., PAPANDREOU G., SCHROFF F., ADAM H.: Re-thinking atrous convolution for semantic image segmentation. *arXiv preprint arXiv:1706.05587* (2017). 8
- [DN21] DHARIWAL P., NICHOL A.: Diffusion models beat GANs on image synthesis. *Advances in Neural Information Processing Systems* 34 (2021), 8780–8794. 2, 3, 4
- [FSW\*19] FANG H.-S., SUN J., WANG R., GOU M., LI Y.-L., LU C.: Instaboost: Boosting instance segmentation via probability map guided copy-pasting. In *Proceedings of the IEEE/CVF International Conference on Computer Vision* (2019), pp. 682–691. 2
- [GMBK17] GEORGAKIS G., MOUSAVIAN A., BERG A. C., KOSECKA J.: Synthesizing training data for object detection in indoor scenes. *arXiv preprint arXiv:1702.07836* (2017). 2
- [GPAM\*20] GOODFELLOW I., POUGET-ABADIE J., MIRZA M., XU B., WARDE-FARLEY D., OZAI R., COURVILLE A., BENGIO Y.: Generative adversarial networks. *Communications of the ACM* 63, 11 (2020), 139–144. 1
- [HJA20] HO J., JAIN A., ABBEEL P.: Denoising diffusion probabilistic models. *Advances in Neural Information Processing Systems* 33 (2020), 6840–6851. 2, 3, 4
- [HRU\*17] HEUSEL M., RAMSAUER H., UNTERTHINER T., NESSLER B., HOCHREITER S.: GANs trained by a two time-scale update rule converge to a local nash equilibrium. *Advances in Neural Information Processing Systems* 30 (2017). 6
- [HS22] HO J., SALIMANS T.: Classifier-free diffusion guidance. *arXiv preprint arXiv:2207.12598* (2022). 2, 3
- [HZO\*23] HACHNOCHI R., ZHAO M., ORZECH N., GAL R., MAHDAVI-AMIRI A., COHEN-OR D., BERMANO A. H.: Cross-domain compositing with pretrained diffusion models. *arXiv preprint arXiv:2302.10167* (2023). 3
- [HZRS16] HE K., ZHANG X., REN S., SUN J.: Deep residual learning for image recognition. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition* (2016), pp. 770–778.
- [JSZ\*15] JADERBERG M., SIMONYAN K., ZISSERMAN A., ET AL.: Spatial transformer networks. *Advances in Neural Information Processing Systems* 28 (2015). 5
- [KB14] KINGMA D. P., BA J.: Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980* (2014). 6
- [KMR\*23] KIRILLOV A., MINTUN E., RAVI N., MAO H., ROLLAND C., GUSTAFSON L., XIAO T., WHITEHEAD S., BERG A. C., LO W.-Y., ET AL.: Segment anything. *arXiv preprint arXiv:2304.02643* (2023).
- [LLG\*18] LEE D., LIU S., GU J., LIU M.-Y., YANG M.-H., KAUTZ J.: Context-aware synthesis and placement of object instances. *Advances in Neural Information Processing Systems* 31 (2018). 1, 2
- [LLZ\*21] LIU L., LIU Z., ZHANG B., LI J., NIU L., LIU Q., ZHANG L.: OPA: Object placement assessment dataset. *arXiv preprint arXiv:2107.01889* (2021). 2, 5
- [LMB\*14] LIN T.-Y., MAIRE M., BELONGIE S., HAYS J., PERONA P., RAMANAN D., DOLLÁR P., ZITNICK C. L.: Microsoft coco: Common objects in context. In *Computer Vision—ECCV 2014: 13th European Conference, Zurich, Switzerland, September 6–12, 2014, Proceedings, Part V 13* (2014), Springer, pp. 740–755. 5
- [LPA\*23] LIU X., PARK D. H., AZADI S., ZHANG G., CHOPIKYAN A., HU Y., SHI H., ROHRBACH A., DARRELL T.: More control for free! Image synthesis with semantic diffusion guidance. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision* (2023), pp. 289–299. 3
- [LSLW16] LARSEN A. B. L., SØNDERBY S. K., LAROCHELLE H., WINTHER O.: Autoencoding beyond pixels using a learned similarity metric. In *International Conference on Machine Learning* (2016), PMLR, pp. 1558–1566. 2
- [LYW\*18] LIN C.-H., YUMER E., WANG O., SHECHTMAN E., LUCEY S.: ST-GAN: Spatial transformer generative adversarial networks for image compositing. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition* (2018), pp. 9455–9464. 1, 2
- [NCL\*21] NIU L., CONG W., LIU L., HONG Y., ZHANG B., LIANG J., ZHANG L.: Making images real again: A comprehensive survey on deep image composition. *arXiv preprint arXiv:2106.14490* (2021). 2
- [ND21] NICHOL A. Q., DHARIWAL P.: Improved denoising diffusion probabilistic models. In *International Conference on Machine Learning* (2021), PMLR, pp. 8162–8171. 2, 3
- [NDR\*21] NICHOL A., DHARIWAL P., RAMESH A., SHYAM P., MISHKIN P., MCGREW B., SUTSKEVER I., CHEN M.: GLIDE: Towards photorealistic image generation and editing with text-guided diffusion models. *arXiv preprint arXiv:2112.10741* (2021). 3
- [PGM\*19] PASZKE A., GROSS S., MASSA F., LERER A., BRADBURY J., CHANAN G., KILLEEN T., LIN Z., GIMELSHAIN N., ANTIGA L., DESMAISON A., KOPF A., YANG E., DEVITO Z., RAISSON M., TEJANI A., CHILAMKURTHY S., STEINER B., FANG L., BAI J., CHINTALA S.: Pytorch: An imperative style, high-performance deep learning library. In *Advances in Neural Information Processing Systems* 32. 2019, pp. 8024–8035. 6
- [RBL\*22] ROMBACH R., BLATTMANN A., LORENZ D., ESSER P., OMMER B.: High-resolution image synthesis with latent diffusion models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition* (2022), pp. 10684–10695. 3
- [RDN\*22] RAMESH A., DHARIWAL P., NICHOL A., CHU C., CHEN M.: Hierarchical text-conditional image generation with CLIP latents. *arXiv preprint arXiv:2204.06125* (2022). 3
- [RHB18] REMEZ T., HUANG J., BROWN M.: Learning to segment via cut-and-paste. In *Proceedings of the European Conference on Computer Vision* (2018), pp. 37–52. 2
- [SCS\*22] SAHARIA C., CHAN W., SAXENA S., LI L., WHANG J., DENTON E. L., GHASEMIPOUR K., GONTIJO LOPES R., KARAGOL AYAN B., SALIMANS T., ET AL.: Photorealistic text-to-image diffusion models with deep language understanding. *Advances in Neural Information Processing Systems* 35 (2022), 36479–36494. 3

