FH Salzburg
MultiMediaTechnology

An Overview of 3D Object Reconstruction Diffusion Models

Seminar Paper

Author: Felix Beer
Advisor: DI Gerlinde Emsenhuber
Repository: https://github.com/felixbeer/3d-diffusion-models-paper

Salzburg, Austria, dd.mm.yyyy

An Overview of 3D Object Reconstruction Diffusion Models

Felix Beer
fbeer.mmt-b2022@fh-salzburg.ac.at
Salzburg University of Applied Sciences

Abstract

1 Introduction

In today's world, nearly every industry uses 3D models to visually represent objects or environments. Whether for entertainment, development, or research, 3D models are essential tools for understanding complex concepts and ideas. Another exemplary field is real estate development, where architects and engineers utilize Building Information Modeling (BIM). BIM improves collaboration and efficiency in construction projects by integrating various aspects of a building's lifecycle, from initial design through maintenance and deconstruction, into 3D models (Azhar, Khalfan, and Maqsood 2020).

With modern interactive media, the demand for high-quality and numerous 3D models has significantly increased. Triple-A games and blockbuster movies heavily rely on detailed 3D models to create true-to-life experiences.

However, creating these models is a time-consuming and costly process that requires skilled artists and designers, especially when compared to capturing images or videos with cameras or smartphones. Recent advancements in Generative 3D AI have made it possible to generate 3D models from a single image, a process known as 3D object or mesh reconstruction.

While the generated models are currently not yet at the level required for Triple-A games or the high standards of the entertainment industry, it holds significant potential for transforming and partially automating the creation and use of 3D models. At present, these technologies are more suitable for smaller prototyping projects or indie productions.

Some examples of 3D mesh reconstruction include the newly released TripoSR (Tochilkin et al. (2024)) as well as established models like Zero-1-to-3 (R. Liu et al. (2023)), One-2-3-45 (M. Liu et al. (2023)), and One-2-3-45++ (M. Liu et al. (2024)).

This paper provides an overview of the various approaches and models used for 3D mesh reconstruction, including a comparison of their performance and visual results. Furthermore, it discusses the different strategies like voxel-based (Zhirong Wu et al. 2015), point cloud-based (Charles et al. 2017), and mesh-based (N. Wang et al. 2018) methods. It also explores the underlying methods and concepts like convolutional neural networks and neural radiance fields (Mildenhall et al. 2021) used in modern neural network-based models. Finally, it offers a brief look at use cases and applications that benefit the most as well as 3D-model datasets such as Objaverse-XL (Deitke et al. 2023) and their impact on model training.

2 Concepts and Functionality

To understand the process of 3D mesh reconstruction, it is essential to explore the various techniques employed to generate 3D models from 2D images.

3D mesh reconstruction involves several methodologies, each contributing uniquely to the overall objective. Initial techniques like Shape from Shading (Horn and Brooks 1989) utilized shading information to infer the 3D structure of objects. Multi-view Stereo (MVS) employed multiple images from different viewpoints to estimate 3D structure through triangulation methods.

The introduction of Convolutional Neural Networks (CNNs) significantly improved the accuracy of 3D reconstructions by providing robust tools for image analysis and feature extraction. Generative Adversarial Networks (GANs (Goodfellow et al. 2014)) further enhanced these capabilities by providing a better training method. Combining CNNs and GANs has led to advanced models such as Pixel2Mesh++ (Wen et al. 2019) and made it possible to accurately estimate a complete three-dimensional model from a single image.

Lastly Neural Implicit Functions, like NeRF (Mildenhall et al. 2021), have further improved 3D reconstruction by using neural networks to model continuous volumetric functions which can generate high-quality 3D reconstructions with smooth surfaces.

The following sections examine each technique in detail, exploring their specific mechanisms, innovations, and contributions to the field.

2.1 Shape from Shading

Shape from Shading (SFS) is an early technique used to estimate the shape of an object from a single image, dating back to the late 80s (Horn and Brooks (1989)). This method aims to reconstruct the 3D shape of an object by analyzing shading information in a 2D image.

