Synthetic data generation using VAE for novel views.

|  |  |  |
| --- | --- | --- |
|  | Jyoti Madake  Department of Electronics and Telecommunication Engineering Vishwakarma Institute of Technology Pune, India jyoti.madake@vit.edu | Shaurya Singh  Department of Electronics and Telecommunication Engineering Vishwakarma Institute of Technology Pune, India  Shaurya.singh19@vit.edu |

**Abstract:** **Synthetic data generation plays a key role in solving many of the complex machine learning problems. Since this problem deals with generation which is non-deterministic unlike discriminators that we are familiar with like for example classifiers, they are especially tricky to deal with. When utilized correctly, proper synthetic generation can aid many problems especially those which lack relevant data so it is especially beneficial if we can generate high fidelity images based on one or a couple of training instances. This paper is an illustration of this concept, the aim is to build a generation system using a variational autoencoder network trained on a chair dataset with 599 classes and about 37000 images which can be used to produce new data on chairs, and further be implemented for using in novel generation, which is useful for single-view 3D object reconstruction. We also train a DCGAN on the same dataset to prove the VAEs efficacy over it in terms of ease of training and much less tuning while also providing subjectively good reconstructions.**

INTRODUCTION

All complex machine learning problems rely on data, and for training supervised learning systems we need not only a large amount of data, but we also need the data to be annotated and labelled so the model knows if it’s doing its given task correctly or not. This deterministic approach has had a massive impact on the world in applications ranging from finance to autonomous cars. On the flipside, an emerging field over the last decade has been the progress made in the generative side of things which is by nature non-deterministic and by and large unsupervised. This field is exciting because as un-supervised systems do, they learn the nature or rules of objects rather than the objects themselves. They pose a somewhat deeper understanding of problems they solve than their deterministic counter parts. The issue we have with data is not the lack thereof for supervised systems, but rather the lack of correctly annotated labelled datasets which are usually done manually and are rather tedious. Generative systems only require data, which we have plenty of, and assumes that under every data instance belonging to a dataset there is a unknown probability distribution that explains why a certain data instance belongs to a certain dataset. Once they learn this probability distribution, they are able to generate new high-fidelity samples that look like they could have been plucked straight from the training dataset. Its applications using un-structured data are endless, and we have recently seen a boom in the domains it excels at like NLP and image processing.

Image-based 3D object reconstruction has many sprawling applications in today’s day and age, from reconstructing tumours from scans to better understand the underlying issue, to improve the cheapest and easiest to work with sensory data in robotics i.e., visual data from cameras to better map its surroundings, to mapping human shapes and poses for better animations. Its importance is almost understated, and its potential is unlimited.

To create an accurate 3D object reconstruction from images we need as many images of the object in question as possible so as to minimize loss of information in the 3D object. The problem of single-view 3D object reconstruction can be generalized into the following subproblems:

1. Encoding the input image for which we wish to generate novel views into a vector with well-preserved information.
2. Use the encoded vector as an input to a trained generator to output novel views which can be used to better map a 3D object.
3. The 3D object generator which takes the multi-view images and encodes them to output a 3D generated decoded mesh/point-cloud/voxel.

This paper aims to aid the 2nd process in the pipeline regarding synthetic data generation which is to be used as a novel view generator in the context of single-view 3D object reconstruction.

LITERATURE REVIEW

Regarding prior works, this problem has garnered increasing attention since the 2000s, [1] is a comprehensive survey on methods prevalent for this challenge in 2006. This problem had its second wave in the mid-2010s with the advent of machine learning, [2] is a comprehensive study detailing the development of this problem since 2015 discussing 149 papers with a broad spectrum of approaches. There are also case studies done on this problem, [3] is a case study that evaluates the accuracy of a particular reconstruction method which is low-cost and not as hardware demanding as some of its contemporary solutions.

