Computer Aided Geometric Design ••• (••••) ••••••

Contents lists available at ScienceDirect

Computer Aided Geometric Design

www.elsevier.com/locate/cagd

Learning geometry-aware joint latent space for simultaneous multimodal shape generation

Artem Komarichev, Jing Hua, Zichun Zhong *

Department of Computer Science, Wayne State University, Detroit, MI 48202, USA

ARTICLE INFO

Article history:
Available online xxxx

Keywords:
Joint latent space (mixer)
Multimodal/cross-modality shape generation/interpolation
Geometry-awareness

ABSTRACT

The real-world objects in our physical environment present with diverse information and multimodal features, including 3D shapes (geometry and topology) and 2D images (appearance and semantics), etc. How to effectively represent and correlate them in a unified way is still very challenging due to different modalities and representations. In this paper, we present a novel method to learn a unified and effective latent space for a joint representation and simultaneous generation of 3D point clouds and 2D images. We propose a new geometry-aware autoencoder for 3D shapes with a full-resolution shape feature extractor and a multi-resolution geometric feature extractor at different scales, which can enhance the geometric variability and scalability of the latent representation. Then, the proposed mixer, i.e., a joint latent space, can synergically integrate and complement the encoded features from 3D geometry and 2D contents through our intermodality feature mapping and intramodality feature consistency design. It is noted that our joint latent space can simultaneously generate multimodal representations and correlations with high-quality, high-fidelity, and high cross-modality similarity, which the traditional single-modal methods cannot handle. The extensive experiments demonstrate that our approach outperforms the state-of-the-art methods in shape auto-encoding as well as simultaneous multimodal (SMM) shape and color image generation and interpolation, etc. Furthermore, our joint-learning of 2D and 3D facets of a shape for the novel SMM semantic-aware generation task can enhance the capability of the corresponding single-modality and single-tasking to the next level.

1. Introduction

3D objects can be described by different modal representations, such as 3D shapes, 2D view images, and descriptive texts. 3D object representation and generation are the most fundamental problems in computer graphics, visualization, and computer vision with a wide range of applications in modeling, rendering, vision, robotics, medicine, augmented reality and virtual reality, etc. To date, deep learning Goodfellow et al. (2016) based data-driven object analysis has become effective and successful in each individual domain, such as 3D shape-based object reconstruction, classification, segmentation, and generation Wu et al. (2016); Qi et al. (2017a,b); Masci et al. (2015); Boscaini et al. (2016); Kostrikov et al. (2018); Hanocka et al. (2019); Yang et al. (2018); Achlioptas et al. (2018); Li et al. (2018a); 2D view image-based object reconstruction, classification, and segmentation Su et al. (2015); Qi et al. (2016); Kalogerakis et al. (2017); Huang et al. (2017); Lun et al.

* Corresponding author.
E-mail addresses: artem.komarichev@wayne.edu (A. Komarichev), jinghua@wayne.edu (J. Hua), zichunzhong@wayne.edu (Z. Zhong).

https://doi.org/10.1016/j.cagd.2022.102076
0167-8396/

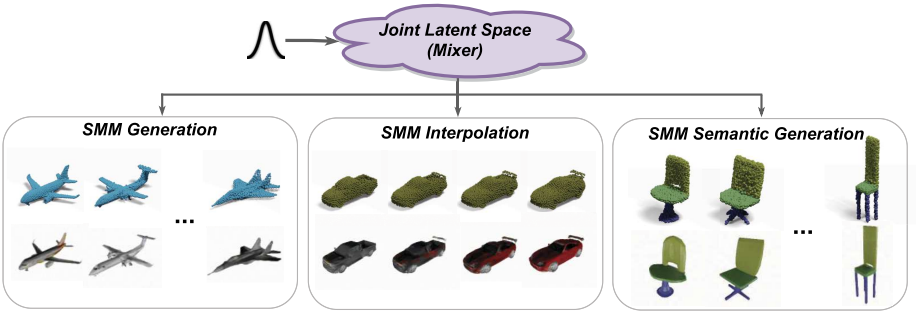


Fig. 1. The proposed method to learn the novel joint latent space for simultaneous multimodal (SMM) generation/interpolation, and SMM semantic-aware generation (i.e., generation with part-level semantic annotations) of new objects. Here we show some samples from our results (top row: point clouds, bottom row: images).

(2017); Wang and Gupta (2016); Fan et al. (2017); etc. However, this process is easily overwhelmed by tons of involved features and objects, especially when considering multimodality information. From our observations, it is very difficult to express an object with single modality, due to the complexity and complementarity of the available modalities. A theoretical way to solve this problem is to map the multimodal representations and features onto a universal high-dimensional (high-d) encoding space, where the representation and computation are under the same metric. In this case, the *joint latent space* is a viable scheme to investigate the cross-modality representations of objects. While there has been some prior work on image-text embedding Gong et al. (2014a,b); Kiros et al. (2014); Klein et al. (2015); Reed et al. (2016); Wang et al. (2016); Peng et al. (2016), shape-image embedding via distance metrics Li et al. (2015), and 3D volumetric image-text embedding Chen et al. (2018), Shape Unicode Muralikrishnan et al. (2019), Cross-Modal Deep Variational Spurr et al. (2018), there exist limited works on constructing a high-quality joint latent space for effective and universal joint 3D shape and 2D image representation and generation, due to its complexity and difficulty.

In this work, we propose a new framework to build a mixed but unified representation and generation for multimodal data (e.g., 3D point clouds and 2D images) through the geometry-aware data-driven joint latent space. It is noted that representing the multimodality in the high-dimensional shape-image space and mapping it into a low-dimensional joint latent space (i.e., a compact and effective representation based on the Minimum Description Length principle Hinton and Zemel (1994); Grünwald (2007)) is notoriously difficult since essentially the shape and image latent spaces are totally different through being encoded from different types of data as well as different types of neural networks. In order to address this challenge, we develop new approaches (overview is shown in Fig. 1) and the key *contributions* are:

- We propose a new *joint latent space – mixer* approach for learning the high-quality 3D object multimodal representation and generative models. It provides an intrinsic and unified representation and correlation for cross-modality data by synergically integrating and complementing the encoded features from 3D geometry and 2D contents via the proposed intermodality feature mapping and intramodality feature consistency design;
- We design a new *geometry-aware autoencoder* for 3D shapes through the developed full-resolution shape feature extractor and multi-resolution geometric feature extractor, which can enhance the geometric variability and scalability of the joint latent representation;
- Our network model presents novel performance on shape (point cloud) auto-encoding by outperforming the several state-of-the-art methods (e.g., AE-EMD Achlioptas et al., 2018, AE-CD Achlioptas et al., 2018, LOGAN-AE Yin et al., 2019, and ShapeGF-AE Cai et al., 2020). Also, our method presents several novel 3D shape tasks, such as *simultaneous multimodal (SMM)* shape and color image generation and interpolation, where we outperform the state-of-the-arts (e.g., Shape Unicode Muralikrishnan et al., 2019, multimodality feature concatenation, etc.) and SMM semantic-aware generation (i.e., generation with part-level semantic annotations on shape and image), which can enhance the capability of the corresponding single-modality and single-tasking to the next level.

To our knowledge, there are many real-world applications where our proposed joint latent space could be applied. In this paper, we are focusing on shape-image joint generation tasks for creating and augmenting new multimodality datasets. Besides that, there are some potential follow-up applications and extensions, such as joint learning for shape completion and image inpainting, etc.

2. Related work

Due to the scope of our work, we focus only on recent related deep learning methods on multimodal/joint embedding, and point cloud auto-encoding and generation.

Multimodal/Joint Embedding. In order to learn and explore the multimodal/joint representation from a data-driven perspective, the embedding-based methods are preferred to be used. In recent years, multimodal embeddings have been used

in computer vision to establish image-text relationships Jason et al. (2010, 2011); Gong et al. (2014a,b); Kiros et al. (2014); Klein et al. (2015); Reed et al. (2016); Wang et al. (2016); Peng et al. (2016); Wu et al. (2017), but they are limitedly applied to 3D shapes in computer graphics and computer vision domains. Graphics related work Herzog et al. (2015) starts by creating a common embedding space for 3D models and keywords, and then, adds images and sketches. Recently, Li et al. (2015) proposed a joint embedding space with convolutional neural network populated by both 3D shapes and 2D images of objects. This work is capable of shape and image retrieval via such space, but cannot be feasible to generate new multimodal data of 3D objects. Hu et al. (2015) proposed a learning-based 3D object template method by quantizing geometry and appearance spaces with an And-Or Tree representation. Both of their shape-image spaces are still designed and applied in an image domain without considering 3D shape geometric features to build a geometric embedding. Girdhar et al. (2016) introduced a novel TL-embedding network of learning an embedding space for 3D objects with predictive and generative capabilities from 2D images. Mandikal et al. (2018) proposed a 3D reconstruction method from single view image through learning corresponding latent embedding. Both these methods are lacking of simultaneously generating different modalities with high fidelity. Chen et al. (2018) presented a method for generating colored 3D voxelized shapes from natural language via the 3D volumetric image-text embedding. Muralikrishnan et al. (2019) proposed a unified code for 3D shapes between different representations. The main drawback of this approach is a need of translating between (during training) all possible encoder-decoder pairs in every modality (with a high computational complexity and training time) and not designed for high-quality joint generation tasks; while our method only needs a mixer (i.e., a low-dimensional joint latent space) that unifies different representations effectively. We have provided the detailed comparison experiments in Sec. 4.2. Spurr et al. (2018) proposed a method that maps each modality of the same object in a latent space to minimize the distance between them. However, they did not develop a joint latent space like us. Therefore, their mapping is not effective as shown in their visualization results, especially from 3D to 2D, such as the generated RGB images are of low quality and blurry. This is because the generated latent vector of 3D model cannot well match/represent the latent vector of 2D RGB image in the same latent space. Also, it is worth mentioning that they focused mainly on mapping 2D images to 3D shapes, where we focus on generating both modalities simultaneously. Schwarz et al. (2021) mainly focused on image synthesis via the generative radiance field, which does not work on the 3D shape generation simultaneously. They provided some simple 3D reconstruction/consistency results by using a postprocessing multi-view stereo method and their 3D shape quality does not look good as ours.