The idea of SFS is to use variations in shading to infer the 3D geometry of an object's surface. This method operates under several key assumptions:

1. Lambertian Reflectance: The surface of the object reflects light uniformly in all directions. This means the intensity of reflected light depends only on the angle between the light source and the surface normal.

2. Single Light Source: The object is illuminated by a single distant light source.

3. Known Light Source Direction: The direction of the light source relative to the camera is known.

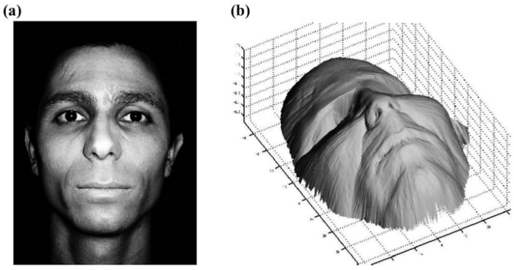

Figure 1: (a) A real face image. (b) Surface recovered from (a) by the generic SFS algorithm with the perspective model, with the light source located at the optical center. (taken from He and Chen (2018))

When all assumptions are met, the intensity of the reflected light in relation to the incoming light direction can be used to estimate the surface normals at each point of the object. By integrating these normals, the depth of the object can be estimated, providing a 3D representation of the object.

While Shape from Shading is a robust technique, it has limitations. It requires accurate knowledge of the light source direction and assumes Lambertian reflectance, which does not hold up for real-world objects and results in a significant deviation from real light behavior (Wolff 1996). This problem is not just present in the SFS method but also in a lot of other computer graphics and computer vision appliances, as a compromise for computational efficiency and simplicity. Additionally, the SFS is sensitive to noise and may produce inaccurate results in complex scenes.

2.2 Multi-view Stereo

Multi-view Stereo (MVS) uses multiple images of an object taken from different angles to estimate its 3D structure (see Figure 2). This concept, known as Structure from Motion (SfM), cannot be traced back to a single publication but is rather a collection of various works. Ullman (Ullman and Brenner 1997) was among the first to describe the process in a computational context, noting that the structure of four non-coplanar points can be recovered from three orthographic projections.

SfM involves the following steps:

1. Camera Calibration: Estimating intrinsic and extrinsic camera parameters. This involves focal length, principal point as well as distortion of each camera.

2. Feature Detection and Matching: Identifying and matching features using techniques like Difference of Gaussians (DoG) and Harris corner detection. Furukawa and Ponce (2010) for example, proposed a robust and efficient algorithm based on these well-established techniques.

3. Pose Estimation: Determining relative positions and orientations of cameras based on features.

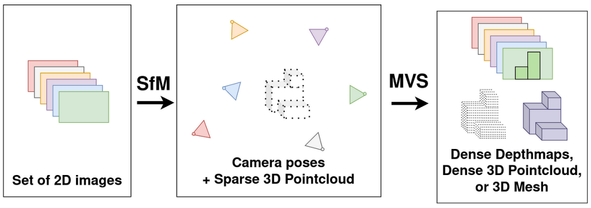

Figure 2: Multi-view Stereo Pipeline (taken from Rouch (2023))

4. Triangulation: Using poses and features to estimate 3D points in the scene.

5. Dense Reconstruction: Interpolating the sparse 3D points to create a solid 3D model.

MVS has also gained significant relevance in augmented and virtual reality for real-time environment reconstruction and mapping.

Despite its effectiveness, MVS has limitations, such as requiring multiple images from different viewpoints, which may not always be available. Moreover, MVS can struggle with textureless surfaces and repetitive patterns, which can make feature matching challenging. These limitations have driven the development and adoption of deep learning methods, which can overcome some of these challenges by learning from large datasets of images and 3D models.

2.3 Convolutional Neural Networks

Convolutional Neural Networks (CNNs) are deep learning models especially useful when working with image data. They have significantly impacted the field of computer vision and have found applications in many areas, including 3D mesh reconstruction.

CNNs usually operate on grid-like structures, processing data with convolutional layers that capture spatial hierarchies of features through a series of filters. This grid-based operation is particularly effective for image data, where the spatial arrangement of pixels is crucial for understanding the content.

Compared to traditional methods, like MVS, CNNs and the other following deep learning models can learn complex features from images and automatically extract relevant information for 3D reconstruction. They are also more robust to noise and variations in lighting conditions.