These are exciting times for synthetic data generation, illustrated by recent developments in prompt-based image generation [4] [5]. Some key advancements were made in the last decade to aid 2D, and 3D-based synthetic data generation. [6] demonstrated over-reliance on CNNs in image classification tasks and exemplified the efficacy of pure transformers. [7] illustrated the utility of Fourier features bettering results for various high-frequency low-dimensional tasks like image and 3D shape regression. Effectively encapsulating an input signal is a crucial challenge in computer vision, [8] demonstrates the challenges in existing implicit neural network designs whenever it concerns modeling signals with intricate details. They propose a solution that depicts intricate details of objects more precisely.

Human posture and form regression is a 3D reconstruction application with a large body of research. [9] SCAPE, [10] SMPL, and [11] SMPL-X are solutions for identifying and rebuilding human shapes and poses.

This sub-domain has many novel applications; [12] showcases SHAPY, a method for accurately predicting the shape of a human body from an RGB image by predicting parameters of the SMPL-X model that maps shape, pose, and expression into a 3D mesh.[13] demonstrate BANMo, an approach for generating precise and animated 3D object models from monocular videos. [14] is a 2003 method delving into extracting, reconstructing, and modeling human figures from a single image or a monocular video sequence. [15] propose ICON, a technique designed to create an avatar entirely from 2D pictures of real individuals in natural poses. This model generates a pixel-aligned 3D shape reconstruction of the clothed person from an SMPL estimated body and an RGB image of segmented clothes.[16] Discusses the challenges of collecting geometric and aesthetic information in 3D morphable faces, proposing IMAvatar, а novel approach for inferring implicit head avatars from monocular videos.[17] describe a novel technique for capturing the motion of the main body parts for several interacting people.

Image-based 3D reconstruction has an intriguing use in archaeological excavation, [18] demonstrated the efficacy of image-based 3D modeling for recording, documenting, and displaying archaeological heritage.

Existing image-based 3D reconstruction methods are as diverse as their applications. Looking into single-view 3D reconstruction, [19] uses a dynamic Bayesian network (DBN) to approximate a distribution across probable scene structures to illustrate an autonomous 3D reconstruction system of interior scenes from a single image.[20] Create a 3D geometry using a point set generation network and presents a solution to the ground truth ambiguity problem for 3D reconstruction from a single image.[21] builds a voxel model from a single RGB picture using a conditional adversarial volumetric Z-GAN framework and outperforms other state-of-the-art approaches such as MarrNet and 3D-R2N2. [22] present a CNN architecture that infers a 3D model from an image and predicts the object's geometry and depth map; several views are merged into a complete 3D point cloud, which is subsequently optimized into a mesh. [23] illustrates a novel single-view architecture that creates a 3D mesh model. This network predicts 3D geometry in a rough to fine manner as it is easier to train and more consistent. [24] illustrate the effectiveness of surface-based predicted representations over volumetric-based predicted representations for a 3D shape. Making structured predictions can help with the problem of unobserved voxels in a depth picture. This is accomplished via an algorithm [25] that completes the unseen geometry of tabletop-sized objects using a supervised model trained on volumetric components. [26] presents 3D-RecGAN++, a novel technique that uses generative adversarial networks to reconstruct the 3D structure of an object fully from a single-depth photograph. [27] was possibly the first technique that achieved autonomous object reconstruction from a single image on a large and realistic dataset. The method proposed employs estimated instance segmentations and predicted viewpoints to reconstruct a complete 3D mesh and high-frequency 2.5D depth maps. [28] solves estimating depths from just a single image, the proposed method is a deep convolutional neural field for depth estimation by exploring CNN and continuous CRF. [29] cite the overabundance of research on depth estimate using stereo photos while criticizing the dearth of research at the time on depth estimation using single RGB images. In their innovative approach, the authors of this work directly regress depth using data from a neural network.

Coming to single/multi-view 3D reconstruction, [30] presents a 3D recurrent reconstruction neural network that extends the traditional LSTM architecture and integrates single and multi-view 3D reconstruction. This method required little monitoring during training and testing; all that is needed are bounding boxes. [31] demonstrates that modeling volumetric objects in a generative adversarial approach is a potential technique for creating innovative and realistic objects. [32] propose a novel approach for single-image 3D object reconstruction that directly models 2.5D sketches. Including 2.5D sketches improves the model's performance and adaptability to images from various categories and domains. [33] propose a novel, efficient 2D encoding for 3D geometry that allows the reconstruction of entire 3D structures from a single high-resolution image. [34] employ an estimated gradient for mesh rendering, enabling rendering to be integrated into neural networks. [35] Propose Pix2Vox++, a unified framework for single and multi-view 3D reconstruction, with an encoder, decoder, and refiner capable of handling 3D reconstruction on real-world and synthetic images.