Point Cloud Auto-Encoding and Generation. In recent years, researchers have focused on developing more advanced techniques for 3D computer vision and geometry analysis using deep learning methods Xiao et al. (2020). Prior research has explored deep architectures for feature learning over 3D point clouds on classification and segmentation Qi et al. (2017a,b); Tatarchenko et al. (2018); Li et al. (2018b); Komarichev et al. (2019); Thomas et al. (2019); Gao et al. (2020); Xu et al. (2021), on single-view 3D shape reconstruction Fan et al. (2017); Tatarchenko et al. (2019); Li et al. (2020), and on point cloud upsampling/completion Li et al. (2019); Liu et al. (2020); Huang et al. (2020), etc. Recently, the shape auto-encoding and generative modeling Fan et al. (2017); Li et al. (2018a); Achlioptas et al. (2018); Valsesia et al. (2018); Shu et al. (2019); Yang et al. (2019); Yin et al. (2019); Cai et al. (2020) for 3D point clouds gain more attention in the computer graphics and vision domains. All the generation methods focus on generating 3D point clouds directly in the raw space or learning the distribution of shapes in the latent space of the pre-trained autoencoder. Fan et al. (2017) developed a point set generation network for 3D object reconstruction from a single image. Li et al. (2018a) proposed a twofold modification to generative adversarial network (GAN) algorithm for learning to generate point clouds (PC-GAN). Achlioptas et al. (2018) proposed two generators for 3D point clouds in both raw space (r-GAN) and latent space (l-GAN) of the pre-trained autoenocoder. Valsesia et al. (2018) presented the unsupervised problem of a generative model exploiting graph convolution on 3D point clouds. Then, Shu et al. (2019) introduced a tree-structured graph convolution network (TreeGCN) in a generator for TreeGAN on 3D point clouds generation. Yang et al. (2019) proposed a principled probabilistic framework to generate 3D point clouds by modeling them as a distribution of distributions. Yin et al. (2019) proposed a multi-scale overcomplete autoencoder on point clouds and shape translator. Cai et al. (2020) proposed a point cloud auto-encoding and generation method by learning the gradient field of the shape log-density. Li et al. (2021) proposed an unsupervised sphere-guided point cloud generative model for a 3D shape generation in raw space. Previous generative methods generate only 3D shape with measurably high fidelity and good coverage, but are not able to generate corresponding image views. To the best of our knowledge, our work represents a first attempt to design a joint generative model for simultaneous generation of high-quality multimodal shape representations, such as 3D point clouds and corresponding 2D images. The related comparison experiments are shown in Secs. 4.1 and 4.3.

In this paper, we aim to fill the gap, i.e., to find a practical way to compute a joint multimodal latent space for simultaneous 3D shape and 2D image generations with comprehensive geometric and data features.

3. Geometry-aware joint latent space

Motivation. In this work, we propose a new neural network framework to learn an effective low-dimensional joint latent space (i.e., a compact and effective neural representation) for simultaneously generating high-fidelity 3D point cloud and its corresponding rendered image with high-quality textures, lighting, or semantics. The main goal of two proposed autoencoders on raw 3D point clouds and 2D images is to extract discriminative features with the same dimension, which can

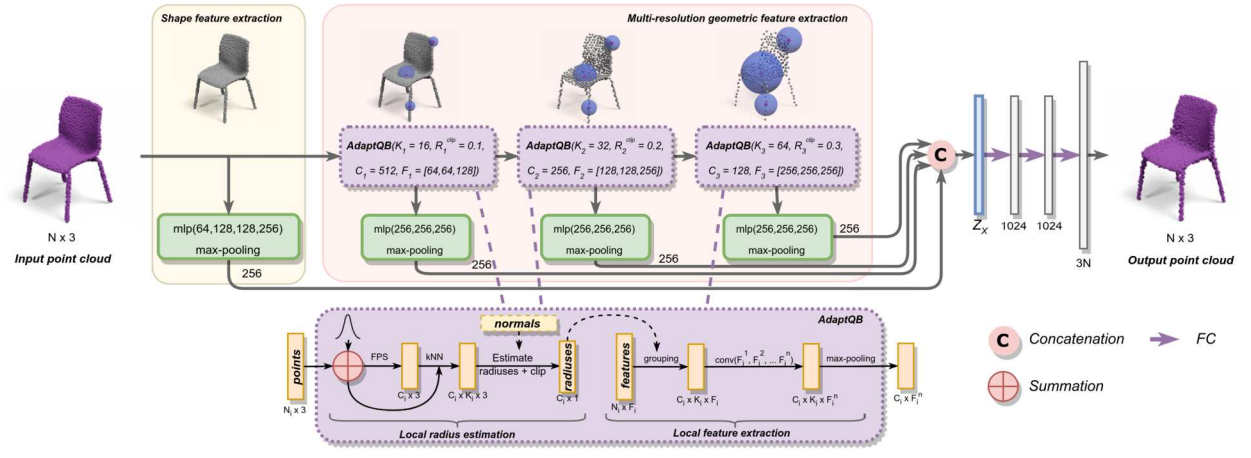


Fig. 2. The architecture of the geometry-aware autoencoder on point clouds. Normals are only used for radius estimation in adaptive query ball (*AdaptQB*) subnetwork and projections in particle loss. They are not used as additional features in the neural network computation. Z_X is a 1024-dim feature vector that encodes a given point cloud. $conv(F_i^1, F_i^2, ..., F_i^n)$ stands for convolutions with the kernel size 1×1 applied sequentially with corresponding feature map sizes F_i^j, $j \in 1, ..., n, i \in 1, 2, 3$.

effectively encode shape geometry and image texture/semantics, respectively. The main goal of the proposed mixer is to synergistically and complementarily combine the extracted latent codes from different representation domains (i.e., shape and image) into the unified joint latent space, which can map and reconstruct the encoded (geometric, appearance, and semantic) information from both modalities. Finally, the proposed joint generative model can simultaneously generate new 3D shape multimodal representations, i.e., 3D shape (semantic) point cloud and its corresponding 2D rendered/semantic image via the joint latent code. The framework enables novel simultaneous multimodal shape and color image generation/interpolation and cross-modality semantic-aware generation, besides improving the quality of traditional single-modal generation applications.

Overview. Given a set of point clouds $\mathcal{X}$ and a set of its corresponding images $\mathcal{Y}$. Our aim is to effectively learn the joint latent space between these two modality sets. Our entire deep neural network framework is comprised of four parts, which are trained separately. The architecture of geometry-aware autoencoder on point clouds is shown in Fig. 2 and the architecture of autoencoder on images is presented in Supplementary Material. These autoencoders produce two separate latent codes $\mathcal{Z}_\mathcal{X}$ and $\mathcal{Z}_\mathcal{Y}$ from the given point clouds and images, respectively. We define both latent codes as $\mathcal{Z}_\mathcal{X} = \{E_\mathcal{X}(X) \mid X \in \mathcal{X}\}$ and $\mathcal{Z}_\mathcal{Y} = \{E_\mathcal{Y}(Y) \mid Y \in \mathcal{Y}\}$. After that, our proposed mixer network (in Fig. 3) in both latent spaces $\mathcal{Z}_\mathcal{X}$ and $\mathcal{Z}_\mathcal{Y}$ is learned to map them into a joint latent space $\mathcal{Z}_{\mathcal{XY}}$, where $\mathcal{Z}_{\mathcal{XY}} = \{E_{\mathcal{XY}}([X, Y]) \mid X \in \mathcal{X}, Y \in \mathcal{Y}\}$. Through such joint latent space, diverse multimodal features from 3D shapes (geometry and topology) and 2D images (appearance and semantics) can be effectively encoded and shared together. Then, our joint generative model as shown in Fig. 3 is to generate joint latent vectors ($\hat{\mathcal{Z}}_{\mathcal{XY}} = \{G(I_{256}) \mid I_{256} = noise\}$) that can be reconstructed from joint (multimodal) latent space back to the separate (single-modal) latent spaces on point clouds and images through $D_{\mathcal{XY}}(\hat{Z}_{XY}) \rightarrow \hat{Z}_X, \hat{Z}_Y$, where $\hat{Z}_{XY} \in \hat{\mathcal{Z}}_{\mathcal{XY}}$. Finally, we obtain generated point cloud and image through $\hat{X} = D_\mathcal{X}(\hat{Z}_X)$ and $\hat{Y} = D_\mathcal{Y}(\hat{Z}_Y)$.

In the following, we will introduce the main technical components of our method: a geometry-aware autoencoder on 3D point clouds, an autoencoder on 2D images, a mixer (i.e., a joint latent space), a joint generative model, and a cross-modality similarity score evaluation.

3.1. Geometry-aware autoencoder on shapes

Our proposed geometry-aware autoencoder (GAE) is depicted in Fig. 2. The input to the encoder is a set of point clouds $\mathcal{X}$. The encoder contains two main components: a *full-resolution shape feature extractor* and a *multi-resolution geometric feature extractor*. The *full-resolution shape feature extractor* is represented by a multilayer perceptron (MLP) layer and a max-pooling layer to form a single 256-dim vector Z_0. The *multi-resolution geometric feature extractor* passes input point cloud $\mathcal{X}$ through a set of abstraction *AdaptQB* layers (in Sec. 3.1.1). The output features from each of these three resolutions are further processed by an MLP followed by max-pooling layer to form a single 256-dim vectors (Z_1, Z_2, Z_3). We concatenate all four extracted vectors to form a 1024-dim latent code Z_X. Our decoder $D_\mathcal{X}$ is a set of three fully-connected (FC) layers, where first two layers followed by a ReLU layer and the last FC layer directly outputs point clouds.