AlexNet by Krizhevsky, Sutskever, and Hinton (2012) was one of the first successful applications of CNNs to image classification, achieving substantial performance improvements over previous methods. Specifically, AlexNet reduced top-1 error rates to 37.5% and top-5 error rates to 17.0%, which were significantly better than the previous state-of-the-art. Top-1 and top-5 error rates are common metrics used to evaluate the performance of image classification models. The top-1 error rate is the percentage of images for which the correct label is not in the top-1 predicted labels, while the top-5 error rate is the percentage of images for which the correct label is not in the top-5 predicted labels.

2.4 Generative Adversarial Networks

Generative Adversarial Networks (GAN) (Goodfellow et al. 2014) are a type of deep learning model that consists of two neural networks: a generator and a discriminator. The generator is responsible for generating new data samples, while the discriminator is responsible for distinguishing between real and generated data samples. The two networks are trained together in competition, where the generator tries to generate realistic data samples to fool the discriminator, and the discriminator tries to distinguish between real and generated data samples.

In the context of 3D mesh reconstruction, CNNs and GANs are often used together to generate 3D models from 2D images. The CNN is used to extract features from the input image, and the GAN is used to generate the 3D model from these features. One example of this is the Pixel2Mesh++ model by Wen et al. (2019).

2.5 Neural Implicit Functions

Neural Implicit Functions represent an alternative approach to 3D reconstruction by using neural networks to model continuous volumetric functions. Unlike traditional methods that use explicit representations like meshes or voxels, neural implicit functions encode the geometry of 3D objects in a continuous function that can be evaluated at any point in 3D space.

Neural Implicit Functions, such as DeepSDF (Park et al. 2019) and NeRF (Mildenhall et al. 2021), use neural networks to map spatial coordinates to implicit representations of 3D shapes. These models learn a function that outputs a value indicating whether a point lies inside or outside the object (in the case of signed distance functions — DeepSDF) or the density and color at a given spatial location (in the case of radiance fields — NeRF).

Neural Implicit Functions usually work in the following steps:

1. Data Collection: Collect dataset of 3D shaped and corresponding, annotated 2D images. An example of such a dataset is ShapeNet (Chang et al. 2015).

2. Network Training: Training the network to minimize error between the predicted and actual shape representation using the loss function. This process is similar to the fooling of the discriminator in GANs (2.4).

3. Shape Inference: Once trained, the network can infer the 3D shape from new inputs by querying the implicit function at various spatial locations to reconstruct the geometry.

The main advantages of neural implicit functions include their continuous nature, which allows for high-quality reconstructions with smooth surfaces. Another advantage is the fact that they typically require less memory and computation compared to explicit representations like meshes or voxels created by deep convolutional networks like the aforementioned CNNs (2.3). NeRFs — Neural Radiance Fields (Mildenhall et al. 2021) have been particularly successful in generating high-quality 3D reconstructions from 2D images (see Figure 3) and are used in almost all modern neural network-based 3D reconstruction models.

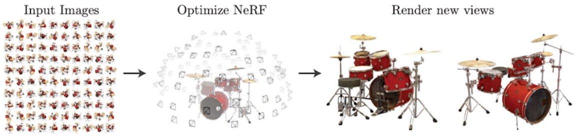

Figure 3: NeRF - 100 input views to NeRF representation (Mildenhall et al. 2021)

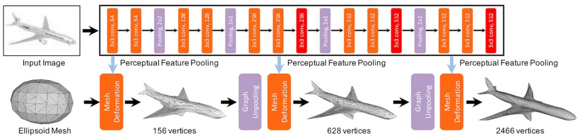

Figure 4: Pixel2Mesh Pipeline (Horn and Brooks 1989)

3 Models

In the recent years several models have been developed to generate 3D models from 2D images. Some of the most prominent models include:

3.1 Pixel2Mesh

Pixel2Mesh is a model developed by N. Wang et al. (2018). It is built with two main components. The image feature network which is a convolutional neural network (CNN) that extracts perceptual features from the input image. The second component is a cascaded mesh deformation network which is a graph-based convolution network.