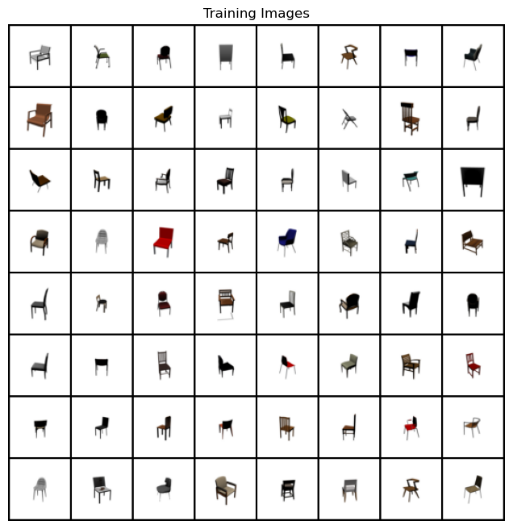
Some other novel applications include [36], which proposes a solution for the 3D construction of objects rotated by hand. [37] is a method for reasoning about the physical stability of 3D volumetric objects reconstructed using depth pictures recorded by a range camera or a large-scale point cloud scene rebuilt using the SLAM methodology. [38] Hypothesize that to learn a good vector representation, it needs to be generative and predictable since both properties are crucial for image understanding tasks. [39] demonstrate that neural networks can predict 3D shapes from a single image without using ground truth 3D volumetric data in training. [40] demonstrate the automatic 3D reconstruction of objects depicted in web images. [41] SoftRas, a truly differentiable rendering framework capable of directly rendering a mesh in a fully differentiable way, is presented in this study. [42] investigates the PASCAL VOC dataset's recognition and reconstruction challenge.

METHODOLOGY

Synthetic data generation is done using two architectures to demonstrate the efficacy of one over the other for this constrained example. The first technique talked about is the DCGAN (Deep Convolutional Generative Adversarial Network) which is a state-of-the-art architecture for GANs and demonstrate the difficulties it faces for this scenario. The second technique is the VAE (Variational Autoencoder) which illustrates its strengths over GANs for this particular problem.

1. *Dataset and pre-processing:*

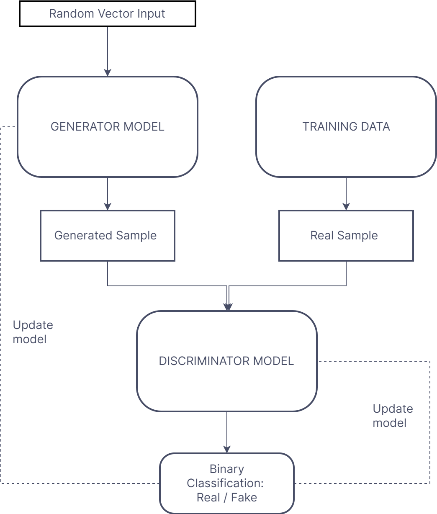
The dataset used is a subset of the large dataset of chair CAD models[43] containing 599 classes and 37138 images.



The original images are 600x600 RGB images, for both architectures we feed the models a resized image of 64x64 pixels. For the DCGAN the images are further pre-processed to be centre cropped, normalized and converted to tensors. They are fed to the model in training in batches of 128 as is suggested by the paper [44]. For VAE the images are only converted to tensors and fed to the network in batches of 64, no further pre-processing is done. Here we do split the images into a train and validation set, containing 32138 images and 5000 respectively. The validation loop does not adjust the model parameters and is only there to evaluate the model’s performance on new instances it has not seen before.