3.1.1. Adaptive query ball layers

Our proposed Adaptive Query Ball subnetwork learns the optimal query ball size using local geometric information (i.e., sampling density and curvature) from the point cloud. This approach helps our geometric-aware autoencoder to adaptively focus on challenging parts of the shapes which include rich geometry and topology details for more accurate representation

and reconstruction. This subnetwork includes two components: a *local radius estimation* and a *local feature estimation* modules. The adaptive query ball is represented as *AdaptQB*$(K_i, R_i^{clip}, C_i, F_i)$, where K_i is a number of neighbors for k-NN, R_i^{clip} is a upper boundary radius for clipping, C_i is the number of query points, F_i is a layer size of an MLP layer for feature extraction at a given resolution $i \in \{1, 2, 3\}$. The ablation study on Adaptive Query Ball is given in Supplementary Material.

Local Radius Estimation. In order to estimate radiuses from the local geometry for query ball computations, our proposed idea is inspired from the traditional geometry computation Gumhold et al. (2001); Mitra and Nguyen (2003) and then extended to neural network computation design. Initially, we perturb the given point cloud $\mathcal{X}$ with a small Gaussian noise along with zero mean and 0.01 standard deviation. The reason of adding the noise is to avoid the case in which the local surface is flat leading to producing an infinite radius. After that, we use k-NN algorithm to find K neighbors for each query point $\mathbf{q}_i$ subsampled by Farthest Point Sampling (FPS) Qi et al. (2017b). Then, we calculate the centroid position of the local neighborhood $\mathcal{N}_i$ for a query point $\mathbf{q}_i$ as: $\mathbf{c}_i = \frac{1}{|\mathcal{N}_i|}\sum_{\mathbf{p}_j \in \mathcal{N}_i} \mathbf{p}_j$. After calculating centroid, we can find the distance d_i between the centroid and the query point $\mathbf{q}_i$, along normal direction $\mathbf{n}(\mathbf{q}_i)$, by $d_i = \|\mathbf{n}(\mathbf{q}_i) \cdot (\mathbf{c}_i - \mathbf{q}_i)\|$. Let us define μ_j as the distance between the query point $\mathbf{q}_i$ and its j-th neighboring point $\mathbf{p}_j$ as $\mu_j = \|\mathbf{p}_j - \mathbf{q}_i\|$. Then, the average distance of the query point $\mathbf{q}_i$ to its neighbors can be calculated by $\mu_i^{avg} = \frac{1}{|\mathcal{N}_i|}\sum_{\mu_j \in \mathcal{N}_i} \mu_j$. Additionally, the local sampling density ρ of the given query point $\mathbf{q}_i$ is equal to the number of points defined in the area of the local disk, i.e., $\rho = \frac{|\mathcal{N}_i|}{\pi \mu_{max}^2}$, where $\mu_{max} = \max_{\mu_j \in \mathcal{N}_j}(\mu_j)$. Finally, the local curvature can be estimated as $\kappa_i = \frac{2d_i}{(\mu_i^{avg})^2}$. In order to find the optimal radius r_i for an *optimal ball size* at the query point $\mathbf{q}_i$, we use:

$$r_i \approx \left(\frac{a_1\sigma_n}{\kappa_i\sqrt{\epsilon\rho}} + \frac{a_2\sigma_n^2}{\kappa_i}\right)^{1/3}, \tag{1}$$

where $a_1 = a_2 = 0.5$ are coefficients and σ_n is a standard deviation of a noise which we employ to perturb point clouds locally. The value of ϵ is 0.1. The related module layers design is shown in Fig. 2.

Local Feature Extraction. After estimating local radius for each centroid, we group neighbors according to each query ball size and run a set of convolutional layers with feature map sizes F_i followed by max-pooling on each local query ball as shown in Fig. 2.

Our proposed adaptive query ball subnetwork is adaptive to the point cloud resolution and the local curvature of the point cloud surface for a given shape as illustrated in Fig. 2 (e.g., the examples of the adaptive query balls on a chair shape).

3.1.2. GAE loss

We define two geometric-based losses to constrain the shape surface property of the output point clouds to produce more appealing and high-fidelity reconstructions as follows.

Reconstruction Loss. Our decoder produces 2048 points and we choose the Earth Mover's Distance (EMD) Rubner et al. (2000) as a reconstruction loss because EMD can capture better geometric shape as compared to Chamfer Distance Fan et al. (2017). The EMD transforms one point set X_1 to the other X_2 according to: $L_{EMD} = \min_{\varsigma: X_1 \to X_2} \sum_{\mathbf{x} \in X_1} \|\mathbf{x} - \varsigma(\mathbf{x})\|_2$, where ς is a bijection from X_1 to X_2. It is differentiable almost everywhere.

Particle-Based Loss. Inspired by the approaches Bossen and Heckbert (1996); Zhong et al. (2013) for surface approximation and meshing, where each generated point is considered as a particle. When the inter-particle force reaches equilibrium, the particle loss produces uniformly distributed point clouds, which is an effective design for accurate surface reconstruction and regularization.

The particle energy E_{ij} between particles i and j is defined as: $E_{ij} = e^{-\frac{\|\mathbf{x}_i - \mathbf{x}_j\|^2}{2\sigma^2}}$, where $\mathbf{x}_i$ and $\mathbf{x}_j$ are neighboring points in a generated point cloud and $i \neq j, i = 1, \ldots, N, j = 1, \ldots, N$. σ is kernel width, which is computed by $\sigma = c_\sigma\sqrt{|\Omega|/N}$, where c_σ is a constant coefficient, $|\Omega|$ is the surface area of a given point cloud, and N is the number of points in a given point cloud. As suggested in Zhong et al. (2013), the best value for c_σ is 0.3 for generating a high-quality isotropic particle distribution.

Then, each point from the generated point cloud needs to be projected onto the shape surface represented by a local disk around a ground truth point $\mathbf{q}_i$. The surface is represented by a ground truth point cloud with estimated normals. Then, the final total particle loss is: $L_{PL} = \frac{1}{N}\sum_{i=1}^{N}\sum_{j=1, j\neq i}^{K} E_{ij}$, where K is a number of neighbors for k-NN. In our experiments we set $K = 20$. Finally, at the backpropagation step in our neural network, the gradients of the particle loss are differentiable with respect to the point locations in the tangent space T_Ω of the shape surface: $\frac{\partial L_{PL}}{\partial \mathbf{x}_i}|_{T_\Omega} = \frac{\partial L_{PL}}{\partial \mathbf{x}_i} - \left[\frac{\partial L_{PL}}{\partial \mathbf{x}_i} \cdot \mathbf{n}(\mathbf{x}_i)\right]\mathbf{n}(\mathbf{x}_i)$, where $\mathbf{n}(\mathbf{x}_i)$ is the unit normal of the shape surface at $\mathbf{x}_i$.

Overall 3D Shape Autoencoder Loss. The overall loss is differentiable with a weighted sum of the above two losses:

$$L_{AE_{pc}} = L_{EMD} + \lambda_{PL} L_{PL}, \tag{2}$$

where λ_{PL} is set to 5.0 in our experiments. The geometry-aware autoencoder for point cloud reconstruction results are given in Sec. 4.1 and ablation study on the particle-based loss are provided in Supplementary Material.

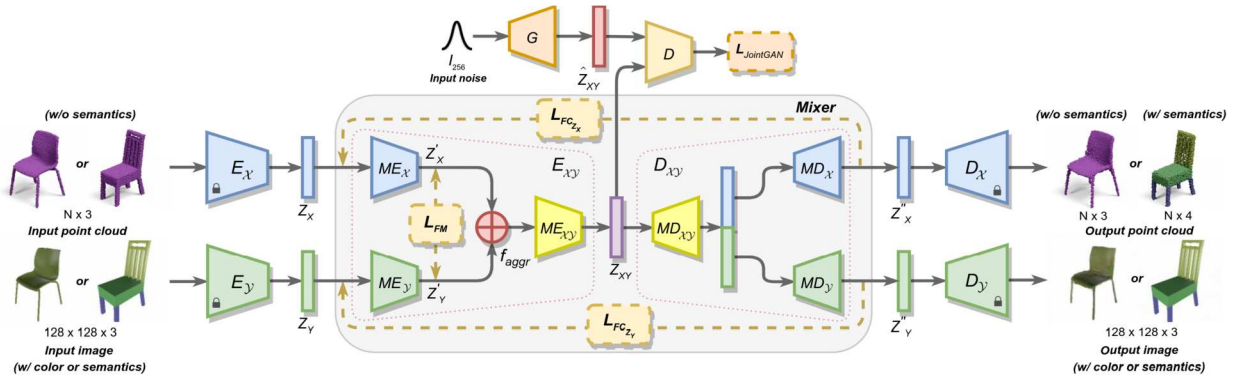


Fig. 3. The architecture of joint generative model via mixer. The proposed mixer blends and complements two different latent vectors Z_X and Z_Y into a joint latent vector Z_{XY}. Then, our generative model in the joint latent space, comprising a generator G and a discriminator D, generates a new joint latent vector $\hat{Z}_{XY}$, which includes both 3D point cloud and 2D image modalities. Finally, the proposed joint generative model can be employed for several novel simultaneous multimodal (SMM) shape generation tasks. f_{aggr} – an aggregation function between two inputs Z'_X and Z'_Y. We choose summation as the aggregation function in our experiments.

