

# BuilDiff: 3D Building Shape Generation using Single-Image Conditional Point Cloud Diffusion Models

Yao Wei  
University of Twente  
The Netherlands  
yao.wei@utwente.nl

George Vosselman  
University of Twente  
The Netherlands  
george.vosselman@utwente.nl

Michael Ying Yang  
University of Twente  
The Netherlands  
michael.yang@utwente.nl

## Abstract

*3D building generation with low data acquisition costs, such as single image-to-3D, becomes increasingly important. However, most of the existing single image-to-3D building creation works are restricted to those images with specific viewing angles, hence they are difficult to scale to general-view images that commonly appear in practical cases. To fill this gap, we propose a novel 3D building shape generation method exploiting point cloud diffusion models with image conditioning schemes, which demonstrates flexibility to the input images. By cooperating two conditional diffusion models and introducing a regularization strategy during denoising process, our method is able to synthesize building roofs while maintaining the overall structures. We validate our framework on two newly built datasets and extensive experiments show that our method outperforms previous works in terms of building generation quality.*

## 1. Introduction

Buildings play a role in various applications including urban modeling [34], industrial design [28], and virtual reality [38]. Since buildings usually have far more complicated 3D shapes than the clean models in common 3D object datasets, e.g., ShapeNet [2], efficient acquisition of 3D buildings remains an open problem in both synthetic and real scenarios. On one hand, designers typically use CAD technologies to create crafted 3D building models for synthetic scenarios (e.g., games and films), which inevitably requires tremendous manual effort. On the other hand, 3D shapes of real-world buildings are mostly acquired using mobile platforms such as airplanes, which are equipped with LiDAR sensors that capture the 3D coordinates of observed objects. Despite the recent release of several 3D city models where buildings are the primary components [14, 26], obtaining high-quality 3D building models is still a costly process, leading to limited availability in many parts of the world. Therefore, there is an urgent need for auto-

mated methods to generate realistic 3D buildings.

As an alternative to LiDAR point clouds, optical images obtained by regular cameras that have a lower acquisition cost, capture rich information about buildings as well. Over the past few decades, image-based 3D building reconstruction has long been a research focus [36, 6, 44]. Most of the existing works rely on photogrammetric technologies (e.g., Structure-from-Motion) leveraging multiple overlapping images taken from different viewpoints. Such requirements, however, restrict the applicability of those approaches to certain situations where multi-view images are infeasible to obtain. In contrast, single-image 3D reconstruction [20, 47, 24] offers a low-cost solution towards efficient 3D building shape generation, which could reduce the time-consuming manual design of synthetic buildings and facilitate the digital simulations of real buildings.

Driven by deep learning techniques, generative models [15, 10, 30, 12] have shown promising results on 3D computer vision, e.g., single image-to-3D point cloud [8, 9, 39]. However, these methods are mainly designed and trained on synthetic and symmetric 3D objects like airplanes, and they have limited applicability when generating 3D building models with complex structures in real-world scenarios. Besides, it is observed that single image-to-3D building methods [35, 20, 17] mostly require remote sensing images taken from specific perspectives, e.g., nadir-view, and the generated building models exhibit low level-of-details (LoDs) [26] without roof structures.

In this work, we present **BuilDiff** that takes a step into single general-view image-to-3D **building** synthesis by leveraging **diffusion** models in a coarse-to-fine manner. Our **main contributions** are summarized as follows:

- A novel hierarchical framework named *BuilDiff* is proposed to generate realistic 3D shapes of buildings with roof structures, i.e., at LoD2, given their single general-view images.
- Guided by an image auto-encoder, a base diffusion model coarsely identifies the overall structures of

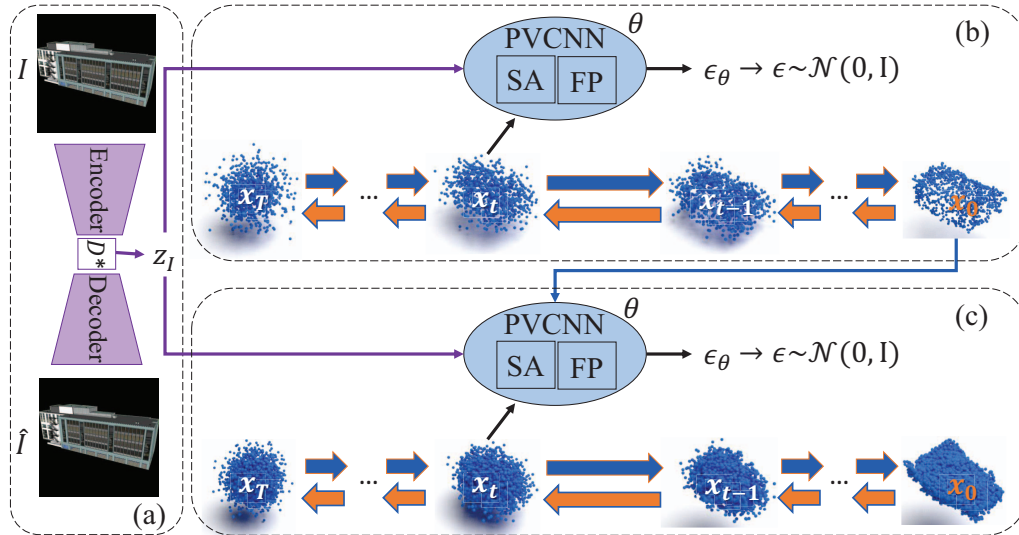


Figure 1. Overall pipeline of BuildDiff. (a) First, a CNN-based image auto-encoder is pre-trained to derive image embedding  $z_I$  in the latent space. (b) Second, a point cloud diffusion model  $\theta$  conditioned on  $z_I$  is trained for denoising from noise, i.e.,  $x_T$ , to a desired shape  $x_0$ . (c) Finally, another point cloud diffusion model is trained to sample a high-resolution shape conditioned on  $z_I$  and the low-resolution shape inferred by the previous diffusion model.

buildings, and a upsampler diffusion then derives higher resolution point clouds.

- A weighted building footprint-based regularization loss is introduced to constrain building structures and avoid ambiguous guidance during denoising process.

Experiments demonstrate the effectiveness of the proposed method on synthetic and real-world scenarios.

## 2. Related Work

**Single Image 3D Reconstruction.** Reconstructing the 3D shape of an object from its single-view image is a long-standing research problem in computer vision [3, 8]. This is motivated by human perception: given a single object image, humans can infer its potential 3D structure based on the cue image as well as prior knowledge of the 3D world. Many deep learning-based methods [9, 42, 40, 39] explore such prior knowledge by learning from abundant pairs of 3D shapes and single images. However, most of these methods concentrate on 3D benchmarks such as ShapeNet [2], which are restricted to synthetic objects with simple and symmetric 3D shapes. To capture the complexity of real-world objects, recent works [7, 21] exploit camera poses of images to reconstruct 3D shapes that are well aligned with the input images. While the camera poses contribute to 2D-3D correspondences, this formulation usually suffers from inflexibility for images without pose information. There can potentially be multiple solutions for pose estimation, but an inaccurate pose may result in low-resolution geometries.