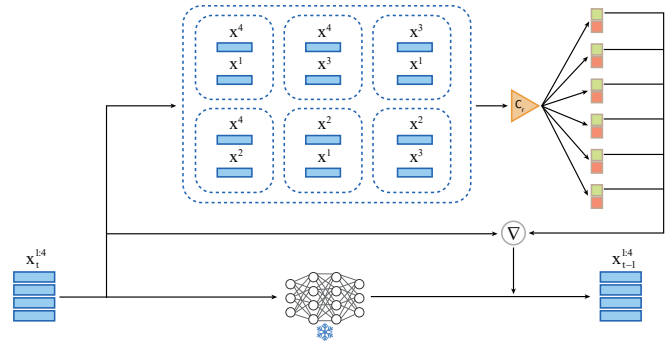
- [SME20] SONG J., MENG C., ERMON S.: Denoising diffusion implicit models. *arXiv preprint arXiv:2010.02502* (2020). 2, 3
- [TCA\*19] TRIPATHI S., CHANDRA S., AGRAWAL A., TYAGI A., REHG J. M., CHARI V.: Learning to generate synthetic data via compositing. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition* (2019), pp. 461–470. 1, 2, 7
- [WY\*19] WANG H., WANG Q., YANG F., ZHANG W., ZUO W.: Data augmentation for object detection via progressive and selective instance-switching. *arXiv preprint arXiv:1906.00358* (2019). 2
- [ZIE\*18] ZHANG R., ISOLA P., EFROS A. A., SHECHTMAN E., WANG O.: The unreasonable effectiveness of deep features as a perceptual metric. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition* (2018), pp. 586–595. 6
- [ZLC\*23] ZHU S., LIN Z., COHEN S., KUEN J., ZHANG Z., CHEN C.: TopNet: Transformer-based object placement network for image compositing. *arXiv preprint arXiv:2304.03372* (2023). 1, 3
- [ZLNZ22] ZHOU S., LIU L., NIU L., ZHANG L.: Learning object placement via dual-path graph completion. In *Proceedings of the European Conference on Computer Vision* (2022), Springer, pp. 373–389. 1, 2, 6, 7
- [ZMZ\*22] ZHOU H., MA R., ZHANG L.-X., GAO L., MAHDAVI-AMIRI A., ZHANG H.: SAC-GAN: Structure-aware image composition. *IEEE Transactions on Visualization and Computer Graphics* (2022). 1, 2, 3, 5, 7
- [ZSQ\*17] ZHAO H., SHI J., QI X., WANG X., JIA J.: Pyramid scene parsing network. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition* (2017), pp. 2881–2890. 8
- [ZWM\*20] ZHANG L., WEN T., MIN J., WANG J., HAN D., SHI J.: Learning object placement by inpainting for compositional data augmentation. In *Proceedings of the European Conference on Computer Vision* (2020), Springer, pp. 566–581. 1, 2, 5, 7
- [ZZL\*20] ZHANG S.-H., ZHOU Z.-P., LIU B., DONG X., HALL P.: What and where: A context-based recommendation system for object insertion. *Computational Visual Media* 6 (2020), 79–93. 2

