Artificial Intelligence Review (2024) 57:42
https://doi.org/10.1007/s10462-023-10687-x

Variational autoencoders for 3D data processing

Szilárd Molnár[1] · **Levente Tamás**[1]

Accepted: 30 December 2023 / Published online: 8 February 2024

Abstract
Variational autoencoders (VAEs) play an important role in high-dimensional data generation based on their ability to fuse the stochastic data representation with the power of recent deep learning techniques. The main advantages of these types of generators lie in their ability to encode the information with the possibility to decode and generalize new samples. This capability was heavily explored for 2D image processing; however, only limited research focuses on VAEs for 3D data processing. In this article, we provide a thorough review of the latest achievements in 3D data processing using VAEs. These 3D data types are mostly point clouds, meshes, and voxel grids, which are the focus of a wide range of applications, especially in robotics. First, we shortly present the basic autoencoder with the extensions towards the VAE with further subcategories relevant to discrete point cloud processing. Then, the 3D data specific VAEs are presented according to how they operate on spatial data. Finally, a few comprehensive table summarizing the methods, codes, and datasets as well as a citation map is presented for a better understanding of the VAEs applied to 3D data. The structure of the analyzed papers follows a taxonomy, which differentiates the algorithms according to their primary data types and application domains.

Keywords Variational autoencoder · Survey · 3D data

1 Introduction

Training deep learning models on higher dimensional spaces is a challenging task due to the runtime complexity. This is especially true for the 3D dimensional space with non-Euclidean constraints and irregular data distribution for which the direct application of the 2D convolution operators is not feasible. With the recent advances in the field of deep learning for generative models, the VAEs (Kingma and Welling 2014) has come into focus thanks to their ability to generalize data even in high-dimensional spaces. Variational Autoencoders are a class of deep learning generative methods based on variational methods

✉ Levente Tamás
levente.tamas@aut.utcluj.ro

Szilárd Molnár
szilard.molnar@aut.utcluj.ro

[1] Automation Department, Technical University of Cluj-Napoca, Memorandumului st. 28, 400114 Cluj-Napoca, Romania

(Mittal and Behl 2018) and enable the compressed representation of higher dimension spaces efficiently. Furthermore, with almost real-time processing capabilities, the VAEs prove to be efficient in scene completion, recognition, and segmentation tasks as well.

1.1 Motivation

The recent advances in inexpensive 2.5D depth sensors such as RealSense, Kinect, or Apple Prime, resulted in an increased interest in point cloud processing. As these sensors provide relatively high frame rate depth information, it is crucial for the fast processing of the spatial image stream. Currently, numerous methods are created to process 3D information, to reach the level of optimization and performance seen in the 2D domain. Compared to the 2D domain, the main disadvantage of the 3D methods, in addition to the later start, is the computational complexity needed for higher dimensionality of input data. Tutorials and surveys exist about VAEs, although they focus on the simpler task of 2D image generation with limited resolution. This paper provides an overview of existing 3D data generation methods using VAE architecture, which to the best of our knowledge is the first systematic overview of the VAEs for the 3D domain. Instead of including the wide range of 2D VAE methods, we provide cornerstone review papers from the 2D VAE domain highlighting the common aspects between the 2D and 3D domains.

In this article, we provide a systematic overview of the recent VAE-based spatial data processing methods. To ensure a proper focus of our investigation, we narrowed down the analysis to the VAEs operating on 3D discrete data such as point clouds or meshes. With this selection, we hope to provide a good starting point for readers interested in spatial data processing using Variational Autoencoders.

1.2 Structure of the article

The article gives an overview of the fundamental principles of the VAE in Sect. 2 assuming the basic knowledge of probabilistic distributions and neuronal networks. Besides the basic neural network variants, the extension towards specific VAEs, such as β-VAE, are discussed too, focusing on the problem of hyperparameter tuning for these methods. In the next Section, a thorough analysis of existing VAE methods for spatial data processing is presented by grouping the methods into different sub-classes according to the types of data on which they are operating. The taxonomy of this grouping is further presented in Sect. 3. The presented articles are summarized in a table according to the data types used on input, code availability, and the references on which these methods are based. Further on, a reference graph is provided highlighting the connections in time among different articles of VAE dealing with 3D data processing. Finally, the paper concludes with the current state-of-the-art overview and the future possibilities within this domain.

2 Background

Machine learning algorithms are *discriminative* or *generative* (Ng and Jordan 2001). A discriminative model extracts information about the data analyzed, such as classification. Generative models, on the other hand, generate new data based on the specific class distribution. In other words, generative models are often seen as the opposite of discriminative models. For this type of algorithm, we differentiate two major architectures: *Generative*

Adversarial Networks (GANs) (Goodfellow et al. 2014; Creswell et al. 2018) and *Variational Autoencoders (VAEs)* (Kingma and Welling 2014).

GANs have two competing modules: *generator* and *analyzer* (the latter is a discriminator model). The generative part is trained to create meaningful data that are similar to observed data, while the analyzer is trained to differentiate between generated and real data. Training them as competitors results in a generator model capable of creating high-quality synthetic data. Such an example is available in the work of Antal and Bodó (2021), where the authors created a method, that generates realistic faces, while also learning facial features. This method is useful for drawing composite sketches or data augmentation. Learning facial features has multiple significant applications, such as identity recognition. Further surveys exist on comparing face recognition methods (Li et al. 2022a).

Typically, GANs produce more photorealistic images than VAEs; however, GANs require more data and tuning than VAEs (Mi et al. 2018). The increased training efforts are caused by two main problems: unstable convergence and mode collapse. The latter means that if one of the modules (usually the discriminator) is too powerful, the other module fails to learn every aspect of the input data, as the training gradient vanishes. In other words, even if the generator creates higher-quality images than in the previous iteration, from the point of view of the discriminator, the real data are equally superior to the generated image as in the previous iteration.

VAEs also have two modules: *encoder* and *decoder*; however, in this case, they are not competitors. The encoder part is searching for few, yet meaningful variables to describe the characteristics of the input data, while the decoder is trained to reconstruct the original data from these variables. Different flavors of VAE produce comparable results to GANs, for example, Vector Quantized Variational Autoencoder (VQ-VAE-2) (Razavi et al. 2019). Furthermore, combinations of VAEs and GANs are proposed by Larsen et al. (2016), Makhzani et al. (2015), Zamorski et al. (2020).

The customizability of VAEs makes them suitable for higher-level generation tasks; therefore, the center of this work is the VAE architecture, thus we start by providing a theoretical background about VAEs (Doersch 2016; Kingma and Welling 2019).

2.1 Theoretical foundations of Variational Autoencoders

Variational Autoencoders, as the name suggests, evolved from autoencoders; therefore, our starting point is the autoencoder by Kramer (1991). It has two components: the encoder and the decoder, as shown in Fig. 1. The encoder compresses the data into a latent space using latent variables which are highly representative of the original data but more difficult to understand for the human observer. Meanwhile, the decoder extracts these variables, in other words,

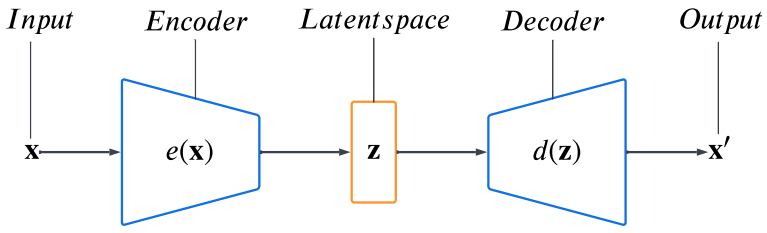


Fig. 1 The architecture of the AE model

reconstructs the original data. Hence, one of the main applications of autoencoders is data compressing.

The second use case is the denoising of data. Mafi et al. (2019) provide a survey on different denoising applications, among them, autoencoder-based methods are recommended due to their performance.

Third, autoencoder-based video generation methods are used in video forensic forgery detection applications (Javed et al. 2021). Similarly, Dhiman and Vishwakarma (2019) collect methods on abnormal human activity detectors both in 2D and 3D domains. The authors claim that autoencoder-based human activity detectors are sensitive to variations in the view.

The goal of the autoencoder is to recreate the original data; therefore, the metric used compares the input data with the output data, calculating the difference between them. This metric is called *reconstruction loss* (1), and in most cases *mean squared error* is used:

$$loss = ||\mathbf{x} - \mathbf{x}'||^2 = ||\mathbf{x} - d(\mathbf{z})||^2 = ||\mathbf{x} - d(e(\mathbf{x}))||^2 \tag{1}$$

where $\mathbf{x}$ is the input data, $\mathbf{x}'$ is the reconstructed data, $\mathbf{z}$ is the latent space, $e(\cdot)$ is the encoder and $d(\cdot)$ is the decoder.

For simple data reconstruction, this architecture is adequate; however, it is not sufficient for new data generation. The main reason is that the obtained latent space is not uniformly distributed but contains discrete data patches (Kingma and Welling 2014). The interpolation between these patches in the latent space results in highly meaningless and incomprehensible data at the output of the decoder. Therefore, to generate previously unseen data, the latent space must be organized (Kingma and Welling 2014).

The solution to the problem of the non-uniformly distributed latent space is approximating latent distributions rather than distinct values but finding these distributions takes an exponential amount of time, hence they are intractable in most cases. To solve this problem, Kingma and Welling (2014) in their pioneering work created the base of VAE. They proposed to find a simpler probabilistic distribution that is tractable and approximates the original probabilistic distribution, or, in other words, optimizes the *marginal likelihood* (4). This causes the latent space to be filled in a predictive manner, according to the chosen simplified probabilistic distribution, thus the interpolation between latent patches becomes possible, and results in comprehensible data at generation. This is called mean-field variational inference, and it is commonly used with the Gaussian probabilistic distribution, also known as the normal distribution: $\mathcal{N}(0, \mathbf{I})$.

As in (4) the marginal likelihood is composed of two terms: the *Kullback- Leibler divergence* (D_{KL}) (2) and the *Evidence Lower Bound (ELBO)* (3), also called *variational lower bound* (Kullback and Leibler 1951; Bulinski and Dimitrov 2021). These two components are inversely proportional, thus maximizing the ELBO results in minimizing the D_{KL}.

$$D_{KL}(q_{\boldsymbol{\phi}}(\mathbf{z}|\mathbf{x})||p_{\boldsymbol{\theta}}(\mathbf{z}|\mathbf{x})) = \mathbb{E}_{q_{\boldsymbol{\phi}}(\mathbf{z}|\mathbf{x})}[\log q_{\boldsymbol{\phi}}(\mathbf{z}|\mathbf{x}) - \log p_{\boldsymbol{\theta}}(\mathbf{z}|\mathbf{x})] \tag{2}$$

$$\mathcal{L}(\boldsymbol{\theta}, \boldsymbol{\phi}; \mathbf{x}^{(i)}) = \mathbb{E}_{q_{\boldsymbol{\phi}}(\mathbf{z}|\mathbf{x})}[\log p_{\boldsymbol{\theta}}(\mathbf{x}, \mathbf{z}) - \log q_{\boldsymbol{\phi}}(\mathbf{z}|\mathbf{x})] \tag{3}$$

$$\log p_{\boldsymbol{\theta}}(\mathbf{x}) = D_{KL}(q_{\boldsymbol{\phi}}(\mathbf{z}|\mathbf{x})||p_{\boldsymbol{\theta}}(\mathbf{z}|\mathbf{x})) + \mathcal{L}_{\boldsymbol{\theta},\boldsymbol{\phi}}(\mathbf{x}) \tag{4}$$

where $\mathcal{L}(\boldsymbol{\theta}, \boldsymbol{\phi}; \mathbf{x}^{(i)})$ is the ELBO, $\mathbb{E}_{q_{\boldsymbol{\phi}}(\mathbf{z}|\mathbf{x})}$ is the expected value operator with respect to the distribution q, $\boldsymbol{\phi}, \boldsymbol{\theta}$ are the parameters (weights) of the encoder and decoder, respectively, and $q_{\boldsymbol{\phi}}(\cdot), p_{\boldsymbol{\theta}}(\cdot)$, are the probabilistic distributions over the parameters $\boldsymbol{\phi}, \boldsymbol{\theta}$.

While training, the encoder produces two vectors describing the distribution: the mean $\boldsymbol{\mu}$ and the standard deviation $\boldsymbol{\sigma}$. Initially, the decoder samples several values from the latent space to reconstruct the information. But this sampling creates a nondeterministic node $\mathbf{z}$ on the path of data, which makes backpropagation impossible through that node. Kingma and Welling (2014) propose *reparameterization trick* (5) to solve this problem. They extract the sampling to a separate node with normal distribution $p(\boldsymbol{\epsilon})$, therefore, the backpropagation becomes possible.

$$\mathbf{z} = \boldsymbol{\sigma}\boldsymbol{\epsilon} + \boldsymbol{\mu} \tag{5}$$

The architecture of a VAE is shown in Fig. 2.

2.2 Variational autoencoders extensions

Although, the basic VAE is considered a powerful architecture compared to simple autoencoders, room for improvement by expanding the architecture exists. The first variant is β -VAE, which balances the capacity of the latent channels and the independence constraints, which describe the relations of the features decoded inside the latent space. Further architectural modifications such as Adversarial Autoencoder (AAE), two-stage VAEs, and hierarchical VAEs also exist.

2.2.1 β-VAE with hyperparameters

A method that is focused on generating new 2D images that are similar to existing images is called β-VAE proposed by Higgins et al. (2017). The β is a hyperparameter that is used to balance latent channel capacity and independence constraints while discovering more latent features. The drawback of this method is the dependence on the sampling density and the correct choice of β. In this β-VAE method, the authors mention that the data generation of VAEs is often referred to as learning a disentangled representation of the latent space, as latent units are sensitive to one generative factor, but insensitive to other factors, that is, the insight for a generative VAE considered disentangled VAE (Burgess et al. 2018). In Fig. 3, handwritten digits generated using the basic VAE and β-VAE methods are shown. The digits look like a mixture of many other digits, proving that a VAE interpolates the latent space. Furthermore, the basic VAE also generates unrealistic-looking digits, which shows its entangled nature, whereas the β-VAE generates more realistic digits by disentangling the latent space.

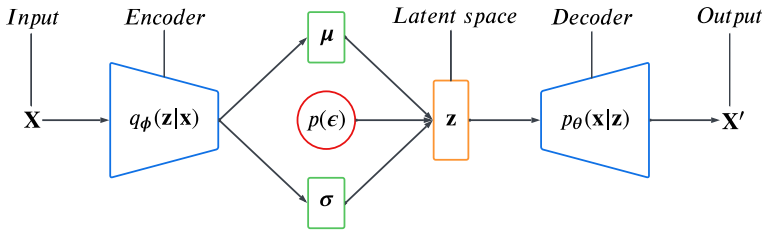


Fig. 2 The architecture of the VAE model. The probabilistic encoder and decoder are marked with blue, the latent space $\mathbf{z}$ with yellow, the mean $\boldsymbol{\mu}$ and standard deviation $\boldsymbol{\sigma}$ with green, and the stochastic node $p(\boldsymbol{\epsilon})$ with red

(a) VAE (b) β-VAE ($\beta = 150$)

Fig. 3 Different digits were generated using basic VAE, and β-VAE by Spanopoulos and Konstantinidis (2021)

The hyperparameter β is an additional extension for reparameterizing the VAE for creating disentangled latent space representation. Another hyperparameter called C is used to control the information capacity of the latent space $q(\mathbf{z}|\mathbf{x})$. These parameters are proposed by Higgins et al. (2017) as follows in (6):

$$\mathcal{L}(\boldsymbol{\theta}, \boldsymbol{\phi}; \mathbf{x}, \mathbf{z}, \beta, C) = \mathbb{E}_{q_{\boldsymbol{\phi}}(\mathbf{z}|\mathbf{x})}[\log p_{\boldsymbol{\theta}}(\mathbf{x}|\mathbf{z}) - \beta |D_{KL}(q(\mathbf{z}|\mathbf{x})||p(\mathbf{z})) - C|] \tag{6}$$

2.2.2 Adversarial autoencoders

Architectural modifications are also used to expand the capabilities of VAEs. Mi et al. (2018) provide a comparison between GANs and VAEs, and their derivatives such as Wasserstein GAN (Arjovsky et al. 2017) or the best resulting quality method, VAE-GAN (Larsen et al. 2016). Another mixture of a VAE and a GAN is called AAE architecture (Makhzani et al. 2015). The difference lies in the introduction of a discriminator module (D). This module decides whether a given object is real or fake in the *regularization* phase. In this phase only the discriminator (D) is updated, the encoder and decoder modules are updated in the next cycle, called *reconstruction* phase. Also, in this case, the decoder is called generator (G). Therefore, the reconstruction loss is supplemented by the adversarial training criterion from the GAN architectures(7):

$$\min_G \max_D E_{\mathbf{x}\sim p_{data}}[\log D(\mathbf{x})] + E_{\mathbf{z}\sim p(\mathbf{z})}[\log(1 - D(G(\mathbf{z})))] \tag{7}$$

2.2.3 Two-stage VAE

The basic intuition behind the Two-stage VAE (Dai and Wipf 2019) is that the Gaussian encoder and decoder assumption reduces the effectiveness of the model, hence the authors propose the two-stage VAE architecture, where the first stage learns the manifold

for each allowance, thus providing a mapping to a lower-dimensional representation. The second stage is smaller and converts the low-dimensional data into latent values. This work ensures that a two-stage VAE produces high-quality images without the blurred edges specific for VAEs, although GANs are still superior in this context to the two-stage VAE.

2.2.4 Hierarchical VAE

A hierarchical VAE, proposed under the name of Nouveau VAE (Vahdat and Kautz 2020), is similar to a two-stage VAE. This method ensures that the architecture has multiple latent spaces on different levels. All of them have the usual elements, like *reparameterization trick*. The residual blocks are situated between the latent layers, and they act like very simple encoder or decoder between the layers. Inside the model, two paths are defined: bottom-up and top-down. The advantage of multiple latent spaces is that each of them learns a specific feature; thus, in a generative process, the generative task itself is controlled to a much higher degree, resulting in more realistic and sharper data.

2.2.5 Other VAE variants

The recent advances of the generative models has an impact on the VAE models, such as VAEs, as described so far. However, more alterations of this architecture exist Wei et al. (2020); Wei and Mahmood (2021a), that we enumerate in a non exhaustive manner:

1. *f-VAEGAN-D2* Xian et al. (2019)
2. *InfoVAEGAN* Ye and Bors (2021)
3. *Adversarial Symmetric VAE*Pu et al. (2017)
4. *Lifelong VAE-GAN*Ye and Bors (2020)

2.3 Overview of existing VAE surveys

Multiple applications benefit from the capabilities of VAEs in the field of 2D and 3D data processing. For the 3D domain, the main applications include model generation, object deformation, classification, and segmentation. To our knowledge, no 3D specific VAE survey exists, although considerable research output on 2D domain specific VAEs is available. The existing 2D-specific VAE surveys are presented in this section as a preliminary step to the 3D domain-specific part.

2.3.1 Wei and Mahmood: recent advances in variational autoencoders with representation learning for biomedical informatics: a survey

A survey by Wei and Mahmood (2021b) focuses on different biomedical measurements. The use of VAEs in biomedical informatics goes from data generation to representation learning. The authors differentiate three major categories, such as molecular design, sequence dataset analysis, medical imaging and image analysis, and several different subcategories, including string representation design, graph representation design, sequence engineering, dimensionality reduction, integrated multi-omics data analysis, mutation effect prediction, gene expression analysis, DNA methylation analysis, image augmentation for a downstream task, representation learning for decision making. One important notice is that a molecule can be seen as a graph and treated as one using a VAE.

2.3.2 Asperti et al.: a survey on variational autoencoders from a green AI perspective

The next survey, by Asperti et al. (2021), compares the different methods mainly from a power efficiency standpoint, like current consumption, describing the basic concepts of the VAEs and possible improvements. After testing the different methods, the authors of the survey concluded notable characteristics of the VAE. Firstly, in their experience, the decoder is more important than the encoder, and increasing the number of the latent variables improves reconstruction, but does not necessarily affect the generation.

The idea of comparing architectures from the efficiency standpoint and current consumption comes from Green AI Schwartz et al. (2020), which states the importance of the trade-off between the performance and efficiency of a model, and how this is affected by design, model description, and mathematical formulation. The metric for modeling the efficiency from a Green AI perspective is the computation of the Floating Point Operations (FLOPS), which does not depend on the hardware, yet it correlates with the runtime, even though it is highly affected by memory accessing time. So the goal is to achieve a higher number of FLOPS.

Furthermore, Asperti et al. mention that the training of a VAE is expected to be improved by using *von Mises-Fisher* from the *Hyperspherical VAE* (Davidson et al. 2018) instead of the Gaussian distribution. Besides this, Hou et al. (2017) states that using deeply hidden features extracted from other pretrained image processing modules overcomes blurriness. The authors of this survey also emphasize the importance of the *running average* method, which maintains a balance between the reconstruction loss and the D_{KL}, since if the D_{KL} does not decrease while the reconstruction loss does, the improvement of the model will be prevented. This is known as the phenomenon *variable collapse*.

2.3.3 Kovenko and Bogach: a comprehensive study of autoencoders' applications related to images

The third survey on autoencoder-based 2D image processing methods was created by Kovenko and Bogach (2020). The authors notice, that Bernoulli distribution was not considered before in VAEs, but it works better for grayscale images, such as the MNIST dataset, while for RGB data the Gaussian distribution is suggested. A so-called *warmup* phase is mentioned, where they gradually modify a β variable (Sønderby et al. 2016), unlike the original article (Higgins et al. 2017), where β is kept constant. This is done to prevent latent units from becoming inactive during training, which would otherwise lead to a poorly trained model. The authors show that the *warmup* phase improves training and validation losses by approximately 7%.