Unlike common objects, buildings have much more challenging structures [33]. For real-world 3D building reconstruction from a single remote sensing image, existing works [35, 20, 17] roughly regard it as a combination of building footprint extraction and height prediction from nadir-view images. To generate more detailed 3D buildings, [24] performs image-to-mesh by leveraging street-view images aided by nadir-view images, and [47] develops implicit representations of object-level buildings. A recent work [29] specifically focuses on generating 3D point clouds of building roofs. Although these methods could generate relatively simple buildings, they fail to deal with complex buildings. Moreover, the applicability remains limited to view-specific input images, and general-purpose single-image 3D building reconstruction is less explored.

**Deep Generative Models for 3D Point Cloud.** In recent years, many generative models have been proposed and developed, including Variational Autoencoders (VAE) [15], Generative Adversarial Networks (GANs) [10], Normalizing Flows (NFs) [30], and Diffusion models [12, 31], which have achieved impressive results on data synthesis, especially for image synthesis tasks [27]. With great success in the 2D domain, deep generative models are gradually applied in the 3D domain. Besides using GANs [1] and NFs [43], a growing number of works [19, 48, 46, 23, 21] leverage diffusion models for 3D point cloud synthesis.

Parameterized by two Markov chains, denoising diffusion probabilistic models (DDPMs) [12] consist of a forward process and a reverse process. The former gradually adds random noise to input data  $x_0$ , and the latter recon-

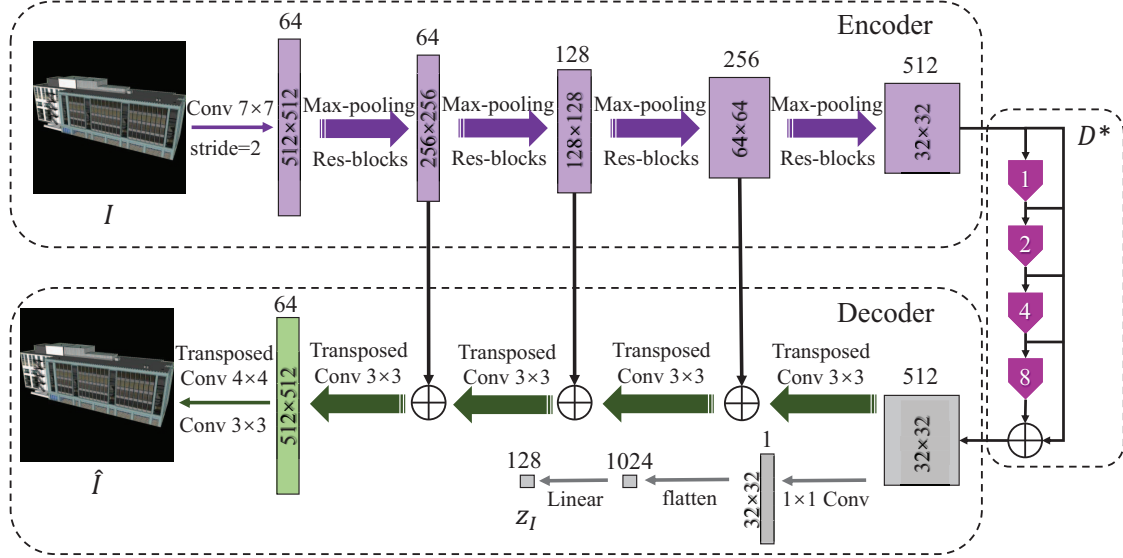


Figure 2. The architecture of image auto-encoder network. The input is RGB image  $I \in \mathbb{R}^{H \times W \times 3}$ , and the output contains reconstruction  $\hat{I} \in \mathbb{R}^{H \times W \times 3}$  and an image embedding  $z_I \in \mathbb{R}^d$ . The channel numbers and sizes of the feature maps are annotated above and inside them, respectively.  $\oplus$  indicates addition process, and the black arrows between encoder and decoder are skip connections.

constructs  $x_0$  starting from the random noise  $x_T$  where  $T$  indicates the total number of time steps. Compared with other generative models, denoising diffusion models demonstrate the advantages of high quality and diversity, which offer a promising avenue for the 3D shape synthesis of complex objects such as buildings. Hence, we perform 3D building reconstruction from a single image by scaling diffusion models from unconditional to conditional settings. To decrease the diversity while increasing the quality of each individual sample, two types of guidance are presented for diffusion models: classifier guidance [5] and classifier-free guidance [13].

### 3. Method

Instead of only exploiting images from specific perspectives, e.g., nadir- or street-views, our goal is to generate 3D point clouds of buildings from a single general-view image, aiming to improve the applicability of the proposed method. As shown in Fig. 1, we introduce a hierarchical framework BuilDiff which consists of three components: (a) image auto-encoder, (b) image-conditional point cloud base diffusion, and (c) image-conditional point cloud up-sampler diffusion.

#### 3.1. Image Auto-encoder Pre-training

A common and straightforward way to condition diffusion models is to compress the cue images into a latent space. Rather than using an encoder pre-trained on public image databases (e.g., ImageNet [4]) directly map an input image to a latent feature vector that serves as ambiguous conditions from buildings, we instead fine-tune the encoder

and train an additional decoder with the building images of the training set. As shown in Fig. 2, the overall network can be regarded as an image auto-encoder, which learns to reconstruct the input building images and extracts the features of buildings acting as representative conditions.

Taking an RGB image  $I$  of size  $H \times W$  as input, our image auto-encoder employs a ResNet-34 [11] based encoder to output feature maps of size  $H/32 \times W/32$ , which is then fed into stacked dilated convolution layers [45] (denoted as  $D^*$ ). There are four dilated convolution layers with dilation rates of 1, 2, 4, 8, which are stacked in cascade mode and parallel mode to aggregate features from different scales. For the feature maps derived from  $D^*$ , there are two processing ways in the decoding stage. Through transposed convolution layers, the feature maps can be upsampled to  $\hat{I}$  that exhibits the same size of the input image  $I$ . On the other hand, they are linearly projected into a 1-D image embedding  $z_I$  with a dimensionality of  $d$  via a  $1 \times 1$  convolution layer and a linear layer.

The image auto-encoder is trained by minimizing  $\mathcal{L}_{AE}$  defined as,

$$\mathcal{L}_{AE} = \mathcal{L}_{rec}(I, \hat{I}) + \mathcal{L}_{con}(z_I, z_I^a) \quad (1)$$

where  $\mathcal{L}_{rec}$  is a reconstruction loss between  $I$  and  $\hat{I}$ ,  $\mathcal{L}_{con}$  is a consistency loss which encourages the embedding  $z_I$  of an image  $I$  to be as close as possible to the embedding  $z_I^a$  of the augmented version  $I^a$  of the image  $I$ . After training, we use the frozen pre-trained image auto-encoder to map the image to an embeddings  $z_I$ , which acts as the image-dependent conditions for the following diffusion models.