## 6. Supplementary Material

In this supplementary material, we provide additional details on the creation of Cityscapes-OP dataset and more technical details on how to extend our framework for multi-object placement.

### 6.1. More details on Cityscapes-OP dataset

The Cityscapes [COR\*16] dataset, initially designed for 2D semantic segmentation of street scenes, consists of 2,975 training and 500 validation images. Due to its complex background scenes, this dataset has been widely used in previous object placement methods. To enable training of the plausibility classifier which needs both positive and negative samples we process the Cityscapes dataset to create a dataset specifically for object placement, referred to as Cityscapes-OP. When constructing the Cityscapes-OP dataset, we first extract 1,300 intact cars from the Cityscapes dataset to form the total object library. From this library, we randomly select 50 cars for the training object library and 25 cars for the test object library. Additionally, we randomly choose 500 images from the Cityscapes training set and 100 images from the validation set to create the training background library and test background library, respectively. Given the complexity of the background scenes, our focus in building the dataset lies primarily on the background images rather than the foreground objects.

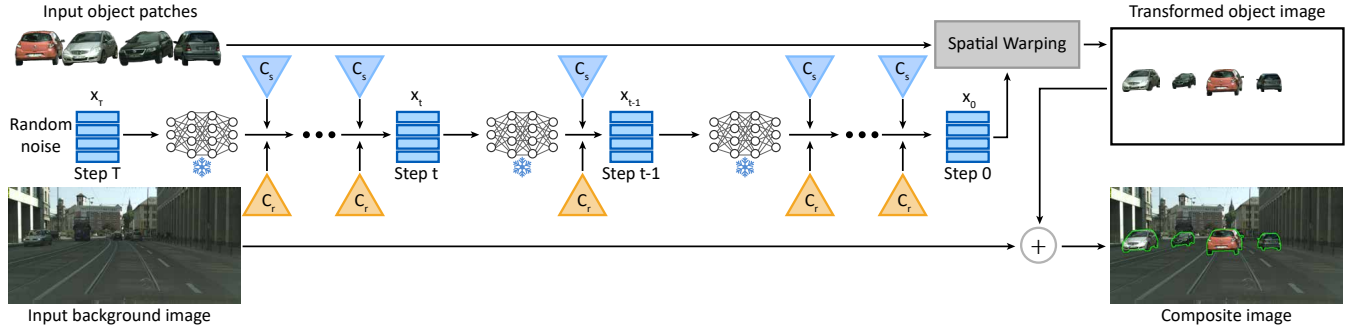


**Figure 9:** The details of  $C_r$  during inference. Taking the example of placing 4 objects, we first pair up the locations of the 4 objects obtained from the diffusion model sampling at time step  $t$ , resulting in 6 location pairs. Then, the  $C_r$  guides the sampling results of the diffusion model at time step  $t - 1$  based on the relational plausibility score of these 6 location pairs.

To build the training set, we utilize our pre-trained unguided diffusion model as the generator to produce 10,000 sizes and locations for object placement. By randomly composing the one of 50 cars from the training object library with the one of the 500 background images from the training background library, we obtain 10,000 composite images. These images are then manually labeled as plausible or implausible, resulting in 3,869 positive and 6,131 negative samples. To further enrich the dataset, we include 2,000 additional simple negative samples, such as composite images with objects that are over large or over small, or placed in unreasonable locations. These simple negative samples are created without overlapping with the previous 10,000 samples in terms of objects and backgrounds. Finally, the labeled 12,000 composite images constitute the training set of the Cityscapes-OP dataset. The test set is constructed in a similar way to the training set, where we generate 1,000 sizes and locations to create composite images by randomly combining one of the 25 objects from the test object library with one of the 100 backgrounds from the test background library. The resulting 1,000 composite images are manually labeled, yielding 395 positive samples and 605 negative samples.