A graph-based convolution network differs from traditional CNNs in that it operates on a graph rather than a grid. In the context of Pixel2Mesh, the graph represents the 3D mesh model with vertices and edges.

The Pixel2Mesh model works in the following steps:

1. The input image is passed through the image feature network to extract features.

2. The cascaded mesh deformation network initializes with an ellipsoid mesh model.

3. The features extracted from the image are then taken to refine the shape of the mesh model.

4. The mesh model gets refined iteratively in 3 blocks, with each iteration refining the shape and increasing the mesh resolution. (see Figure 4)

5. The vertex positions get estimated each step, which are then used to look up the features from the image feature network for the next iteration.

3.2 Pixel2Mesh++

Pixel2Mesh++ (Wen et al. 2019) is an adaption of the original Pixel2Mesh model. Instead of using a single image, it relies on multiple images to generate a more accurate 3D model. It mixes the already well-known MVS systems with the Pixel2Mesh model.

Pixel2Mesh++ operates by sampling deformation around each vertex of the base mesh and selecting the optimal deformation using perceptual features pooled from multiple input images. This multi-view approach enables the model to capture more detailed and accurate geometries, especially in occluded or less visible areas, where single-view methods cannot capture the full extent of an object and need to estimate the shape by hallucinating the missing parts — hence this model is included in this overview.

3.3 Zero-1-to-3

The Zero-1-to-3 (Zero123) model, developed by R. Liu et al. (2023), is designed to generate 3D models from a single image in a zero-shot manner, meaning it can produce accurate 3D reconstructions without needing extensive per-object training. Compared to Pixel2Mesh, this model created not just the mesh, but also a colored texture based on the given image. This model highlights its robustness and versatility across various types of input images.

The process involves fine-tuning a pre-trained diffusion model, in this case Stable Diffusion, to learn how to control camera extrinsics (the external parameters defining the camera's position and orientation). By using a 3D model dataset, the model learns to manipulate the camera viewpoint. This allows it to create views from certain angles — also called novel view synthesis.

By merging the input image with the synthesized views, the model can effectively learn the 3D structure of the object and generate a high-quality 3D mesh.

The retrained version, Zero123-XL, further improves the performance by including additional training data. Zero-1-to-3-XL makes use of the extensive dataset Objaverse-XL (Deitke et al. 2023), which contains over 10 million 3D objects. This larger dataset allows the model to enhance its zero-shot capabilities and overall reconstruction accuracy.

3.4 One-2-3-45

One-2-3-45, developed by M. Liu et al. (2023), is a state-of-the-art model designed to generate 3D meshes from single 2D images with exceptional efficiency.

It can reconstruct a full 360-degree 3D textured mesh in just 45 seconds, significantly faster than other models, which often require lengthy optimization processes. This rapid processing is achieved without the need for per-shape optimization, making it far more suitable for real-time consumer applications. One-2-3-45 uses the aforementioned Zero123 (3.3) model to generate multi-view images from the input image. These images are then compared to the input image for pose estimation. The results are poses for each of the generated views. These results are lastly fed into a signed distance function based neural surface reconstruction model (similar to DeepSDF mentioned in 2.5) which generated the finished 3D mesh.

An improved version, One-2-3-45++ (M. Liu et al. 2024), was later released, which further optimized the model's performance. The biggest improvement was the switch from Zero123 to a custom 2D diffusion model. Zero123 tried to predict each view individually, which resulted in inconsistencies. The new model uses a fine-tuned 2D diffusion model, which can attend all views simultaneously, resulting in more accurate results.

3.5 TripoSR

TripoSR, developed by Tochilkin et al. (2024), is an another model for fast 3D object reconstruction from a single image. It was created as a group effort by Stability AI (the creators of Stable Diffusion) as well as Tripo AI. Building upon the Large Reconstruction Model (LRM) architecture by Hong et al. (2024), TripoSR uses transformer technology for feed-forward 3D generation, producing high-quality 3D meshes in under 0.5 seconds.

The LRM architecture consists of three main parts: an image encoder, a triplane representation, and a NeRF decoder. The image encoder, based on a vision transformer, takes an input image and converts it into a set of detailed features called latent vectors. These latent vectors capture important information about the object's shape and appearance in a compressed form.