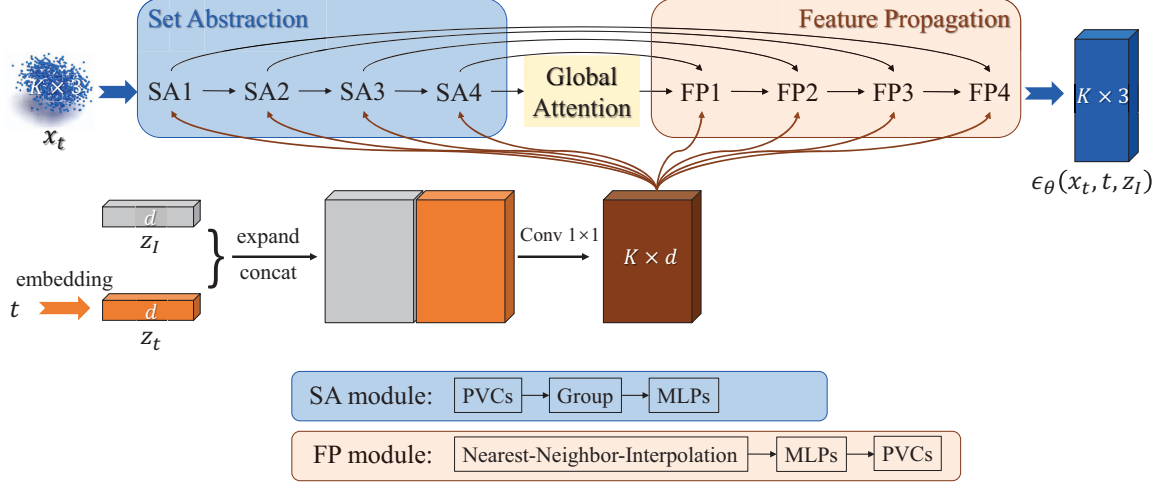


Figure 3. The architecture of the image conditional denoising network  $\theta$ .

### 3.2. Image-conditional Point Cloud Diffusion

A conditional diffusion model contains a forward diffusion process (see orange arrows in Fig. 1) and a denoising diffusion process (see blue arrows in Fig. 1). Given 3D point cloud  $x_0 \sim q(x_0)$  of a building with  $K$  points, the forward diffusion gradually adds noise by  $q(x_t|x_{t-1})$  using a sequence of increasing noise schedules  $\beta_t \in \{\beta_1, \dots, \beta_T\}$  where  $T$  is the final time step. Assuming  $\alpha_t := 1 - \beta_t$  and  $\bar{\alpha}_t := \prod_{s=1}^t \alpha_s$ , noisy point cloud  $x_t$  can be represented by,

$$x_t = \sqrt{\bar{\alpha}_t}x_0 + \sqrt{1 - \bar{\alpha}_t}\epsilon \quad (2)$$

where the time step  $t$  is sampled from discrete values  $\{1, \dots, T-1, T\}$ .

The denoising diffusion starts from a random noise tensor  $x_T \in \mathbb{R}^{K \times 3}$  sampled from a Gaussian prior distribution  $p(x_T) \sim \mathcal{N}(0, \mathcal{I})$ . The noise is progressively removed by  $q(x_{t-1}|x_t, z_I)$ , which can be approximated by  $p_\theta(x_{t-1}|x_t, z_I)$  parameterized with a denoising network  $\theta$ . For each denoising step,  $\theta$  takes as input noisy point cloud  $x_t \in \mathbb{R}^{K \times 3}$ , time step  $t$  and image embedding  $z_I \in \mathbb{R}^d$ . It outputs a noise  $\epsilon_\theta(x_t, t, z_I) \in \mathbb{R}^{K \times 3}$  and the target is a standard Gaussian noise  $\epsilon \sim \mathcal{N}(0, \mathcal{I})$ . The denoising loss  $\mathcal{L}_{eps}$ , which is commonly used as a simplified training objective in DDPMs, is denoted as,

$$\mathcal{L}_{eps} = \|\epsilon - \epsilon_\theta(x_t, t, z_I)\|^2 \quad (3)$$

As shown in Fig. 3,  $\theta$  is basically built on PVCNNs [18] with two major components, i.e., set abstraction (SA) modules and feature propagation (FP) modules. SA modules typically consist of point-voxel convolutions (PVConvs) and multi-layer perceptrons (MLPs); while FP modules typically consist of nearest neighbor interpolation, MLPs and PVConvs. By using both point-based and voxel-based branches, PVConvs can capture global and local structures of point clouds. Before sending input into the SA or FP

modules, the image embedding  $z_I$  is concatenated with the temporal embedding (denoted as  $z_t$ ) derived from  $t$ . Similar to [48], we employ sinusoidal positional embedding [37] to produce  $z_t$  that consists of pairs of sines and cosines with varying frequencies, followed by two linear layers with LeakyReLU activation function. Here,  $z_I$  and  $z_t$  have the same dimension  $d$ , and they are concatenated after being expanded to the size of  $K \times d$  where  $K$  is the number of points. We feed the fused feature map ( $K \times 2d$ ) through two convolution layers with LeakyReLU activation function, resulting in a feature map of size  $K \times d$  which are then concatenated with point features in SA and FP modules.

During training, we additionally introduce a weighted building footprint-based regularization strategy. Based on the predicted  $\epsilon_\theta$ , the desired  $x_0$  can be reconstructed by,

$$\hat{x}_0 = \frac{1}{\sqrt{\bar{\alpha}_t}}(x_t - \sqrt{1 - \bar{\alpha}_t}\epsilon_\theta(x_t, t, z_I)) \quad (4)$$

Considering the nature of buildings standing on the ground, the reconstructed  $\hat{x}_0$  and the target  $x_0$  are projected (denoted as  $proj$ ) to the ground, i.e.,  $z = 0$ , to obtain footprints  $proj(\hat{x}_0)$  and  $proj(x_0)$ . Then, we adopt a point-based metric  $\Omega$  to measure the similarity between two footprints. The regularization loss  $\mathcal{L}_{reg}$  can be formulated as,

$$\mathcal{L}_{reg} = \lambda(t)\Omega(proj(x_0), proj(\hat{x}_0)) \quad (5)$$

where  $\lambda$  is the weight depending on time step  $t$ . When  $t$  is close to  $T$ ,  $x_t$  could be noisy as  $x_T$  is standard Gaussian noise, so  $\Omega$  is assigned with lower weight  $\lambda$ . On the contrary,  $\Omega$  is assigned with higher weight  $\lambda$  when  $t$  is close to 1. Overall, the denoising network  $\theta$  is optimized by minimizing,

$$\mathcal{L}_\theta = \mathcal{L}_{eps} + \rho\mathcal{L}_{reg} \quad (6)$$

where regularization weight  $\rho$  balances these two terms.