3 Relevant 3D methods using variational autoencoders

The most common data types that generative models, like VAEs, learn are 2D images, 1D audio signals, and text semantics.

However, being in a time, when the advance in 3D image processing techniques is fast, our main focus is on generative models that work with 3D objects, represented with depth images, meshes, or point clouds. An overview of the different 3D representations is provided by Friedrich et al. (2018). Each of them has an advantage for a specific task,

depending on the available resources or the given architecture. The conversion is possible between different types of representations if needed. The conversion is simple for specific cases, for instance from depth image to point cloud, and more tedious for other cases, for instance from point cloud to polygon mesh.

In the following part, we provide an overview of different techniques that are using VAE architectures for processing 3D data. We organized them into categories according to the type of data and scope in order to have a valid taxonomy. The first dimension characterizes the primary data types used in the analyzed papers, while the second dimension focuses on different types of applications and data processing domains. In 3D, we can specify 3 popular data formats, which are: polygon mesh, voxel grid, and point cloud. These 3 formats represent 3 distinct categories in the first taxonomical dimension, while the other data types build up a fourth category. This division is presented in Fig. 4. Later, an overview is provided about the analyzed methods, where multiple tables correspond to the first dimension of the taxonomy.

3.1 Polygon mesh transformations

In computer graphics and CAD applications, 3D objects are often stored using polygon meshes. This representation technique is slightly more complex than other types, such as depth images or point clouds, but the relationships of the edges, faces, and vertices are well-defined, and therefore the complex objects are stored with high accuracy. This well-defined nature of a mesh makes the objects easily deformable by applying a transformation on a given vertex or edge.

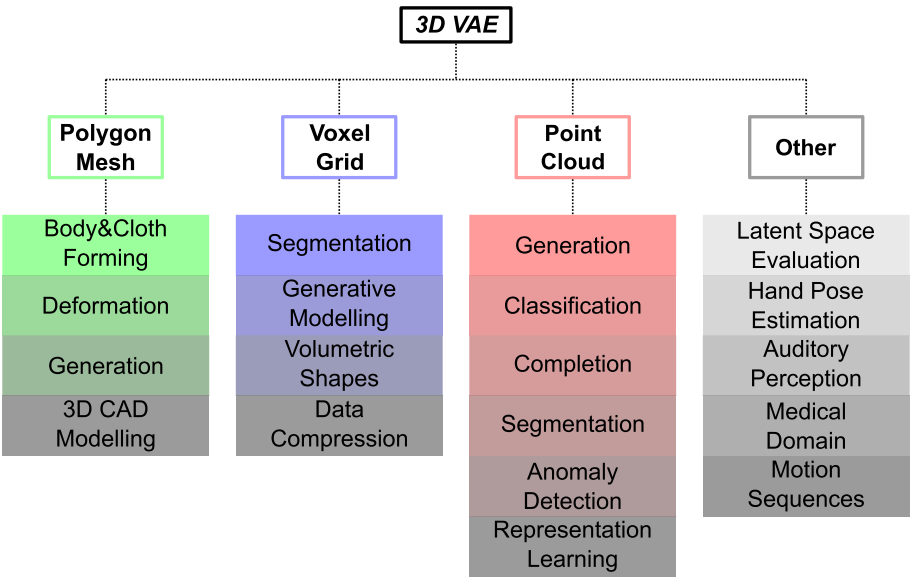


Fig. 4 The taxonomy of this survey. The first level represents the 4 major data type categories, while the next level refers to the application domains

3.1.1 Mesh deformation

The first method that we analyzed was created by Tan et al. (2018), and it deforms the 3D mesh models into other shapes; more specifically, the method deforms human poses. The method, called *mesh VAE*, explores the probabilistic latent space of 3D mesh surfaces. The reasoning behind the choice of the data type is that the deformation operation on meshes is a straightforward process, while the voxel representation creates too rough edges. This method is based on an advanced surface representation technique, *Rotation Invariant Mesh Difference (RIMD)* (Gao et al. 2016), which, as the name suggests, is translation and rotation invariant. This representation recovers the shapes accurately and efficiently by solving the nonlinear optimization *as-rigid-as-possible* (Sorkine and Alexa 2007), and applying the rotation transformation Murray et al. (1994). The combination of mean-squared error as a loss function (8) and the hyperbolic tangent (*tanh*) as the activation function results in a method that generates new poses from the existing meshes of human bodies. As an extension, this method is completed by an adjustable parameter, enforcing specific features in the latent space. Thus, mesh VAE becomes more controllable. A further update of this method is to develop the ability to process non-homogeneous meshes, since the time of writing, the architecture only works with homogeneous meshes.

$$L_{meshVAE} = \alpha \frac{1}{2MK} \sum_{j=1}^{M} \sum_{i=1}^{K} (\mathbf{g}_i^j - \mathbf{h}_i^j)^2 + D_{KL}(q(\mathbf{z}|\mathbf{h})||p(\mathbf{z})) \tag{8}$$

where $\mathbf{h}$ is the preprocessed RIMD feature, $\mathbf{g}$ is the output of the VAE framework, M is the number of models, K is the number of feature dimensions and α is the tuning parameter.

Furthermore, RIMD states that the difference in the deformation of a model from the reference model is the energy needed for the transformation (9). Then the deformation is decomposed into a rotation and a shearing part.

$$E(\mathbf{T}_i) = \sum_{j \in N_i} c_{i,j} ||\mathbf{e'}_{ij} - \mathbf{T}_i \mathbf{e}_{ij}||^2 = \sum_{j \in N_i} c_{ij} \sum_{t \in N_i} c'_i ||\mathbf{e'}_{ij} - \mathbf{R}_t dR_{ti} \mathbf{S}_i \mathbf{e}_{ij}||^2 \tag{9}$$

where $\mathbf{T}_i$—deformation gradient in the one-ring neighborhood; N_i—one-ring neighbors of vertex v_i; $\mathbf{e'}_{ij} = \mathbf{p'}_i - \mathbf{p'}_j$—for the deformed model; $\mathbf{e}_{ij} = \mathbf{p}_i - \mathbf{p}_j$—for the reference model; $c_{ij} = \cot \alpha_{i,j} + \cot \beta_{ij}$—cotangent weights ($\alpha_{ij}$ and β_{ij} are angels opposite the edge connecting v_i and v_j); $\mathbf{R}_i$—rotation; dR_{ti}—rotation difference; $\mathbf{S}_i$—scaling or shearing; $c'_i = \frac{1}{|N_i|}$.

By minimizing the energy, one gets the feature representation as (10):

$$g = \{\log dR_{ij}; \mathbf{S}_i\} (\forall i, j \in N_i) \tag{10}$$

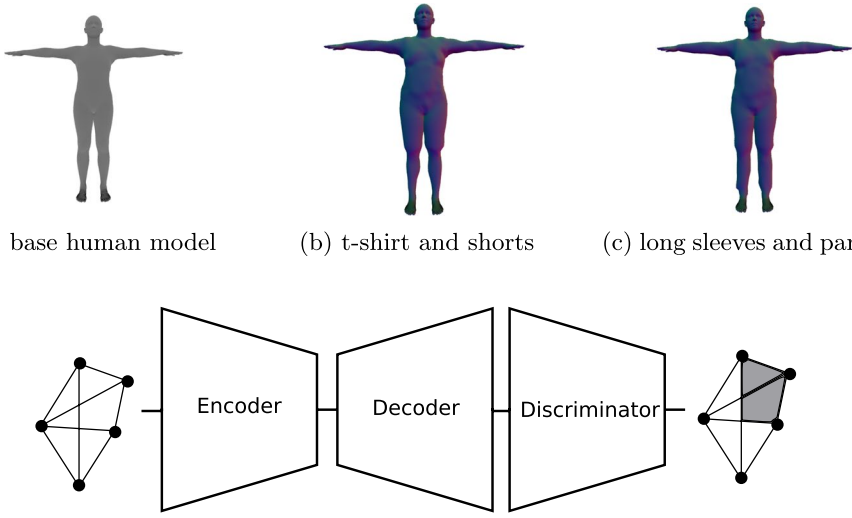
3.1.2 Body and cloth moulding with CAPE

Ma et al. (2020b) describe a method, called *Clothed Auto Person Encoding (CAPE)*, that generates different human poses, specifically with different styles of clothing on, modeling even the wrinkles, Fig. 5. The basis of this method is the *Skinned Multi-person Linear Model (SMPL)* (Loper et al. 2015), which is a novel vertex-based body representation format. Unlike the usually regression-based methods, the CAPE is rather a probabilistic generation task, which results in a model that generates multiple clothing deformations for a single pose and body type, unlike the previously achieved one-to-one

correspondence. By expanding the SMPLify method (Bogo et al. 2016), they achieve better estimates of the generation of human bodies and clothes. Ma et al. (2020b) describe clothing with the clothing displacement graph, which is processed by an architecture that is a combination of the VAE and the GAN architectures; thus, the authors gave this method a second name, Mesh-VAE-GAN. This means that after the encoder-decoder part of VAE, they introduce a discriminator, to improve the deformation quality. However, we can mention a few limitations to this method. Firstly, CAPE has an offset-based limitation in representing clothes, hence a multi-layer model is in need of development to overcome this drawback. Secondly, the mesh resolution of SMPL limits the quality of the CAPE, hence, depending on the scenario, a higher mesh resolution is required. Thirdly, this method is based on poses, while dynamic motions are currently out of reach.

3.1.3 Mesh generation and edge contraction pooling

For better 3D shape generation and a better understanding of probabilistic latent spaces, we need a reliable pooling operation. This is what Yuan et al. (2020) proposed for meshes using VAEs, based on (Garland and Heckbert 1997). The method uses mesh simplification to create a hierarchy, avoiding irregular-sized triangles in the mesh. The method is effective for denser meshes because the number of parameters used is less than usual. The hierarchy of different layers means that different levels of detail are feasible for the model to analyze, thus keeping track of the mapping between coarser and finer meshes. In (Garland and Heckbert 1997), repeated edge simplification is used, which is modified to produce evenly



(a) base human model (b) t-shirt and shorts (c) long sleeves and pants

(d) The architecture: a mesh of subtracting the base human model from the clothed model is fed into a VAE, which is the generator part (with the E and D modules). Then the discriminator module performs the patchwise classification

Fig. 5 Dressing as a human in clothing (Ma et al. 2020b)

sized triangles by introducing the length of the edges into the metrics. Yuan et al. (2020) defined the de-pooling operation as the inverse of the pooling operation. The structure of the method is completed with spectral graph convolutions from (Defferrard et al. 2016) instead of fully connected layers. This method is expanded by the possibility to generate shapes, by implementing a Conditional Variational Autoencoder (CVAE) architecture, where the user defines the parameters that must be taken into account when generating a new shape. Also at the input, this method uses vertex-based deformation feature representation (Gao et al. 2019). While Yuan et al. (2020) produce better quality meshes than (Garland and Heckbert 1997), only homogeneous meshes are processable due to the mesh simplification-based pooling.

The industry lacks research and data consisting of 3D faces or surfaces. This is the reason why Park and Kim (2021) created the Face-based Variational Autoencoder (Face-VAE) model that generates new 3D face data for industrial sites. This method uses polygon meshes, as the authors consider this type of data representation for faces to be a rarity, unlike 2D image-based, voxel-based, or point-cloud-based models. The main building components of this method are adjacency matrices and feature matrices with the graph data structure. The data are converted into binary format, still, good results were obtained by focusing on the structurization aspect of the VAEs instead of the pure generation aspect of them. The binary conversion is similar to a voxelization step, although this is in a specific matrix form. This model works optimally with vertices of 300 or less, for higher resolutions the octree concept might be a suitable candidate.

3.1.4 3D CAD model retrieval

CAD models are commonly based on polygon meshes, and they are often used in engineering, such as designing parts and assemblies in production. For these parts, the existence of the schematic is required, together with the loops of the task done to achieve the given part. For the occasion of a missing schematic or missing process tree, Qin et al. (2022) proposes a method, that retrieves information about a given 3D CAD model, including loop attributes, loop structures, and loop relation trees. This method also describes the structural semantic information of a part, by training a VAE-based recursive neural network (RvNN) (Socher et al. 2011) with backpropagation through structure (BPTS) (Goller and Küchler 1996). Therefore at the output of the model, a sketch/view is obtained about the input CAD model. This paper also utilizes a type of parameter tuning to a great extent, namely the cyclical learning rate (CLR) (Smith 2017). This approach further requires model optimizations and an implementation of more than one retrieval mode, such as coarse-grained and fine-grained retrievals.

3.2 Voxel grid processing

Another well-known 3D data representation format is the voxel grid. The 3D space is partitioned into a grid, where each cell stores a point or not. This type of organization is similar to a 2D image, where each pixel has a defined place. Although voxel grids are considered well organized, the stored information is not continuous enough. Still, because they are well organized, voxel grids are widely used, especially in discriminative tasks, such as segmentation.

3.2.1 Voxel segmentation

Voxel segmentation is simpler than data generation, yet an optimized method significantly helps other methods by supplying fast and accurately segmented labels. Such a segmentation method is the work of Meng et al. (2019). As a basis of the architecture, they used VAE and Radial Basis Function (RBF) (Buhmann 2000) (11), (12). The authors complain about the ineffectiveness of point clouds due to their unordered nature and lack of symmetry; thus, a voxel grid is proposed. Typically, a voxel only contains a boolean occupancy status, which is not descriptive enough, but with RBF, they are interpolated, which leads to a representation with more resolution.

$$rbf(\mathbf{c_p}) = \sum_{j=1}^{N} w_j \eta(||\mathbf{c_p} - \mathbf{v}_j||_2^2) \tag{11}$$

where N is the number of data points, w_j is the scalar value, $\eta(\cdot)$ is the symmetry function, $\mathbf{c_p}$ is the center point of a voxel, and $\mathbf{v}$ are the data points and:

$$rbf(\mathbf{c_p}) = \max_{v \in \mathbf{V}} \left(\exp \frac{||\mathbf{c_p} - \mathbf{v}_j||_2^2}{2\delta^2} \right) \tag{12}$$

where V is the set of points and δ is the predefined parameter (usually a multiple of the size of the subvoxel).

Furthermore, in this method, *Group Equivariant Convolutional Neural Network (G-CNN)* (Cohen and Welling 2016) is used, which improves the invariance against rotation, translation, and scaling, by capturing the intrinsic symmetry of point clouds. In general, G-CNNs increase the expressive capacity of the given network while maintaining the number of parameters. In this particular case, their role more specifically is to detect co-occurrences in the latent space. By doing so, the method reduces the number of parameters without reducing the information density.

In addition, the Multilayer Perceptron (MLP) functions (Ruck et al. 1990) are used to combine the per-point features. RBF-VAE is the first module, and the segmentation (group convolution) is the second one, they are trained separately. As the loss function, they have used cross-entropy loss for segmentation (13):

$$l_{ce} = -\sum_{l}^{L} \mathbf{p}_{\mathbf{gt}_l} \log \mathbf{p}_l \tag{13}$$

where L is the number of labels, $\mathbf{p}_{\mathbf{gt}_l}$ is the probability of the ground truth label and $\mathbf{p}_l$ is the probability of each label.

While the work of Meng et al. (2019) presents a robust segmentation method, its performance is reduced in certain shapes, most probably due to the encoded 90° symmetry. Further experiments are planned to verify if this method is suitable for surface normal estimation.

3.2.2 Generative and discriminative voxel modeling

Brock et al. (2016) realized the importance of choosing the correct representation for 3D data. Hence, they are experimenting with the viability of using voxel grids with VAE

architecture. They analyze not only the quality of the latent space created by the encoder but also the interpolation capabilities of the decoder in the task of shape generation and classification.

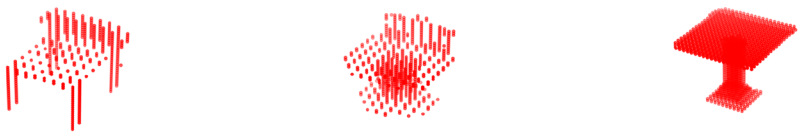
The created model is well suited for generating dense objects but struggles with thin shapes, like table legs and sharp edges. The reasoning is that these features fail to be encoded correctly in the latent space, so a loss of local features is expected. The smoothness of the interpolation between two shapes suggests that the grouping and coverage of the latent space meet expectations. The classification task tends to perform slightly worse with few training data than the Orientation-boosted Voxel Net (ORION) (Sedaghat et al. 2017) method but outperforms it after introducing more training data. Proposed future works consist of improved resolution, spatial occupancy data, adding data augmentation, and experimenting with different Voxception architectures.

3.2.3 Volumetric shape generation

The next method, proposed by Guan et al. (2020), is derived from the work of Wang et al. (2014), by lifting it from conventional 2D images to 3D voxel grids. This architecture is superior to a simple autoencoder in that it further analyzes the data connections, and instead of mapping each data to a discrete point in the latent space, a set of instances is mapped, so this architecture lies between the traditional AE and VAE architectures. Thus, we achieve a model that interpolates between shapes but maintains edge sharpness better than a simple VAE, as shown in Fig. 6. The model is capable of extrapolating to a certain degree, creating new unseen data. The latent space, as expected, shows a good grouping of similar characteristics. The drawback of this method, compared to VAE is the longer training time, and larger memory consumption, thus adapting it to large 3D models leads to complex training. The usage of voxel grids helps to improve the performance, but it leads to a fragmented look. As a future work, the authors propose to optimize the model to reduce the memory and time required for training the model, or to use higher resolution voxels. Another idea is to experiment with semantic relations besides the Chamfer distance.

3.2.4 Data compression

One of the main applications of autoencoders is compression. Depending on the data types different architectures are used for compressing images, such as CNNs, Recurrent Neural Networks (RNNs), GANs, or VAEs. Mishra et al. (2022) provide a comprehensive study on these methods; unfortunately, 3D applications are rarely mentioned. Wang et al. (2021b)



(a) Example 1 - bed (b) Example 2 - chair (c) Example 3 - table

Fig. 6 Generating objects as voxel grids using 3D-GAE (Guan et al. 2020)

expanded the idea of data compression based on autoencoders for 3D using VAEs, to obtain better performance by applying them to point clouds. Although this method is officially for point clouds, pre-processing and post-processing phases are necessary to voxelize the point clouds, and optionally scale or partition them. The basic 3D convolutional unit of this method is *Voxeption-ResNet (VRN)* (Brock et al. 2016). Nine of these are used for analysis and synthesis transforms, which are then used in the encoder and decoder. Quantization is the process when instead of rounding a representation, they add a uniform noise, thus maintaining the ability to backpropagate through the model. As a metric, *Weighted Binary Cross-ntropy (WBCE)* loss is used. Furthermore, an adaptive module takes care of the distortions and removes unnecessary voxels according to a threshold. Standardized geometry-based compression algorithms such as *Moving Picture Experts Group (MPEG)* exist that are outperformed by this method by at least 60% in the perspective of the *Bjöntegaard Delta Rate (BDR)*. While traditional codecs perform better in the evaluation of *human visual systems*, deep neural network-based approaches are better at reconstructing spatial, textual, and structural features. On the other hand, this model struggles with *joint rate-distortion optimization*, which is a feature of traditional codecs.

Working with voxels leads to high-performance requirements, which are unfeasible in lightweight scenarios like robotic navigation. Liu et al. (2020a) tackle this problem by introducing a VAE-based voxel compressor. The authors studied different 3D data capture systems, calibrations, and data formats, and concluded that for capturing 3D data, the stereo or depth cameras are optimal and that despite the larger amount of data in the case of adding dimensions for voxel grids, they tend to be relatively easy to work with. Furthermore, using *octomap* from Robot Operating System (ROS), they convert the voxels into *octree* representation (Wurm et al. 2010), as homogeneous *octants*. This lossless compression is then followed by the compression with VAE. One minor drawback is that the reconstructed voxel grid is usually sparser than the original grid, most probably due to overfitting.

Data compression in a real-time application is a desired task, especially if a given robot needs to pass information to another one. For remotely controlled robots, the data needs to reach the operator in near real-time, but with information captured with a Light Detection and Ranging (LIDAR), depth camera, or even with a conventional 2D camera, this task is expected to be slow. Yu and Oh (2022) created a data compression method that takes advantage of the capabilities of VAEs by combining this architecture with an approach *anytime estimation* (Larsson et al. 2017), thus achieving over-compression of the data if necessary. The authors studied category-specific multimodal prior VAEs (Yu and Lee 2018; Yu et al. 2019b; Yu and Lee 2019) to obtain missing elements that occur in harsh environments. In the training phase, missing data points are simulated by applying dropouts to the model. For better scene understanding, the approximation of the correct modals is important, so that they are separated in the latent space; thus, distinguishing categories becomes easier. This method slightly outperforms simple AE and VAE models in most tests, but if the missing rate is below 10% or above 90%, the accuracy of the method of Yu and Oh (2022) could drop below competing models.

3.3 Point cloud processing

Perhaps the most important task a VAE performs is to generate new data. By analyzing existing shapes, the model learns basic features and principles about the objects with which we are training. In many cases, the operator is allowed to set multiple variables to force

the desired characteristic on the generated object. Most of these methods work with point clouds since their unordered nature ensures that the models generate the coordinates of each point independently, but the relationship between neighboring points is maintained. Below we present several methods that generate new point clouds using VAE architectures.

3.3.1 Point cloud generation

CompoNet by Schor et al. (2019) is a part-based 2D or 3D object generator, where the final object is composed of the smaller priors, that are generated separately, thus they are considered to be deformable. The part-based nature improves the variability and quality of the final object. The model has two parts, first, the distinct generative module at the part level, and second, the conditional part composition network, which is based on a spatial transformer network (Jaderberg et al. 2015), which creates new shapes. The main advantage of this method is that the training is specified for lightweight datasets, whereas other methods struggle to learn a proper representation due to the lack of data. As future ideas, the authors mention limitations such as part diversity, part structure modification, and feature transfer between classes.