3.2. Autoencoder on images

The goal of the autoencoder on 2D images is to encode images $\mathcal{Y}$ into a feature space $\mathcal{Z}_{\mathcal{Y}}$, which is different from a point cloud latent space. The detailed architecture of the autoencoder on images is shown in Supplementary Material. Our image autoencoder contains the encoder $E_{\mathcal{Y}}$ and the decoder $D_{\mathcal{Y}}$. The encoder is implemented as a set of five convolutional layers where each of them followed by a batch normalization (BN) layer, a ReLU layer, and a max-pooling at the end. The decoder comprises one FC and four deconvolutional layers with a BN and ReLU layers and one convolutional layer with a Tanh layer. In our architecture, it is noted that no matter (RGB) color image generation or semantic-aware image generation (i.e., the part-level semantic annotations are represented by RGB values), the input and output image dimensions are the same, i.e., $128 \times 128 \times 3$. To train the 2D image autoencoder, we use the following loss:

$$L_{AE_{img}} = \mathbb{E}_{Y\sim\mathcal{Y}}[||D_{\mathcal{Y}}(E_{\mathcal{Y}}(Y)) - Y||_1]. \tag{3}$$

3.3. Joint generative model via mixer

Mixer – Joint Latent Space. In this section, we propose a novel *mixer* that learns the mapping from two disjoint latent spaces $\mathcal{Z}_{\mathcal{X}}$ and $\mathcal{Z}_{\mathcal{Y}}$, representing two modalities, respectively, into a joint latent space $\mathcal{Z}_{\mathcal{XY}}$. We use pre-trained encoders $E_{\mathcal{X}}$ and $E_{\mathcal{Y}}$ on point clouds and images (as proposed in Sec. 3.1 and Sec. 3.2) to extract latent codes $\mathcal{Z}_{\mathcal{X}}$ and $\mathcal{Z}_{\mathcal{Y}}$, respectively. For evaluations, we use pre-trained decoders $D_{\mathcal{X}}$ and $D_{\mathcal{Y}}$.

Our proposed novel method of learning mixer is aiming to learn effective and efficient low-dimensional joint latent space that encodes both geometry and appearance/texture features, 3D and 2D information between two different modalities. The joint shape-image generation results are shown in Sec. 4.2 and Sec. 4.3. Additionally, we demonstrate that our proposed mixer is capable of sharing semantic information from 2D images to 3D point clouds without needing segmentation information in the latent space. The cross-modal shape-image semantic-aware generation results are provided in Supplementary Material.

The architecture of the mixer is shown in Fig. 3. Our mixer is comprised of the encoder $E_{\mathcal{XY}}$ and the decoder $D_{\mathcal{XY}}$. The encoder $E_{\mathcal{XY}}$ includes two components $ME_{\mathcal{X}}$ and $ME_{\mathcal{Y}}$ on each latent vector followed by the aggregation function f_{aggr}, and then by the encoder $ME_{\mathcal{XY}}$ on joint latent space. Both components $ME_{\mathcal{X}}$ and $ME_{\mathcal{Y}}$ have two FC layers with the hidden dimensions {1024, 1024} followed by the linear activation function without a BN layer. $ME_{\mathcal{XY}}$ has one FC layer with the {1024} hidden dimension. The decoder $D_{\mathcal{XY}}$ includes three main parts, i.e., one decoder on latent space $MD_{\mathcal{XY}}$ and two separate decoders $MD_{\mathcal{X}}$ and $MD_{\mathcal{Y}}$ to reconstruct the input latent vectors from the joint latent space into two modalities. $MD_{\mathcal{XY}}$ also has one FC layer with {2048} hidden dimension. Both $ME_{\mathcal{XY}}$ and $MD_{\mathcal{XY}}$ components include a ReLU and a BN layer followed after an FC layer. The output from $MD_{\mathcal{XY}}$ components is split into two equal parts with the same 1024 dimension and then we feed each of them separately to $MD_{\mathcal{X}}$ and $MD_{\mathcal{Y}}$ decoders. Both components $MD_{\mathcal{X}}$ and $MD_{\mathcal{Y}}$ have two FC layers with the hidden dimensions {1024, 1024} followed by a ReLU layer and a BN layer. The last layer only has the linear activation.

In order to better integrate and complement two modal latent vectors, we design two types of the losses in the proposed mixer: *intermodality feature mapping loss* (L_{FM}) and *intramodality feature consistency loss* ($L_{FC_{Z_X}}$ and $L_{FC_{Z_Y}}$). These two losses play the critical role in learning effective joint latent space across and within two different modality latent spaces. The goal of the *intermodality feature mapping loss* is to bring two modal feature vectors into the unified joint latent space. It forces two

outputs from the encoders $ME_{\mathcal{X}}$ (i.e., Z'_X) and $ME_{\mathcal{Y}}$ (i.e., Z'_Y) to be as similar as possible. Intermodality feature mapping loss is the key for our mixer to learn meaningful mapping between two modal feature latent codes. *Intramodality feature consistency loss* is also important for our mixer. It can preserve to learn the good decoding from the joint space back to the original latent spaces by enforcing the output of the mixer to be similar to the input of the mixer within the modality. This loss helps to reconstruct the joint latent vector Z_{XY} into two different latent codes Z''_X and Z''_Y in their original modalities (i.e., 3D shape and 2D image space, respectively). The detailed loss functions of mixer are:

$$\begin{aligned} L_{FC_{Z_X}} &= \mathbb{E}_{Z_X \sim \mathcal{Z}_{\mathcal{X}}}[||D_{\mathcal{XY}}(E_{\mathcal{XY}}(Z_X)) - Z_X||_1], \\ L_{FC_{Z_Y}} &= \mathbb{E}_{Z_Y \sim \mathcal{Z}_{\mathcal{Y}}}[||D_{\mathcal{XY}}(E_{\mathcal{XY}}(Z_Y)) - Z_Y||_1], \\ L_{FM} &= \mathbb{E}_{Z_X \sim \mathcal{Z}_{\mathcal{X}}}[||ME_{\mathcal{X}}(Z_X) - Z'_Y||_1] + \mathbb{E}_{Z_Y \sim \mathcal{Z}_{\mathcal{Y}}}[||ME_{\mathcal{Y}}(Z_Y) - Z'_X||_1], \\ L_{AE_{mixer}} &= c_1 L_{FC_{Z_X}} + c_2 L_{FC_{Z_Y}} + c_3 L_{FM}, \end{aligned} \tag{4}$$

where $Z_X \in \mathcal{Z}_{\mathcal{X}}$ and $Z_Y \in \mathcal{Z}_{\mathcal{Y}}$ represent feature vectors of point clouds and images, $Z''_X = D_{\mathcal{XY}}(E_{\mathcal{XY}}(Z_X))$ and $Z''_Y = D_{\mathcal{XY}}(E_{\mathcal{XY}}(Z_Y))$, respectively. We set $c_1 = 400$, $c_2 = 10$, and $c_3 = 10$ in our experiments.

Joint Generative Model. Our generative model works in the joint latent space and comprises a generator G and a discriminator D with an input of a random noise vector as shown in the top of Fig. 3. Similar to the mixer, generator and discriminator are implemented as a set of FC layers. Particularly, generator G has three FC layers with the hidden dimensions {256, 512, 1024}. Discriminator D has two FC layers with the hidden dimensions {512, 256}. Each hidden FC layer in generator and discriminator is followed by a ReLU. Taking a pair of point cloud and its color/semantic image, we encode them into one joint latent vector Z_{XY}, which is in joint latent space $\mathcal{Z}_{\mathcal{XY}}$. During the training, in our generative model, the pre-trained encoders ($E_{\mathcal{X}}$, $E_{\mathcal{Y}}$, and $E_{\mathcal{XY}}$) are fixed. For evaluation, we use pre-trained decoders ($D_{\mathcal{X}}$, $D_{\mathcal{Y}}$, and $D_{\mathcal{XY}}$). Wasserstein GAN Gulrajani et al. (2017) is used in the joint adversarial loss:

$$L_{JointGAN} = \mathbb{E}_{\hat{Z}_{XY} \sim \hat{\mathcal{Z}}_{\mathcal{XY}}}\left[D(\hat{Z}_{XY})\right] - \mathbb{E}_{Z_{XY} \sim \mathcal{Z}_{\mathcal{XY}}}\left[D\left(Z_{XY}\right)\right] + \lambda L_{GP}, \tag{5}$$

where L_{GP} is a regularization gradient penalty loss and λ is a scalar weight set to 10 by default. Finally, we can simultaneously generate new joint 3D shape multimodal representations, i.e., 3D shape (semantic) point cloud and its corresponding 2D rendered/semantic image through the computed joint latent code $\hat{Z}_{XY}$. The ablation study on our *mixer* for the joint generation is provided in Supplementary Material.

3.4. Cross-modality similarity evaluation

One of the advantages of our proposed mixer over traditional approaches is that we can intrinsically learn the bijection between two modalities in the unified joint latent space and well preserve this mapping for the generation task through the joint latent space. In this case, we propose a novel *Cross-Modality Similarity Score (CMSS)* metric to measure the bijection accuracy between the generated shape and image in the latent space. Given the generated latent shape vector and latent image vector, we evaluate whether these two latent vectors belong to the same object or not through a binary classification network. This new metric shows the percentage of generated pairs (shape and image) belonging to the same object as follows: $CMSS = \mathbb{E}_{\left(\hat{Z}_X, \hat{Z}_Y\right) \sim \left(\hat{\mathcal{Z}}_{\mathcal{X}}, \hat{\mathcal{Z}}_{\mathcal{Y}}\right)} \mathcal{F}_{BC}(\hat{Z}_X, \hat{Z}_Y)$, where $\mathcal{F}_{BC}()$ is a pre-trained binary classification network. Similar to the mixer, this binary classification network has two parallel branches with two FC layers (one for the shape latent code and one for the image latent code) with the hidden dimension {1024, 1024} followed by a ReLU and a BN layer. After that, the aggregation function (i.e., summation) is applied and then followed by the set of two FC layers, where the first layer with the hidden dimension {1024} is followed by a ReLU layer and the last FC layer directly outputs a binary classification value.

4. Experiments

In this section, we evaluate our models on various tasks in detail, such as point cloud auto-encoding, simultaneous multimodal (SMM) shape and color image generation, single-modal (shape or image) generation, joint latent space interpolation and visualization, etc. Additionally, we evaluate our method on novel SMM semantic-aware generation task in Supplementary Material. The ablation study on the proposed GAE components and analysis on mixer for SMM generation is shown in Supplementary Material. The network configurations, the training details, and all quantitative evaluation metrics in each task are provided in Supplementary Material. In all our experiments and comparisons, the number of points in the output point clouds is 2048 and the resolution of the output images is 128×128. For the comparison experiments, best results in the tables are shown in bold font. All models in this paper are trained on a single NVIDIA Titan Xp GPU with 12GB GDDR5X. The source code of our framework and data will be released later.