The overall training process is described in the Algorithm 1. Similar to [23], we employ a classifier-free guidance strategy [13] which jointly learns a conditional and



an unconditional model. The conditioning image embedding  $z_I$  is randomly dropped, thus the conditional output  $\epsilon_\theta(x_t, t, z_I)$  is randomly replaced by the unconditional one  $\epsilon_\theta(x_t, t, \emptyset)$ .

---

**Algorithm 1** Training of conditional diffusion model

---

```

while not converge do
  sample  $x_0 \sim q(x_0)$ ,  $\epsilon \sim \mathcal{N}(0, \mathcal{I})$ 
  sample  $t \sim \mathcal{U}(\{1, \dots, T-1, T\})$ 
   $x_t = \sqrt{\bar{\alpha}_t}x_0 + \sqrt{1 - \bar{\alpha}_t}\epsilon$ 
   $\mathcal{L}_{eps} = \|\epsilon - \epsilon_\theta(x_t, t, z_I)\|^2$ 
   $\hat{x}_0 = \frac{1}{\sqrt{\bar{\alpha}_t}}(x_t - \sqrt{1 - \bar{\alpha}_t}\epsilon_\theta(x_t, t, z_I))$ 
   $\mathcal{L}_{reg} = \lambda(t)\Omega(\text{proj}(x_0), \text{proj}(\hat{x}_0))$ 
   $\mathcal{L}_\theta = \mathcal{L}_{eps} + \rho\mathcal{L}_{reg}$ 
  update model parameter  $\theta$  with  $\nabla_\theta \mathcal{L}_\theta$ 
end while

```

---

During sampling, the denoising diffusion begins from a Gaussian noise (i.e.,  $x_T \sim \mathcal{N}(0, \mathcal{I})$ ), and denoises with the output of network  $\theta$  step by step.  $x_{T-1}$  can be predicted by,

$$x_{t-1} = \frac{1}{\sqrt{\bar{\alpha}_t}}(x_t - \frac{1-\alpha_t}{\sqrt{1-\bar{\alpha}_t}}\epsilon_{guided}(x_t, t, z_I)) + \sigma_t \mathbf{z} \quad (7)$$

where  $t$  begins from  $T$  to 1, and  $\mathbf{z}$  is sampled from standard Gaussian distribution when  $t > 1$ . Leveraging a guidance scale  $\gamma$ , the guided noise output is,

$$\epsilon_{guided} := (1 + \gamma)\epsilon_\theta(x_t, t, z_I) - \gamma\epsilon_\theta(x_t, t, \emptyset) \quad (8)$$

Eventually, the desired  $x_0$  can be sampled when  $t = 1$ .

### 3.3. Point Cloud Upsampler Diffusion

For point cloud base diffusion, the key objective is generating low-resolution point clouds ( $K$  points) that could coarsely capture the overall structure of buildings. We train another diffusion model conditioned on the image embedding  $z_I$  derived by a frozen pre-trained auto-encoder and the low-resolution point cloud inferred by base diffusion. The goal lies in generating high-resolution point cloud with fine-grained structure. Thus, the second diffusion model is called upsampler diffusion.

Our upsampler diffusion leverages similar architecture as our base diffusion model. Assuming the desired point cloud  $x_0 \in \mathbb{R}^{N \times 3}$  consists of  $N$  points ( $N > K$ ), we randomly sample a noise tensor  $x_T \in \mathbb{R}^{N \times 3}$  from a Gaussian prior distribution  $p(x_T) \sim \mathcal{N}(0, \mathcal{I})$ . During training, the denoising network  $\theta$  takes as input  $K$  points (i.e., low-resolution point cloud) and  $(N - K)$  points sampled from noisy  $x_t$ , time step  $t$  and conditioning image embedding  $z_I$ . At each step, the first  $K$  of  $N$  points sampled by  $\theta$  is replaced by the low-resolution point cloud, and the updated  $N$  points are used as input in the next time step. In short, to arrive at  $N$  points, our upsampler conditions on  $K$  points and denoises the rest  $(N - K)$  points.

## 4. Experiments

### 4.1. Datasets

To validate the performance of the proposed method, we create two datasets, **BuildingNet-SVI** and **BuildingNL3D**, providing thousands of image-3D pairs of buildings.

**BuildingNet-SVI.** Built on BuildingNet [33] dataset that covers a variety of synthetic 3D building models (e.g., churches, houses, and office buildings), we collect the corresponding single-view RGB synthetic images of buildings and obtained 406 image-3D pairs after quality checks w.r.t. completeness and consistency. Besides, we manually annotate the foreground object (i.e., individual building) at the pixel level, and crop each image centered around the building. Each building point cloud has 100,000 uniformly distributed 3D points. We follow the official splitting rules from BuildingNet [33], thus 321 and 85 image-3D pairs are used for training and testing, respectively.

**BuildingNL3D.** We collect 2,769 pairs of aerial RGB images and Airborne Laser Scanning (ALS) point clouds of buildings which are located in the urban area of a city in the Netherlands. Unlike synthetic images that have individual buildings and relatively clean backgrounds, aerial images usually face more challenges such as multiple buildings appearing in a single image. Hence, the buildings are manually labeled so that only one building of interest appears in each image. Raw ALS point clouds are normalized from their real geographic coordinates to  $xyz$  coordinates within the range  $[-1, 1]^3$ . The dataset is divided into 2,171 image-3D training pairs and 598 test pairs according to tile-based splitting rules where buildings in the training and test sets are not repeated.

### 4.2. Evaluation Metrics

We adopt Chamfer distance (CD) [8], Earth mover’s distance (EMD) [32] and F1-Score [16] to evaluate the pairwise similarity between a generated building and its reference building. Specifically, CD and EMD are multiplied by  $10^2$ , and the threshold  $\tau$  of F1 is set as 0.001. The point clouds are normalized into  $[-1, 1]^3$  before calculating with these metrics. The visualization of 3D point clouds is achieved by using Mitsuba Renderer [22].

### 4.3. Implementation Details

The models are implemented with PyTorch [25] on a NVIDIA A40 GPU with 45GB memory. The batch size is set as 8. The images are resized into  $1024 \times 1024$  pixels, i.e.,  $H=W=1024$ . Each 3D point cloud has normalized 100,000 points for representing the shape of the building.