Next, the triplane representation takes these latent vectors and arranges them in a three-dimensional grid. This grid captures the object's complex shapes and textures. The triplane representation uses layers that help the model focus on different parts of the image and learn how they relate to each other.

Finally, the NeRF decoder takes the 3D grid and predicts the color and density of each point in space, creating a smooth and detailed 3D model of the object.

TripoSR introduces several key improvements to this architecture. For example, it uses only a hand-selected higher quality subset of the Objaverse dataset, which improves generalization and reconstruction quality. It also introduces a mask loss function that penalizes differences between predicted and ground-truth mask images, reducing artifacts and improving reconstruction fidelity.

3.6 Comparison

3.6.1 Model Training

Most models are trained on a 3D model dataset. A recent dataset, Objaverse-XL (Deitke et al. 2023) contains over 10 million 3D objects and is used by models like Zero123-XL to improve their zero-shot capabilities. The term zero-shot in machine learning refers to the ability of a model to handle elements it has never seen before. Just like with training Large Language Models, the 3D models in these datasets are crawled from various sources from across the internet. This leads to ethical concerns regarding copyright and licensing. In the case of Objaverse-XL, most models are sourced from open-source GitHub repositories but also some more specialized sites like Sketchfab, Thingiverse or Polycam.

3.6.2 Model Performance

Based on the order of the models listed above, each models reconstruction accuracy increases compared to the previous one. Hence TripoSR, the newest model, is the most accurate and efficient model.

Compared to other models like One-2-3-45 and Zero-1-to-3, TripoSR excels in both speed and performance, as evidenced by lower Chamfer Distance (CD) and higher F-score (FS) metrics on benchmark datasets.

4 Applications of 3D Mesh Reconstruction

As the field is still relatively new, no mainstream applications have been established yet. However, the potential is great and some possible applications have already been identified.

4.1 Development and Entertainment

The most prominent application of 3D mesh reconstruction could be in the development and entertainment industry. The ability to generate 3D models from 2D images could revolutionize the asset creation process. This could be especially beneficial for indie developers or small studios that do not have the resources to create high-quality 3D models from scratch. The generated models could be used in video games, movies, animations, and other forms of media. This could significantly reduce the time and cost associated with creating 3D assets, allowing developers and animators to focus on other aspects of their projects. (see Figure 6)

4.2 Medical

Z.-Y. Wang et al. (2019) have shown that 3D mesh reconstruction can be used in the medical field to generate 3D models of organs from medical images. These models can be used for diagnosis, treatment planning, and medical education. For example, 3D models of the heart can be used to plan surgeries and visualize complex anatomical structures.

Diagnosing based on the 3D model also comes with risks and can potentially be problematic. The model is only an estimation based on training data ant not a real representation of the patient's anatomy.

4.3 Other Applications

4.3.1 Cultural Heritage

5 Discussion and Future Direction

Discussion of results and their implications. What are the limitations current works? What are the next steps in this research area?

6 Conclusion

References

Azhar, Salman, Malik Khalfan, and Tayyab Maqsood. 2020. "Building Information Modeling (BIM): Now and beyond." Publisher: UTS ePress, *The Australasian Journal of Construction Economics and Building* 12, no. 4 (August 20, 2020): 15–28. https://doi.org/10.5130/ajceb.v12i4.3032.

Chang, Angel X., Thomas Funkhouser, Leonidas Guibas, Pat Hanrahan, Qixing Huang, Zimo Li, Silvio Savarese, et al. 2015. *ShapeNet: An Information-Rich 3D Model Repository,* December 9, 2015. https://doi.org/10.48550/arXiv.1512.03012.

Charles, R. Qi, Hao Su, Mo Kaichun, and Leonidas J. Guibas. 2017. "PointNet: Deep Learning on Point Sets for 3D Classification and Segmentation." In *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR),* 77–85. 2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR). Honolulu, HI: IEEE, July. ISBN: 978-1-5386-0457-1. https://doi.org/10.1109/CVPR.2017.16.

Deitke, Matt, Ruoshi Liu, Matthew Wallingford, Huong Ngo, Oscar Michel, Aditya Kusupati, Alan Fan, et al. 2023. "Objaverse-XL: A Universe of 10M+ 3D Objects." In *Advances in Neural Information Processing Systems 37.* https://doi.org/10.48550/arXiv.2307.05663.