The next method by Saha et al. (2020), which was later expanded in (Saha et al. 2022), called *Point Cloud Variational Autoencoder (PC-VAE)*, uses VAEs to help 3D car designers by generating new unseen models from existing models. Naturally, the method is not limited to car body shapes, but in itself, due to its complexity, the body of the car deserves higher attention, since it has to be designed to be aerodynamically competent, safe, and pleasant to the eye. The VAE is used to specify local features in the latent space, not just global ones. By combining optimization and MLP, the operator instructs the model to generate a specific car type that is different but real-like. This method works with point clouds since they are flexible and require less memory than a voxel grid or polygon mesh. The authors consider (Schor et al. 2019) closest in terms of model generation. The work is further based on a point cloud autoencoder proposed by Achlioptas et al. (2018), that is further advanced in (Rios et al. 2019a, b). The PC-VAE architecture consists of five 1D convolutional layers (as described in (Qi et al. 2017a)), with ReLU after each layer (Nair and Hinton 2010), and at the end a batch normalization layer (Saha et al. 2019). The decoder part consists of three fully connected layers. Although this approach has a huge potential, further optimization is needed to create a complete cooperative framework.

Yang et al. (2019a) tackle the problem of generating realistic 3D point clouds by encoding them as the distributions of shapes, which in turn are the distributions of points. Thus, the method samples according to a shape or the number of points. The method is called *PointFlow*, and it greatly utilizes the advantages of normalizing flows (Rezende and Mohamed 2015) in the VAE architecture. This method might not exceed the accuracy of state-of-the-art approaches, but it has great potential in the usage of image-based point cloud reconstructions.

The problem of generating complex shapes is also addressed by Li et al. (2022b), who propose EditVAE, which introduces part awareness into the model, in other words, the object generation task is divided into the part generation task, where the parts are assembled following a schematic, and each part has its own disentangled latent representation, which also learns the dependencies between parts. With standardized transformation, the authors achieved part-aware point cloud generation and shape editing by interchanging parts between objects. Previously, an attempt was made to do this using (Gal et al. 2021), but that method generated semantically insufficient parts. The authors use

a primitive-based point cloud segmentation derived from shapes created by Paschalidou et al. (2019). Objects are built from generic primitives, such as cubes and spheres; on them, a superquadric parameterization and other deformation parameters (Barr 1984) are computed. An inductive bias (Locatello et al. 2019) is used to disentangle the latent representation, where semantically meaningful parts are classified by shape or pose. An approximate posterior is achieved via a simple deterministic mapping (Nielsen et al. 2020), and via the usage of PointNet architecture (Qi et al. 2017a) as the posterior, and TreeGAN (Shu et al. 2019) as the decoder-generator module. EditVAE in most cases outperforms the state-of-the-art methods, but further optimization is required to expect the best accuracy across the board.

3.3.2 Point cloud classification and generation

The next method is from Gadelha et al. (2018), which has an encoder capable of classifying point clouds, as well as a decoder to generate new point clouds from conventional RGB images. The latter is also called *image-to-shape inference* (Simonyan and Zisserman 2015). Both branches are based on *kd-trees* (Klokov and Lempitsky 2017), which divides the data in such a way that each split is along one of the axes; in other words, the split is alternated between the three axes. Instead of kd-trees, *rp-trees* (Dasgupta and Freund 2008) are also possibilities, the difference being that in this case, the splits are not strictly along an axis, but more arbitrary, for instance along an edge. The tree structure represents a locality preserving a 1D ordered list of points, that is efficient in feed-forward processes using 1D convolutions. The splitting results in a probabilistic kd-trees, which performs multiplicative transformations while maintaining a good scaling quality.

The main reason why this method is based on point clouds rather than voxel representation is that voxel grids scale poorly and do not model surface details, without additional extensions like RBF. Also, because of self-occlusions, image-based representations are insufficient for modeling anything other than the surfaces seen from the sensor. Although the accuracies are decent, further optimization is possible.

The architecture of this method is built using various implementations and ideas: *multigrid* network (Ke et al. 2017), *multiscale* (Lin et al. 2017; He et al. 2015), *dilated* (Yu and Koltun 2016) and *atreus* (Chen et al. 2018) filters. Multigrid networks mean that the data are converted into multiple resolutions, and the architecture of the model is the connections of pyramids with different resolutions. This results in an architecture where at the feed-forward processing, a change in a certain resolution influences the change in other ones, adjacent resolution layers as well. In other words, scaling and learning of the global-to-local feature ratio is managed. The dilated filters in the convolutions are specifically meant for dense predictions, as in the case of point clouds, where the multiscale information is relatively hard to gather without loss of resolution. Atreus filters or upsampled convolutions are working with dilated filters as they make it easier to choose the correct resolution, without drastically increasing the parameters. Often, neural networks require specific layer sizes or scales, but multiscale networks cope with arbitrary layered sizes.

For the shape classification task for each point cloud, approximately 10^3 points are sampled using Poisson disk sampling (Bowers et al. 2010). The splitting into probabilistic kd-trees comes next. In training, the model minimizes cross-entropy loss. Anisotropic scaling factors are applied as scale augmentation. The model creates the kd-trees of a given point cloud 16 times, and the final version is the average of these trees. For this task, the ModelNet (Wu et al. 2015) dataset is used.

For image-to-shape inference, the encoder is borrowed from Visual Geometry Group (VGG) (Simonyan and Zisserman 2015); however, the decoder is proprietary. The generated shapes are set to have $4 \cdot 10^3$ points. For this task, the ShapeNet dataset is used, along with the Chamfer Distance (CD) (14) as the loss function. The combination of these two tasks is used for unsupervised learning applications.

3.3.3 Point cloud completion

By Han et al. (2019) PC-VAE was proposed, which trains its VAE architecture by creating multiple viewpoints for each 3D object, then the method divides them into a front half and a back half, based on the geodesic distance. The Euclidean distance is ignored because it causes non-semantic splitting, as suggested by Crane et al. (2017). By learning to reconstruct parts of the objects from different angles, this method simultaneously learns global and local features, along with the relationship between them, like geometric and structural information, resulting in unsupervised training, which makes the model capable of generating new objects as well.

The authors chose point clouds in this case because point clouds are relatively easy to create, collect, and process. Therefore, the encoding of the aforementioned front and back halves is based on PointNet++ (Qi et al. 2017b), which is an extension of PointNet (Qi et al. 2017a). Then, given the encoding of the front half, the decoder is trained to reconstruct the back half. This method considers the Earth Mover's distance (EMD) (Rubner et al. 2000) (15) to be more suitable than the CD (14) since it results in better visual quality.

This half-to-half prediction and self-reconstruction are achieved using three branches: aggregation, reconstruction, and prediction. The aggregation branch contains two encoder, one for local features and another for global features, extracted from the front half of each sample point cloud. Additionally, this branch is completed by an aggregation RNN (Liu et al. 2019b). The reconstruction branch recreates the full object from the front half, while the prediction branch generates only the other half for the given front half. The reconstruction branch is a decoder while the prediction branch is a prediction RNN.

The RNN structure is mentioned and used by Liu et al. (2019b), in which they created a sequence learning model for point clouds formed into an encoder-decoder structure. By using an RNN architecture, the method has the advantage of capturing fine-grained information about smaller sequences of data and then transferring it to another sequence, discovering new features and relationships along the way.

The contribution of Kim et al. (2021) to the 3D VAE models is the *SetVAE* architecture, which is specifically created for working with unorganized sets, such as point clouds. This is a hierarchical model, that is capable of learning *coarse-to-fine* surface features on different levels. Their main focus is on the two principles of working with set-structured data, which are exchangeability and variable cardinality, in other words, the model should be invariant to the ordering of the elements and their number. The authors maintain the interactions between the elements of a set using attention-based set transformers (Lee et al. 2019) and multi-head attention (Vaswani et al. 2017), to achieve a hierarchy of latent variables (Sønderby et al. 2016; Vahdat and Kautz 2020). Then, their generator is composed of special *attentive bottleneck layers*, which ensures stochastic interactions between latent variables. The input is processed first by the multi-head attention block and then by the induced set attention block. The authors discovered that their model learns disentangled properties for generative purposes while using fewer parameters, but still outperforms *PointFlow* (Yang et al. 2019a) and other GAN-based methods. Further, attention-based

methods are presented in the work of de Santana Correia and Colombini (2022), including VAE-based methods in the 2D domain. For Kim et al. (2021) further optimization is needed to outperform all the state-of-the-art methods.

3.3.4 Point cloud segmentation

In graphics applications and computer vision, part segmentation is a base task. For this reason, Nash and Williams (2017) created Shape Variational Autoencoder (ShapeVAE), which describes the joint distribution over parts. The capability of synthesizing shapes makes it suitable for performing surface reconstruction and also for imputation of missing parts for completion. The authors take advantage of the work of Huang et al. (2015), by applying dense point correspondences in segmented parts. More specifically, this method consists of 3D keypoint modeling using a *beta shape machine*, which is a variant of a multi-layered Boltzmann machine. This results in a powerful latent space representation for both global and local features, which reconstructs good-quality surfaces. 3D objects often contain a large amount of data, like surface points and surface normals, so it is necessary to capture the semantic characteristics. The authors introduce a hierarchical model in which each layer is responsible for a certain type of feature; the higher layers capture global features and the lower layers capture local features. This method is strict from the point of view of the input data. Each type of object has a predefined number of categories for the parts from which at most one category is allowed to be missing. Additionally, the input meshes must be consistently scaled and aligned with each other. The drawback of this approach is the dependency of consistent mesh segmentations and dense correspondences. The proposed solution is to experiment with unordered point sets. Also, the use of GANs might increase the quality of the fine edges.

Neural Radiance Fields (NeRF) (Mildenhall et al. 2020) is created for the incorporation of geometric structures and is a differentiable volume rendering, mainly used in the understanding of the scene. On its own NeRF is a 6D continuous vector function, where the inputs are ray coordinates. These ray coordinates are similar to points from a point cloud, which was converted from a depth image. The problem with the basic NeRF is that too much optimization is required for each scene, and new scenes are hardly generalizable. To solve the problem, starting from (Eslami et al. 2018), Kosiorek et al. (2021) mixed NeRF with VAE to create Neural Radiance Fields VAE (NeRF-VAE), the novelty of which is the ability to generate new scenes from a previously unseen environment. The NeRF-VAE learns a distribution over a radiance field by conditioning them into the latent space. Since the training contains the camera orientation of each scene, the model learns different viewpoints as well, hence better control over the generative phase. The model is improved by creating an attention-based NeRF-VAE. The approach described in Kosiorek et al. (2021) limits the expressivity of each scene to increase the overall accuracy over multiple scenes, which is the opposite of the basic NeRF. Also, low-dimensional latent variables are a promising idea for future work.

Following the trend of presenting discriminator-type methods based on VAEs, Anvekar et al. (2022) created a method, that captures unsupervised hierarchical local and global geometric signatures. The authors propose a Geometric Proximity Correlator (GPC) and variational sampling to extract and analyze the morphology of the point clouds. Their classification tests revealed that the combination of this method with the PointNet (Qi et al. 2017a) method outperforms several other methods in accuracy, losing out to only the Point

Transformer Network (Zhao et al. 2021). This approach outperforms competing models in some of the classes while falling back on others.

Yu et al. (2022) generalize the idea of transformers from BERT (Devlin et al. 2019) for point clouds. The created method, Point-BERT, is pre-trained using an MPM task. This is followed by dividing the point cloud into local patches, and then a discrete VAE (Rolfe 2017) is used to generate discrete tokens storing the required information. After randomly masking out patches, the transformer improves the quality of the model, also learning to estimate the information hidden behind the masked patches. For the division, DGCNN (Wang et al. 2019) tokenizer with FoldingNET (Yang et al. 2018) is used. Time optimization is needed for possible future implementations.

3.3.5 Anomaly detection

For point cloud generation, detecting outlier data, or anomalies is an important step. Masuda et al. (2021) created a method, based on VAEs, that detects anomalies in point clouds without the need for supervision. The model trains to extract the distribution of the different characteristics of the given shapes using a graph-based encoder (Shen et al. 2018), then tries to reconstruct them using the FoldingNet-based decoder (Yang et al. 2018). This is completed with a spherical shape, like a grid, rather than a plane. Two distance metrics were considered, the EMD (Rubner et al. 2000) (15) and the CD (14). But because of faster convergence, the CD was preferred. The detected level of anomaly is expressed as the area under the curve of the receiver operating characteristic, which verifies whether a sample point cloud is correctly classified as normal or abnormal. This approach is currently showing only theoretical advantages, while real-life experiments are planned in the future.

Czerniawski et al. (2021) propose a change detector method for 3D computer models of buildings, similar to the anomaly detection method. They implement point cloud completion with hierarchical VAE, which improves the change detection by about 0.2 of the total area under the curve. The hierarchical structure is ensured by skipping connections between the layers from the encoder and decoder. Buildings are often stored using Building Information Modeling (BIM) (Eastman et al. 2021), which is a digital automated platform. Scanning buildings is a complex task since current data acquisition devices, such as laser scanners and depth cameras, have many shortcomings; for example, covered parts or non-optimal reflective surfaces often cause the surfaces to be missing or deformed. In addition, capturing every detail of a building leads to impractically large datasets. Thus, the perceptual completion of the proposed method helps compare two computer models or point clouds. The basis of this method is the Variational Shape Learner (VSL) network (Liu et al. 2018). The authors create their dataset, using a LIDAR or a timeof-flight (ToF) camera to scan buildings and create point clouds (Ma et al. 2020a). Furthermore, they modified the scans (Liu et al. 2018) to be incomplete. However, the VSL network is pretrained on the ModelNet (Wu et al. 2015) synthetic dataset. However, limitations for this approach exist. The lack of automated change detection, limited resolution, and diversity necessitates more real-life data.

For robot autonomy, the possibility of serving erroneous data, called anomalies, is unavoidable for any sensor. To detect these anomalies and resolve potential failures, Ji et al. (2020) proposed the Supervised Variational Autoencoder (SVAE) architecture. The core of this model is the global feature extraction from the high-dimensional inputs. The method is specifically tested for *agbots*, or low-cost robots for agriculture, that roam under the crop canopies, to help manage the ecosystem (Higuti et al. 2019; Kayacan and Chowdhary

2019). For these robots, identifying anomalies and their causes without high computational demands is important. The input is multi-modal consisting of a high-dimensional sensor, like LIDAR or a ToF camera, and a low-dimensional sensor, like the wheel encoder. The output of the model specifies whether an anomaly exists or not, with the necessary control sequence. The two parts of the model are the feature generator and the classifier. The balance between these two parts is maintained with VAE. The encoder, by compressing the high-dimensional signals, filters out the noise and finds the global features. The training of the model is a one-stage procedure, and the extracted features in anomaly detection outperform baseline methods, such as VAE Fixed Features + MLP proposed by Kingma et al. (2014).

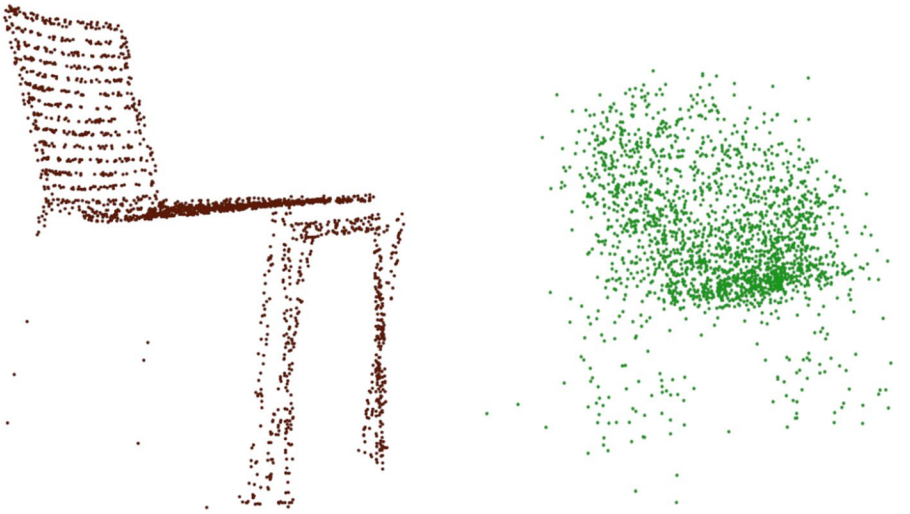
3.3.6 Representation learning

All of the currently available 3D data representation methods have limitations that negatively affect the results (Friedrich et al. 2018). One of the most popular 3D representation formats is the point cloud; however, a point cloud is unstructured, which is a major bottleneck in many applications, since finding the connections between the points is a difficult and time-consuming procedure. These applications are usually limited to a certain number of points per point cloud, usually, 2048, to retain an acceptable runtime. On the other hand depth images offer the same performance and efficiency as more conventional 2D image processing techniques (Molnár et al. 2021). However, depth images suffer from self-occlusion, so they contain only a projection of an object or a scene, without storing information about hidden objects. Capturing the same object from different viewpoints and treating them as a collection of images, or as a *omniview projection*, is a possible solution; however, this results in duplicated data points, while the problem of hidden surfaces remains.

To solve the problem of processing complex data formats, such as point clouds, we further present methods that compress point clouds into a latent representation. Representing the data in the latent space is less intuitively comprehensive for a human observer, but it is more convenient for any neural network to work with, arriving at a solution that is runtime efficient and has acceptable data preservation capabilities at the same time. Such work is proposed by Achlioptas et al. (2018), in which the authors suggest that a GAN performs better if it works in the latent space. First, an autoencoder calculates a latent space, and then a GAN is trained inside that latent space. This reduces the mode collapse and other difficulties in training a GAN and results in newly generated point clouds. One slight drawback is the required separate training for every category. We conducted a small test by introducing a latent vector, which encoded a chair with missing legs. The output was a noisy object, but all four legs were attached, Fig. 7. This method could decode insufficiently some details or partial shapes.

In this work as a metric, the CD (14) and the EMD (Rubner et al. 2000) (15) are considered. The EMD (Rubner et al. 2000) (15) is differentiable almost everywhere, while the CD is differentiable everywhere and computationally less demanding. Despite this Zamorski et al. (2020) state that the EMD (Rubner et al. 2000) (15) leads to better generative capabilities because it compares distributions of points, rather than distances between points.

Zamorski et al. (2020), following the idea of combining GANs and VAEs, created a method, which allows the model to learn latent space representation of objects and to generate point clouds. Latent space is defined as continuous or binary. The architecture is the mixture of VAEs and GANs, which is known as AAE architecture (Makhzani et al. 2015).



(a) a chair with missing legs (b) reconstructing the point cloud

Fig. 7 Experiment with (Achlioptas et al. 2018). The model was trained with conventional chairs, and then we reconstructed a chair with missing legs. The legs in the reconstruction are present, but the point cloud is noisy

(a) ground truth object (b) reconstructed object

Fig. 8 Reconstruction of 3D objects from latent space using *3dAAE* (Zamorski et al. 2020)

In this case, the decoder is called a generator module, and an additional discriminator module is added, which decides whether the latent variables are fake or real. The encoder and decoder are updated in an alternating way, using (7), with the results shown in Fig. 8. This has the advantage over a basic autoencoder in that in many cases new data are created in the latent space and a simple autoencoder struggles to decode previously unseen values. This work proves that using more advanced architectures, such as VAEs and GANs, improves the reconstruction as opposed to simple AEs presented in Achlioptas et al. (2018).

$$Loss_{CD}(\mathbf{S_1}, \mathbf{S_2}) = \sum_{\mathbf{x} \in \mathbf{S_1}} \min_{\mathbf{x}' \in \mathbf{S_2}} ||\mathbf{x} - \mathbf{x}'||_2^2 + \sum_{\mathbf{x}' \in \mathbf{S_2}} \min_{\mathbf{x} \in \mathbf{S_1}} ||\mathbf{x} - \mathbf{x}'||_2^2 \tag{14}$$

$$Loss_{EMD}(\mathbf{S_1}, \mathbf{S_2}) = \min_{f : \mathbf{S_1} \to \mathbf{S_2}} \sum_{\mathbf{x} \in \mathbf{S_1}} ||\mathbf{x} - f(\mathbf{x})||_2 \tag{15}$$

where $\mathbf{S_1}$ and $\mathbf{S_2}$ are two equally sized subsets of points.

Sharing the idea of creating a manageable latent representation format for point clouds, Wang et al. (2020) proposed the usage of geometry images (Gu et al. 2002). Thus, the converted point clouds were processed as basic 2D images, with minimal information loss. An example of the conversion is shown in Fig. 9. A geometry image is a reshaped point cloud, where the x, y, and z coordinates are changed to RGB colors. To increase the training stability of a GAN, an AAE architecture was used by Wang et al. (2020).

The conversion from a polygon mesh to a geometry image is possible to be done without neural networks. As an advantage, we mention that the results are reliably high resolution, but they require more processing time. The original method by Gu et al. (2002) divides the input model into disks and then folds it into an atlas. This means that the information about the connection between different surfaces was lost. To solve this issue, Sinha et al. (2016) proposed a method that introduced planar parameterization by first deforming the mesh into a *genus-0* (Fan et al. 2020) type model, which means that it does not have holes in it. Then the object is transferred into a spherical domain, from which it is sampled into an octahedron. The method traces paths on the surface of the object along the edges of the octahedron and cuts them into pieces. From these pieces, flat 2D geometry images are obtained. The boundaries of the resulting image are without discontinuities. The possibility exists to ransom this complicated procedure with VAE-based networks, unfortunately, at the time of writing, no such method is available.