1. *Architecture of DCGAN*

Generative adversarial network (GAN) is a learning technique with two neural networks, one being the generator model and the other being the discriminator model. The idea is simple, the generator model outputs fake data from random vector inputs (noise) the then the discriminator attempts to detect whether the data is fake or real. The discriminator is a regular binary classifier, and the goal is to fool it into thinking fake data is real data as many times as possible.

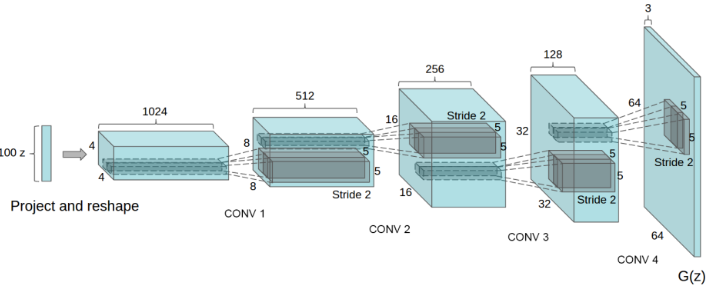


The training happens in tandem, the generator and then the discriminator and then the generator again every epoch. As we will see, GANs are notoriously difficult to train, and are very sensitive to hyperparameters, activation functions and regularization techniques.

***Implementation:***

DCGANs exploit the strengths of convolutional neural networks in the unsupervised domain of image generation with GAN.

It’s stated in the original paper that the model weights need to be randomly initialized from a normal distribution with mean = 0 and standard deviation being = 0.

*Generator G:* 

This module is designed to convert the latent space vector *z* into a dataspace bearing the same dimensions as our training data (3x64x64). This is done by imposing a series of 2D convolutional transpose layers, 5 in total. Each layer follows a 2D batch normalization layer and the ReLU activation function. The output layer of the generator is fed through a tanh function which returns it to the input data range of [-1,1]. (We essentially applied the same transformation to the images in the training dataset because the training dataset was normalized putting each pixel value in the range[-1,1].)

*Discriminator D:*

This is a binary classification network that takes an image as input and outputs the probability that the image is real or fake. Here the discriminator takes a 3x64x64 input image and then processes it through a series of 2D convolutional layers followed by a batch normalization layer and leaky ReLU activation layers. There are 5 layers in total with the final layer outputting the final probability through a sigmoid function. The DCGAN paper mentions it is a good practice to use stride convolution rather than pooling layers to down sample because it lets the network learn its pooling function. Batch normalization and leaky ReLU functions promote healthy gradient flow which is critical for the learning process of both the generator and discriminator.

*Loss functions and optimizers:*

Loss function used here is binary cross entropy (BCE) since it’s a binary classification problem for the discriminator.

We define real label as 1 and the fake label as 0, these labels are used when calculating the losses of the discriminator and the generator. We make use of two separate optimizers for the discriminator and the generator. The paper specifies the learning rate to be at 0.0002. To keep track of the generators learning progression, we generate a fixed batch of random latent vectors (fixed noise). When in the training loop we periodically input this fixed noise into the generator to see the images generated iteratively.

*Training the discriminator:*

The goal of the discriminator is to maximize the probability of correctly classifying a given input as real or fake. We want to maximise the discriminator loss:

First, we compute loss on real samples from the training set log(D(x)), then calculate the gradients. Then we construct a batch of fake samples with the current generator, forward pass it through D, calculate its loss log(1-D(G(z))), compute its gradients. Now that we have the gradients of the real and fake batches, we update the discriminator model parameters using its optimizer.

Text, table

Description automatically generated A picture containing text, crossword puzzle

Description automatically generated

The above image shows a real batch of images and generated batch of images.

*Training the generator.*

We want to train the generator to minimize:

in order to generate better fake images. But this method does not provide enough gradients, so another suggested method is to maximize . In the code implementation this is done by classifying the generator output with the discriminator, then computing its loss using real labels, then its gradients and finally updating its model parameters using the optimizer algorithm. The model was trained for 15 epochs.