**Image auto-encoder details.** We adopt augmentation techniques including image rotation with an angle of  $90^\circ$  and color shifting in the Hue-Saturation-Value (HSV) space

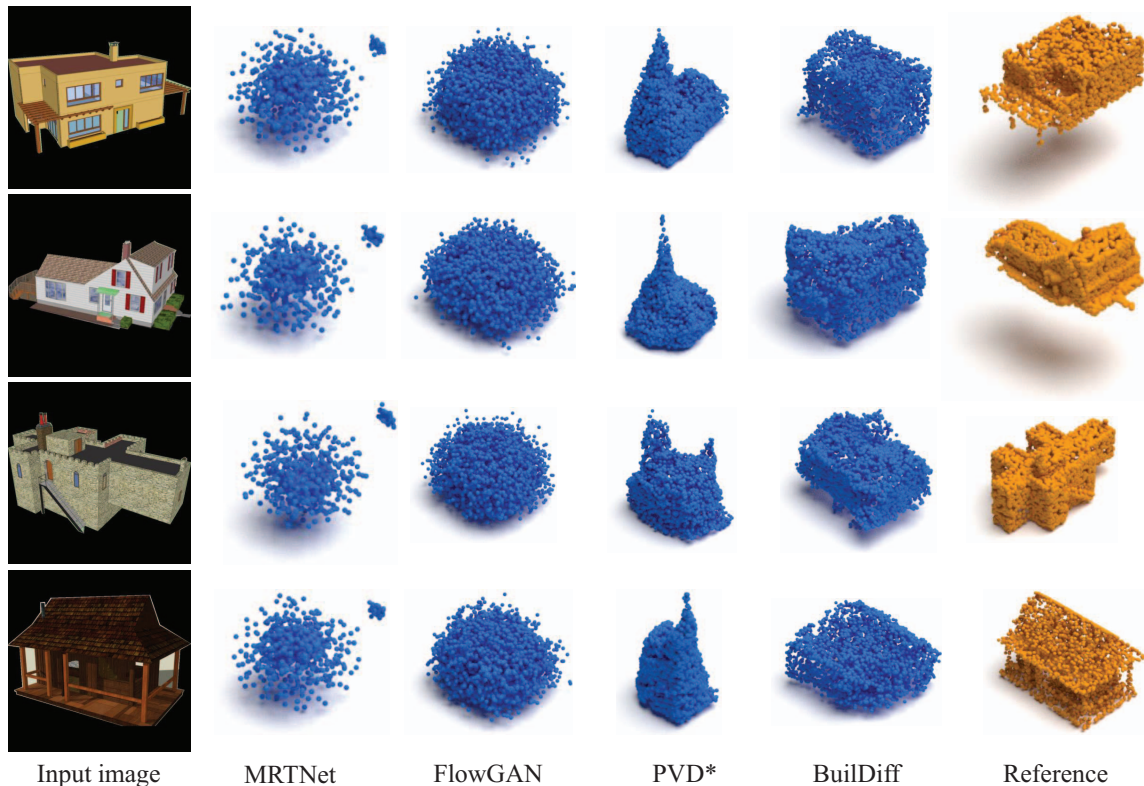


Figure 4. Qualitative comparison of different 3D point cloud generation models on the BuildingNet-SVI dataset. The reference and predicted point clouds are colored by orange and blue, respectively.

ranging from -255 to 255. These are based on the consideration that buildings may appear upside down in the images and building images could exhibit large color variations. The dimensionality  $d$  of image embedding  $z_I$  is 128. We train the auto-encoder for 30 epochs and use the Adam optimizer with a learning rate 0.0002. The image reconstruction loss  $\mathcal{L}_{rec}$  and the embedding consistency loss  $\mathcal{L}_{con}$  are achieved by mean squared error (MSE). During training the following image-conditional diffusion models, we freeze the pre-trained autoencoder to derive  $z_I$ .

**Base diffusion details.** Given 100,000 points per building, we randomly sample  $K=1024$  points (representing low-resolution point clouds) for training and testing. We set  $\beta_0=0.0001$ ,  $\beta_T=0.02$  and linearly interpolate other  $\beta_s$ . Similar to existing works [48, 46], the total number of time steps is set as  $T=1000$  for the base diffusion. The dimensionality of temporal embedding is  $d=128$ . Chamfer distance [8] is employed as the point-based distance metrics  $\Omega$  to measure the similarity between two projections. The regularization weight  $\rho$  is set as 0.001. Regarding the classifier-free guidance strategy, we employ drop probability 0.1 for the training phase, and guidance scale  $\gamma=4$  for the sampling phase. The diffusion model is trained for 700 epochs and optimized by Adam with a learning rate 0.0002. Specifically,  $\lambda(t)$  is defined as,

$$\lambda(t) = \begin{cases} 1, & t = 1 \\ 0.75, & 1 < t \leq \frac{1}{4}T \\ 0.50, & \frac{1}{4}T < t \leq \frac{1}{2}T \\ 0.25, & \frac{1}{2}T < t \leq \frac{3}{4}T \\ 0, & \frac{3}{4}T < t \leq T \end{cases}$$

**Upsampler diffusion details.** For the training and testing phases, we randomly sample  $N=4096$  points per building. The denoising network of the upsampler is trained for 200 epochs and the total number of time steps  $T$  is set as 500.

#### 4.4. Quantitative and Qualitative Results

To demonstrate the performance of the proposed method, BuilDiff is compared with several 3D point cloud generation approaches based on deep generative models. The comparison methods are MRTNet [9] which leverages VAEs for single image 3D reconstruction, FlowGAN [39] which reconstructs 3D point cloud from a single image by combining the advantages of NFs and GANs, and PVD [48] which use diffusion models for 3D point cloud generation. As PVD is originally designed for unconditional point cloud synthesis, we extend its framework by introducing a global embedding extracted from images as an auxiliary input of denoising diffusion models, thus we note it as PVD\*. Our method differs from PVD\* in three main aspects: (1) an