Following the idea of geometry images, Molnár et al. Molnár and Tamás (2022) proposed a VAE-based representation learning method, which is capable of generating a point cloud by generating geometry images first. Their contribution is the hyperparameter tuning of the β-VAE, and the impact of using different geometry image sizes. Additionally, they also introduced an autoencoder-based point cloud to geometry image converter for mitigating the issue of unstable conversion typical for this task. The main drawback of using geometry images is the geometry image creation itself, then the loss of fine-detailed edges.

Finding the optimal representations for working with point clouds is a critical problem. Using other representation data formats, such as depth images or geometry images, together with point clouds yields transfer learning (Molnár et al. 2021) or depth data error detection (Masuda et al. 2021). Therefore, they must be chosen correctly depending on the task. This problem is often referred to as *representation learning*.

Another representation is the simple single-view image and, with the help of the work of Zhang et al. (2021), converting it to a 3D shape. View-aware Geometry-structure Network (VGSNet) learns multimodal feature representation from 2D images, 3D geometry, and structure, producing a method that is capable of reconstructing the geometry and structure of a shape based on a single-view image. The VGSNet is based on StructureNet (Mo et al. 2019a). The main goal is to create 3D point clouds from multimodal input. The model

(a) ToF Image (b) Point Cloud

Fig. 9 The conversion from a geometry image to a point cloud is relatively easy, the inverse transformation is more difficult

trains the images and their 3D point clouds simultaneously, achieving one-to-one mapping while learning a multimodal feature representation (Guo et al. 2019). By learning the shape of parts, this method achieves consistent 3D reconstruction. Similarly to the work of Wang and Yoon (2021), the main branch consists of a VAE, while an auxiliary branch is introduced for separate image encoding. VGSNet is tested with synthetic images, while real-life evaluation is preserved for future work. Further optimizations are planned by experimenting with different losses and data structures.

Wu et al. (2016) initially made a 3D-GAN model, then an extension to it, called 3dvae-gan, which is a VAE and GAN-based 3D shape generator. The novelty is an encoder that takes 2D images, learns their latent representation, and generates a new image with the appropriate generative and discriminative modules. The method is very similar to Larsen et al. (2016), but 3D voxel grids are used. Promising unsupervised method, however, optimization is needed to compete with supervised approaches, such as Sedaghat et al. (2017).

3.4 Other 3D-related applications

We focus on methods that do not rely on one of the previous three data representation techniques (polygon mesh, voxel grid, point cloud) and are specialized in a certain domain. The described methods come from the domain of medicine, motion sequence generation, auditory signal processing, hand pose estimation, and evaluation tasks.

3.4.1 Evaluation of the latent space for different architectures

Talking about point cloud compression into latent space and other representation learning procedures, the question arises how well does this latent space store relevant information? Ali and van Kaick (2021) answer this question by creating a comparison between different models and the quality of their latent spaces. The analyzed architectures are *3D-AE* (Guan et al. 2020), *3D-VAE* (Brock et al. 2016), and modified GANs like *3D-GAE* (Guan et al. 2020) or *3D-PGAE*, an inhouse modification for the *3D-GAE*. In comparison, the latent spaces of these methods are compared in their interpolation capabilities and the level of meaningful organization of the information inside the latent space. These properties are intuitive, but they are rarely optimized.

The authors of this paper create a synthetic dataset, where they know the prior structure and semantic attributes of the shapes. The dataset creation is based on the *split grammar* of Wonka et al. (2003). As metrics, they measure the correlation between the shapes and the latent space, and between the shape attributes and the organization of the latent space, as similar features should be grouped near each other. The idea comes from a similar comparison paper (Hinton and Salakhutdinov 2006), which is for 2D embeddings, in which the learned latent variables are colored according to the class label. As an evaluation metric, Theis et al. (2016) discuss *Parzen window* estimates. Ali and van Kaick (2021) conclude that VAE-based methods provide lower quality latent space organization, than other autoencoder-based methods, while GAN-based methods produce better qualitative results than VAEs. This experiment was limited by the number of shapes and the detail level. Future works include increasing the shapes as well as experimenting with different data structures.

3.4.2 Hand pose estimation in 6DoF

In the case of the work of Spurr et al. (2018), based on the work of Wan et al. (2017), a 3-dimensional hand pose is estimated using 2D RGB images or depth images. Cross-modal training is rarely used, although the nature of VAEs creates the possibility to encode more data types in the same latent space, from which the desired data type is recovered. This method offers significant improvement against Zimmermann and Brox (2017) in RHD, however, falls back in STB. The architecture can be seen in Fig. 10.

This particular example utilizes this capability to a high extent by estimating hand-poses from depth images or even from conventional RGB images, as in Fig. 11. This model is very specific, and although only a finite number of hand poses exist (Santello et al. 1998), this number was further reduced in dimensions by Tagliasacchi et al. (2015). However, the results of this hand pose estimator are still notable. The VAE architecture is very well utilized to project 2D key points into low-dimensional latent space, then estimate the 3D joint posterior, outperforming the similar method of Zimmermann and Brox (2017).

Yang et al. (2019b) complain that the method of Spurr et al. (2018) converges too slowly and compromises the accuracy of pose reconstruction. Yang et al. (2019b), in addition to the RGB data, rather explore other input data types like point clouds and heat maps. As for the architecture, they extend the cross-modal VAE architecture seen in the work of Spurr et al. (2018). Then, the use of two latent space alignment operator strategies causes the model to learn the hand poses.

Gu et al. (2020) highlight the problem of balancing modality-specific representations and shared representations. Additionally, other unnecessary information is stored in the shared latent space, such as the intrinsic camera or background. Gu et al. (2020) propose the disentangling of these features from the latent space. They achieve better *mean joint error* than previous methods, by joining two VAEs by their latent spaces, to process both the 2D modalities and the 3D modalities as well. The architecture can be seen in Fig. 12.

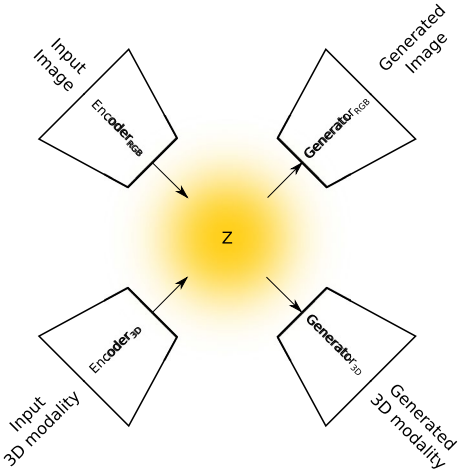
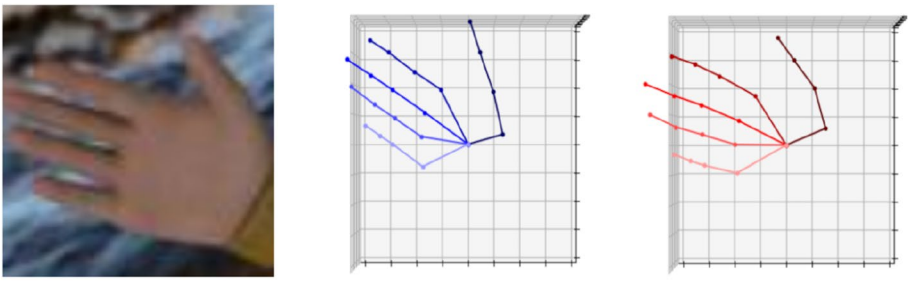


Fig. 10 The architecture from Spurr et al. (2018), which is an example of multimodal data processing. The model has a branch for RGB images and another for 3D hand pose, however, the latent space is common



(a) real image (b) GT hand pose (c) reco hand pose

Fig. 11 Hand pose estimation using (Spurr et al. 2018)

3.4.3 Auditory 3D perception

The majority of the perceived data comes from visuals, by auditive, we estimate the position of the sound source. To exploit this phenomenon, audio device manufacturers need to analyze auditory information. Since everyone is different, the shapes of the ears and the ear canals are different as well. To achieve the best possible surround sound, companies need to develop Head Related Transfer Function (HRTF), which tailors the sound for each individual. Usually, this requires a tedious procedure, where a professional measures the hearing of the person with specific tools. To address this issue, Yamamoto and Igarashi (2017) developed a VAE-based 3D spatial sound individualization technique, where the calibration procedure is composed of only sound tests and the response of the user. Similarly, better spatial audio profiles are created by analyzing audio sources (Karamatlı et al. 2019). Although the work of Yamamoto and Igarashi (2017) is promising, further data variability tests should be conducted, as well as quality assessments and calibration optimization.

3.4.4 Variational Autoencoders for medical domain using 3D data

Besides computer vision and object reconstruction, the adaptation of VAEs in the medical domain is also widespread. Unfortunately, one of the main problems with our health is caused by heart dysfunction. For this, medical specialists must understand the

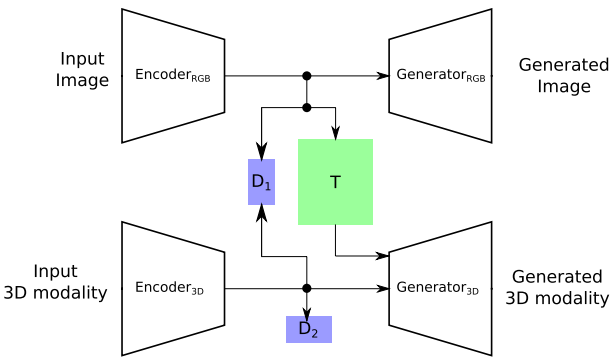


Fig. 12 The architecture from Gu et al. (2020), which is an example of multimodal data processing. The model has a branch for RGB images and another for 3D hand pose. Between the branches, there is a translator (T) layer and two discriminator modules (D_1, D_2)

internal structure of a patient. High-resolution 3DMR sequences provide a detailed model of the heart, but the data acquisition time is very long. Multiplanar breath-hold 2D cites sequences are considered standard, but these do not necessarily hold the required amount of information about the condition of the heart. Biffi et al. (2019) proposed a method based on VAE, which is capable of generating high-resolution 3D segmentation of the left ventricular myocardium from only three segmentations from standard 2D cardiac views. The standard VAE method is slightly modified to achieve a CVAE, with two encoder, one for the 2D data, and another for training 3D data. This way, it is easier to learn the latent space of the required 3D features when we want to encode both data types into the same latent space. In this work, further real-life tests should be conducted.

The work of Lyu and Shu (2021) is also part of the medical domain, more specifically in 3D brain tumor detection and segmentation. They, inspired by the winners of Multimodal Brain Tumor Segmentation Challenge (BraTS) 2018 (Myronenko 2019), created a two-stage cascade network, which provides quantitative and qualitative improvements for model ensembling and stabilizes the predictions. The first stage calculates rough segmentation, while the second stage concatenates preliminary maps from the first stage and MRI input and refines the results. Attention-gates (AGs) (Oktay et al. 2018) are used to suppress irrelevant background data. The first branch is a mixture of U-Net (Ronneberger et al. 2015) and VAE architectures. A large encoder extracts the features, a smaller decoder predicts the segmentation maps, while VAE reconstructs the input images. The second stage is an attention-gated mixture of U-Net (Ronneberger et al. 2015) and VAE architecture. The encoder is similar to the previous stage; the only change is that the input is the segmentation map and the multimodal MRI. Whereas the decoder received the AGs, so each level has a gate, that helps to decide the attention coefficient. The work of Lyu and Shu (2021) requires further optimization to reduce the memory and time constraint limitations. Additionally, future work includes experimenting with combining two attention mechanisms.

3.4.5 Motion sequence generation

So far we have analyzed static 3D shapes; however, for VAEs dynamic models also exist. Starting from the comprehensive survey on human pose estimation created by Wang et al. (2021a), or on face alignments by Gogić et al. (2021), we conclude that human poses, being very complex entities, offer a practical base for dynamic modeling. Approximating motion from a pose is a possibility, as described by Yan et al. (2018), where they created a human motion generator using VAE, called *Motion Transformation Variational Auto-Encoders (MTVAE)*. One way to describe human motion is by short-term dynamics with transformations between them, also known as motion modes. Concatenating these motions, we describe the long-term motion sequence of a body. This model learns the different dynamic states of a body, transfers them to latent space, and then generates plausible, yet diverse, sequences of motions from a starting pose. The method is also expanded to facial expressions. The authors based their work on prediction Long Short-term Memory (LSTM) for sequence generation (Srivastava et al. 2015; Hochreiter and Schmidhuber 1997), so VAE is encapsulated between an LSTM encoder and an LSTM decoder. They create two versions of MT-VAE. The first concatenates the motion features, whereas the second version sums them up. Motions are mainly gathered from 2D images and video sequences, but the extension to 3D pose descriptions is also possible.

Fig. 13 Generating human poses in different actions (Petrovich et al. 2021)

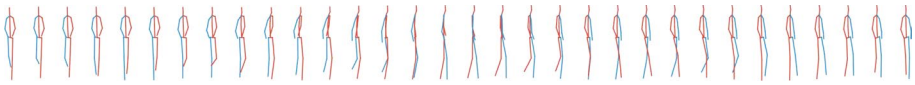


Fig. 14 Generating walking human poses (Cai et al. 2021)

Another motion sequence generator is proposed by Petrovich et al. (2021), which, unlike previous methods, does not need initial poses to estimate diverse motions, Fig. 13. This is achieved by training an action-conditional generative model, with a transformer-based encoder-decoder network, hence the name: Action-conditioned Transformer VAE (ACTOR). Body information is stored using SMPL (Loper et al. 2015), which describes joint positions and surfaces. This allows the usage of several possible reconstruction losses: constraining part rotations in the kinematic tree, joint locations, or surface points. Positional encoding (Mildenhall et al. 2020) is a promising idea, but prior to this method, it was not used in motion planning. The relatively small number of motion capture (MoCap) datasets is a limitation for the 3D domain, which is why the authors opted to use monocular motion estimation methods (Kocabas et al. 2020). Fortunately, this choice makes it possible to learn larger-scale applications, since the data require less memory for training. Since a human body has several independently linked parts, it is necessary to learn a disentangled representation for the pose and shape in the latent space. Then the need to introduce certain variables, such as the category of movement or duration, produces a CVAE (Sohn et al. 2015). For training, SMPL is expanded with a continuous 6D rotation representation (Zhou et al. 2019). Petrovich et al. (2021) proposes to further research the priors on motion estimator or action recognition problems.

A third motion sequence generator, made by Cai et al. (2021), has the advantage over previous methods that, in addition to generating a motion sequence, Fig. 14, it also predicts, completes, and interpolates motion, and even spatial and temporal recovery is possible. These tasks are usually complex in themselves, so most methods tackle only one of them at a time. However, in this article, the authors treat every input as a masked motion series. This means that visible parts are considered input conditions, whereas masked parts are for learning the completion. Based on the input conditions, the CVAE (Sohn et al. 2015), used in this work, samples and generates new motion sequences. The authors introduced an action-adaptive modification to give better semantic guidance for the sequence, thus improving the flexibility. Cross-attention mechanisms are used, similarly to hierarchical VAE, which connect the encoder layers to the decoder layers. This pays global attention to long-term dependencies (Vaswani et al. 2017). Also, the model consists of two parallel VAEs, where one of them learns the latent distribution for the visible parts, while the other one is for synthesizing the missing parts. They share information during training using their decoder weights. Both VAEs are expanded

with adversarial networks to ensure high-quality results. The metrics used are derived from (Mao et al. 2017; Wang et al. 2018; Zheng et al. 2019).

At the time of writing, the most novel article on motion modeling is the work of Bourached et al. (2021). This method also focuses on multiple aspects of human motion, such as trajectory prediction and action classification. Each task suffers from data degradation, which is a common problem in real-world scenarios. This method is meant to generate a holistic model of action over multiple time scales, generate coherent motion sequences, detect outliers, and complete missing data. This is achieved by implementing a hierarchical VAE architecture combined with graph convolutional neural networks, called *Hierarchical Graph-convolutional Variational Autoencoder (HG-VAE)*. The local features depend on the global patterns via a reduced graph size. The latent variables are connected using *rezero residual connections* (Bachlechner et al. 2021), and the authors also took advantage of the fact that spatial graph convolutions (Niepert et al. 2016; Kipf and Welling 2016) provide a natural means of learning contractions and expansions.

Bourached et al. (2021) also mention *Hierarchical Motion VAE (HM-VAE)* (Li et al. 2021), which is very similar to their method, but in juxtaposition to *HM-VAE*, Bourached et al. (2021) use 4 stochastic layers and learn the contraction of joints for each of the latent variables, as shown in Fig. 15. Also, the work of Li et al. (2021) is more focused on video-based human motion analysis, like pose estimation, motion completion, and synthesis, also this method is capable of correcting erroneous body animations. Their method consists of a generalized motion prior (Yang et al. 2021), which analyses body movement from high-quality motion data (Mahmood et al. 2019). The fundamental base of *HM-VAE* is the skeleton-aware architecture (Aberman et al. 2020), allowing to locally capture the body part motions. The work of Bourached et al. (2021) is limited by the complexity of the data pipeline, which requires optimization. On the other hand, the work of Li et al. (2021) tends to accumulate errors over multiple frames, and lacks incorporating physical properties and action conditions.

4 Summary of the methods

To assist the reader in focusing on a certain work in the domain of 3D VAEs, we provide a tabular and graph-based representation for the already presented methods. The first one has the advantages of available resources near the work such as code, and datasets

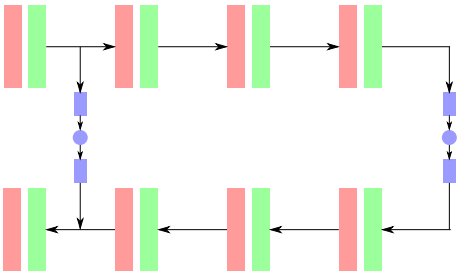


Fig. 15 The architecture of Li et al. (2021), where the top path represents the encoder, the bottom path is the decoder, the red rectangles are the skeleton convolutions, the green rectangles are the skeleton pooling/unpooling layers, the blue rectangles are linear layers, and the blue circles are the latent vectors. This diagram represents a two-stage VAE

while the latter one highlights the impact of the work within the domain in terms of citations.

4.1 Tabular summary

We provide an overview in the form of a table for the 4 major data types: Table 1 for polygon meshes, Table 2 for voxel grids, Table 3 for point clouds, and Table 4 for other types (SMPL, motion descriptor, HRTF, images, depth maps). Each table contains the targeted application, dataset, and whether we found the code or not, as well as the reference work on which the article is based.

4.2 Citation map

In Fig. 16 we present a citation map for the discussed VAE methods focusing on spatial data. The figure on the horizontal axis has the timeline, while on the vertical axis the impact of the work (measured as citations) on a logarithmic scale.

The horizontal texts denote the articles, while the vertical texts include the most relevant connections to the corresponding article. The color encoding is related to the domain, i.e. the data type which is accepted at the input of the methods. As a period, we considered in the reference the work of Kingma and Welling (2014) from 2014 till today covering more than a decade of VAE-based spatial data processing methods.

Besides Kingma and Welling (2014) work, the most impact had Wu et al. (2016) for spatial image-based processing, and Gadelha et al. (2018) for the discrete point cloud processing, Brock et al. (2016) for the voxel grid representation of 3D space and finally Tan et al. (2018) for the deforming mesh variants.

Currently, an increased interest is manifested for the discrete point cloud-specific processing, especially in the works written by Li et al. (2022b), Anvekar et al. (2022) and Yu et al. (2022).

5 Conclusions

In this paper, we summarized the works focusing on spatial data processing based on Variational Autoencoders. We considered as the base method the fundamental paper by Kingma and Welling (2014), which has the highest impact in this domain. For a better understanding, we introduced the main theoretical foundations at the beginning of this review. The more than two hundred works cited either as direct or indirect references show a high interest in this specific domain. The papers were grouped according to their input data-specific characteristics, highlighting the open-source code and dataset availability for the selected ones. Moreover, we considered also the reproduction of the runtime environment for several methods, for which we summarized the current setup configuration on the webpage of the paper (this changed considerably for the older papers during the time).