In the end we provide some statistics during the training loop

A screenshot of a computer

Description automatically generated with medium confidence

We display the discriminator loss, the generator loss, discriminator confidence one real images [D(x)] and discriminator confidence on generated images [D(G(z))]

D(x) should ideally start off at 1 and then converge at 0.5, whereas D(G(z)) should start off at 0 and then slowly converge at 0.5.

1. *Architecture of VAE*

Variational Autoencoders (VAE) are also used to generate new images from a latent vector. Although they reconstruct images similar to the images they are trained on, they can be trained on many variations of the image. In terms of architecture, they reassemble a standard autoencoder, consisting of an encoder and a decoder.

Diagram

Description automatically generated with medium confidence

The major difference between VAE and standard autoencoder is that the latent vector generated by VAEs is continuous which making them a generative neural network.

Diagram

Description automatically generated

Standard encoders produce image data from the latent vectors, but they try to replicate the image data while doing so. VAEs are good at generating new images from the latent vector, although the new data is very similar to the data they are trained on.

*Implementation:*

We don’t need detailed images, nor do we need high resolution images therefore reducing the need for GANs. They are also significantly harder to train, whereas VAEs are straightforward to train. Outputs for VAEs are generally blurry, but they get the job done.

*Preparing the VAE model.*

For the first conv layer we have the number of output channels set to be 64 with it doubling with each conv layer. Image channels is 3. The latent dimensions is 100, which is the number of features we consider while sampling using reparameterization. This is also the same latent size used in the DCGAN.

The encoder has 5 2D conv layers with the last layer having an output channel of 1024. After this we have 4 FC layers, 2 of which provide us the mean and log variance for sampling. After this we have the decoder which consists of 5 transposed layers.

Once the model is defined, we have 2 methods, reparametrize() for sampling and the other one being the general forward function.

*Writing the Loss, training, and validation functions.*

*Loss:* The final loss is BCE (binary cross entropy) and KL divergence. KL divergence simply compares the probability distribution of the input image and the generated image. The goal is finding the probability distribution of an image that tells us why it belongs to a class or dataset. BCE loss is the reconstruction loss. This compares the input images and the images that are reconstructed by the VAE.

We add the BCE with the KL divergence which gives us the final loss value for each training step.

*Training:* We keep the training loop simple to minimize room for error and simplify debugging. We have the train function which accepts the CVAE model, train data loader, the training dataset, computation device, optimizer, and the criterion.

The training steps are the usual, we make predictions, by passing the image data through the model and are returned with reconstruction image, the mean, and the log variance. Then we calculate the loss and then compute gradients, update model parameters, and reset gradients. We train the model for 75 epochs.

*Validation:* In the validation loop we don’t backpropagate the loss or update parameters, we just save the reconstructed images.

RESULTS

*DCGAN:*

Table

Description automatically generatedTable, calendar

Description automatically generated

0 % training output 25% training output

Table

Description automatically generatedTable, calendar

Description automatically generated

75% training output 100% training output

As we can observe, the model through each training phase only really understood a vague understanding of what a chair looks like. The bulk of the images are white blocks and sludges of colour with only a few images representing a vague silhouette of a chair. Also given there are 599 classes of different types of chairs, the final output and the generated images throughout training only seemed to understand a limited amount of chair representations, which is definitely not ideal in the case of synthetic data generation where variety is key.

A screenshot of a game

Description automatically generated with low confidence

Here we compare reconstructed image from a fixed batch, we observe that not even one image vaguely resembles it its counterpart image that it is trying to reconstruct.

Chart

Description automatically generated

Here we plot the loss of the generator against the Discriminator. Ideally, both the plots should converge at a number which usually indicated that the generator has figured out a competent probability distribution, and this would also show that the difference in generator losses compared to discriminator losses is minimal. Here we observe a steady rift in between the discriminator and generator loss, which is illustrated in the quality of the reconstructed images.

*VAE:*

epoch 1
Table

Description automatically generated with low confidence

0TH epoch 5TH epoch

Text

Description automatically generated with medium confidenceText

Description automatically generated with medium confidence

25th epoch 50th epoch

Text

Description automatically generated with medium confidence

75th epoch – The final result.