Furukawa, Yasutaka, and Jean Ponce. 2010. "Accurate, Dense, and Robust Multiview Stereopsis." *IEEE Transactions on Pattern Analysis and Machine Intelligence* 32, no. 8 (August): 1362–1376. ISSN: 0162-8828. https://doi.org/10.1109/TPAMI.2009.161.

Goodfellow, Ian, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, and Yoshua Bengio. 2014. "Generative Adversarial Nets." In *Advances in Neural Information Processing Systems,* vol. 27. Curran Associates, Inc. https://doi.org/10.48550/arXiv.1406.2661.

He, Yu, and Shengyong Chen. 2018. "Advances in sensing and processing methods for three-dimensional robot vision." *International Journal of Advanced Robotic Systems* 15 (March 20, 2018): 172988141876062. https://doi.org/10.1177/1729881418760623.

Hong, Yicong, Kai Zhang, Jiuxiang Gu, Sai Bi, Yang Zhou, Difan Liu, Feng Liu, Kalyan Sunkavalli, Trung Bui, and Hao Tan. 2024. *LRM: Large Reconstruction Model for Single Image to 3D,* March 9, 2024. https://doi.org/10.48550/arXiv.2311.04400.

Horn, Berthold, and Michael Brooks. 1989. *Shape from Shading.* Vol. 2. MIT Press, January 1, 1989.

Krizhevsky, Alex, Ilya Sutskever, and Geoffrey E Hinton. 2012. "ImageNet Classification with Deep Convolutional Neural Networks." In *Advances in Neural Information Processing Systems,* vol. 25. Curran Associates, Inc. https://doi.org/10.1145/3065386.

Liu, Minghua, Ruoxi Shi, Linghao Chen, Zhuoyang Zhang, Chao Xu, Xinyue Wei, Hansheng Chen, Chong Zeng, Jiayuan Gu, and Hao Su. 2024. "One-2-3-45++: Fast Single Image to 3D Objects with Consistent Multi-View Generation and 3D Diffusion," 10072–10083. Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. https://doi.org/10.48550/arXiv.2311.07885.

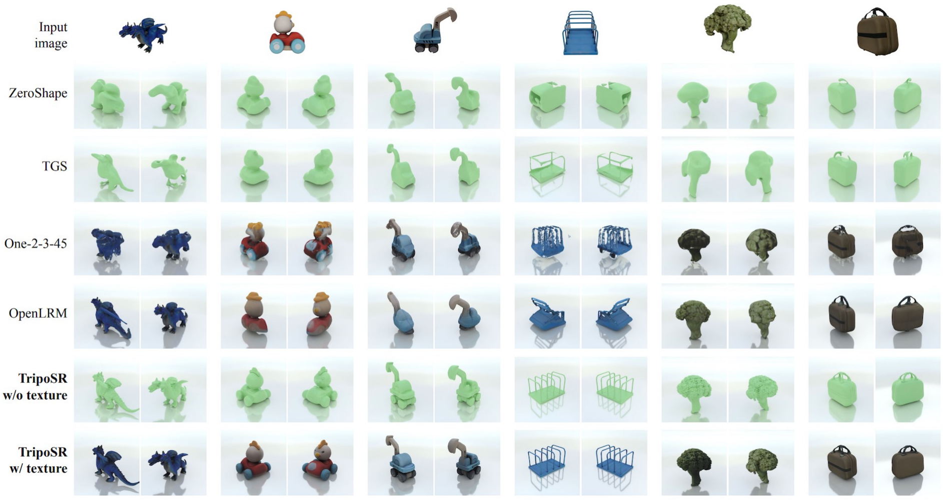

Figure 5: Comparison of TripoSR to other state-of-the-art methods — including One-2-3-45 (Tochilkin et al. 2024)

Figure 6: Models generated by One-2-3-45++ (M. Liu et al. 2024)

Liu, Minghua, Chao Xu, Haian Jin, Linghao Chen, Mukund Varma T, Zexiang Xu, and Hao Su. 2023. "One-2-3-45: Any Single Image to 3D Mesh in 45 Seconds without Per-Shape Optimization." In *Advances in Neural Information Processing Systems 36.* June 29, 2023. https://doi.org/10.48550/arXiv.2306.16928.