The different input data characteristics were addressed in a specific way. An overview of the different 3D representations is provided by Friedrich et al. (2018). Each of these

Table 1 A detailed listing of the studied methods—Polygon meshes

Method	Application	Description	Dataset and code	Based on
Tan et al. (2018)	mesh deformation	Deforms meshes using rotation invariant mesh difference (mesh VAE)	MPI FAUST (Bogo et al. 2014) SCAPE (Anguelov et al. 2005) Dyna (Pons-Moll et al. 2015) Swing (Vlasic et al. 2008) Spacetime faces (Zhang et al. 2004) *Code:* yes	Kingma and Welling (2014), Gao et al. (2016), Sohn et al. (2015), Gregor et al. (2015)
Ma et al. (2020b)	cloth deformation on human bodies	mesh-VAE-GAN learns to dress human models *CAPE*	proprietary *Code:* no	Kingma and Welling (2014), Bogo et al. (2016), Loper et al. (2015), Bao et al. (2017), Larsen et al. (2016), Defferrard et al. (2016), Ranjan et al. (2018), Goodfellow et al. (2014)
Yuan et al. (2020)	shape generation, pooling operation	Mesh contraction pooling, with even-sized triangles	SCAPE (Anguelov et al. 2005) Face (Neumann et al. 2013) Horse and Camel (Sumner and Popović 2004) Swing (Vlasic et al. 2008) MPI FAUST (Bogo et al. 2014) *Code:* yes	Kingma and Welling (2014), Buhmann (2000), Cohen and Welling (2016), Ruck et al. (1990), Defferrard et al. (2016), Garland and Heckbert (1997), Gao et al. (2019)
Park and Kim (2021)	face generation	It converts the polygons into binary data, then new faces are generated (FaceVAE)	ModelNet (Wu et al. 2015) *Code:* no	Kingma and Welling (2014), Cai et al. (2019)
Qin et al. (2022)	CAD model retrieval	3D CAD model retrieval based on sketch and unsupervised VAE	proprietary *Code:* no	Goller and Küchler (1996), Socher et al. (2011), Smith (2017)

Table 2 A detailed listing of the studied methods—Voxel grids

Method	Application	Description	Dataset and code	Based on
Brock et al. (2016)	object generation, classification	Testing interpolation capabilities of the decoder in the task of shape generation and classification	ModelNet (Wu et al. 2015) *Code:* yes	Kingma and Welling (2014), Sedaghat et al. (2017), Team et al. (2016), Clevert et al. (2016), Dumoulin and Visin (2016), Glorot and Bengio (2010), Ioffe and Szegedy (2015)
Meng et al. (2019)	segmentation	Voxel VAE Net (VVNet) with group convolutions for point cloud segmentation and radial basis function for interpolating boolean occupancy grids (VVNet)	S3DIS (Armeni et al. 2016) ShapeNet (Chang et al. 2015) *Code:* no	Kingma and Welling (2014), Buhmann (2000), Qi et al. (2017a), Cohen and Welling (2016)
Guan et al. (2020)	volumetric shape generation	Guidance on the construction of the latent manifold and ability to map input shapes to this manifold (3D-GAE)	COSEG (Wang et al. 2012) ModelNet (Wu et al. 2015) *Code:* yes	Kingma and Welling (2014), Wang et al. (2014), Belkin and Niyogi (2001)
Liu et al. (2020a)	voxel compression	ROS converts voxels into octree representation for voxel compression	ScanNet (Dai et al. 2017) *Code:* no	Kingma and Welling (2014), Wurm et al. (2010), Gower (1975)
Ali and van Kaick (2021)	evaluates the quality of the latent space	Shape generator by recursively splitting the geometry into smaller pieces	proprietary *Code:* yes	Kingma and Welling (2014), Brock et al. (2016), Guan et al. (2020), Wonka et al. (2003), Burgess et al. (2018), Hinton and Salakhutdinov (2006), Theis et al. (2016), Smelik et al. (2014), Edelmann et al. (2021)
Yu and Oh (2022)	data compression	Compresses data using an anytime estimation approach in robot-human connections	ModelNet (Wu et al. 2015) PASCAL3D (Xiang et al. 2014) *Code:* yes	Kingma and Welling (2014), Yu and Lee (2018), Yu et al. (2019b), Yu and Lee (2019), Larsson et al. (2017)

Table 3 A detailed listing of the studied methods—Point clouds

Method	Application	Description	Dataset and code	Based on
Wu et al. (2016)	2D image to 3D voxel generation	Learning a probabilistic latent space of object shapes via 3D generative-adversarial modeling (3D-GAN)	ModelNet (Wu et al. 2015) IKEA (Lim et al. 2013) *Code:* no	Kingma and Welling (2014), Goodfellow et al. (2014), Larsen et al. (2016), Radford et al. (2016), Maturana and Scherer (2015)
Nash and Williams (2017)	segmentation, reconstruction	Part segmentation using multi-layered Boltzmann-machine (ShapeVAE)	Kim et al. (2013) *Code:* no	Kingma and Welling (2014), Huang et al. (2015)
Gadelha et al. (2018)	classification and point generation	Organize the point clouds into kdtree (or rp-tree) for generating shapes (Multiresolution Tree Network (MRTNet))	ShapeNet (Chang et al. 2015) ModelNet (Wu et al. 2015) *Code:* yes	Kingma and Welling (2014), Simonyan and Zisserman (2015), Klokov and Lempitsky (2017), Dasgupta and Freund (2008), Ke et al. (2017), Lin et al. (2017), He et al. (2015), Yu and Koltun (2016), Chen et al. (2018), Bowers et al. (2010)
Han et al. (2019)	point cloud completion	Unsupervised feature learning for 3D point clouds from multiple angles by joint self-reconstruction and half-to-half prediction (Multi-Angle Point Cloud Variational Autoencoder (MAP-VAE))	ModelNet (Wu et al. 2015) *Code:* no	Kingma and Welling (2014), Crane et al. (2017), Qi et al. (2017b)
Schor et al. (2019)	2D and 3D shape generation	Learning to generate the unseen by part synthesis and composition (CompoNet) using point clouds and RGB images	COSEG (Sidi et al. 2011) ShapeNet (Chang et al. 2015) *Code:* yes	Kingma and Welling (2014), Achlioptas et al. (2018), Jaderberg et al. (2015), Nash and Williams (2017), Qi et al. (2017a)
Yang et al. (2019b)	estimating hand poses	Cross-modal hand pose estimation using images, point clouds, and heat maps	RHD (Zimmermann and Brox 2017) *Code:* no	Kingma and Welling (2014), Spurr et al. (2018), He et al. (2016), Higgins et al. (2017), Li and Lee (2019), Pandey and Dukkipati (2017), Radford et al. (2016)

Table 3 (continued)

Method	Application	Description	Dataset and code	Based on
Yang et al. (2019a)	realistic point cloud generation	3D point cloud generator with continuous normalizing flows (PointFlow)	ShapeNet (Chang et al. 2015) *Code:* yes	Kingma and Welling (2014), Rezende and Mohamed (2015), Achlioptas et al. (2018), Grathwohl et al. (2019)
Ji et al. (2020)	anomaly detection	Detects anomalies and their causes in agbot applications (SVAE)	proprietary *Code:* yes	Kingma and Welling (2014), Doersch (2016), Le et al. (2018)
Saha et al. (2020)	generating car shapes	This method generates the guidance design for car bodies (PC-VAE)	ShapeNet (Chang et al. 2015) *Code:* no	Kingma and Welling (2014), Schor et al. (2019), Achlioptas et al. (2018), Rios et al. (2019a, 2019b), Qi et al. (2017a), Gadelha et al. (2018)
Wang et al. (2020)	point clouds reconstruction and generation	Generates new point cloud by working with geometry images	D-FAUST (Bogo et al. 2017) ShapeNet (Chang et al. 2015) *Code:* no	Kingma and Welling (2014), Gu et al. (2002), Goodfellow et al. (2014), Li et al. (2019), Nowozin et al. (2016), Qi et al. (2017a), Shen et al. (2018)
Zamorski et al. (2020)	reconstructs and generates point clouds	3D shape generator using continuous and binary latent representations (3dAAE)	ShapeNet (Chang et al. 2015) ModelNet (Wu et al. 2015) *Code:* yes	Kingma and Welling (2014), Qi et al. (2017a), Gulrajani et al. (2017), Makhzani et al. (2015), Achlioptas et al. (2018)
Czerniawski et al. (2021)	change detection	Detects changes between building models using shape completion	proprietary *Code:* no	Kingma and Welling (2014), Eastman et al. (2021), Ma et al. (2020a), Liu et al. (2018)
Kim et al. (2021)	data generation	The model uses set transformers to hierarchically learn the latent representation for sets (SetVAE)	ShapeNet (Chang et al. 2015) Set-MNIST (Zhang et al. 2019) *Code:* yes	Kingma and Welling (2014), Vaswani et al. (2017), Vahdat and Kautz (2020), Lee et al. (2019), Sønderby et al. (2016)

Table 3 (continued)

Method	Application	Description	Dataset and code	Based on
Kosiorek et al. (2021)	scene generation	Generates new scenes from previously unseen environments using ray scene descriptors from images and camera poses (NeRF-VAE)	GQN (Eslami et al. 2018) CLEVR (Johnson et al. 2017) Jaytracer—proprietary *Code:* no	Kingma and Welling (2014), Mildenhall et al. (2020), Eslami et al. (2018), Vahdat and Kautz (2020)
Masuda et al. (2021)	anomaly detection	Detect anomalies in the reconstructed point clouds	ShapeNet (Chang et al. 2015) *Code:* no	Kingma and Welling (2014), Shen et al. (2018), Yang et al. (2018), Akcay et al. (2019), Kimura et al. (2020), Achlioptas et al. (2018), Fan et al. (2017), Schlegl et al. (2017)
Wang et al. (2021b)	point cloud compression	Point cloud geometry compression by voxel conversion (Learned-PCGC)	ShapeNet (Chang et al. 2015) *Code:* yes	Kingma and Welling (2014), Brock et al. (2016), Ballé et al. (2018), Liu et al. (2019a)
Zhang et al. (2021)	image to shape conversion	View-aware geometry-structure joint learning for single-view 3D shape reconstruction (VGSNet)	ShapeNet (Chang et al. 2015) PartNet (Mo et al. 2019b; Yu et al. 2019a) *Code:* yes	Kingma and Welling (2014), Mo et al. (2019a), Guo et al. (2019), Wang and Yoon (2021)
Anvekar et al. (2022)	point cloud classification	Extract and analyze the morphology of point clouds (VG-VAE)	ModelNet (Wu et al. 2015) *Code:* no	Katageri et al. (2021a), Katageri et al. (2021b), Qi et al. (2017a)
Li et al. (2022b)	shape generator	Parts are learned individually and used to build new shapes (EditVAE)	ShapeNet (Chang et al. 2015) *Code:* no	Kingma and Welling (2014), Paschalidou et al. (2019), Achlioptas et al. (2018), Gal et al. (2021), Qi et al. (2017a), Nielsen et al. (2020), Barr (1984)
Molnár and Tamás (2022)	representation learning	Representation learning of point clouds and hyperparameter tuning using geometry images	ModelNet Wu et al. (2015) *Code:* yes	Kingma and Welling (2014), Zamorski et al. (2020), Higgins et al. (2017), Gu et al. (2002)

Table 3 (continued)

Method	Application	Description	Dataset and code	Based on
Yu et al. (2022)	reconstruction, completion	Pre-training 3D point cloud transformers with masked point modeling for improving reconstruction and completion qualities (Point-BERT)	ModelNet (Wu et al. 2015) *Code:* yes	Devlin et al. (2019), Vaswani et al. (2017), Rolfe (2017), Qi et al. (2017a), Wang et al. (2019), Yang et al. (2018)

Table 4 A detailed listing of the studied methods—SMPL, motion descriptor, HRTF

Method	Application	Description	Dataset and code	Based on
Wan et al. (2017)	estimating hand poses	Hand pose estimation with two deep generative models, common latent space from depth images, and hand poses (Crossing Nets)	NYU (Tompson et al. 2014) MSRA (Sun et al. 2015) ICVL (Tang et al. 2014) *Code:* no	Kingma and Welling (2014), Radford et al. (2016), Salimans et al. (2016)
Spurr et al. (2018)	hand pose estimation	Hand pose estimation with two pairs of encoder and decoder, with a common latent space, using RGB images, depth images, and 2D joint locations	STB (Zhang et al. 2016) RHD (Zimmermann and Brox 2017) NYU (Tompson et al. 2014) MSRA (Sun et al. 2015) ICVL (Tang et al. 2014) *Code:* yes	Kingma and Welling (2014), Wan et al. (2017), Tagliasacchi et al. (2015), He et al. (2016)
Yamamoto and Igarashi (2017)	HRTF optimization	Surround audio signal optimization with HRTFs	CIPIP (Algazi et al. 2001) *Code:* no	Kingma and Welling (2014), Sohn et al. (2015), Rezende et al. (2014)
Yan et al. (2018)	generates possible face expressions and body motion sequences	Motion transformations to generate multimodal human dynamics (MT-VAE) from series of motion modes	Aff-Wild (Zafeiriou et al. 2017) Human3.6M (Ionescu et al. 2014) (Paysan et al. 2009) *Code:* yes	Kingma and Welling (2014), Srivastava et al. (2015), Hochreiter and Schmidhuber (1997), Bregler (1997), Smith and Vul (2013), Lan et al. (2014), Tran et al. (2017), Zhu et al. (2016)
Biffi et al. (2019)	3D left ventricular myocardium creation from 2D views	3D high-resolution cardiac segmentation and reconstruction from 2D views	Bai et al. (2015) *Code:* no	Kingma and Welling (2014), Cerrolaza et al. (2018), Biffi et al. (2018)
Gu et al. (2020)	hand pose estimation	3D hand pose estimation with disentangled cross-modal latent space (DCMLS) from RGB and depth images	STB (Zhang et al. 2016) RHD (Zimmermann and Brox 2017) *Code:* no	Kingma and Welling (2014), Spurr et al. (2018), Goodfellow et al. (2014), Zimmermann and Brox (2017)

Table 4 (continued)

Method	Application	Description	Dataset and code	Based on
Bourached et al. (2021)	trajectory prediction and action classification	Hierarchical graph-convolutional variational autoencoding for generative modeling of human motion (HG-VAE) using SMPL	Human3.6M (Ionescu et al. 2014) AMASS (Mahmood et al. 2019) *Code:* yes	Kingma and Welling (2014), Bachlechner et al. (2021), Mao et al. (2019), Mao et al. (2020), Li et al. (2021), Sønderby et al. (2016), Child (2020), Loper et al. (2015), Romero et al. (2017)
Cai et al. (2021)	predict, complete, interpolate motion	Unified 3D human motion synthesis model via CVAE from motion descriptors and videos	Human3.6M (Ionescu et al. 2014) *Code:* yes	Kingma and Welling (2014), Mao et al. (2017), Sohn et al. (2015), Vaswani et al. (2017), Wang et al. (2018), Zheng et al. (2019)
Li et al. (2021)	pose estimation, motion completion, and animation correction	Task-generic hierarchical human motion prior (HM-VAE) using motion descriptors and RGB images	LAFAN1 (Harvey et al. 2020) AMASS (Mahmood et al. 2019) *Code:* no	Kingma and Welling (2014), Aberman et al. (2020), Yang et al. (2021)
Lyu and Shu (2021)	brain tumor segmentation	Two-stage U-Net and VAE model with attention gates from segmenting brain tumors using multimodal MRI images	BraTS 2020 (Bakas et al. 2019) *Code:* yes	Myronenko (2019), Oktay et al. (2018)
Petrovich et al. (2021)	motion sequence generation	Generates motion sequences for human bodies using SMPL, and ACTOR	NTU-RGB+D (Liu et al. 2020b; Shahroudy et al. 2016) HumanAct12 (Guo et al. 2020) UESTC (Ji et al. 2021) *Code:* yes	Kingma and Welling (2014), Devlin et al. (2019), Dosovitskiy et al. (2021), Kocabas et al. (2020), Loper et al. (2015), Sohn et al. (2015), Zhou et al. (2019)

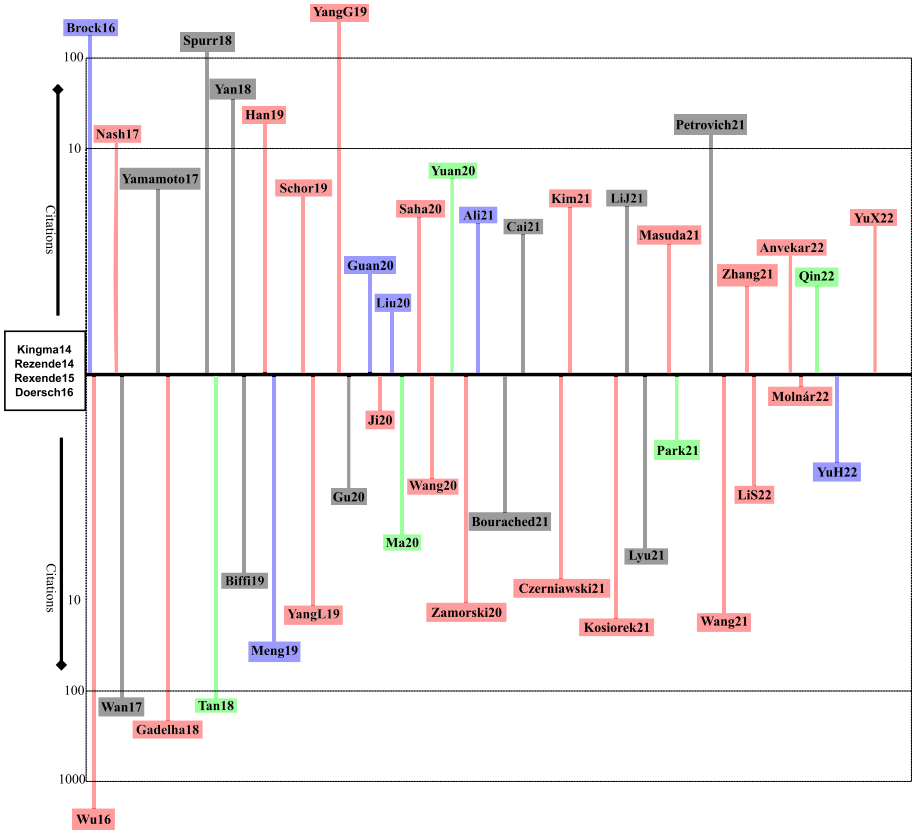


Fig. 16 The evolution of 3D VAE methods. Kingma and Welling (2014), Rezende et al. (2014); Rezende and Mohamed (2015), Doersch (2016) established the base of the models, and combining them with other works, we get this timeline about the methods we have presented in this work. The colors show the processed data: red—point cloud, green—polygon mesh, blue—voxel grid, gray—other (SMPL, motion descriptor, HRTF, image). The length of the line represents the impact of the papers, by the current number of citations they have

representations has an advantage for a specific task, depending on the available resources or the given architecture. The conversion is possible between different types of representations, e.g. from discrete points to meshes, however, the conversion time is often a bottleneck in the processing pipeline. The largest impact for the 3D domain had the work of Kingma and Welling (2014). Based on the input data specific categories, the most impact for the discrete point cloud processing has Gadelha et al. (2018), while for voxel grid representation of 3D space Brock et al. (2016) and Tan et al. (2018) for the deforming meshes. These different data types were the first level for our taxonomy, while the various applications represent the further levels of taxonomy.

The continuous growth of the research papers focusing on VAEs in the last few years is projecting a future increase of interest in the 3D domain as well, especially for compact and fast representation tasks. This later task proves to be doable by carefully designed hyperparameters for the generic β-VAEs. Most models currently require a huge amount of memory and time, therefore optimizations are welcome. Another proposed solution for

lowering the system requirements and increasing the accuracy would be to experiment with different data types and feature descriptors.

Acknowledgements The authors are thankful for the support of Analog Devices GMBH Romania, for the equipment list and Nvidia for graphic cards offered as support to this work. This work was financially supported by the Romanian National Authority for Scientific Research, project number PN-III-P2−2.1-PED-2021-3120 as well as European Union's Horizon 2020 research and innovation programme under grant agreement No. 871295. The authors are also thankful to KMTA and Domus Foundations for their support.

Author Contributions SM and LT wrote the manuscript text. SM prepared all figures. All authors reviewed the manuscript. LT ensured funding, SM and LT made review.

Declarations

Conflict of interest The authors declare no competing interests.