This learning progress illustrated by the above images showcases the prowess VAE has over DCGAN over this constrained problem type. The model is quickly able to learn the high-level features of the training data and does not generate the sludges and white backgrounds in any capacity. It could generate distinct, usable generated images in just 5 epochs, although a little blurry. Further training epochs really sharpen the images.

Chart, line chart

Description automatically generated

Here we plot the validation loss and training loss. The training loss starts at a very high 90000, but quickly tumbles down.

The interesting takeaway here is that the validation curve and the training curve actually meet each other very early in the training process, between 5 and 10 epochs. Before that, the model was only slightly underfitting. Post 10 epochs, the validation curve seems to trail further and further away from the training curve, implying that model is doubling down on overfitting the more we epochs we train it.

CONCLUSION

The lack of meaningful results using DCGAN is not too surprising as this clearly showcases how difficult it is to GANs, given how many variabilities there are to tweak in the model and the dataset. Given our problem set, VAE is able to reconstruct high fidelity images with a simpler training procedure.

The results could still be tweaked for VAE, more classes could be trained and also in higher dimensions which is refrained here due to training time and hardware constraints.

As illustrated in the above plot, overtraining the model with the learning rate and epochs used lead to overfitting. Using a learning rate scheduler to train for longer and/or stop the training when the loss plateaus should help circumvent this issue.

REFERENCES

[1] Han, Xian-Feng, Hamid Laga, and Mohammed Bennamoun. "Image-based 3D object reconstruction: State-of-the-art and trends in the deep learning era." *IEEE transactions on pattern analysis and machine intelligence* 43, no. 5 (2019): 1578-1604.

[2] Remondino, Fabio, and Sabry El‐Hakim. "Image‐based 3D modelling: a review." *The photogrammetric record* 21, no. 115 (2006): 269-291.

[3] Koutsoudis, Anestis, Blaž Vidmar, George Ioannakis, Fotis Arnaoutoglou, George Pavlidis, and Christodoulos Chamzas. "Multi-image 3D reconstruction data evaluation." *Journal of cultural heritage* 15, no. 1 (2014): 73-79.

[4] Ramesh, Aditya, Mikhail Pavlov, Gabriel Goh, Scott Gray, Chelsea Voss, Alec Radford, Mark Chen, and Ilya Sutskever. "Zero-shot text-to-image generation." In *International Conference on Machine Learning*, pp. 8821-8831. PMLR, 2021.

[5] Dhariwal, Prafulla, and Alexander Nichol. "Diffusion models beat gans on image synthesis." *Advances in Neural Information Processing Systems* 34 (2021): 8780-8794.

[6] Dosovitskiy, Alexey, Lucas Beyer, Alexander Kolesnikov, Dirk Weissenborn, Xiaohua Zhai, Thomas Unterthiner, Mostafa Dehghani et al. "An image is worth 16x16 words: Transformers for image recognition at scale." *arXiv preprint arXiv:2010.11929* (2020).

[7] Tancik, Matthew, Pratul Srinivasan, Ben Mildenhall, Sara Fridovich-Keil, Nithin Raghavan, Utkarsh Singhal, Ravi Ramamoorthi, Jonathan Barron, and Ren Ng. "Fourier features let networks learn high frequency functions in low dimensional domains." *Advances in Neural Information Processing Systems* 33 (2020): 7537-7547.

[8] Sitzmann, Vincent, Julien Martel, Alexander Bergman, David Lindell, and Gordon Wetzstein. "Implicit neural representations with periodic activation functions." *Advances in Neural Information Processing Systems* 33 (2020): 7462-7473.

[9] Anguelov, Dragomir, Praveen Srinivasan, Daphne Koller, Sebastian Thrun, Jim Rodgers, and James Davis. "Scape: shape completion and animation of people." In *ACM SIGGRAPH 2005 Papers*, pp. 408-416. 2005.

[10] Loper, Matthew, Naureen Mahmood, Javier Romero, Gerard Pons-Moll, and Michael J. Black. "SMPL: A skinned multi-person linear model." *ACM transactions on graphics (TOG)* 34, no. 6 (2015): 1-16.