Dataset. We evaluate our models on *ShapeNet Core* dataset, which provides meshes with the texture (if there is no texture provided, we use the plain gray color for image rendering). We test our framework on three different object classes: *chair*, *airplane*, and *car*. For experiments on *ShapeNet Core*, we sample 10,000 uniform points with normals from meshes, and

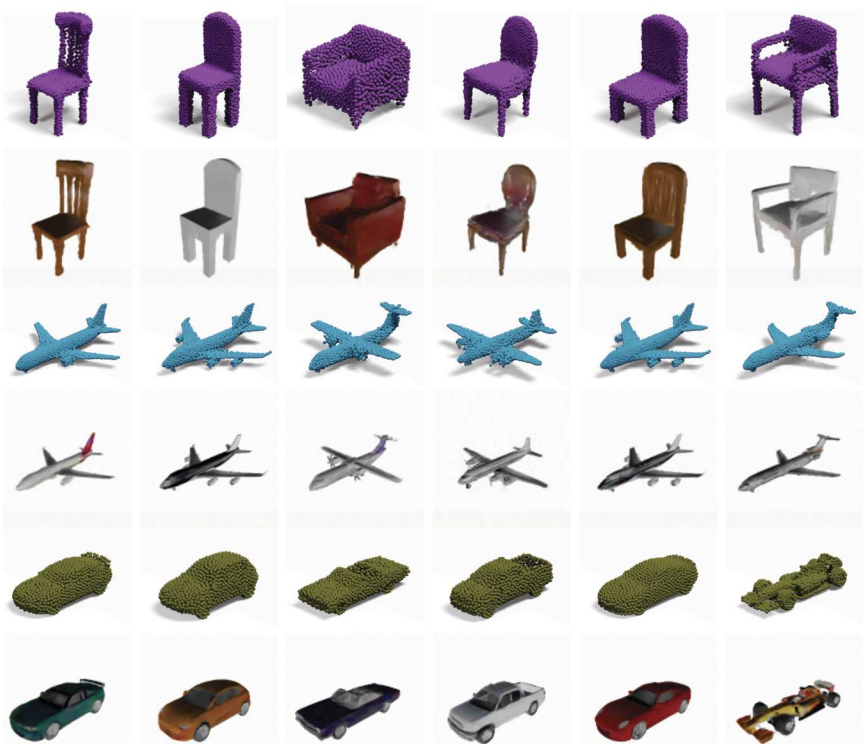


Fig. 4. Qualitative SMM generation results on 3D point clouds and corresponding 2D color images from three different categories: *chair*, *airplane*, and *car* by our method. Top row: point clouds, bottom row: images. (For interpretation of the colors in the figure(s), the reader is referred to the web version of this article.)

normalize and orient them the same as in the process of Achlioptas et al. (2018). We approximate the surface area for each shape point cloud from dataset for particle loss calculation and we shuffle points for a better generalization. For each object, we render one view 128×128 color image from a fixed viewpoint. Finally, a pair of a point cloud and its color image from a given 3D object is made.

4.1. Shape auto-encoding evaluation

We evaluate the quality of reconstructed point clouds that demonstrates how well our proposed geometry-aware autoencoder can encode the given point clouds. We compare our model with several state-of-the-art methods in the shape auto-encoding task on three shape categories (i.e., *chair*, *airplane*, and *car*) from *ShapeNet Core* dataset as shown in Table 1. Our model consistently outperforms AE-EMD Achlioptas et al. (2018), AE-CD Achlioptas et al. (2018) and LOGAN-AE Yin et al. (2019) on all evaluation metrics. The training time T_{train} of AE-CD is lowest because it uses CD loss for training. Our architecture with the proposed components (i.e., adaptive query ball and particle loss) does not increase training time. Our GAE also outperforms ShapeGF-AE Cai et al. (2020) (one of the latest works) on most of the metrics across different categories, and shows competitive performance on CD metric on *chair* class and F2 metric on *airplane* class. But the training time of our model is significantly lower than the training time of ShapeGF-AE across different categories.

4.2. Simultaneous multimodal generation

We evaluate our method on the simultaneous multimodal (SMM) shape and image generation task. Our SMM generation results are shown in Fig. 4 and more provided in Supplementary Material. We compare our proposed framework through the SMM generation task with the state-of-the-art methods, which can generate 3D point clouds and 2D images in different ways (i.e. *with* or *without* using joint latent space). Both the quantitative evaluation (Table 2) and qualitative evaluation (Fig. 5) are provided. More qualitative evaluation results are provided in Supplementary Material.

Firstly, we compare our method with the state-of-the-art Shape Unicode Muralikrishnan et al. (2019) which is one of the closest methods to ours that *uses joint latent space* to unify different modalities. We train Shape Unicode model on *ShapeNet Core* dataset, with two modalities (i.e., 3D point clouds and 2D images). After training Shape Unicode we extract separately image and point cloud unicodes to train our joint generative model. In the first experiment (i.e., shown in Fig. 5 (1st row)), we generate image unicodes with our joint generative model and reconstruct two representations (i.e., 3D point cloud and 2D image) using Shape Unicode's decoders. In the second experiment (i.e., shown in Fig. 5 (2nd row)), we generate point

Table 1
Comparison of the state-of-the-art point cloud reconstruction methods on different classes from *ShapeNet Core* dataset.

	chair					*airplane*					*car*				
	F1 ↑	F2 ↑	CD ↓	EMD ↓	T_{train}	F1 ↑	F2 ↑	CD ↓	EMD ↓	T_{train}	F1 ↑	F2 ↑	CD ↓	EMD ↓	T_{train}
AE-EMD Achlioptas et al. (2018)	57.25	80.65	0.705	0.873	3h	87.84	94.44	0.254	0.589	1h 50 m	62.03	88.74	0.452	0.617	3h 15 m
AE-CD Achlioptas et al. (2018)	62.12	84.09	0.536	1.378	55 m	89.76	95.73	0.213	0.841	40 m	63.77	90.07	0.417	0.808	1h 5 m
LOGAN-AE Yin et al. (2019)	55.37	78.53	0.776	0.820	12h	87.88	94.02	0.281	0.549	6h 55 m	61.62	88.75	0.461	0.584	12h 45 m
ShapeGF-AE Cai et al. (2020)	62.84	85.67	**0.475**	0.823	45h	91.18	**96.91**	0.189	0.571	26h 30 m	53.64	83.64	0.492	0.704	49h 30 m
our (GAE)	**68.49**	**88.37**	0.504	**0.635**	3h 40 m	**92.36**	96.84	**0.182**	**0.438**	2h 20 m	**66.73**	**91.74**	**0.402**	**0.536**	3h 55 m

Table 2
Comparison of our method with the state-of-the-art methods on simultaneous multimodal generation task on *chair* class. Note: the source code of the original Shape Unicode (ShUn) Muralikrishnan et al. (2019) is not publicly available and we have implemented their method following the paper's settings and architectures.

	point cloud					*image*		
Model	JSD ↓	MMD-CD ↓	MMD-EMD ↓	COV-CD ↑	COV-EMD ↑	FID ↓	KID ↓	CMSS ↑
ShUn (img feats) Muralikrishnan et al. (2019)	0.813	**0.122**	1.076	59.56	18.30	21.27	18.71	-
ShUn (pc feats) Muralikrishnan et al. (2019)	0.853	0.125	1.081	55.13	18.23	21.00	18.46	-
$\mathcal{Z}_\mathcal{X}$ concat $\mathcal{Z}_\mathcal{Y}$	1.936	0.271	0.820	46.13	55.20	11.83	8.16	1.64
$\mathcal{Z}_\mathcal{X}$ + rendering	**0.042**	0.133	0.531	65.31	67.68	18.69	16.36	84.95
$\mathcal{Z}_\mathcal{Y}$ + PSGN Fan et al. (2017)	0.593	0.126	0.932	60.81	15.21	11.73	7.91	65.61
our (joint)	0.060	0.128	**0.522**	**66.94**	**69.30**	**10.53**	**6.53**	**98.67**

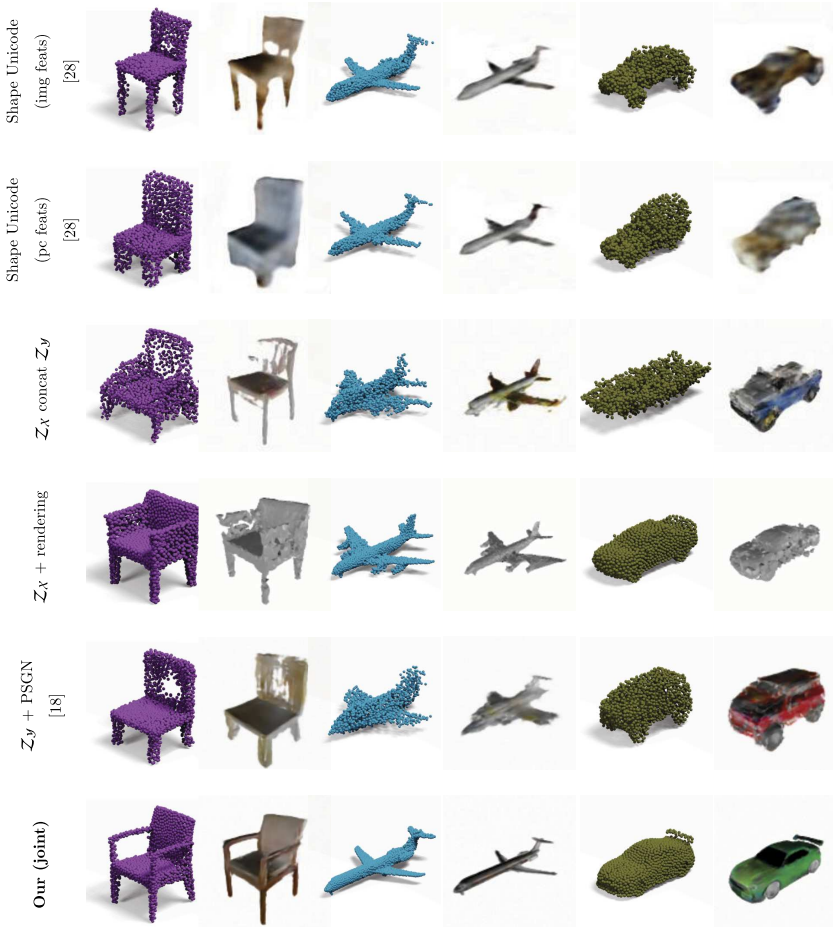


Fig. 5. Comparison of our method with the state-of-the-art methods on multimodal shape and image generation. $\mathcal{Z}_\mathcal{X}$ - the shape latent space, $\mathcal{Z}_\mathcal{Y}$ - the image latent space. Left: point clouds, right: images.