References

Aberman K, Li P, Lischinski D et al (2020) Skeleton-aware networks for deep motion retargeting. ACM Trans Graph 39(4):62:1-62:14

Achlioptas P, Diamanti O, Mitliagkas I et al (2018) Learning Representations and Generative Models for 3D Point Clouds. In: Dy JG, Krause A (eds) Proceedings of the 35th International Conference on Machine Learning, ICML 2018, Stockholmsmässan, Stockholm, Sweden, July 10-15, 2018, Proceedings of Machine Learning Research, vol 80. Proceedings of Machine Learning Research, pp 40–49

Akcay S, Atapour-Abarghouei A, Breckon TP (2019) GANomaly: Semi-supervised Anomaly Detection via Adversarial Training. In: Computer Vision—ACCV 2018—14th Asian Conference on Computer Vision, Perth, Australia, December 2–6, 2018, Revised Selected Papers, Part III, vol 11363. Springer, pp 622–637

Algazi V, Duda R, Thompson D et al (2001) The CIPIC HRTF Database. In: Proceedings of the 2001 IEEE Workshop on the Applications of Signal Processing to Audio and Acoustics. IEEE, New Paltz, NY, USA, pp 99–102

Ali S, van Kaick O (2021) Evaluation of Latent Space Learning With Procedurally-Generated Datasets of Shapes. In: IEEE/CVF International Conference on Computer Vision Workshops, ICCVW 2021, Montreal, BC, Canada, October 11-17, 2021. IEEE, online, pp 2086–2094, https://github.com/SharjeelAliCS/3D-latent-space-eval

Anguelov D, Srinivasan P, Koller D et al (2005) SCAPE: shape completion and animation of people. ACM Trans Graph 24(3):408–416

Antal L, Bodó Z (2021) Feature Axes Orthogonalization in Semantic Face Editing. In: 17th IEEE International Conference on Intelligent Computer Communication and Processing, ICCP 2021, Cluj-Napoca, Romania, October 28–30, 2021. IEEE, pp 163–169

Anvekar T, Tabib RA, Hegde D et al (2022) VG-VAE: A Venatus Geometry Point-Cloud Variational Auto-Encoder. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR) Workshops. IEEE, pp 2978–2985

Arjovsky M, Chintala S, Bottou L (2017) Wasserstein Generative Adversarial Networks. In: Proceedings of the 34th International Conference on Machine Learning, ICML 2017, Sydney, NSW, Australia, 6–11 August 2017, vol 70. Proceedings of Machine Learning Research, pp 214–223

Armeni I, Sener O, Zamir AR et al (2016) 3D Semantic Parsing of Large-Scale Indoor Spaces. In: 2016 IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2016, Las Vegas, NV, USA, June 27–30, 2016. IEEE Computer Society, pp 1534–1543

Asperti A, Evangelista D, Loli Piccolomini E (2021) A survey on variational autoencoders from a green AI perspective. SN Comput Sci 2(4):1–23

Bachlechner T, Majumder BP, Mao H et al (2021) ReZero is All You Need: Fast Convergence at Large Depth. In: Proceedings of the Thirty-Seventh Conference on Uncertainty in Artificial Intelligence, UAI 2021, Virtual Event, 27-30 July 2021, vol 161. Association for Uncertainty in Artificial Intelligence Press, pp 1352–1361

Bai W, Shi W, de Marvao A et al (2015) A bi-ventricular cardiac atlas built from 1000+ high resolution MR images of healthy subjects and an analysis of shape and motion. Med Image Anal 26(1):133–145

Bakas S, Reyes M, Jakab A et al (2019) Identifying the Best Machine Learning Algorithms for Brain Tumor Segmentation, Progression Assessment, and Overall Survival Prediction in the BRATS Challenge. Computing Research Repository arxiv:abs/1811.02629

Ballé J, Minnen D, Singh S et al (2018) Variational image compression with a scale hyperprior. In: 6th International Conference on Learning Representations, ICLR 2018, Vancouver, BC, Canada, April 30–May 3, 2018, Conference Track Proceedings

Bao J, Chen D, Wen F et al (2017) CVAE-GAN: fine-grained image generation through asymmetric training. In: IEEE International Conference on Computer Vision, ICCV 2017, Venice, Italy, October 22–29, 2017. IEEE Computer Society, pp 2764–2773

Barr AH (1984) Global and Local Deformations of Solid Primitives. In: Proceedings of the 11th Annual Conference on Computer Graphics and Interactive Techniques, SIGGRAPH 1984, Minneapolis, Minnesota, USA, July 23–27, 1984. Association for Computing Machinery, pp 21–30

Belkin M, Niyogi P (2001) Laplacian Eigenmaps and Spectral Techniques for Embedding and Clustering. In: Advances in Neural Information Processing Systems 14 [Neural Information Processing Systems: Natural and Synthetic, NIPS 2001, December 3–8, 2001, Vancouver, BC, Canada]. MIT Press, pp 585–591

Biffi C, Oktay O, Tarroni G et al (2018) Learning Interpretable Anatomical Features Through Deep Generative Models: Application to Cardiac Remodeling. In: Frangi AF, Schnabel JA, Davatzikos C, et al (eds) Medical Image Computing and Computer Assisted Intervention—MICCAI 2018—21st International Conference, Granada, Spain, September 16-20, 2018, Proceedings, Part I, Proceedings of Machine Learning Research, vol 11071. Springer International Publishing, pp 464–471

Biffi C, Cerrolaza JJ, Tarroni G et al. (2019) 3D High-Resolution Cardiac Segmentation Reconstruction from 2D Views Using Conditional Variational Autoencoders. In: 16th IEEE International Symposium on Biomedical Imaging, ISBI 2019, Venice, Italy, April 8-11, 2019. IEEE, pp 1643–1646

Bogo F, Romero J, Loper M, et al (2014) FAUST: Dataset and Evaluation for 3D Mesh Registration. In: 2014 IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2014, Columbus, OH, USA, June 23-28, 2014. IEEE Computer Society, pp 3794–3801

Bogo F, Kanazawa A, Lassner C, et al (2016) Keep It SMPL: Automatic Estimation of 3D Human Pose and Shape from a Single Image. In: Computer Vision—ECCV 2016—14th European Conference, Amsterdam, The Netherlands, October 11-14, 2016, Proceedings, Part V, vol 9909. Springer International Publishing, pp 561–578

Bogo F, Romero J, Pons-Moll G et al (2017) Dynamic FAUST: Registering Human Bodies in Motion | Perceiving Systems. 2017 IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2017, Honolulu, HI, USA, July 21–26, 2017. IEEE, Honolulu, HI, USA, pp 5573–5582

Bourached A, Gray R, Griffiths RR, et al (2021) Hierarchical Graph-Convolutional Variational AutoEncoding for Generative Modelling of Human Motion. Computing Research Repository arxiv:abs/2111.12602. https://github.com/bouracha/generative_imputation

Bowers J, Wang R, Wei LY et al (2010) Parallel Poisson disk sampling with spectrum analysis on surfaces. ACM Trans Graph 29(6):166:1-166:10

Bregler C (1997) Learning and Recognizing Human Dynamics in Video Sequences. 1997 Conference on Computer Vision and Pattern Recognition (CVPR '97), June 17–19, 1997. Puerto Rico. IEEE Computer Society, San Juan, pp 568–574

Brock A, Lim T, Ritchie JM et al (2016) Generative and Discriminative Voxel Modeling with Convolutional Neural Networks. Computing Research Repository arxiv:abs/1608.04236. https://github.com/ajbrock/Generative-and-Discriminative-Voxel-Modeling

Buhmann MD (2000) Radial basis functions. Acta Numer 9:1–38

Bulinski A, Dimitrov D (2021) Statistical estimation of the Kullback-Leibler divergence. Mathematics 9(5):544

Burgess CP, Higgins I, Pal A et al (2018) Understanding Disentangling in β-VAE. Computing Research Repository arxiv:abs/1804.03599

Cai L, Gao H, Ji S (2019) Multi-Stage Variational Auto-Encoders for Coarse-to-Fine Image Generation. In: Proceedings of the 2019 SIAM International Conference on Data Mining, SDM 2019, Calgary, Alberta, Canada, May 2-4, 2019. Society for Industrial and Applied Mathematics, pp 630–638

Cai Y, Wang Y, Zhu Y et al (2021) A Unified 3D Human Motion Synthesis Model via Conditional Variational Auto-Encoder. In: 2021 IEEE/CVF International Conference on Computer Vision, ICCV 2021, Montreal, QC, Canada, October 10-17, 2021. IEEE, online, pp 11,645–11,655, https://github.com/vanoracai/

Cerrolaza JJ, Li Y, Biffi C et al (2018) 3D Fetal Skull Reconstruction from 2DUS via Deep Conditional Generative Networks. In: Medical Image Computing and Computer Assisted Intervention—MICCAI 2018—21st International Conference, Granada, Spain, September 16-20, 2018, Proceedings, Part I, vol 11070. Springer International Publishing, pp 383–391

Chang AX, Funkhouser T, Guibas L et al (2015) ShapeNet: An Information-Rich 3D Model Repository. Computing Research Repository arxiv:abs/1512.03012

Chen LC, Papandreou G, Kokkinos I et al (2018) DeepLab: Semantic image segmentation with deep convolutional Nets, Atrous convolution, and fully connected CRFs. IEEE Trans Pattern Anal Mach Intell 40(4):834–848

Child R (2020) Very Deep VAEs Generalize Autoregressive Models and Can Outperform Them on Images. In: 9th International Conference on Learning Representations, ICLR 2021, Virtual Event, Austria, May 3-7, 2021

Clevert DA, Unterthiner T, Hochreiter S (2016) Fast and Accurate Deep Network Learning by Exponential Linear Units (ELUs). In: 4th International Conference on Learning Representations, ICLR 2016, San Juan, Puerto Rico, May 2-4, 2016, Conference Track Proceedings

Cohen TS, Welling M (2016) Group Equivariant Convolutional Networks. In: Proceedings of the 33nd International Conference on Machine Learning, ICML 2016, New York City, NY, USA, June 19-24, 2016, JMLR Workshop and Conference Proceedings, vol 48. Journal of Machine Learning Research, pp 2990–2999

Crane K, Weischedel C, Wardetzky M (2017) The heat method for distance computation. Commun ACM 60(11):90–99

Creswell A, White T, Dumoulin V et al (2018) Generative adversarial networks: an overview. IEEE Signal Process Mag 35(1):53–65

Czerniawski T, Ma JW, Leite F (2021) Automated building change detection with amodal completion of point clouds. Autom Constr 124(103):568

Dai A, Chang AX, Savva M et al (2017) ScanNet: Richly-annotated 3D Reconstructions of Indoor Scenes. In: 2017 IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2017, Honolulu, HI, USA, July 21-26, 2017. IEEE Computer Society, pp 2432–2443

Dai B, Wipf D (2019) Diagnosing and Enhancing VAE Models. In: 7th International Conference on Learning Representations, ICLR 2019, New Orleans, LA, USA, May 6-9, 2019

Dasgupta S, Freund Y (2008) Random Projection Trees and Low Dimensional Manifolds. In: Dwork C (ed) Proceedings of the 40th Annual ACM Symposium on Theory of Computing, Victoria, British Columbia, Canada, May 17-20, 2008. Association for Computing Machinery, pp 537–546

Davidson TR, Falorsi L, De Cao N et al (2018) Hyperspherical Variational Auto-Encoders. In: Globerson A, Silva R (eds) Proceedings of the Thirty-Fourth Conference on Uncertainty in Artificial Intelligence, UAI 2018, Monterey, California, USA, August 6-10, 2018. Association for Uncertainty in Artificial Intelligence Press, pp 856–865

de Santana Correia A, Colombini EL (2022) Attention, please! A survey of Neural Attention Models in Deep Learning. Artificial Intelligence Review pp 1–88

Defferrard M, Bresson X, Vandergheynst P (2016) Convolutional Neural Networks on Graphs with Fast Localized Spectral Filtering. In: Lee DD, Sugiyama M, von Luxburg U et al (eds) Advances in Neural Information Processing Systems 29: Annual Conference on Neural Information Processing Systems 2016, December 5-10, 2016, Barcelona, Spain. Curran Associates Inc., pp 3837–3845

Devlin J, Chang MW, Lee K et al (2019) BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding. In: Burstein J, Doran C, Solorio T (eds) Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, NAACL-HLT 2019, Minneapolis, MN, USA, June 2-7, 2019, Volume 1 (Long and Short Papers). Association for Computational Linguistics, pp 4171–4186

Dhiman C, Vishwakarma DK (2019) A review of state-of-the-art techniques for abnormal human activity recognition. Eng Appl Artif Intell 77:21–45

Doersch C (2016) Tutorial on Variational Autoencoders. Computing Research Repository abs/1606.05908

Dosovitskiy A, Beyer L, Kolesnikov A et al (2021) An Image is Worth 16x16 Words: Transformers for Image Recognition at Scale. In: 9th International Conference on Learning Representations, ICLR 2021, Virtual Event, Austria, May 3-7, 2021

Dumoulin V, Visin F (2016) A Guide to Convolution Arithmetic for Deep Learning. Computing Research Repository arxiv:abs/1603.07285

Eastman CM, Eastman C, Teicholz P et al (2021) BIM Handbook: A Guide to Building Information Modeling for Owners. John Wiley & Sons, Managers, Designers, Engineers and Contractors

Edelmann D, Móri TF, Székely GJ (2021) On relationships between the Pearson and the distance correlation coefficients. Stat Probab Lett 169(108):960

Eslami SMA, Jimenez Rezende D, Besse F et al (2018) Neural scene representation and rendering. Science 360(6394):1204–1210

Fan H, Su H, Guibas LJ (2017) A Point Set Generation Network for 3D Object Reconstruction From a Single Image. In: 2017 IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2017, Honolulu, HI, USA, July 21-26, 2017. IEEE Computer Society, pp 2463–2471

Fan H, Wu L, You F (2020) Structures in genus-zero relative Gromov-Witten theory. J Topol 13(1):269–307

Friedrich T, Aulig N, Menzel S (2018) On the Potential and Challenges of Neural Style Transfer for Three-Dimensional Shape Data. In: Rodrigues H, Herskovits J, Mota Soares C et al (eds) EngOpt 2018 Proceedings of the 6th International Conference on Engineering Optimization, vol 1. Springer International Publishing, Lisboa, Portugal, pp 581–592

Gadelha M, Wang R, Maji S (2018) Multiresolution Tree Networks for 3D Point Cloud Processing. In: Computer Vision—ECCV 2018—15th European Conference, Munich, Germany, September 8-14, 2018, Proceedings, Part III, vol 11211. Springer International Publishing, pp 105–122, https://github.com/matheusgadelha/MRTNet

Gal R, Bermano A, Zhang H et al (2021) MRGAN: Multi-Rooted 3D Shape Generation with Unsupervised Part Disentanglement. In: 2021 IEEE/CVF International Conference on Computer Vision Workshops (ICCVW). IEEE, online, pp 2039–2048

Gao L, Lai YK, Liang D et al (2016) Efficient and flexible deformation representation for data-driven surface modeling. ACM Trans Graph 35(5):158:1-158:17

Gao L, Lai YK, Yang J et al (2019) Sparse data driven mesh deformation. IEEE Trans Visual Comput Graph 27(3):2085–2100

Garland M, Heckbert PS (1997) Surface Simplification Using Quadric Error Metrics. In: Owen GS, Whitted T, Mones-Hattal B (eds) Proceedings of the 24th Annual Conference on Computer Graphics and Interactive Techniques, SIGGRAPH 1997, Los Angeles, CA, USA, August 3-8, 1997. Association for Computing Machinery, pp 209–216

Glorot X, Bengio Y (2010) Understanding the Difficulty of Training Deep Feedforward Neural Networks. In: Proceedings of the Thirteenth International Conference on Artificial Intelligence and Statistics, AISTATS 2010, Chia Laguna Resort, Sardinia, Italy, May 13-15, 2010, JMLR Proceedings, vol 9. Journal of Machine Learning Research, pp 249–256

Gogić I, Ahlberg J, Pandzic I (2021) Regression-based methods for face alignment: a survey. Signal Process 157(107):755

Goller C, Küchler A (1996) Learning Task-Dependent Distributed Representations by Backpropagation Through Structure. In: Proceedings of International Conference on Neural Networks (ICNN'96), Washington, DC, USA, June 3-6, 1996. IEEE, pp 347–352

Goodfellow I, Pouget-Abadie J, Mirza M et al (2014) Generative Adversarial Nets. In: Ghahramani Z, Welling M, Cortes C et al (eds) Advances in Neural Information Processing Systems 27: Annual Conference on Neural Information Processing Systems 2014, December 8-13 2014, Montreal, Quebec, Canada. Curran Associates Inc., pp 2672–2680

Gower JC (1975) Generalized procrustes analysis. Psychometrika 40(1):33–51

Grathwohl W, Chen RTQ, Bettencourt J et al (2019) FFJORD: Free-form Continuous Dynamics for Scalable Reversible Generative Models. In: 7th International Conference on Learning Representations, ICLR 2019, New Orleans, LA, USA, May 6-9, 2019

Gregor K, Danihelka I, Graves A et al (2015) DRAW: A Recurrent Neural Network For Image Generation. In: Proceedings of the 32nd International Conference on Machine Learning, ICML 2015, Lille, France, 6-11 July 2015. Journal of Machine Learning Research, JMLR Workshop and Conference Proceedings, pp 1462–1471

Gu J, Wang Z, Ouyang W et al (2020) 3D Hand Pose Estimation with Disentangled Cross-Modal Latent Space. In: IEEE Winter Conference on Applications of Computer Vision, WACV 2020, Snowmass Village, CO, USA, March 1-5, 2020. IEEE, pp 380–389

Gu X, Gortler SJ, Hoppe H (2002) Geometry Images. ACM Transactions on Graphics 21(3)

Guan Y, Jahan T, van Kaick O (2020) Generalized Autoencoder for Volumetric Shape Generation. In: 2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition, CVPR Workshops 2020, Seattle, WA, USA, June 14-19, 2020. IEEE, online, pp 1082–1088, https://github.com/IsaacGuan/3D-GAE

Gulrajani I, Ahmed F, Arjovsky M et al (2017) Improved Training of Wasserstein GANs. In: Guyon I, von Luxburg U, Bengio S et al (eds) Advances in Neural Information Processing Systems 30: Annual Conference on Neural Information Processing Systems 2017, December 4-9, 2017, Long Beach, CA, USA. Curran Associates Inc., pp 5767–5777

Guo C, Zuo X, Wang S et al (2020) Action2Motion: Conditioned Generation of 3D Human Motions. In: Chen CW, Cucchiara R, Hua X et al (eds) MM '20: The 28th ACM International Conference on Multimedia, Virtual Event / Seattle, WA, USA, October 12-16, 2020. Association for Computing Machinery, pp 2021–2029

Guo W, Wang J, Wang S (2019) Deep multimodal representation learning: a survey. IEEE Access 7:63,373-63,394

Han Z, Wang X, Liu YS et al (2019) Multi-Angle Point Cloud-VAE: Unsupervised Feature Learning for 3D Point Clouds From Multiple Angles by Joint Self-Reconstruction and Half-to-Half Prediction. In: 2019 IEEE/CVF International Conference on Computer Vision, ICCV 2019, Seoul, Korea (South), October 27—November 2, 2019. IEEE, pp 10,441–10,450

Harvey FG, Yurick M, Nowrouzezahrai D et al (2020) Robust motion in-betweening. ACM Trans Graph 39(4):60:60:1-60:60:12

He K, Zhang X, Ren S et al (2015) Spatial pyramid pooling in deep convolutional networks for visual recognition. IEEE Trans Pattern Anal Mach Intell 37(9):1904–1916

He K, Zhang X, Ren S et al (2016) Deep Residual Learning for Image Recognition. In: 2016 IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2016, Las Vegas, NV, USA, June 27-30, 2016. IEEE Computer Society, pp 770–778

Higgins I, Matthey L, Pal A et al (2017) Beta-VAE: Learning Basic Visual Concepts with a Constrained Variational Framework. In: 5th International Conference on Learning Representations, ICLR 2017, Toulon, France, April 24-26, 2017, Conference Track Proceedings

Higuti VAH, Velasquez AEB, Magalhaes DV et al (2019) Under canopy light detection and ranging-based autonomous navigation. J Field Robot 36(3):547–567

Hinton GE, Salakhutdinov RR (2006) Reducing the dimensionality of data with neural networks. Science 313(5786):504–507

Hochreiter S, Schmidhuber J (1997) Long short-term memory. Neural Comput 9(8):1735–1780

Hou X, Shen L, Sun K et al (2017) Deep Feature Consistent Variational Autoencoder. In: 2017 IEEE Winter Conference on Applications of Computer Vision, WACV 2017, Santa Rosa, CA, USA, March 24-31, 2017. IEEE Computer Society, pp 1133–1141

Huang H, Kalogerakis E, Marlin B (2015) Analysis and synthesis of 3D shape families via deep-learned generative models of surfaces. Comput Graph Forum 34(5):25–38

Ioffe S, Szegedy C (2015) Batch Normalization: Accelerating Deep Network Training by Reducing Internal Covariate Shift. In: Bach FR, Blei DM (eds) Proceedings of the 32nd International Conference on Machine Learning, ICML 2015, Lille, France, 6-11 July 2015, JMLR Workshop and Conference Proceedings, vol 37. Journal of Machine Learning Research, pp 448–456

Ionescu C, Papava D, Olaru V et al (2014) Human3.6M: large scale datasets and predictive methods for 3D human sensing in natural environments. IEEE Trans Pattern Anal Mach Intell 36(7):1325–1339

Jaderberg M, Simonyan K, Zisserman A et al (2015) Spatial Transformer Networks. In: Cortes C, Lawrence ND, Lee DD et al (eds) Advances in Neural Information Processing Systems 28: Annual Conference on Neural Information Processing Systems 2015, December 7-12, 2015, Montreal, Quebec, Canada. Curran Associates Inc., pp 2017–2025

Javed AR, Jalil Z, Zehra W et al (2021) A comprehensive survey on digital video forensics: taxonomy, challenges, and future directions. Eng Appl Artif Intell 106(104):456

Ji T, Vuppala ST, Chowdhary G et al (2020) Multi-Modal Anomaly Detection for Unstructured and Uncertain Environments. In: Kober J, Ramos F, Tomlin CJ (eds) 4th Conference on Robot Learning, CoRL 2020, 16-18 November 2020, Virtual Event / Cambridge, MA, USA. Proceedings of Machine Learning Research, Proceedings of Machine Learning Research, pp 1443–1455, https://sites.google.com/illinois.edu/supervised-vae

Ji Y, Yang Y, Shen F et al (2021) Arbitrary-view human action recognition: a varying-view RGB-D action dataset. IEEE Trans Circuits Syst Video Technol 31(1):289–300

Johnson J, Hariharan B, van der Maaten L et al (2017) CLEVR: A Diagnostic Dataset for Compositional Language and Elementary Visual Reasoning. In: 2017 IEEE Conference on Computer Vision and