It is worthy noting that the Cityscapes-OP dataset primarily focuses on single foreground category, specifically for cars. In contrast, the OPA dataset encompasses multiple foreground classes. On the other hand, the complex background scenes of Cityscapes-OP dataset allow us to effectively evaluate object placement methods in challenging background scenarios. During the dataset construction, we leverage the annotated bounding boxes and semantic/instance segmentation in the Cityscapes to create the Cityscapes-OP. These annotations provide valuable information for the foreground object generation and enhance the quality and realism of the composite images. Taking these factors into consideration, we believe the Cityscapes-OP dataset will be valuable for advancing research in the field of object placement.





**Figure 10:** Illustration of how to use  $C_r$  and  $C_s$  together to guide the diffusion model sampling.

## 6.2. More details on relational plausibility classifier

We propose a relational plausibility classifier  $C_r$  to guide the diffusion model sampling so that the sampling direction is biased towards the direction of plausible spatial relationship between multiple objects, enabling the diffusion model to generate appropriate sizes and locations for multiple foreground objects.

The network structure of  $C_r$  consists of four fully connected layers. Its goal is to determine whether the spatial relationship between two objects is plausible or not.  $C_r$  takes the placements of two objects as input and outputs a two-dimensional vector representing the plausibility of their spatial relationship.

$C_r$  is trained in a supervised manner with cross-entropy loss function. The training data are also manually labeled based on the Cityscapes-OP dataset. Firstly, 5 objects are randomly selected from the training object library of the Cityscapes-OP dataset, and 80 background images are randomly selected from the training background library. Then, each 2 of these 5 objects is paired together, resulting in 10 pairs of objects. For each pair of objects, image composition is performed under the same background using the  $C_s$  guided diffusion sampling, yielding composite images with two objects placed within the scene. Thereby, 800 composite images are obtained. These images are then manually labeled, following the same annotation process as the Cityscapes-OP dataset, and result in 407 positive samples and 393 negative samples. The 800 labeled composite images are then used to train  $C_r$ . This training is conducted on an RTX 3060 GPU using the Adam optimizer, with a batch size of 100 and 400 training epochs at a learning rate of  $10^{-4}$ .

In the case of one background and two foreground objects, the noisy placements  $\mathbf{x}_t^1, \mathbf{x}_t^2$  for the two objects sampled by the diffusion model at step  $t$  are fed into the relational plausibility classifier  $C_r$ . The gradient of the classifier’s output with respect to its input is then computed. This gradient is used to correct the mean of the distribution predicted by the diffusion model, thereby steering the sampling direction of the diffusion model towards a more plausible spatial relationship between the two objects. The sampling equation is as follows:

$$(\mathbf{x}_{t-1}^1, \mathbf{x}_{t-1}^2) \leftarrow N\left(\mu + \lambda_r \Sigma \nabla_{(\mathbf{x}_t^1, \mathbf{x}_t^2)} \log p_\tau(\mathbf{y} | \mathbf{x}_t^1, \mathbf{x}_t^2), \Sigma\right), \quad (10)$$

where  $\lambda_r$  is the guidance scale factor, used to control the degree of distribution correction.  $\Sigma$  is a fixed constant obtained from the diffusion process calculation.

Fig. 9 shows the case of taking 4 objects for multi-object placement. Specifically, the diffusion model samples four noisy placements at step  $t$ :  $\mathbf{x}_t^1, \mathbf{x}_t^2, \mathbf{x}_t^3, \mathbf{x}_t^4$ . These placements are paired to obtain six placement combinations. Then, for each combination, the gradient is computed by Eq. (10). The gradients of all combinations are summed and used to correct the mean  $\mu$  of the distribution predicted by the diffusion model.

Fig. 10 shows how to use the  $C_r$  together with the  $C_s$  to guide the sampling of the diffusion model, thereby accomplishing the task of placing multiple foreground objects on a single background image. The  $C_s$  ensures that the size and location of each object are plausible, while the  $C_r$  ensures that the spatial relationships between the objects are plausible.