cloud unicode vectors, and reconstruct 3D point clouds and 2D images from that latent vectors. Fig. 5 shows that 3D point clouds and 2D images generated using Shape Unicode's latent vectors produce blurry 2D images and low-quality 3D point clouds, with poor evaluation metrics on both modalities as shown in Table 2. However, our proposed joint latent space can simultaneously generate high-quality 3D point clouds and 2D images as shown in Fig. 5 (6th row) and Table 2. We know that each modality of the *ShapeNet Core* dataset has some complementary/different information about the object that does not share with other modalities. And when Shape Unicode enforces the latent vectors of different modalities to be as close as possible it results in the low quality of the separate modalities. However, our joint latent space can learn a unified and effective latent space for a joint representation, where each modality represented by its own latent vector without losing individual/discriminative information can be mixed together into a unified joint latent vector.

Table 3
Comparison of the state-of-the-art single-modal generations.

		Model	JSD ↓	MMD-CD ↓	MMD-EMD ↓	COV-CD ↑	COV-EMD ↑	FID ↓	KID ↓
airplane	*point cloud*	r-GAN Achlioptas et al. (2018)	0.414	0.04	0.65	59	16	-	-
		l-GAN (EMD) Achlioptas et al. (2018)	0.232	0.05	0.34	57	65	-	-
		l-GAN (CD) Achlioptas et al. (2018)	0.257	0.04	0.41	62	34	-	-
		TreeGAN Shu et al. (2019)	0.337	0.05	0.51	63	25	-	-
		ShapeGF-GAN Cai et al. (2020)	0.222	0.05	0.39	40	38	-	-
		SP-GAN Li et al. (2021)	0.170	0.05	0.35	51	45	-	-
	image	DCGAN Radford et al. (2015)	-	-	-	-	-	19.58	21.54
		PlatonicGAN Henzler et al. (2019)	-	-	-	-	-	18.35	19.95
		WGAN-GP Gulrajani et al. (2017)	-	-	-	-	-	11.65	10.51
		our (joint)	**0.082**	**0.039**	**0.31**	**68**	**69**	**8.32**	**5.57**
chair	*point cloud*	r-GAN Achlioptas et al. (2018)	0.340	0.14	0.83	65	28	-	-
		l-GAN (EMD) Achlioptas et al. (2018)	0.114	0.14	0.56	64	65	-	-
		l-GAN (CD) Achlioptas et al. (2018)	0.247	0.14	0.70	65	29	-	-
		TreeGAN Shu et al. (2019)	0.233	0.15	0.74	65	31	-	-
		ShapeGF-GAN Cai et al. (2020)	0.089	0.15	0.61	48	47	-	-
		SP-GAN Li et al. (2021)	0.210	0.17	0.58	44	39	-	-
	image	DCGAN Radford et al. (2015)	-	-	-	-	-	16.25	15.64
		PlatonicGAN Henzler et al. (2019)	-	-	-	-	-	24.15	25.39
		WGAN-GP Gulrajani et al. (2017)	-	-	-	-	-	13.81	10.33
		our (joint)	**0.060**	**0.128**	**0.52**	**67**	**69**	**10.53**	**6.53**

Secondly, we compare our method with the group of the state-of-the-art methods that *does not use joint latent space*. First, we show the advantage of learning joint latent space using our mixer against simple feature concatenation of the extracted latent vectors from pre-trained autoencoders on point clouds and images. In this case, the quality of the generated point clouds suffers the most as shown in Table 2 as well as in Fig. 5 (3rd row). The major reason is that the two modal autoencoders are in different latent spaces, which cannot work meaningfully and effectively by directly bringing them together without any neural processing. Another alternative is to generate point clouds through the latent space $\mathcal{Z}_{\mathcal{X}}$, in which we train the generation model. After generating point clouds, we reconstruct their surface meshes by the ball-pivoting algorithm Bernardini et al. (1999) with additional surface refinement technique such as holes closing. Finally we render the reconstructed meshes for image generation as shown in Fig. 5 (4th row). There are two main drawbacks: (1) the rendered object in images appears to be prone to have artifacts and missing parts, since the surface is reconstructed by some complicated mesh post-processing steps; (2) this approach renders images without color/texture, since images are generated from point clouds (without colors). Vice versa, we show another way, in which we generate images through the latent space $\mathcal{Z}_{\mathcal{Y}}$ first, and then use one of the existing methods for image to point cloud reconstruction (i.e., PSGN Fan et al. (2017)) to generate corresponding point clouds. If there are artifacts (i.e., missing parts or holes as shown in Fig. 5 (5th row)) in the generated images, the PSGN method may reconstruct point cloud with the defects. Additionally, our method significantly outperforms other methods on CMSS metric as shown in Table 2. This new metric confirms the importance of the proposed mixer in our method. It is worth mentioning that in Table 2 we mark CMSS score for Shape Unicode Muralikrishnan et al. (2019) as '-', because by the nature their embedding loss encourages the embeddings generated by each encoder to be similar constraining them by the L_1 loss. However, the Fig. 5 (1st and 2nd rows) shows that the reconstructed 3D point clouds and 2D images from the unicode vectors cannot guarantee to produce high-quality pairs of different modalities from their unicode space.

In conclusion, from Table 2 and Fig. 5, it is noted that our mixer can effectively learn joint latent space for generating two modality representation sets with higher fidelity 3D shapes, better quality 2D images, and their implicit bijection simultaneously as compared to other approaches. Shape Unicode (Muralikrishnan et al., 2019) builds their joint latent space to generate different modalities and produces blurry 2D images and poor-quality 3D point clouds. Their proposed unicode space is not effective as our joint latent space on cross-modality generation task. The other methods that do not use joint latent space have some major limitations in our cross-modality generation task. Essentially, they can only manipulate the generation task in a single-modal latent space so that the generated results of the other modality highly depend on the quality of the given modality's result.

4.3. Comparison with single-modal generation

In order to further analyze the advantage of using our mixer for joint generation, we also provide experiments for the single-modal generation task in Table 3. Our mixer helps to improve the quality on all metrics compared to single-modal generation (point clouds or images) methods, which lack another modality representation shown as '-'.

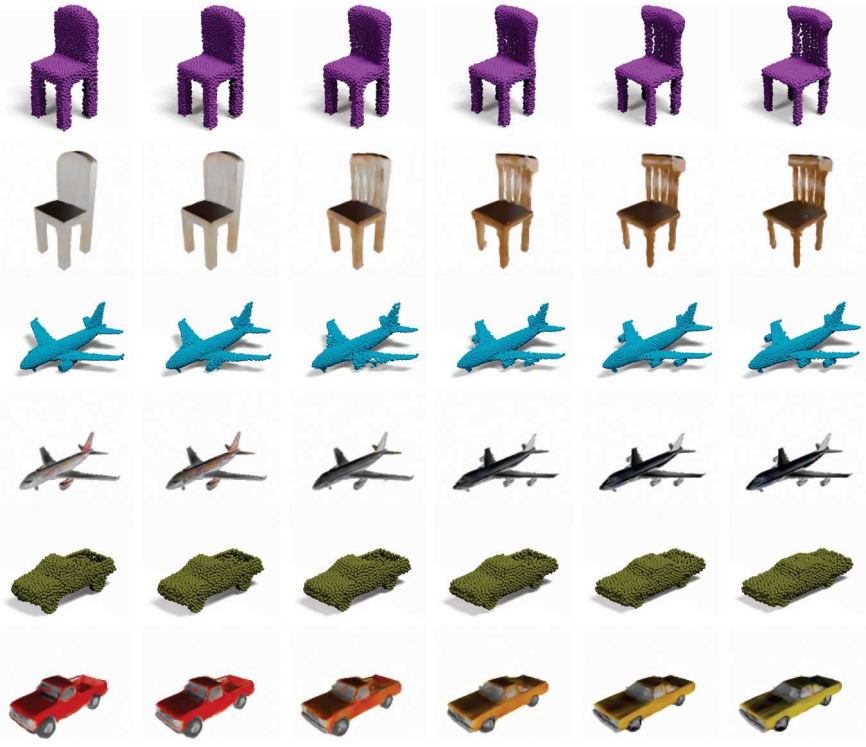


Fig. 6. Joint latent space interpolation and generation. Top row: point clouds, bottom row: images.