[11] Pavlakos, Georgios, Vasileios Choutas, Nima Ghorbani, Timo Bolkart, Ahmed AA Osman, Dimitrios Tzionas, and Michael J. Black. "Expressive body capture: 3d hands, face, and body from a single image." In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 10975-10985. 2019.

[12] Choutas, Vasileios, Lea Müller, Chun-Hao P. Huang, Siyu Tang, Dimitrios Tzionas, and Michael J. Black. "Accurate 3D Body Shape Regression Using Metric and Semantic Attributes." In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 2718-2728. 2022.

[13] Yang, Gengshan, Minh Vo, Natalia Neverova, Deva Ramanan, Andrea Vedaldi, and Hanbyul Joo. "Banmo: Building animatable 3d neural models from many casual videos." In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 2863-2873. 2022.

[14] Remondino, Fabio, and Andreas Roditakis. "Human figure reconstruction and modeling from single image or monocular video sequence." In *Fourth International Conference on 3-D Digital Imaging and Modeling, 2003. 3DIM 2003. Proceedings.*, pp. 116-123. IEEE, 2003.

[15] Xiu, Yuliang, Jinlong Yang, Dimitrios Tzionas, and Michael J. Black. "Icon: Implicit clothed humans obtained from normals." In *2022 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 13286-13296. IEEE, 2022.

[16] Zheng, Yufeng, Victoria Fernández Abrevaya, Marcel C. Bühler, Xu Chen, Michael J. Black, and Otmar Hilliges. "Im avatar: Implicit morphable head avatars from videos." In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 13545-13555. 2022.

[17] Joo, Hanbyul, Tomas Simon, and Yaser Sheikh. "Total capture: A 3d deformation model for tracking faces, hands, and bodies." In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 8320-8329. 2018.

[18] De Reu, Jeroen, Philippe De Smedt, Davy Herremans, Marc Van Meirvenne, Pieter Laloo, and Wim De Clercq. "On introducing an image-based 3D reconstruction method in archaeological excavation practice." *Journal of Archaeological Science* 41 (2014): 251-262.

[19] Delage, Erick, Honglak Lee, and Andrew Y. Ng. "A dynamic bayesian network model for autonomous 3d reconstruction from a single indoor image." In *2006 IEEE computer society conference on computer vision and pattern recognition (CVPR'06)*, vol. 2, pp. 2418-2428. IEEE, 2006.

[20] Fan, Haoqiang, Hao Su, and Leonidas J. Guibas. "A point set generation network for 3d object reconstruction from a single image." In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 605-613. 2017.

[21] Knyaz, Vladimir A., Vladimir V. Kniaz, and Fabio Remondino. "Image-to-voxel model translation with conditional adversarial networks." In *Proceedings of the European Conference on Computer Vision (ECCV) Workshops*, pp. 0-0. 2018.

[22] Tatarchenko, Maxim, Alexey Dosovitskiy, and Thomas Brox. "Multi-view 3d models from single images with a convolutional network." In *European Conference on Computer Vision*, pp. 322-337. Springer, Cham, 2016.

[23] Wang, Nanyang, Yinda Zhang, Zhuwen Li, Yanwei Fu, Wei Liu, and Yu-Gang Jiang. "Pixel2mesh: Generating 3d mesh models from single rgb images." In *Proceedings of the European conference on computer vision (ECCV)*, pp. 52-67. 2018.

[24] Shin, Daeyun, Charless C. Fowlkes, and Derek Hoiem. "Pixels, voxels, and views: A study of shape representations for single view 3d object shape prediction." In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 3061-3069. 2018.

[25] Firman, Michael, Oisin Mac Aodha, Simon Julier, and Gabriel J. Brostow. "Structured prediction of unobserved voxels from a single depth image." In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 5431-5440. 2016.

[26] Yang, Bo, Stefano Rosa, Andrew Markham, Niki Trigoni, and Hongkai Wen. "Dense 3D object reconstruction from a single depth view." *IEEE transactions on pattern analysis and machine intelligence* 41, no. 12 (2018): 2820-2834.