Liu, Ruoshi, Rundi Wu, Basile Van Hoorick, Pavel Tokmakov, Sergey Zakharov, and Carl Vondrick. 2023. "Zero-1-to-3: Zero-shot One Image to 3D Object," 9298–9309. Proceedings of the IEEE/CVF International Conference on Computer Vision. https://doi.org/10.1109/ICCV51070.2023.00853.

Mildenhall, Ben, Pratul P. Srinivasan, Matthew Tancik, Jonathan T. Barron, Ravi Ramamoorthi, and Ren Ng. 2021. "NeRF: representing scenes as neural radiance fields for view synthesis." *Communications of the ACM*

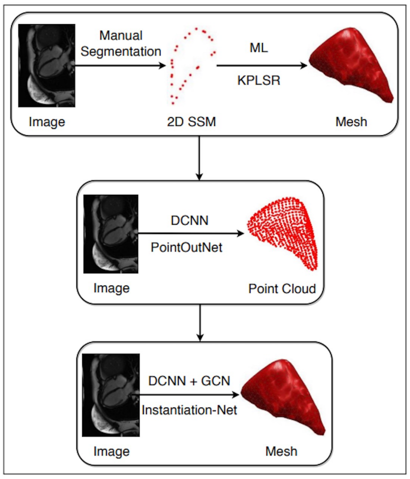

Figure 7: Medical Image Pipeline of Instantiation-Net (Z.-Y. Wang et al. 2019).

65, no. 1 (December 17, 2021): 99–106. ISSN: 0001-0782. https://doi.org/10.1145/3503250.

Park, Jeong Joon, Peter Florence, Julian Straub, Richard Newcombe, and Steven Lovegrove. 2019. "DeepSDF: Learning Continuous Signed Distance Functions for Shape Representation." In *2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR),* 165–174. 2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR). Long Beach, CA, USA: IEEE, June. ISBN: 978-1-72813-293-8. https://doi.org/10.1109/CVPR.2019.00025.

Rouch, Thomas. 2023. "PatchMatch Multi-View Stereo." Medium, April 10, 2023. Accessed June 30, 2024. https://betterprogramming.pub/patchmatch-multi-view-stereo-1-2-fc46e5dfe912.

Tochilkin, Dmitry, David Pankratz, Zexiang Liu, Zixuan Huang, Adam Letts, Yangguang Li, Ding Liang, Christian Laforte, Varun Jampani, and Yan-Pei Cao. 2024. *TripoSR: Fast 3D Object Reconstruction from a Single Image,* March 4, 2024. https://doi.org/10.48550/arXiv.2403.02151.

Ullman, S., and Sydney Brenner. 1997. "The interpretation of structure from motion." Publisher: Royal Society, *Proceedings of the Royal Society of London. Series B. Biological Sciences* 203, no. 1153 (January): 405–426. https://doi.org/10.1098/rspb.1979.0006.

Wang, Nanyang, Yinda Zhang, Zhuwen Li, Yanwei Fu, Wei Liu, and Yu-Gang Jiang. 2018. "Pixel2Mesh: Generating 3D Mesh Models from Single RGB Images," 52–67. Proceedings of the European Conference on Computer Vision (ECCV). https://doi.org/10.1007/978-3-030-01252-6_4.

Wang, Zhao-Yang, Xiao-Yun Zhou, Peichao Li, Celia Riga, and Guang-Zhong Yang. 2019. *Instantiation-Net: 3D Mesh Reconstruction from Single 2D Image for Right Ventricle,* September 16, 2019. https://doi.org/10.1007/978-3-030-59719-1_66.

Wen, Chao, Yinda Zhang, Zhuwen Li, and Yanwei Fu. 2019. "Pixel2Mesh++: Multi-View 3D Mesh Generation via Deformation." In *2019 IEEE/CVF International Conference on Computer Vision (ICCV),* 1042–1051. 2019 IEEE/CVF International Conference on Computer Vision (ICCV). Seoul, Korea (South): IEEE, October. ISBN: 978-1-72814-803-8. https://doi.org/10.1109/ICCV.2019.00113.