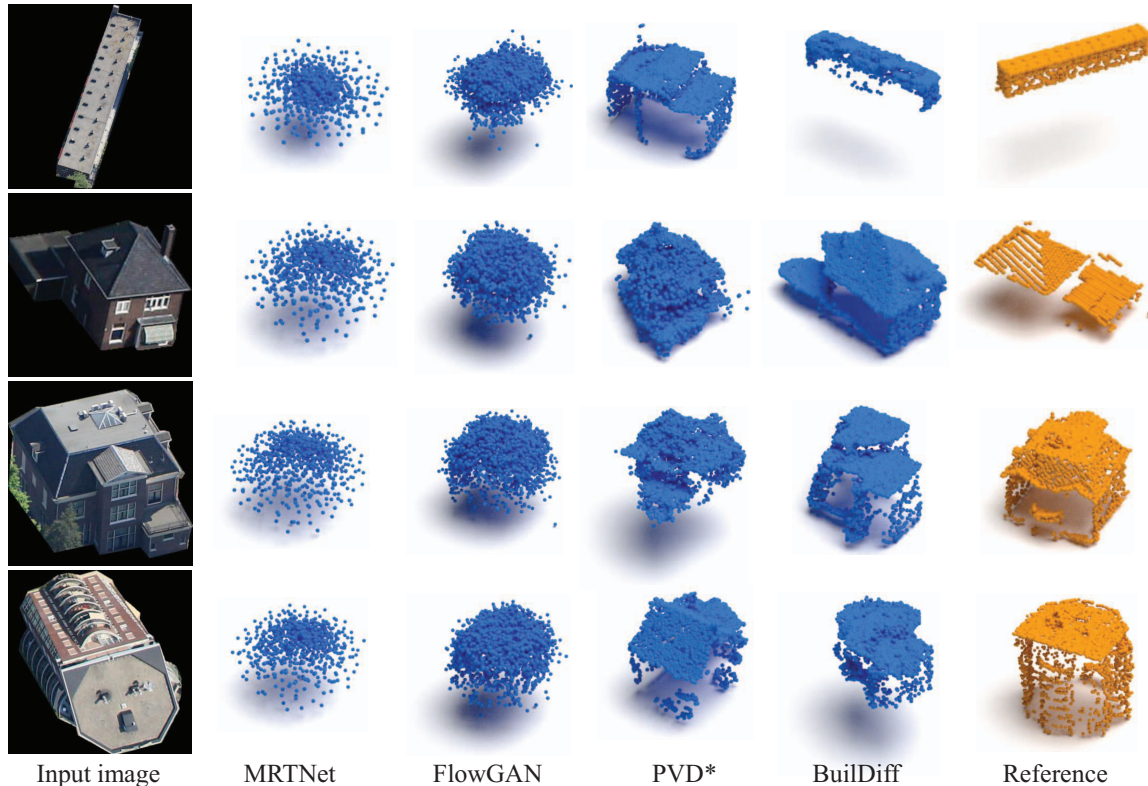


Figure 5. Qualitative comparison of different 3D point cloud generation models on the BuildingNL3D dataset. The reference and predicted point clouds are colored by orange and blue, respectively.

auto-encoder is pre-trained to extract more representative conditions  $z_I$  to avoid ambiguous guidance; (2) a weighted footprint-based regularization strategy is introduced to the training of denoising network  $\theta$ ; (3) a point cloud upsampler is used to produce high-resolution point clouds of buildings.

The implementation of the comparison methods is based on their officially released code. All the models are trained and evaluated on the BuildingNet-SVI dataset and BuildingNL3D dataset, the quantitative results of which are shown in Table 1 and Table 2. Here, we sample 4,096 points per shape to ensure a fair comparison.

Regarding the synthetic data, it can be seen from Table 1 that BuilDiff achieves the best EMD and F1 as well as the second-best CD. Unlike common 3D objects (e.g., ShapeNet) that other methods focus on, buildings have more complex structures that are not symmetrical. Such natures make it difficult to learn the shape of buildings in canonical object space. Correspondingly, directly using VAEs or GANs to constrain the network output may prevent the network from learning valuable information.

In Table 2, BuilDiff reports the lowest EMD, second-highest F1, and comparable CD with the baseline method PVD\*. Regarding other methods, MRTNet directly employs CD loss for training, it obtains low CD while performs poorly on other metrics. FlowGAN obtains the high-

Methods	CD↓	EMD↓	F1↑
MRTNet [9]	6.11	49.07	6.89
FlowGAN [39]	<b>2.00</b>	21.21	<u>21.17</u>
PVD [48] *	6.18	<u>16.08</u>	20.02
BuilDiff	<u>3.14</u>	<b>10.84</b>	<b>21.41</b>

Table 1. Quantitative results on BuildingNet-SVI dataset. The best results are indicated in **bold** and the second best are marked with underlines.

Methods	CD↓	EMD↓	F1↑
MRTNet [9]	<u>2.84</u>	44.06	5.18
FlowGAN [39]	<b>2.33</b>	24.06	<b>22.26</b>
PVD [48] *	5.69	<u>14.74</u>	13.01
BuilDiff	3.81	<b>10.43</b>	<u>22.08</u>

Table 2. Quantitative results on BuildingNL3D dataset. The best results are indicated in **bold** and the second best are marked with underlines.

est F1 but inferior EMD, demonstrating its preference for generating uniform global distributed points while ignoring local details. As EMD is stricter with local quality [41], our results show robust performance in terms of both global shapes and local structures.

The qualitative results of several generated 3D buildings are shown in Fig. 4 and Fig. 5. MRTNet and FlowGAN yield unsatisfactory results, which demonstrates their poor



pts	pre-train AE	regularization	upsampler	CD↓	EMD↓	F1↑
1024	×	×	-	4.523	12.244	14.042
1024	✓	×	-	4.759	12.847	<b>14.631</b>
1024	✓	✓	-	<b>3.556</b>	11.747	13.889
1024	✓	✓†	-	3.827	<b>10.682</b>	13.376
4096	×	×	-	6.181	16.075	20.022
4096	✓	✓†	-	3.678	15.127	21.128
4096	✓	✓†	✓	<b>3.142</b>	<b>10.840</b>	<b>21.413</b>

Table 3. Ablation studies on the BuildingNet-SVI dataset. The best results are shown in **bold**.

pts	pre-train AE	regularization	upsampler	CD↓	EMD↓	F1↑
1024	×	×	-	11.109	24.027	5.180
1024	✓	×	-	6.406	16.770	8.531
1024	✓	✓	-	<b>5.680</b>	15.813	8.928
1024	✓	✓†	-	5.766	<b>15.625</b>	<b>9.601</b>
4096	×	×	-	5.691	14.735	13.012
4096	✓	✓†	-	4.046	13.373	15.227
4096	✓	✓†	✓	<b>3.810</b>	<b>10.427</b>	<b>22.081</b>

Table 4. Ablation studies on the BuildingNL3D dataset. The best results are shown in **bold**.

generalization ability in 3D building reconstruction. In contrast, diffusion-based methods perform better in capturing the shape of buildings. Due to the simple conditioning strategy used by PVD\*, the generated 3D buildings are inconsistent with the input condition images. By the proposed conditioning schemes, BuilDiff decreases the diversity of diffusion models and increases the quality of generated buildings. Furthermore, it can be observed from the last row of Fig. 5 that our method demonstrates the robustness even when buildings appear upside down in the input image.

#### 4.5. Ablation Study

We verify the effectiveness of the proposed components, i.e., pre-training image auto-encoder, weighted footprint-based regularization, and point cloud upsampler. The comparison results are shown in Table 3 and Table 4. The first four rows are tested on 1,024 points in which our upsampler is not used. We sequentially compare the effects of pre-training image auto-encoder, footprint-based regularization, and weighted footprint-based regularization that is denoted with †. The last three rows are tested on 4,096 points.