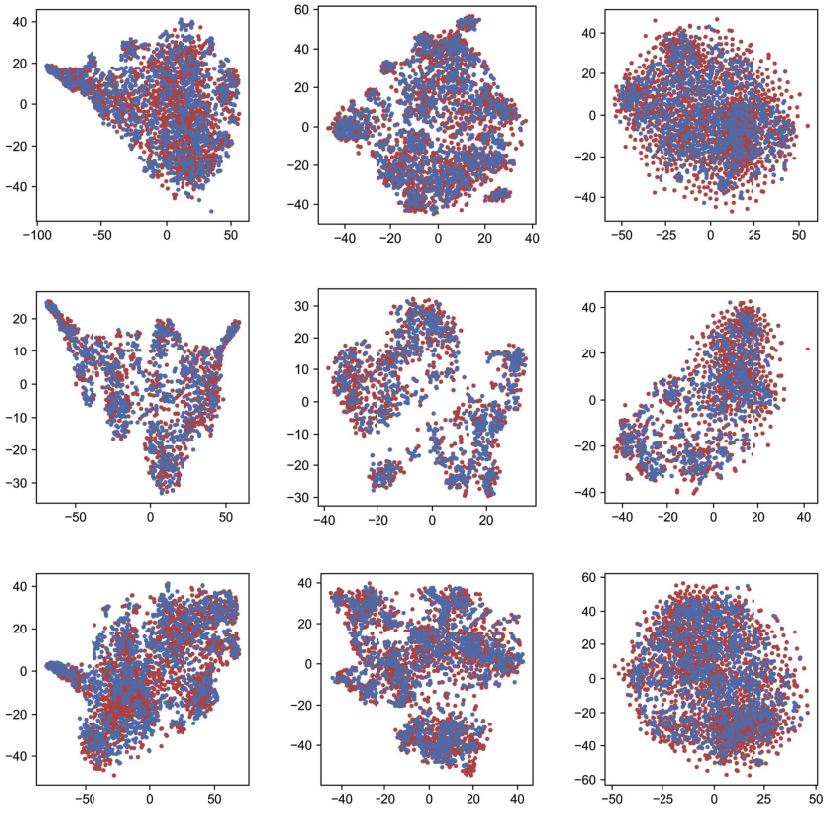
Shape Generation. We compare our model with other state-of-the-art generative methods in both raw point cloud space and latent space. We quantitatively compare our method's performance on the shape generation with r-GAN Achlioptas et al. (2018), l-GAN (AE-EMD) Achlioptas et al. (2018), l-GAN (AE-CD) Achlioptas et al. (2018), TreeGAN Shu et al. (2019), ShapeGF-GAN Cai et al. (2020), and SP-GAN Li et al. (2021) as shown in Table 3. We run all these methods on our prepared datasets to make the comparison fair. We evaluate all these methods following the evaluation scheme in Achlioptas et al. (2018). Our method outperforms all these methods. These experiments demonstrate that the proposed joint latent space $\mathcal{Z}_{\mathcal{XY}}$ can effectively combine and integrate two different modal features from the same object to improve the quality of generated point clouds. We provide qualitative comparison of different methods with our method in Supplementary Material. It shows that our method outperforms other single-modal generation methods in the quality of generated point clouds on both *ShapeNet Core chair* and *airplane* classes with respect to the high-fidelity geometry and topology of a variety of 3D objects.

Image Generation. We compare our approach with some state-of-the-art and well-known generative models, i.e., DCGAN Radford et al. (2015), WGAN-GP Gulrajani et al. (2017), PlatonicGAN Henzler et al. (2019) on images. Our generative model results outperforms them as shown in Table 3. These experiments show that the learned joint latent space can effectively combine and integrate two different modal features from the same object to improve the quality of generated images. We provide visualization results of each method in Supplementary Material. It shows that our approach generates better quality images compared to other alternatives on *ShapeNet Core chair* and *airplane* classes. Our results have less artifacts and more realistic textures/colors.

4.4. Joint latent space interpolation and analysis

Joint Latent Space Interpolation. In Fig. 6, we show the linear interpolation in the proposed joint latent space between the selected left- and right-most images and shapes on *chair*, *airplane*, and *car* classes. More results with large variations in shape geometry and topology, and image texture are provided in Supplementary Material. The experiments show that our method can learn a joint latent space with smooth transitions in both modalities, i.e., 3D point clouds and 2D color images for different object classes. To the best of our knowledge, we are the first to show that interpolating joint latent space can result in smooth transition both in shape and image simultaneously. Through this approach, we can explore and generate new knowledge from multimodal 3D datasets.

Joint Latent Space Analysis. Fig. 7 shows the t-SNE visualization Maaten and Hinton (2008) of the different latent spaces in our method. The first column visualizes the ground truth of joint latent codes $\mathcal{Z}_{\mathcal{XY}}$ and the generated joint latent codes



(a) $\mathcal{Z}_{\mathcal{XY}}$ (joint) (b) $\mathcal{Z}_{\mathcal{X}}$ (point clouds) (c) $\mathcal{Z}_{\mathcal{Y}}$ (images)

Fig. 7. The t-SNE visualization of the joint latent space and latent spaces of point clouds and images. Top row - *chair*, middle row - *airplane*, bottom row - *car*. The left column represents joint latent space, the middle column - latent space of point clouds, the right column - latent space of images. Blue - generated latent codes, red - latent codes produced by mixer or AEs, which are considered as reference.

$\hat{\mathcal{Z}}_{\mathcal{XY}}$. The second column visualizes point cloud latent codes $\mathcal{Z}_{\mathcal{X}}$ produced by AE on point clouds and the latent codes $\hat{\mathcal{Z}}_{\mathcal{X}}$ produced by the mixer decoder from the generated joint latent codes. The third column visualizes image latent codes $\mathcal{Z}_{\mathcal{Y}}$ produced by AE on images and the latent codes $\hat{\mathcal{Z}}_{\mathcal{Y}}$ produced by the mixer decoder from the generated joint latent codes. This visualization shows that the distribution of the generated joint latent codes well matches the latent codes produced by AEs and mixer. Also, it shows that mixer can successfully reconstruct separate latent codes from the generated joint latent codes. This visualization can also well explain the superior performance in our SMM generation for both point clouds and images.

5. Conclusion

In this work, we have proposed a new geometry-aware joint latent space framework for both 3D shapes and 2D images, which can better capture the 3D object information from cross-modality, such as shape geometry, image texture, shape/image semantics, etc. Our approach can simultaneously generate high-quality 3D shapes and 2D images with high diversity. Through extensive experiments on the benchmark datasets, our method has achieved the state-of-the-art performance on point cloud auto-encoding, and several novel 3D shape tasks, such as simultaneous multimodal (SMM) shape and color image generation and interpolation, and SMM semantic-aware generation.

Limitation and Future Work. In the joint latent space interpolation and generation task, we noticed that sometimes our interpolated shapes and images have some inconsistencies with each other. For example, the interpolation between a chair with arms and a chair without might produce such issue. In general, the interpolation of the shape is smoother than the interpolation of the corresponding image in some cases. Since our work does not focus on image generation task, some more advanced image autoencoder and neural rendering techniques can be applied to further improve the current results. Incorporating the arbitrary- or multi-view 2D images as well as applying our model to some more complex datasets in the current framework will be our future work.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Acknowledgements

We would like to thank the reviewers for their valuable comments. This work was partially supported by the NSFC 61972353 and the NSF under Grant Numbers IIS-1816511, OAC-1845962, and OAC-1910469.

Appendix A. Supplementary material

Supplementary material related to this article can be found online at https://doi.org/10.1016/j.cagd.2022.102076.