Pattern Recognition, CVPR 2017, Honolulu, HI, USA, July 21-26, 2017. IEEE Computer Society, pp 1988–1997
Karamatlı E, Cemgil AT, Kırbız S (2019) Audio source separation using variational autoencoders and weak class supervision. IEEE Signal Process Lett 26(9):1349–1353
Katageri S, Kudari SV, Gunari A et al (2021a) ABD-Net: Attention Based Decomposition Network for 3D Point Cloud Decomposition. In: IEEE/CVF International Conference on Computer Vision Workshops, ICCVW 2021, Montreal, BC, Canada, October 11-17, 2021. IEEE, pp 2049–2057
Katageri S, Kulmi S, Tabib RA et al (2021b) PointDCCNet: 3D Object categorization Network Using Point Cloud Decomposition. In: IEEE Conference on Computer Vision and Pattern Recognition Workshops, CVPR Workshops 2021, virtual, June 19-25, 2021. IEEE, pp 2200–2208
Kayacan E, Chowdhary G (2019) Tracking error learning control for precise mobile robot path tracking in outdoor environment. J Intell Robot Syst 95(3):975–986
Ke TW, Maire M, Yu SX (2017) Multigrid Neural Architectures. In: 2017 IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2017, Honolulu, HI, USA, July 21-26, 2017. IEEE Computer Society, pp 4067–4075
Kim J, Yoo J, Lee J et al (2021) SetVAE: Learning Hierarchical Composition for Generative Modeling of Set-Structured Data. In: IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2021, virtual, June 19-25, 2021. IEEE, pp 15,059–15,068, https://github.com/jw9730/setvae
Kim VG, Li W, Mitra NJ et al (2013) Learning part-based templates from large collections of 3D shapes. ACM Trans Graph 32(4):70:1-70:12
Kimura D, Chaudhury S, Narita M et al (2020) Adversarial Discriminative Attention for Robust Anomaly Detection. In: IEEE Winter Conference on Applications of Computer Vision, WACV 2020, Snowmass Village, CO, USA, March 1-5, 2020. IEEE, pp 2172–2181
Kingma DP, Welling M (2014) Auto-Encoding Variational Bayes. In: Bengio Y, LeCun Y (eds) 2nd International Conference on Learning Representations, ICLR 2014, Banff, AB, Canada, April 14-16, 2014, Conference Track Proceedings
Kingma DP, Welling M (2019) An Introduction to Variational Autoencoders. Found Trends® Mach Learn 12(4):307–392
Kingma DP, Rezende DJ, Mohamed S et al (2014) Semi-Supervised Learning with Deep Generative Models. In: Ghahramani Z, Welling M, Cortes C et al (eds) Advances in Neural Information Processing Systems 27: Annual Conference on Neural Information Processing Systems 2014, December 8-13 2014, Montreal, Quebec, Canada. Curran Associates Inc., pp 3581–3589
Kipf TN, Welling M (2016) Semi-Supervised Classification with Graph Convolutional Networks. Computing Research Repository arxiv:abs/1609.02907
Klokov R, Lempitsky V (2017) Escape from Cells: Deep Kd-Networks for the Recognition of 3D Point Cloud Models. In: IEEE International Conference on Computer Vision, ICCV 2017, Venice, Italy, October 22-29, 2017. IEEE Computer Society, pp 863–872
Kocabas M, Athanasiou N, Black MJ (2020) VIBE: Video Inference for Human Body Pose and Shape Estimation. In: 2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition, CVPR 2020, Seattle, WA, USA, June 13-19, 2020. IEEE, online, pp 5253–5263
Kosiorek AR, Strathmann H, Zoran D et al (2021) NeRF-VAE: A Geometry Aware 3D Scene Generative Model. In: Meila M, Zhang T (eds) Proceedings of the 38th International Conference on Machine Learning, ICML 2021, 18-24 July 2021, Virtual Event, Proceedings of Machine Learning Research, vol 139. Proceedings of Machine Learning Research, pp 5742–5752
Kovenko V, Bogach I (2020) A Comprehensive Study of Autoencoders' Applications Related to Images. In: Snytyuk V, Anisimov A, Krak I et al (eds) Proceedings of the 7th International Conference "Information Technology and Interactions" (IT &I-2020). Workshops Proceedings, Kyiv, Ukraine, December 02-03, 2020, CEUR Workshop Proceedings, vol 2845. CEUR-WS.org, pp 43–54
Kramer MA (1991) Nonlinear principal component analysis using autoassociative neural networks. AIChE J 37(2):233–243
Kullback S, Leibler RA (1951) On information and sufficiency. Ann Math Stat 22(1):79–86
Lan T, Chen TC, Savarese S (2014) A Hierarchical Representation for Future Action Prediction. In: Fleet DJ, Pajdla T, Schiele B et al (eds) Computer Vision—ECCV 2014—13th European Conference, Zurich, Switzerland, September 6-12, 2014, Proceedings, Part III, Proceedings of Machine Learning Research, vol 8691. Springer International Publishing, pp 689–704
Larsen ABL, Sønderby SK, Larochelle H et al (2016) Autoencoding Beyond Pixels Using a Learned Similarity Metric. In: Proceedings of the 33nd International Conference on Machine Learning, ICML 2016, New York City, NY, USA, June 19-24, 2016. Proceedings of Machine Learning Research, JMLR Workshop and Conference Proceedings, pp 1558–1566

Larsson G, Maire M, Shakhnarovich G (2017) FractalNet: Ultra-Deep Neural Networks without Residuals. In: 5th International Conference on Learning Representations, ICLR 2017, Toulon, France, April 24-26, 2017, Conference Track Proceedings

Le L, Patterson A, White M (2018) Supervised Autoencoders: Improving Generalization Performance with Unsupervised Regularizers. In: Bengio S, Wallach HM, Larochelle H et al (eds) Advances in Neural Information Processing Systems 31: Annual Conference on Neural Information Processing Systems 2018, NeurIPS 2018, December 3-8, 2018, Montreal, Canada. Curran Associates Inc., pp 107–117

Lee J, Lee Y, Kim J et al (2019) Set Transformer: A Framework for Attention-based Permutation-Invariant Neural Networks. In: Proceedings of the 36th International Conference on Machine Learning, ICML 2019, 9-15 June 2019, Long Beach, California, USA, Proceedings of Machine Learning Research, vol 97. Proceedings of Machine Learning Research, pp 3744–3753

Li J, Villegas R, Ceylan D et al (2021) Task-Generic Hierarchical Human Motion Prior using VAEs. In: International Conference on 3D Vision, 3DV 2021, London, United Kingdom, December 1-3, 2021. IEEE, pp 771–781

Li M, Huang B, Tian G (2022) A comprehensive survey on 3D face recognition methods. Eng Appl Artif Intell 110(104):669

Li S, Lee D (2019) Point-To-Pose Voting Based Hand Pose Estimation Using Residual Permutation Equivariant Layer. In: IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2019, Long Beach, CA, USA, June 16-20, 2019. IEEE, pp 11,927–11,936

Li S, Luo Z, Zhen M et al (2019) Cross-Atlas Convolution for Parameterization Invariant Learning on Textured Mesh Surface. In: IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2019, Long Beach, CA, USA, June 16-20, 2019. IEEE, pp 6143–6152

Li S, Liu M, Walder C (2022) EditVAE: Unsupervised Parts-Aware Controllable 3D Point Cloud Shape Generation. Proc AAAI Conf Artif Intell 36(2):1386–1394

Lim JJ, Pirsiavash H, Torralba A (2013) Parsing IKEA Objects: Fine Pose Estimation. In: IEEE International Conference on Computer Vision, ICCV 2013, Sydney, Australia, December 1-8, 2013. IEEE Computer Society, pp 2992–2999

Lin TY, Dollar P, Girshick R et al (2017) Feature Pyramid Networks for Object Detection. In: 2017 IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2017, Honolulu, HI, USA, July 21-26, 2017. IEEE Computer Society, pp 936–944

Liu H, Chen T, Guo P et al (2019a) Non-Local Attention Optimized Deep Image Compression. Computing Research Repository arxiv:abs/1904.09757

Liu J, Mills S, McCane B (2020a) Variational Autoencoder for 3D Voxel Compression. In: 35th International Conference on Image and Vision Computing New Zealand, IVCNZ 2020, Wellington, New Zealand, November 25-27, 2020. IEEE, pp 1–6

Liu J, Shahroudy A, Perez M et al (2020) NTU RGB+D 120: a large-scale benchmark for 3D human activity understanding. IEEE Trans Pattern Anal Mach Intell 42(10):2684–2701

Liu S, Giles L, Ororbia A (2018) Learning a Hierarchical Latent-Variable Model of 3D Shapes. In: 2018 International Conference on 3D Vision, 3DV 2018, Verona, Italy, September 5-8, 2018. IEEE Computer Society, pp 542–551

Liu X, Han Z, Liu YS et al (2019b) Point2Sequence: Learning the Shape Representation of 3D Point Clouds with an Attention-based Sequence to Sequence Network. In: The Thirty-Third AAAI Conference on Artificial Intelligence, AAAI 2019, The Thirty-First Innovative Applications of Artificial Intelligence Conference, IAAI 2019, The Ninth AAAI Symposium on Educational Advances in Artificial Intelligence, EAAI 2019, Honolulu, Hawaii, USA, January 27—February 1, 2019. Association for the Advancement of Artificial Intelligence Press, pp 8778–8785

Locatello F, Bauer S, Lucic M et al (2019) Challenging Common Assumptions in the Unsupervised Learning of Disentangled Representations. In: Chaudhuri K, Salakhutdinov R (eds) Proceedings of the 36th International Conference on Machine Learning, ICML 2019, 9-15 June 2019, Long Beach, California, USA, Proceedings of Machine Learning Research, vol 97. Proceedings of Machine Learning Research, pp 4114–4124

Loper M, Mahmood N, Romero J et al (2015) SMPL: a skinned multi-person linear model. ACM Trans Graph 34(6):248:1-248:16

Lyu C, Shu H (2021) A Two-Stage Cascade Model with Variational Autoencoders and Attention Gates for MRI Brain Tumor Segmentation. In: Brainlesion: Glioma, Multiple Sclerosis, Stroke and Traumatic Brain Injuries—6th International Workshop, BrainLes 2020, Held in Conjunction with MICCAI 2020, Lima, Peru, October 4, 2020, Revised Selected Papers, Part I, Proceedings of Machine Learning Research, vol 12658. Springer International Publishing, pp 435–447, https://github.com/shu-hai/two-stage-VAE-Attention-gate-BraTS2020

Ma JW, Czerniawski T, Leite F (2020) Semantic segmentation of point clouds of building interiors with deep learning: augmenting training datasets with synthetic BIM-based point clouds. Autom Constr 113(103):144

Ma Q, Yang J, Ranjan A et al (2020b) Learning to Dress 3D People in Generative Clothing. In: 2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition, CVPR 2020, Seattle, WA, USA, June 13-19, 2020. IEEE, online, pp 6468–6477, https://cape.is.tue.mpg.de/

Mafi M, Martin H, Cabrerizo M et al (2019) A comprehensive survey on impulse and Gaussian denoising filters for digital images. Signal Process 157:236–260

Mahmood N, Ghorbani N, Troje NF et al (2019) AMASS: Archive of Motion Capture As Surface Shapes. In: 2019 IEEE/CVF International Conference on Computer Vision, ICCV 2019, Seoul, Korea (South), October 27—November 2, 2019. IEEE, pp 5442–5451

Makhzani A, Shlens J, Jaitly N et al (2015) Adversarial Autoencoders. Computing Research Repository arxiv:abs/1511.05644

Mao W, Liu M, Salzmann M et al (2019) Learning Trajectory Dependencies for Human Motion Prediction. In: 2019 IEEE/CVF International Conference on Computer Vision, ICCV 2019, Seoul, Korea (South), October 27—November 2, 2019. IEEE, pp 9489–9497

Mao W, Liu M, Salzmann M (2020) History Repeats Itself: Human Motion Prediction via Motion Attention. In: Vedaldi A, Bischof H, Brox T et al (eds) Computer Vision—ECCV 2020—16th European Conference, Glasgow, UK, August 23-28, 2020, Proceedings, Part XIV, Proceedings of Machine Learning Research, vol 12359. Springer International Publishing, pp 474–489

Mao X, Li Q, Xie H et al (2017) Least Squares Generative Adversarial Networks. In: IEEE International Conference on Computer Vision, ICCV 2017, Venice, Italy, October 22-29, 2017. IEEE Computer Society, pp 2813–2821

Masuda M, Hachiuma R, Fujii R et al (2021) Toward Unsupervised 3D Point Cloud Anomaly Detection Using Variational Autoencoder. In: 2021 IEEE International Conference on Image Processing, ICIP 2021, Anchorage, AK, USA, September 19-22, 2021. IEEE, pp 3118–3122

Maturana D, Scherer S (2015) VoxNet: A 3D Convolutional Neural Network for Real-Time Object Recognition. In: 2015 IEEE/RSJ International Conference on Intelligent Robots and Systems, IROS 2015, Hamburg, Germany, September 28—October 2, 2015. IEEE, pp 922–928

Meng HY, Gao L, Lai Y et al (2019) VV-Net: Voxel VAE Net with Group Convolutions for Point Cloud Segmentation. In: 2019 IEEE/CVF International Conference on Computer Vision, ICCV 2019, Seoul, Korea (South), October 27—November 2, 2019. IEEE, pp 8499–8507

Mi L, Shen M, Zhang J (2018) A Probe Towards Understanding GAN and VAE Models. Computing Research Repository arxiv:abs/1812.05676

Mildenhall B, Srinivasan PP, Tancik M et al (2020) NeRF: Representing Scenes as Neural Radiance Fields for View Synthesis. In: Vedaldi A, Bischof H, Brox T et al (eds) Computer Vision—ECCV 2020—16th European Conference, Glasgow, UK, August 23-28, 2020, Proceedings, Part I, Proceedings of Machine Learning Research, vol 12346. Springer International Publishing, pp 405–421

Mishra D, Singh S, Singh R (2022) Deep architectures for image compression: a critical review. Signal Process 191(108):346

Mittal M, Behl HS (2018) Variational Autoencoders: A Brief Survey. https://www.semanticscholar.org/paper/Variational-Autoencoders%3A-A-Brief-Survey-Mittal-Behl/c1630a31e3aa24c9876aa956907a1ea86e9934f4

Mo K, Guerrero P, Yi L et al (2019) StructureNet: hierarchical graph networks for 3D shape generation. ACM Trans Graph 38(6):242:1-242:19

Mo K, Zhu S, Chang AX et al (2019b) PartNet: A Large-Scale Benchmark for Fine-Grained and Hierarchical Part-Level 3D Object Understanding. In: IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2019, Long Beach, CA, USA, June 16-20, 2019. IEEE, pp 909–918

Molnár S, Tamás L (2022) Representation Learning for Point Clouds with Variational Autoencoders. In: Karlinsky L, Michaeli T, Nishino K (eds) Computer Vision—ECCV 2022 Workshops—Tel Aviv, Israel, October 23–27, 2022, Proceedings, Part VI. Lecture Notes in Computer Science, vol 13806. Springer, pp 727–737

Molnár S, Kelényi B, Tamás L (2021) ToFNest: Efficient Normal Estimation for Time-of-Flight Depth Cameras. In: IEEE/CVF International Conference on Computer Vision Workshops, ICCVW 2021, Montreal, BC, Canada, October 11-17, 2021. IEEE, online, pp 1791–1798

Murray RM, Li Z, Sastry SS (1994) A mathematical introduction to robotic manipulation, vol 1. CRC Press, Boca Raton

Myronenko A (2019) 3D MRI Brain Tumor Segmentation Using Autoencoder Regularization. In: Brainlesion: Glioma, Multiple Sclerosis, Stroke and Traumatic Brain Injuries—4th International

Workshop, BrainLes 2018, Held in Conjunction with MICCAI 2018, Granada, Spain, September 16, 2018, Revised Selected Papers, Part II, Proceedings of Machine Learning Research, vol 11384. Springer International Publishing, pp 311–320

Nair V, Hinton GE (2010) Rectified Linear Units Improve Restricted Boltzmann Machines. In: Proceedings of the 27th International Conference on Machine Learning (ICML-10), June 21-24, 2010, Haifa, Israel. Omnipress, pp 807–814

Nash C, Williams CKI (2017) The shape variational autoencoder: a deep generative model of part-segmented 3D objects. Comput Graph Forum 36(5):1–12

Neumann T, Varanasi K, Wenger S et al (2013) Sparse localized deformation components. ACM Trans Graph 32(6):179:1-179:10

Ng A, Jordan MI (2001) On Discriminative vs. Generative Classifiers: A Comparison of Logistic Regression and Naive Bayes. In: Dietterich TG, Becker S, Ghahramani Z (eds) Advances in Neural Information Processing Systems 14 [Neural Information Processing Systems: Natural and Synthetic, NIPS 2001, December 3-8, 2001, Vancouver, BC, Canada]. MIT Press, pp 841–848

Nielsen D, Jaini P, Hoogeboom E et al (2020) SurVAE Flows: Surjections to Bridge the Gap between VAEs and Flows. In: Larochelle H, Ranzato M, Hadsell R et al (eds) Advances in Neural Information Processing Systems 33: Annual Conference on Neural Information Processing Systems 2020, NeurIPS 2020, December 6-12, 2020, virtual. Curran Associates Inc

Niepert M, Ahmed M, Kutzkov K (2016) Learning Convolutional Neural Networks for Graphs. In: Balcan M, Weinberger KQ (eds) Proceedings of the 33nd International Conference on Machine Learning, ICML 2016, New York City, NY, USA, June 19-24, 2016, JMLR Workshop and Conference Proceedings, vol 48. Journal of Machine Learning Research, pp 2014–2023

Nowozin S, Cseke B, Tomioka R (2016) f-GAN: Training Generative Neural Samplers using Variational Divergence Minimization. In: Lee DD, Sugiyama M, von Luxburg U et al (eds) Advances in Neural Information Processing Systems 29: Annual Conference on Neural Information Processing Systems 2016, December 5-10, 2016, Barcelona, Spain, pp 271–279

Oktay O, Schlemper J, Folgoc LL et al (2018) Attention U-Net: Learning Where to Look for the Pancreas. Computing Research Repository arxiv:abs/1804.03999

Pandey G, Dukkipati A (2017) Variational methods for Conditional Multimodal Deep Learning. In: 2017 International Joint Conference on Neural Networks, IJCNN 2017, Anchorage, AK, USA, May 14-19, 2017. IEEE, pp 308–315

Park S, Kim H (2021) FaceVAE: generation of a 3D geometric object using variational autoencoders. Electronics 10(22):2792

Paschalidou D, Ulusoy AO, Geiger A (2019) Superquadrics Revisited: Learning 3D Shape Parsing Beyond Cuboids. In: IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2019, Long Beach, CA, USA, June 16-20, 2019. IEEE, pp 10,344–10,353

Paysan P, Knothe R, Amberg B et al (2009) A 3D Face Model for Pose and Illumination Invariant Face Recognition. In: Tubaro S, Dugelay J (eds) Sixth IEEE International Conference on Advanced Video and Signal Based Surveillance, AVSS 2009, 2–4 September 2009. IEEE Computer Society, Genova, Italy, pp 296–301

Petrovich M, Black MJ, Varol G (2021) Action-Conditioned 3D Human Motion Synthesis with Transformer VAE. In: 2021 IEEE/CVF International Conference on Computer Vision, ICCV 2021, Montreal, QC, Canada, October 10-17, 2021. IEEE, pp 10,965–10,975, https://imagine.enpc.fr/~petrovim/actor/

Pons-Moll G, Romero J, Mahmood N et al (2015) Dyna: a model of dynamic human shape in motion. ACM Trans Graph 34(4):120:1-120:14

Pu Y, Wang W, Henao R et al (2017) Adversarial Symmetric Variational Autoencoder. In: Guyon I, von Luxburg U, Bengio S et al (eds) Advances in Neural Information Processing Systems 30: Annual Conference on Neural Information Processing Systems 2017, December 4-9, 2017, Long Beach, CA, USA, pp 4330–4339

Qi CR, Su H, Mo K et al (2017a) PointNet: Deep Learning on Point Sets for 3D Classification and Segmentation. In: 2017 IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2017, Honolulu, HI, USA, July 21-26, 2017. IEEE Computer Society, pp 77–85

Qi CR, Yi L, Su H et al (2017b) PointNet++: Deep Hierarchical Feature Learning on Point Sets in a Metric Space. In: Guyon I, von Luxburg U, Bengio S et al (eds) Advances in Neural Information Processing Systems 30: Annual Conference on Neural Information Processing Systems 2017, December 4-9, 2017, Long Beach, CA, USA. Curran Associates Inc., pp 5099–5108

Qin F, Qiu S, Gao S et al (2022) 3D CAD model retrieval based on sketch and unsupervised variational autoencoder. Adv Eng Inform 51(101):427

Radford A, Metz L, Chintala S (2016) Unsupervised Representation Learning with Deep Convolutional Generative Adversarial Networks. In: Bengio Y, LeCun Y (eds) 4th International Conference on

Learning Representations, ICLR 2016, San Juan, Puerto Rico, May 2-4, 2016, Conference Track Proceedings

Ranjan A, Bolkart T, Sanyal S et al (2018) Generating 3D faces Using Convolutional Mesh Autoencoders. In: Ferrari V, Hebert M, Sminchisescu C et al (eds) Computer Vision—ECCV 2018—15th European Conference, Munich, Germany, September 8-14, 2018, Proceedings, Part III, Proceedings of Machine Learning Research, vol 11207. Springer International Publishing, pp 725–741

Razavi A, van den Oord A, Vinyals O (2019) Generating Diverse High-Fidelity Images with VQ-VAE-2. In: Wallach HM, Larochelle H, Beygelzimer A et al (eds) Advances in Neural Information Processing Systems 32: Annual Conference on Neural Information Processing Systems 2019, NeurIPS 2019, December 8-14, 2019, Vancouver, BC, Canada. Curran Associates Inc., pp 14,837–14,847

Rezende DJ, Mohamed S (2015) Variational Inference with Normalizing Flows. In: Bach FR, Blei DM (eds) Proceedings of the 32nd International Conference on Machine Learning, ICML 2015, Lille, France, 6-11 July 2015, JMLR Workshop and Conference Proceedings, vol 37. Journal of Machine Learning Research, pp 1530–1538

Rezende DJ, Mohamed S, Wierstra D (2014) Stochastic Backpropagation and Approximate Inference in Deep Generative Models. In: Proceedings of the 31th International Conference on Machine Learning, ICML 2014, Beijing, China, 21-26 June 2014, JMLR Workshop and Conference Proceedings, vol 32. Journal of Machine Learning Research, pp 1278–1286

Rios T, Sendhoff B, Menzel S et al (2019a) On the Efficiency of a Point Cloud Autoencoder as a Geometric Representation for Shape Optimization. In: IEEE Symposium Series on Computational Intelligence, SSCI 2019, Xiamen, China, December 6-9, 2019. IEEE, pp 791–798

Rios T, Wollstadt P, van Stein B et al (2019b) Scalability of Learning Tasks on 3D CAE Models Using Point Cloud Autoencoders. In: IEEE Symposium Series on Computational Intelligence, SSCI 2019, Xiamen, China, December 6-9, 2019. IEEE, pp 1367–1374