First, it can be observed that our image auto-encoder greatly increases the performance on real data from BuildingNL3D, which means that it could alleviate the domain gap between real-world building images and common images learned by image classification networks. Such expressive condition benefits from multi-scale features and latent consistency performed by image augmentation. Second, the model is further improved by introducing building footprint-based regularization in terms of CD and EMD. The weighted regularization guides the network focus on later denoising steps. Finally, it can be seen from the last two rows that our BuilDiff performs better than the baseline (i.e., pre-train AE + regularization, without upsampling), demonstrating the effect of our upsampler diffusion. Visualizations can be found in Fig. 6. Baseline method sometimes

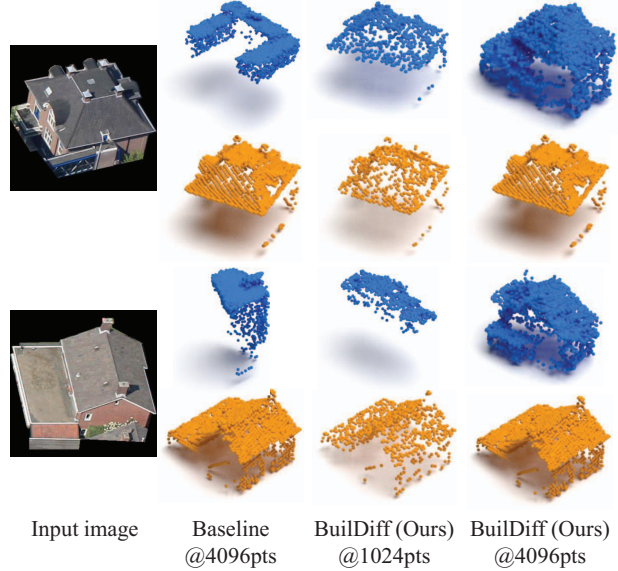


Figure 6. The illustration of the effect of our upsampler diffusion. The reference and predicted point clouds are colored by orange and blue, respectively.

has difficulty focusing on the basic structure of the buildings due to too many points to consider, leading to 3D buildings that are completely unrelated to the input images. In contrast, BuilDiff employs a base diffusion model that coarsely identifies the overall structure of buildings, and then uses an upsampler diffusion to derive high-resolution point clouds conditioned on low-resolution point clouds and input images. Our method can even generate more complete buildings than the reference point clouds, for example, the final results in the last column contain realistic building facades.

## 5. Conclusions

In this paper, we present a diffusion-based method **BuilDiff** for generating 3D point clouds of buildings from single general-view images. To control the diffusion models generating 3D shape consistent with the input image, an image embedding is derived by pre-training a CNN-based image auto-encoder, which extracts multi-scale features of buildings and constrains the latent consistency using augmentation. Then, a conditional denoising diffusion network which takes as input the image embedding and learns to gradually remove the noise from Gaussian noise distribution assisted by weighted building footprint-based regularization. A point cloud upsampler diffusion is finally leveraged to produce high-resolution point clouds conditioned on the low-resolution point clouds sampled from the base diffusion. The effectiveness of the proposed method has been demonstrated by the experimental results on both synthetic and real-world scenarios. We believe our work could bridge the rapidly developing generative modeling techniques and the urgent problem of 3D building generation.



## References

- [1] Panos Achlioptas, Olga Diamanti, Ioannis Mitliagkas, and Leonidas Guibas. Learning representations and generative models for 3d point clouds. In *International conference on machine learning*, pages 40–49. PMLR, 2018. 2
- [2] Angel X Chang, Thomas Funkhouser, Leonidas Guibas, Pat Hanrahan, Qixing Huang, Zimo Li, Silvio Savarese, Manolis Savva, Shuran Song, Hao Su, et al. Shapenet: An information-rich 3d model repository. *arXiv preprint arXiv:1512.03012*, 2015. 1, 2
- [3] Christopher B Choy, Danfei Xu, JunYoung Gwak, Kevin Chen, and Silvio Savarese. 3d-r2n2: A unified approach for single and multi-view 3d object reconstruction. In *Computer Vision—ECCV 2016: 14th European Conference, Amsterdam, The Netherlands, October 11–14, 2016, Proceedings, Part VIII 14*, pages 628–644. Springer, 2016. 2
- [4] Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, and Li Fei-Fei. Imagenet: A large-scale hierarchical image database. In *2009 IEEE conference on computer vision and pattern recognition*, pages 248–255. Ieee, 2009. 3
- [5] Prafulla Dhariwal and Alexander Nichol. Diffusion models beat gans on image synthesis. *Advances in Neural Information Processing Systems*, 34:8780–8794, 2021. 3
- [6] Liuyun Duan and Florent Lafarge. Towards large-scale city reconstruction from satellites. In *Computer Vision—ECCV 2016: 14th European Conference, Amsterdam, The Netherlands, October 11–14, 2016, Proceedings, Part V 14*, pages 89–104. Springer, 2016. 1
- [7] Shivam Duggal and Deepak Pathak. Topologically-aware deformation fields for single-view 3d reconstruction. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 1536–1546, 2022. 2
- [8] Haoqiang Fan, Hao Su, and Leonidas J Guibas. A point set generation network for 3d object reconstruction from a single image. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 605–613, 2017. 1, 2, 5, 6
- [9] Matheus Gadelha, Rui Wang, and Subhransu Maji. Multiresolution tree networks for 3d point cloud processing. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pages 103–118, 2018. 1, 2, 6, 7
- [10] Ian J Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, and Yoshua Bengio. Generative adversarial networks. *arXiv preprint arXiv:1406.2661*, 2014. 1, 2
- [11] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 770–778, 2016. 3
- [12] Jonathan Ho, Ajay Jain, and Pieter Abbeel. Denoising diffusion probabilistic models. *Advances in Neural Information Processing Systems*, 33:6840–6851, 2020. 1, 2
- [13] Jonathan Ho and Tim Salimans. Classifier-free diffusion guidance. *arXiv preprint arXiv:2207.12598*, 2022. 3, 4
- [14] Qingyong Hu, Bo Yang, Sheikh Khalid, Wen Xiao, Niki Trigoni, and Andrew Markham. Towards semantic segmentation of urban-scale 3d point clouds: A dataset, benchmarks and challenges. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 4977–4987, 2021. 1
- [15] Diederik P Kingma and Max Welling. Auto-encoding variational bayes. *arXiv preprint arXiv:1312.6114*, 2013. 1, 2
- [16] Arno Knapitsch, Jaesik Park, Qian-Yi Zhou, and Vladlen Koltun. Tanks and temples: Benchmarking large-scale scene reconstruction. *ACM Transactions on Graphics (ToG)*, 36(4):1–13, 2017. 5
- [17] Weijia Li, Lingxuan Meng, Jinwang Wang, Conghui He, Gui-Song Xia, and Dahua Lin. 3d building reconstruction from monocular remote sensing images. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 12548–12557, 2021. 1, 2
- [18] Zhijian Liu, Haotian Tang, Yujun Lin, and Song Han. Point-voxel cnn for efficient 3d deep learning. *Advances in Neural Information Processing Systems*, 32, 2019. 4
- [19] Shitong Luo and Wei Hu. Diffusion probabilistic models for 3d point cloud generation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 2837–2845, 2021. 2
- [20] Jisan Mahmud, True Price, Akash Bapat, and Jan-Michael Frahm. Boundary-aware 3d building reconstruction from a single overhead image. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 441–451, 2020. 1, 2
- [21] Luke Melas-Kyriazi, Christian Rupprecht, and Andrea Vedaldi. *pc2*: Projection-conditioned point cloud diffusion for single-image 3d reconstruction. *arXiv preprint arXiv:2302.10668*, 2023. 2
- [22] Mitsuba. Physically based renderer. <http://www.mitsuba-renderer.org/>. [Online; accessed 04-March-2023]. 5
- [23] Alex Nichol, Heewoo Jun, Prafulla Dhariwal, Pamela Mishkin, and Mark Chen. Point-e: A system for generating 3d point clouds from complex prompts. *arXiv preprint arXiv:2212.08751*, 2022. 2, 4
- [24] Hui En Pang and Filip Biljecki. 3d building reconstruction from single street view images using deep learning. *International Journal of Applied Earth Observation and Geoinformation*, 112:102859, 2022. 1, 2
- [25] Adam Paszke, Sam Gross, Francisco Massa, Adam Lerer, James Bradbury, Gregory Chanan, Trevor Killeen, Zeming Lin, Natalia Gimelshein, Luca Antiga, et al. Pytorch: An imperative style, high-performance deep learning library. *Advances in neural information processing systems*, 32, 2019. 5
- [26] Ravi Peters, Balázs Dukai, Stelios Vitalis, Jordi van Liempt, and Jantien Stoter. Automated 3d reconstruction of lod2 and lod1 models for all 10 million buildings of the netherlands. *Photogrammetric Engineering & Remote Sensing*, 88(3):165–170, 2022. 1
- [27] Aditya Ramesh, Prafulla Dhariwal, Alex Nichol, Casey Chu, and Mark Chen. Hierarchical text-conditional image generation with clip latents. *arXiv preprint arXiv:2204.06125*, 2022. 2
- [28] Maria E Presa Reyes and Shu-Ching Chen. A 3d virtual environment for storm surge flooding animation. In 2017