References

Achlioptas, P., Diamanti, O., Mitliagkas, I., Guibas, L., 2018. Learning representations and generative models for 3D point clouds. In: Proceedings of the International Conference on Machine Learning, pp. 40–49.
Bernardini, F., Mittleman, J., Rushmeier, H., Silva, C., Taubin, G., 1999. The ball-pivoting algorithm for surface reconstruction. IEEE Trans. Vis. Comput. Graph. 5 (4), 349–359.
Boscaini, D., Masci, J., Rodolà, E., Bronstein, M., 2016. Learning shape correspondence with anisotropic convolutional neural networks. In: Proceedings of the Advances in Neural Information Processing Systems, pp. 3189–3197.
Bossen, F., Heckbert, P., 1996. A pliant method for anisotropic mesh generation. In: Proceedings of the International Meshing Roundtable, p. 6376.
Cai, R., Yang, G., Averbuch-Elor, H., Hao, Z., Belongie, S., Snavely, N., Hariharan, B., 2020. Learning gradient fields for shape generation. In: Proceedings of the European Conference on Computer Vision, pp. 364–381.
Chen, K., Choy, C., Savva, M., Chang, A., Funkhouser, T., Savarese, S., 2018. Text2Shape: generating shapes from natural language by learning joint embeddings. In: Proceedings of the Asian Conference on Computer Vision, pp. 100–116.
Fan, H., Su, H., Guibas, L., 2017. 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, pp. 605–613.
Gao, H., Zhu, X., Lin, S., Dai, J., 2020. Deformable kernels: adapting effective receptive fields for object deformation. In: International Conference on Learning Representations.
Girdhar, R., Fouhey, D., Rodriguez, M., Gupta, A., 2016. Learning a predictable and generative vector representation for objects. In: Proceedings of the European Conference on Computer Vision, pp. 484–499.
Gong, Y., Ke, Q., Isard, M., Lazebnik, S., 2014b. A multi-view embedding space for modeling Internet images, tags, and their semantics. Int. J. Comput. Vis. 106 (2), 210–233.
Gong, Y., Wang, L., Hodosh, M., Hockenmaier, J., Lazebnik, S., 2014a. Improving image-sentence embeddings using large weakly annotated photo collections. In: Proceedings of the European Conference on Computer Vision, pp. 529–545.
Goodfellow, I., Bengio, Y., Courville, A., 2016. Deep Learning. MIT Press.
Grünwald, P., 2007. The Minimum Description Length Principle. MIT Press.
Gulrajani, I., Ahmed, F., Arjovsky, M., Dumoulin, V., Courville, A., 2017. Improved training of Wasserstein GANs. In: Proceedings of the Advances in Neural Information Processing Systems, pp. 5767–5777.
Gumhold, S., Wang, X., MacLeod, R., 2001. Feature extraction from point clouds. In: Proceedings of the International Meshing Roundtable, pp. 293–305.
Hanocka, R., Hertz, A., Fish, N., Giryes, R., Fleishman, S., Cohen-Or, D., 2019. MeshCNN: a network with an edge. ACM Trans. Graph. 38 (4), 1–12.
Henzler, P., Mitra, N., Ritschel, T., 2019. Escaping Plato's cave: 3D shape from adversarial rendering. In: Proceedings of the IEEE International Conference on Computer Vision, pp. 9984–9993.
Herzog, R., Mewes, D., Wand, M., Guibas, L., Seidel LeSSS, H.-P., 2015. Learned shared semantic spaces for relating multi-modal representations of 3D shapes. Comput. Graph. Forum 34 (5), 141–151.
Hinton, G., Zemel, R., 1994. Autoencoders, minimum description length, and Helmholtz free energy. Adv. Neural Inf. Process. Syst. 6, 3–10.
Hu, W., Zhu, S., 2015. Learning 3D object templates by quantizing geometry and appearance spaces. IEEE Trans. Pattern Anal. Mach. Intell. 37 (6), 1190–1205.
Huang, H., Kalogerakis, E., Chaudhuri, S., Ceylan, D., Kim, V.G., Yumer, E., 2017. Learning local shape descriptors from part correspondences with multiview convolutional networks. ACM Trans. Graph. 37 (1), 1–14.
Huang, Z., Yu, Y., Xu, J., Ni, F., Le, X., 2020. PF-Net: point fractal network for 3D point cloud completion. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 7662–7670.
Jason, W., Samy, B., Nicolas, U., 2010. Large scale image annotation: learning to rank with joint word-image embeddings. Mach. Learn. 81 (1), 21–35.
Jason, W., Samy, B., Nicolas, U., 2011. WSABIE: scaling up to large vocabulary image annotation. In: Proceedings of the International Joint Conference on Artificial Intelligence, pp. 2764–2770.
Kalogerakis, E., Averkiou, M., Maji, S., Chaudhuri, S., 2017. 3D shape segmentation with projective convolutional networks. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 3779–3788.
Kiros, R., Salakhutdinov, R., Zemel, R., 2014. Unifying visual-semantic embeddings with multimodal neural language models. arXiv preprint: arXiv:1411.2539.
Klein, B., Lev, G., Sadeh, G., Wolf, L., 2015. Associating neural word embeddings with deep image representations using Fisher vectors. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 4437–4446.
Komarichev, A., Zhong, Z., Hua, J., 2019. A-CNN: annularly convolutional neural networks on point clouds. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 7421–7430.
Kostrikov, I., Jiang, Z., Panozzo, D., Zorin, D., Bruna, J., 2018. Surface networks. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 2540–2548.
Li, C.-L., Zaheer, M., Zhang, Y., Poczos, B., Salakhutdinov, R., 2018a. Point cloud GAN. arXiv preprint: arXiv:1810.05795.
Li, R., Li, X., Fu, C.-W., Cohen-Or, D., Heng, P.-A., 2019. PU-GAN: a point cloud upsampling adversarial network. In: Proceedings of the IEEE International Conference on Computer Vision.
Li, R., Li, X., Hui, K.-H., Fu, C.-W., 2021. SP-GAN: sphere-guided 3D shape generation and manipulation. ACM Trans. Graph. 40 (4), 151.
Li, X., Liu, S., Kim, K., De Mello, S., Jampani, V., Yang, M.-H., Kautz, J., 2020. Self-supervised single-view 3D reconstruction via semantic consistency. In: Proceedings of the European Conference on Computer Vision, pp. 677–693.

Li, Y., Bu, R., Sun, M., Wu, W., Di, X., Chen, B., 2018b. PointCNN: convolution on x-transformed points. Adv. Neural Inf. Process. Syst. 31, 820–830.
Li, Y., Su, H., Qi, C., Fish, N., Cohen-Or, D., Guibas, L., 2015. Joint embeddings of shapes and images via CNN image purification. ACM Trans. Graph. 34 (6), 234.
Liu, M., Sheng, L., Yang, S., Shao, J., Hu, S.-M., 2020. Morphing and sampling network for dense point cloud completion. In: Proceedings of the AAAI Conference on Artificial Intelligence, vol. 34, pp. 11596–11603.
Lun, Z., Gadelha, M., Kalogerakis, E., Maji, S., Wang, R., 2017. 3D shape reconstruction from sketches via multi-view convolutional networks. In: Proceedings of the International Conference on 3D Vision, pp. 67–77.
Maaten, L., Hinton, G., 2008. Visualizing data using t-SNE. J. Mach. Learn. Res. 9, 2579–2605.
Mandikal, P., Navaneet, K., Agarwal, M., Babu, R., 2018. 3D-LMNet: latent embedding matching for accurate and diverse 3D point cloud reconstruction from a single image. In: Proceedings of the British Machine Vision Conference.
Masci, J., Boscaini, D., Bronstein, M., Vandergheynst, P., 2015. Geodesic convolutional neural networks on Riemannian manifolds. In: Proceedings of the IEEE International Conference on Computer Vision Workshops, pp. 37–45.
Mitra, N., Nguyen, A., 2003. Estimating surface normals in noisy point cloud data. In: Proceedings of the Annual Symposium on Computational Geometry, pp. 322–328.
Muralikrishnan, S., Kim, V., Fisher, M., Chaudhuri, S., 2019. Shape unicode: a unified shape representation. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 3790–3799.
Peng, Y., Huang, X., Qi, J., 2016. Cross-media shared representation by hierarchical learning with multiple deep networks. In: Proceedings of the International Joint Conference on Artificial Intelligence, pp. 3846–3853.
Qi, C., Su, H., Nießner, M., Dai, A., Yan, M., Guibas, L., 2016. Volumetric and multi-view cnns for object classification on 3D data. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 5648–5656.
Qi, C., Su, H., Mo, K., Guibas, L., 2017a. PointNet: deep learning on point sets for 3D classification and segmentation. Proc. IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recognit. 1 (2), 4.
Qi, C., Yi, L., Su, H., Guibas, L., 2017b. PointNet++: deep hierarchical feature learning on point sets in a metric space. In: Proceedings of the Advances in Neural Information Processing Systems, pp. 5099–5108.
Radford, A., Metz, L., Chintala, S., 2015. Unsupervised representation learning with deep convolutional generative adversarial networks. arXiv preprint: arXiv:1511.06434.
Reed, S., Akata, Z., Lee, H., Schiele, B., 2016. Learning deep representations of fine-grained visual descriptions. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 49–58.
Rubner, Y., Tomasi, C., Guibas, L., 2000. The Earth mover's distance as a metric for image retrieval. Int. J. Comput. Vis. 40 (2), 99–121.
Schwarz, K., Liao, Y., Niemeyer, M., Geiger, A., 2021. GRAF: generative radiance fields for 3D-aware image synthesis. arXiv preprint: arXiv:2007.02442.
Shu, D.W., Park, S.W., Kwon, J., 2019. 3D point cloud generative adversarial network based on tree structured graph convolutions. In: Proceedings of the IEEE International Conference on Computer Vision, pp. 3859–3868.
Spurr, A., Song, J., Park, S., Hilliges, O., 2018. Cross-modal deep variational hand pose estimation. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 89–98.
Su, H., Maji, S., Kalogerakis, E., Learned-Miller, E., 2015. Multi-view convolutional neural networks for 3D shape recognition. In: Proceedings of the IEEE International Conference on Computer Vision, pp. 945–953.
Tatarchenko, M., Park, J., Koltun, V., Zhou, Q.-Y., 2018. Tangent convolutions for dense prediction in 3D. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 3887–3896.
Tatarchenko, M., Richter, S., Ranftl, R., Li, Z., Koltun, V., Brox, T., 2019. What do single-view 3D reconstruction networks learn? In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 3405–3414.
Thomas, H., Qi, C., Deschaud, J.-E., Marcotegui, B., Goulette, F., Guibas, L., 2019. KPConv: flexible and deformable convolution for point clouds. In: Proceedings of the IEEE International Conference on Computer Vision, pp. 6411–6420.
Valsesia, D., Fracastoro, G., Magli, E., 2018. Learning localized generative models for 3D point clouds via graph convolution. In: Proceedings of the International Conference on Learning Representations.
Wang, L., Li, Y., Lazebnik, S., 2016. Learning deep structure-preserving image-text embeddings. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 5005–5013.
Wang, X., Gupta, A., 2016. Generative image modeling using style and structure adversarial networks. In: Proceedings of the European Conference on Computer Vision, pp. 318–335.
Wu, J., Zhang, C., Xue, T., Freeman, B., Tenenbaum, J., 2016. Learning a probabilistic latent space of object shapes via 3D generative-adversarial modeling. In: Proceedings of the Advances in Neural Information Processing Systems, pp. 82–90.
Wu, J., Lin, Z., Zha, H., 2017. Joint latent subspace learning and regression for cross-modal retrieval. In: Proceedings of the International ACM SIGIR Conference on Research and Development in Information Retrieval, pp. 917–920.
Xiao, Y.-P., Lai, Y.-K., Zhang, F.-L., Li, C., Gao, L., 2020. A survey on deep geometry learning: from a representation perspective. Comput. Vis. Media 6 (2), 113–133.
Xu, M., Ding, R., Zhao, H., Qi, X., 2021. PAConv: position adaptive convolution with dynamic kernel assembling on point clouds. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 3173–3182.
Yang, G., Huang, X., Hao, Z., Liu, M.-Y., Belongie, S., Hariharan, B., 2019. PointFlow: 3D point cloud generation with continuous normalizing flows. In: Proceedings of the IEEE International Conference on Computer Vision, pp. 4541–4550.
Yang, Y., Feng, C., Shen, Y., Tian, D., 2018. FoldingNet: point cloud auto-encoder via deep grid deformation. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 206–215.
Yin, K., Chen, Z., Huang, H., Cohen-Or, D., Zhang, H., 2019. LOGAN: unpaired shape transform in latent overcomplete space. ACM Trans. Graph. 38 (6), 1–13.
Zhong, Z., Guo, X., Wang, W., Lévy, B., Sun, F., Liu, Y., Mao, W., 2013. Particle-based anisotropic surface meshing. ACM Trans. Graph. 32 (4), 99.