Wolff, Lawrence B. 1996. "Generalizing Lambert's Law for smooth surfaces." In *Computer Vision — ECCV '96,* edited by Bernard Buxton and Roberto Cipolla, 40–53. Berlin, Heidelberg: Springer. ISBN: 978-3-540-49950-3. https://doi.org/10.1007/3-540-61123-1_126.

Zhirong Wu, Shuran Song, Aditya Khosla, Fisher Yu, Linguang Zhang, Xiaoou Tang, and Jianxiong Xiao. 2015. "3D ShapeNets: A deep representation for volumetric shapes." In *2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR),* 1912–1920. 2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR). Boston, MA, USA: IEEE, June. ISBN: 978-1-4673-6964-0. https://doi.org/10.1109/CVPR.2015.7298801.

Appendices

A AI Methodology

You can delete the explanatory text in this section and add your AI documentation.

Document in this section, how AI was utilized to support the creation of this seminar paper or thesis. Which tools were used? Which steps of the thesis creation were supported (e.g., brain storming, literature search, quality analysis of papers, text generation, text summaries, paraphrasing, spellchecking, ...). If generative AI was utilized (e.g., ChatGPT, CoPilot), also provide key text prompts that influenced the thesis.

In the following, list each tool and provide a brief description how the tool as utilized. For each tool, estimate in % of the word count of the thesis, how much text was generated/modified/etc. by the AI tool. For the practical part of the thesis, estimate the % of the results (e.g., code) was generated/mofidied/etc. by the AI tool. If a generative AI was used, provide exemplary prompts.

There is no requirement to highlight text in the thesis.

This section should have a maximum length of 2 pages.

A.1 DeepL

I used the tool (https://addressoftool.com) to [paraphrase—translate—...] text for this work. I estimate that XX % of the text was paraphrased with this tool.

A.2 Copilot

I used Copilot (https://addressoftool.com) to learn how to utilize RUST as a new programming language. .... I estimate taht XXX % of the source code was created with this tool.

A.3 AI Tool 2

... document tool usage ...

A.4 Prompts

I utilized the generative AI https://bot.com to create drafts for texts, support brainstorming, ...

... list exemplary prompts here ...

This work has the following word count (counted by texcount):

```
File: body.tex
Encoding: utf8
Sum count: 3232
Words in text: 3102
Words in headers: 47
Words outside text (captions, etc.): 83
Number of headers: 27
Number of floats/tables/figures: 7
Number of math inlines: 0
Number of math displayed: 0
Subcounts:
  text+headers+captions (#headers/#floats/#inlines/#displayed)
  354+1+0 (1/0/0/0) Section: Introduction
  193+3+0 (1/0/0/0) Section: Concepts and Functionality
  278+3+33 (1/1/0/0) Subsection: Shape from Shading
  255+2+8 (1/1/0/0) Subsection: Multi-view Stereo
  224+3+0 (2/0/0/0) Subsection: Convolutional Neural Networks} \label{sec
  137+3+0 (2/0/0/0) Subsection: Generative Adversarial Networks} \label{s
  296+3+10 (2/1/0/0) Subsection: Neural Implicit Functions} \label{sectio
  23+1+0 (1/0/0/0) Section: Models
  181+1+5 (2/1/0/0) Subsection: Pixel2Mesh
  116+1+0 (1/0/0/0) Subsection: Pixel2Mesh++
  198+1+0 (1/0/0/0) Subsection: Zero-1-to-3} \label{section:zero123
  189+1+0 (1/0/0/0) Subsection: One-2-3-45
  252+1+0 (1/0/0/0) Subsection: TripoSR
  171+5+12 (3/1/0/0) Subsection: Comparison
  27+5+0 (1/0/0/0) Section: Applications of 3D Mesh Reconstruction
  98+3+7 (1/1/0/0) Subsection: Development and Entertainment
  89+1+8 (1/1/0/0) Subsection: Medical
  0+4+0 (2/0/0/0) Subsection: Other Applications
  21+4+0 (1/0/0/0) Section: Discussion and Future Direction
  0+1+0 (1/0/0/0) Section: Conclusion
```