Rolfe JT (2017) Discrete Variational Autoencoders. In: 5th International Conference on Learning Representations, ICLR 2017, Toulon, France, April 24-26, 2017, Conference Track Proceedings

Romero J, Tzionas D, Black MJ (2017) Embodied hands: modeling and capturing hands and bodies together. ACM Trans Graph 36(6):1–17

Ronneberger O, Fischer P, Brox T (2015) U-Net: Convolutional Networks for Biomedical Image Segmentation. In: Navab N, Hornegger J, III WMW et al (eds) Medical Image Computing and Computer-Assisted Intervention—MICCAI 2015—18th International Conference Munich, Germany, October 5—9, 2015, Proceedings, Part III, Proceedings of Machine Learning Research, vol 9351. Springer International Publishing, pp 234–241

Rubner Y, Tomasi C, Guibas L (2000) The earth mover's distance as a metric for image retrieval. Int J Comput Vision 40(2):99–121

Ruck D, Rogers S, Kabrisky M et al (1990) The multilayer perceptron as an approximation to a Bayes optimal discriminant function. Trans Neural Netw 1(4):296–298

Saha S, Rios T, Menzel S et al (2019) Learning Time-Series Data of Industrial Design Optimization using Recurrent Neural Networks. In: Papapetrou P, Cheng X, He Q (eds) 2019 International Conference on Data Mining Workshops, ICDM Workshops 2019, Beijing, China, November 8-11, 2019. IEEE, pp 785–792

Saha S, Menzel S, Minku LL et al (2020) Quantifying the Generative Capabilities of Variational Autoencoders for 3D Car Point Clouds. In: 2020 IEEE Symposium Series on Computational Intelligence, SSCI 2020, Canberra, Australia, December 1-4, 2020. IEEE, pp 1469–1477

Saha S, Minku LL, Yao X et al (2022) Exploiting 3D variational autoencoders for interactive vehicle design. Proc Des Soc 2:1747–1756

Salimans T, Goodfellow IJ, Zaremba W et al (2016) Improved Techniques for Training GANs. In: Lee DD, Sugiyama M, von Luxburg U et al (eds) Advances in Neural Information Processing Systems 29: Annual Conference on Neural Information Processing Systems 2016, December 5-10, 2016, Barcelona, Spain. Curran Associates Inc., pp 2226–2234

Santello M, Flanders M, Soechting J (1998) Postural hand synergies for tool use. J Neurosci 18(23):10,105-10,115

Schlegl T, Seeböck P, Waldstein SM et al (2017) Unsupervised Anomaly Detection with Generative Adversarial Networks to Guide Marker Discovery. In: Niethammer M, Styner M, Aylward SR et al (eds) Information Processing in Medical Imaging—25th International Conference, IPMI 2017, Boone, NC, USA, June 25-30, 2017, Proceedings, Proceedings of Machine Learning Research, vol 10265. Springer International Publishing, pp 146–157

Schor N, Katzir O, Zhang H et al (2019) CompoNet: Learning to Generate the Unseen by Part Synthesis and Composition. In: 2019 IEEE/CVF International Conference on Computer Vision, ICCV 2019,

Seoul, Korea (South), October 27—November 2, 2019. IEEE, pp 8758–8767, https://github.com/nschor/CompoNet
Schwartz R, Dodge J, Smith NA et al (2020) Green AI. Commun ACM 63(12):54–63
Sedaghat N, Zolfaghari M, Brox T (2017) Orientation-Boosted Voxel Nets for 3D Object Recognition. In: British Machine Vision Conference 2017, BMVC 2017, London, UK, September 4-7, 2017. British Machine Vision Association
Shahroudy A, Liu J, Ng T et al (2016) NTU RGB+D: A Large Scale Dataset for 3D Human Activity Analysis. In: 2016 IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2016, Las Vegas, NV, USA, June 27-30, 2016. IEEE Computer Society, pp 1010–1019
Shen Y, Feng C, Yang Y et al (2018) Mining Point Cloud Local Structures by Kernel Correlation and Graph Pooling. In: 2018 IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2018, Salt Lake City, UT, USA, June 18-22, 2018. IEEE Computer Society, pp 4548–4557
Shu DW, Park SW, Kwon J (2019) 3D Point Cloud Generative Adversarial Network Based on Tree Structured Graph Convolutions. In: 2019 IEEE/CVF International Conference on Computer Vision, ICCV 2019, Seoul, Korea (South), October 27—November 2, 2019. IEEE, pp 3858–3867
Sidi O, van Kaick O, Kleiman Y et al (2011) Unsupervised co-segmentation of a set of shapes via descriptor-space spectral clustering. ACM Trans Graph 30(6):1–10
Simonyan K, Zisserman A (2015) Very Deep Convolutional Networks for Large-Scale Image Recognition. In: Bengio Y, LeCun Y (eds) 3rd International Conference on Learning Representations, ICLR 2015, San Diego, CA, USA, May 7-9, 2015, Conference Track Proceedings
Sinha A, Bai J, Ramani K (2016) Deep Learning 3D Shape Surfaces Using Geometry Images. In: Leibe B, Matas J, Sebe N et al (eds) Computer Vision—ECCV 2016—14th European Conference, Amsterdam, The Netherlands, October 11-14, 2016, Proceedings, Part V, Proceedings of Machine Learning Research, vol 9910. Springer International Publishing, pp 223–240
Smelik RM, Tutenel T, Bidarra R et al (2014) A survey on procedural modelling for virtual worlds. Comput Graph Forum 33(6):31–50
Smith KA, Vul E (2013) Sources of uncertainty in intuitive physics. Top Cogn Sci 5(1):185–199
Smith LN (2017) Cyclical Learning Rates for Training Neural Networks. In: 2017 IEEE Winter Conference on Applications of Computer Vision, WACV 2017, Santa Rosa, CA, USA, March 24-31, 2017. IEEE Computer Society, pp 464–472
Socher R, Lin CC, Ng AY et al (2011) Parsing Natural Scenes and Natural Language with Recursive Neural Networks. In: Getoor L, Scheffer T (eds) Proceedings of the 28th International Conference on Machine Learning, ICML 2011, Bellevue, Washington, USA, June 28—July 2, 2011. Omnipress, pp 129–136
Sohn K, Lee H, Yan X (2015) Learning Structured Output Representation using Deep Conditional Generative Models. In: Cortes C, Lawrence ND, Lee DD et al (eds) Advances in Neural Information Processing Systems 28: Annual Conference on Neural Information Processing Systems 2015, December 7-12, 2015, Montreal, Quebec, Canada. Curran Associates Inc., pp 3483–3491
Sønderby CK, Raiko T, Maaløe L et al (2016) Ladder Variational Autoencoders. In: Lee DD, Sugiyama M, von Luxburg U et al (eds) Advances in Neural Information Processing Systems 29: Annual Conference on Neural Information Processing Systems 2016, December 5-10, 2016, Barcelona, Spain. Curran Associates Inc., pp 3738–3746
Sorkine O, Alexa M (2007) As-Rigid-As-Possible Surface Modeling. In: Belyaev AG, Garland M (eds) Proceedings of the Fifth Eurographics Symposium on Geometry Processing, Barcelona, Spain, July 4-6, 2007, ACM International Conference Proceeding Series, vol 257. Eurographics Association, pp 109–116
Spanopoulos A, Konstantinidis D (2021) Disentangled variational autoencoder. https://github.com/AndrewSpano/Disentangled-Variational-Autoencoder
Spurr A, Song J, Park S et al (2018) Cross-Modal Deep Variational Hand Pose Estimation. In: 2018 IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2018, Salt Lake City, UT, USA, June 18-22, 2018. IEEE Computer Society, pp 89–98, https://ait.ethz.ch/projects/2018/vae_hands/
Srivastava N, Mansimov E, Salakhutdinov R (2015) Unsupervised Learning of Video Representations using LSTMs. In: Bach FR, Blei DM (eds) Proceedings of the 32nd International Conference on Machine Learning, ICML 2015, Lille, France, 6-11 July 2015, JMLR Workshop and Conference Proceedings, vol 37. Journal of Machine Learning Research, pp 843–852
Sumner RW, Popović J (2004) Deformation transfer for triangle meshes. ACM Trans Graph 23(3):399–405
Sun X, Wei Y, Liang S et al (2015) Cascaded Hand Pose Regression. In: IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2015, Boston, MA, USA, June 7-12, 2015. IEEE Computer Society, pp 824–832

Tagliasacchi A, Schröder M, Tkach A et al (2015) Robust articulated-ICP for real-time hand tracking. Comput Graph Forum 34(5):101–114
Tan Q, Gao L, Lai YK et al (2018) Variational Autoencoders for Deforming 3D Mesh Models. In: 2018 IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2018, Salt Lake City, UT, USA, June 18-22, 2018. IEEE Computer Society, pp 5841–5850, https://qytan.com/publication/vae/
Tang D, Chang HJ, Tejani A et al (2014) Latent Regression Forest: Structured Estimation of 3D Articulated Hand Posture. In: 2014 IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2014, Columbus, OH, USA, June 23-28, 2014. IEEE Computer Society, pp 3786–3793
Team TTD, Al-Rfou R, Alain G et al (2016) Theano: A Python Framework for Fast Computation of Mathematical Expressions. Computing Research Repository arxiv:abs/1605.02688
Theis L, van den Oord A, Bethge M (2016) A Note on the Evaluation of Generative Models. In: Bengio Y, LeCun Y (eds) 4th International Conference on Learning Representations, ICLR 2016, San Juan, Puerto Rico, May 2-4, 2016, Conference Track Proceedings
Tompson J, Stein M, Lecun Y et al (2014) Real-time continuous pose recovery of human hands using convolutional networks. ACM Trans Graph 33(5):169:1-169:10
Tran AT, Hassner T, Masi I et al (2017) Regressing Robust and Discriminative 3D Morphable Models with a Very Deep Neural Network. In: 2017 IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2017, Honolulu, HI, USA, July 21-26, 2017. IEEE Computer Society, pp 1493–1502
Vahdat A, Kautz J (2020) NVAE: A Deep Hierarchical Variational Autoencoder. In: Larochelle H, Ranzato M, Hadsell R et al (eds) Advances in Neural Information Processing Systems 33: Annual Conference on Neural Information Processing Systems 2020, NeurIPS 2020, December 6-12, 2020, virtual. Curran Associates Inc
Vaswani A, Shazeer N, Parmar N et al (2017) Attention Is All You Need. In: Guyon I, von Luxburg U, Bengio S et al (eds) Advances in Neural Information Processing Systems 30: Annual Conference on Neural Information Processing Systems 2017, December 4-9, 2017, Long Beach, CA, USA. Curran Associates Inc., pp 5998–6008
Vlasic D, Baran I, Matusik W et al (2008) Articulated mesh animation from multi-view silhouettes. ACM Trans Graph 27(3):1–9
Wan C, Probst T, Gool LV et al (2017) Crossing Nets: Combining GANs and VAEs with a Shared Latent Space for Hand Pose Estimation. In: 2017 IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2017, Honolulu, HI, USA, July 21-26, 2017. IEEE Computer Society, pp 1196–1205
Wang C, Zhang F, Ge S (2021) A comprehensive survey on 2D multi-person pose estimation methods. Eng Appl Artif Intell 102(104):260
Wang J, Zhu H, Liu H et al (2021) Lossy point cloud geometry compression via end-to-end learning. IEEE Trans Circ Syst Video Technol 31(12):4909–4923
Wang L, Yoon KJ (2021) Knowledge Distillation and Student-Teacher Learning for Visual Intelligence: A Review and New Outlooks. IEEE Transactions on Pattern Analysis and Machine Intelligence pp 1–1
Wang L, Huang Y, Tao P et al (2020) Learning Geometry-Image Representation for 3D Point Cloud Generation. Computing Research Repository arxiv:abs/2011.14289
Wang TC, Liu MY, Zhu JY et al (2018) High-Resolution Image Synthesis and Semantic Manipulation With Conditional GANs. In: 2018 IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2018, Salt Lake City, UT, USA, June 18-22, 2018. IEEE Computer Society, pp 8798–8807
Wang W, Huang Y, Wang Y et al (2014) Generalized Autoencoder: A Neural Network Framework for Dimensionality Reduction. In: IEEE Conference on Computer Vision and Pattern Recognition, CVPR Workshops 2014, Columbus, OH, USA, June 23-28, 2014. IEEE Computer Society, pp 496–503
Wang Y, Asafi S, van Kaick O et al (2012) Active co-analysis of a set of shapes. ACM Trans Graph 31(6):165:1-165:10
Wang Y, Sun Y, Liu Z et al (2019) Dynamic graph CNN for learning on point clouds. ACM Trans Graph 38(5):146:1-146:12
Wei R, Mahmood A (2021) Optimizing few-shot learning based on variational autoencoders. Entropy 23(11):1390
Wei R, Mahmood A (2021) Recent advances in variational autoencoders with representation learning for biomedical informatics: a survey. IEEE Access 9:4939–4956
Wei R, Garcia C, ElSayed A et al (2020) Variations in variational autoencoders—a comparative evaluation. IEEE Access 8:153,651-153,670
Wonka P, Wimmer M, Sillion F et al (2003) Instant architecture. ACM Trans Graph 22(3):669–677
Wu J, Zhang C, Xue T et al (2016) Learning a Probabilistic Latent Space of Object Shapes via 3D Generative-Adversarial Modeling. In: Lee DD, Sugiyama M, von Luxburg U et al (eds) Advances in

Neural Information Processing Systems 29: Annual Conference on Neural Information Processing Systems 2016, December 5-10, 2016, Barcelona, Spain. Curran Associates Inc., pp 82–90

Wu Z, Song S, Khosla A et al (2015) 3D ShapeNets: A Deep Representation for Volumetric Shapes. In: IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2015, Boston, MA, USA, June 7-12, 2015. IEEE Computer Society, pp 1912–1920

Wurm KM, Hornung A, Bennewitz M et al (2010) OctoMap: A Probabilistic, Flexible, and Compact 3D Map Representation for Robotic Systems. In: Proc. of the ICRA 2010 Workshop on Best Practice in 3D Perception and Modeling for Mobile Manipulation. IEEE, Anchorage, AK, USA

Xian Y, Sharma S, Schiele B et al (2019) F-VAEGAN-D2: A Feature Generating Framework for Any-Shot Learning. In: IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2019, Long Beach, CA, USA, June 16-20, 2019. Computer Vision Foundation / IEEE, pp 10,275–10,284

Xiang Y, Mottaghi R, Savarese S (2014) Beyond PASCAL: A benchmark for 3D object detection in the wild. In: IEEE Winter Conference on Applications of Computer Vision, Steamboat Springs, CO, USA, March 24-26, 2014. IEEE Computer Society, pp 75–82

Yamamoto K, Igarashi T (2017) Fully perceptual-based 3D spatial sound individualization with an adaptive variational autoencoder. ACM Trans Graph 36(6):1–13

Yan X, Rastogi A, Villegas R et al (2018) MT-VAE: Learning Motion transformations to Generate Multi-modal Human Dynamics. In: Ferrari V, Hebert M, Sminchisescu C et al (eds) Computer Vision—ECCV 2018—15th European Conference, Munich, Germany, September 8-14, 2018, Proceedings, Part III, Proceedings of Machine Learning Research, vol 11209. Springer International Publishing, pp 276–293, https://github.com/xcyan/eccv18_mtvae

Yang G, Huang X, Hao Z et al (2019a) PointFlow: 3D Point Cloud Generation with Continuous Normalizing Flows. In: 2019 IEEE/CVF International Conference on Computer Vision, ICCV 2019, Seoul, Korea (South), October 27—November 2, 2019. IEEE, pp 4540–4549, https://www.guandaoyang.com/PointFlow/

Yang L, Li S, Lee D et al (2019b) Aligning Latent Spaces for 3D Hand Pose Estimation. In: 2019 IEEE/CVF International Conference on Computer Vision, ICCV 2019, Seoul, Korea (South), October 27—November 2, 2019. IEEE, pp 2335–2343

Yang M, Wen Y, Chen W et al (2021) Deep Optimized Priors for 3D Shape Modeling and Reconstruction. In: IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2021, virtual, June 19-25, 2021. IEEE, pp 3269–3278

Yang Y, Feng C, Shen Y et al (2018) FoldingNet: Point Cloud Auto-Encoder via Deep Grid Deformation. In: 2018 IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2018, Salt Lake City, UT, USA, June 18-22, 2018. IEEE Computer Society, pp 206–215

Ye F, Bors AG (2020) Learning Latent Representations Across Multiple Data Domains Using Lifelong VAEGAN. In: Vedaldi A, Bischof H, Brox T et al (eds) Computer Vision—ECCV 2020–16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part XX. Lecture Notes in Computer Science, vol 12365. Springer, pp 777–795

Ye F, Bors AG (2021) Learning joint latent representations based on information maximization. Inf Sci 567:216–236

Yu F, Koltun V (2016) Multi-Scale Context Aggregation by Dilated Convolutions. In: Bengio Y, LeCun Y (eds) 4th International Conference on Learning Representations, ICLR 2016, San Juan, Puerto Rico, May 2-4, 2016, Conference Track Proceedings

Yu F, Liu K, Zhang Y et al (2019a) PartNet: A Recursive Part Decomposition Network for Fine-Grained and Hierarchical Shape Segmentation. In: IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2019, Long Beach, CA, USA, June 16-20, 2019. IEEE, pp 9491–9500

Yu H, Lee B (2019) Zero-shot Learning via Simultaneous Generating and Learning. In: Wallach HM, Larochelle H, Beygelzimer A et al (eds) Advances in Neural Information Processing Systems 32: Annual Conference on Neural Information Processing Systems 2019, NeurIPS 2019, December 8-14, 2019, Vancouver, BC, Canada. Curran Associates Inc., pp 46–56

Yu H, Oh J (2022) Anytime 3D Object Reconstruction Using Multi-Modal Variational Autoencoder. IEEE Robotics and Automation Letters 7(2):2162–2169. https://github.com/bogus2000/anytime-3D-reconstruction

Yu HW, Lee BH (2018) A Variational Feature Encoding Method of 3D Object for Probabilistic Semantic SLAM. In: 2018 IEEE/RSJ International Conference on Intelligent Robots and Systems, IROS 2018, Madrid, Spain, October 1-5, 2018. IEEE, pp 3605–3612

Yu HW, Moon JY, Lee BH (2019b) A Variational observation Model of 3D Object for Probabilistic Semantic SLAM. In: International Conference on Robotics and Automation, ICRA 2019, Montreal, QC, Canada, May 20-24, 2019. IEEE, pp 5866–5872

Yu X, Tang L, Rao Y et al (2022) Point-BERT: Pre-Training 3D Point Cloud Transformers With Masked Point Modeling. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR) Workshops. IEEE, New Orleans, Louisiana, USA, pp 19,313–19,322, https://github.com/lulutang0608/Point-BERT

Yuan YJ, Lai YK, Yang J et al (2020) Mesh Variational Autoencoders with Edge Contraction Pooling. In: 2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition, CVPR Workshops 2020, Seattle, WA, USA, June 14-19, 2020. IEEE, online, pp 1105–1112, https://github.com/IGLICT/MeshPooling

Zafeiriou S, Kollias D, Nicolaou MA et al (2017) Aff-Wild: Valence and Arousal 'In-the-Wild' Challenge. In: 2017 IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2017, Honolulu, HI, USA, July 21-26, 2017. IEEE Computer Society, pp 1980–1987

Zamorski M, Zięba M, Klukowski P et al (2020) Adversarial Autoencoders for Compact Representations of 3D Point Clouds. Computer Vision and Image Understanding 193:102,921. https://github.com/MaciejZamorski/3d-AAE

Zhang J, Jiao J, Chen M et al (2016) 3D Hand Pose Tracking and Estimation Using Stereo Matching. Computing Research Repository arxiv:abs/1610.07214

Zhang L, Snavely N, Curless B et al (2004) Spacetime faces: high-resolution capture for modeling and animation. ACM Trans Graph 23(3):548–558

Zhang X, Ma R, Zou C et al (2021) View-Aware Geometry-Structure Joint Learning for Single-View 3D Shape Reconstruction. IEEE Transactions on Pattern Analysis and Machine Intelligence pp 1–1. https://github.com/Mehooz/VGSNet

Zhang Y, Hare JS, Prügel-Bennett A (2019) Deep Set Prediction Networks. In: Wallach HM, Larochelle H, Beygelzimer A et al (eds) Advances in Neural Information Processing Systems 32: Annual Conference on Neural Information Processing Systems 2019, NeurIPS 2019, December 8-14, 2019, Vancouver, BC, Canada. Curran Associates Inc., pp 3207–3217

Zhao H, Jiang L, Jia J et al (2021) Point Transformer. In: 2021 IEEE/CVF International Conference on Computer Vision, ICCV 2021, Montreal, QC, Canada, October 10-17, 2021. IEEE, pp 16,239–16,248

Zheng C, Cham TJ, Cai J (2019) Pluralistic Image Completion. In: IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2019, Long Beach, CA, USA, June 16-20, 2019. IEEE, pp 1438–1447

Zhou Y, Barnes C, Lu J et al (2019) On the Continuity of Rotation Representations in Neural Networks. In: IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2019, Long Beach, CA, USA, June 16-20, 2019. IEEE, pp 5745–5753

Zhu X, Lei Z, Liu X et al (2016) Face Alignment Across Large Poses: A 3D Solution. In: 2016 IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2016, Las Vegas, NV, USA, June 27-30, 2016. IEEE Computer Society, pp 146–155

Zimmermann C, Brox T (2017) Learning to Estimate 3D Hand Pose from Single RGB Images. In: IEEE International Conference on Computer Vision, ICCV 2017, Venice, Italy, October 22-29, 2017. IEEE Computer Society, pp 4913–4921