- IEEE third international conference on multimedia big data (BigMM)*, pages 244–245. IEEE, 2017. 1
- [29] Yoones Rezaei and Stephen Lee. sat2pc: Estimating point cloud of building roofs from 2d satellite images. *arXiv preprint arXiv:2205.12464*, 2022. 2
- [30] Danilo Rezende and Shakir Mohamed. Variational inference with normalizing flows. In *International conference on machine learning*, pages 1530–1538. PMLR, 2015. 1, 2
- [31] Robin Rombach, Andreas Blattmann, Dominik Lorenz, Patrick Esser, and Björn Ommer. High-resolution image synthesis with latent diffusion models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 10684–10695, 2022. 2
- [32] Yossi Rubner, Carlo Tomasi, and Leonidas J Guibas. The earth mover’s distance as a metric for image retrieval. *International journal of computer vision*, 40(2):99, 2000. 5
- [33] Pratheba Selvaraju, Mohamed Nabail, Marios Loizou, Maria Maslioukova, Melinos Averkiou, Andreas Andreou, Siddhartha Chaudhuri, and Evangelos Kalogerakis. Buildingnet: Learning to label 3d buildings. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 10397–10407, 2021. 2, 5
- [34] Narushige Shiode. 3d urban models: Recent developments in the digital modelling of urban environments in three-dimensions. *GeoJournal*, 52:263–269, 2000. 1
- [35] Shivangi Srivastava, Michele Volpi, and Devis Tuia. Joint height estimation and semantic labeling of monocular aerial images with cnns. In *2017 IEEE International Geoscience and Remote Sensing Symposium (IGARSS)*, pages 5173–5176. IEEE, 2017. 1, 2
- [36] Akihiko Torii, Michal Havlena, and Tomáš Pajdla. From google street view to 3d city models. In *2009 IEEE 12th international conference on computer vision workshops, ICCV Workshops*, pages 2188–2195. IEEE, 2009. 1
- [37] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. Attention is all you need. *Advances in neural information processing systems*, 30, 2017. 4
- [38] Sa Wang, Zhengli Mao, Changhai Zeng, Huili Gong, Shanshan Li, and Beibei Chen. A new method of virtual reality based on unity3d. In *2010 18th international conference on Geoinformatics*, pages 1–5. IEEE, 2010. 1
- [39] Yao Wei, George Vosselman, and Michael Ying Yang. Flow-based gan for 3d point cloud generation from a single image. *33rd British Machine Vision Conference 2022, BMVC 2022, London, UK, November 21-24, 2022*, 2022. 1, 2, 6, 7
- [40] Xin Wen, Junsheng Zhou, Yu-Shen Liu, Hua Su, Zhen Dong, and Zhizhong Han. 3d shape reconstruction from 2d images with disentangled attribute flow. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 3803–3813, 2022. 2
- [41] Tong Wu, Liang Pan, Junzhe Zhang, Tai Wang, Ziwei Liu, and Dahua Lin. Density-aware chamfer distance as a comprehensive metric for point cloud completion. *arXiv preprint arXiv:2111.12702*, 2021. 7
- [42] Qiangeng Xu, Weiyue Wang, Duygu Ceylan, Radomir Mech, and Ulrich Neumann. Disn: Deep implicit surface network for high-quality single-view 3d reconstruction. *Advances in neural information processing systems*, 32, 2019. 2
- [43] Guandao Yang, Xun Huang, Zekun Hao, Ming-Yu Liu, Serge Belongie, and Bharath Hariharan. Pointflow: 3d point cloud generation with continuous normalizing flows. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 4541–4550, 2019. 2
- [44] Dawen Yu, Shunping Ji, Jin Liu, and Shiqing Wei. Automatic 3d building reconstruction from multi-view aerial images with deep learning. *ISPRS Journal of Photogrammetry and Remote Sensing*, 171:155–170, 2021. 1
- [45] Fisher Yu and Vladlen Koltun. Multi-scale context aggregation by dilated convolutions. *arXiv preprint arXiv:1511.07122*, 2015. 3
- [46] Xiaohui Zeng, Arash Vahdat, Francis Williams, Zan Gojcic, Or Litany, Sanja Fidler, and Karsten Kreis. Lion: Latent point diffusion models for 3d shape generation. *arXiv preprint arXiv:2210.06978*, 2022. 2, 6
- [47] Chunhui Zhao, Chi Zhang, Yiming Yan, and Nan Su. Shape reconstruction of object-level building from single image based on implicit representation network. *IEEE Geoscience and Remote Sensing Letters*, 19:1–5, 2021. 1, 2
- [48] Linqi Zhou, Yilun Du, and Jiajun Wu. 3d shape generation and completion through point-voxel diffusion. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 5826–5835, 2021. 2, 4, 6, 7