[27] Kar, Abhishek, Shubham Tulsiani, Joao Carreira, and Jitendra Malik. "Category-specific object reconstruction from a single image." In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 1966-1974. 2015.

[28] Liu, F., Shen, C. and Lin, G., 2015. Deep convolutional neural fields for depth estimation from a single image. In *Proceedings of the IEEE conference on computer vision and pattern recognition* (pp. 5162-5170).

[29] Eigen, David, Christian Puhrsch, and Rob Fergus. "Depth map prediction from a single image using a multi-scale deep network." *Advances in neural information processing systems* 27 (2014).

[30] Choy, Christopher B., Danfei Xu, JunYoung Gwak, Kevin Chen, and Silvio Savarese. "3d-r2n2: A unified approach for single and multi-view 3d object reconstruction." In *European conference on computer vision*, pp. 628-644. Springer, Cham, 2016.

[31] Wu, Jiajun, Chengkai Zhang, Tianfan Xue, Bill Freeman, and Josh Tenenbaum. "Learning a probabilistic latent space of object shapes via 3d generative-adversarial modeling." *Advances in neural information processing systems* 29 (2016).

[32] Wu, Jiajun, Yifan Wang, Tianfan Xue, Xingyuan Sun, Bill Freeman, and Josh Tenenbaum. "Marrnet: 3d shape reconstruction via 2.5 d sketches." *Advances in neural information processing systems* 30 (2017).

[33] Richter, Stephan R., and Stefan Roth. "Matryoshka networks: Predicting 3d geometry via nested shape layers." In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 1936-1944. 2018.

[34] Kato, Hiroharu, Yoshitaka Ushiku, and Tatsuya Harada. "Neural 3d mesh renderer." In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 3907-3916. 2018.

[35] Xie, Haozhe, Hongxun Yao, Shengping Zhang, Shangchen Zhou, and Wenxiu Sun. "Pix2Vox++: multi-scale context-aware 3D object reconstruction from single and multiple images." *International Journal of Computer Vision* 128, no. 12 (2020): 2919-2935.

[36] Tzionas, Dimitrios, and Juergen Gall. "3d object reconstruction from hand-object interactions." In *Proceedings of the IEEE International Conference on Computer Vision*, pp. 729-737. 2015.

[37] Zheng, Bo, Yibiao Zhao, Joey C. Yu, Katsushi Ikeuchi, and Song-Chun Zhu. "Beyond point clouds: Scene understanding by reasoning geometry and physics." In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 3127-3134. 2013.

[38] Girdhar, Rohit, David F. Fouhey, Mikel Rodriguez, and Abhinav Gupta. "Learning a predictable and generative vector representation for objects." In *European Conference on Computer Vision*, pp. 484-499. Springer, Cham, 2016.

[39] Yan, Xinchen, Jimei Yang, Ersin Yumer, Yijie Guo, and Honglak Lee. "Perspective transformer nets: Learning single-view 3d object reconstruction without 3d supervision." *Advances in neural information processing systems* 29 (2016).

[40] Huang, Qixing, Hai Wang, and Vladlen Koltun. "Single-view reconstruction via joint analysis of image and shape collections." *ACM Transactions on Graphics (TOG)* 34, no. 4 (2015): 1-10.

[41] Liu, Shichen, Tianye Li, Weikai Chen, and Hao Li. "Soft rasterizer: A differentiable renderer for image-based 3d reasoning." In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 7708-7717. 2019.

[42] Vicente, S., Carreira, J., Agapito, L. and Batista, J., 2014. Reconstructing pascal voc. In *Proceedings of the IEEE conference on computer vision and pattern recognition* (pp. 41-48).

[43] Aubry, Mathieu, et al. "Seeing 3d chairs: exemplar part-based 2d-3d alignment using a large dataset of cad models." *Proceedings of the IEEE conference on computer vision and pattern recognition*. 2014.

[44] Radford, Alec, Luke Metz, and Soumith Chintala. "Unsupervised representation learning with deep convolutional generative adversarial networks." *arXiv preprint arXiv:1511.06434* (2015).