School of Informatics

Informatics Research Review
Alternatives to Voxel Decoding for Single-Image 3D-Object Reconstruction with Deep-Learning

[redacted]

January 2021

Abstract

First models of deep learning used for single-image 3D reconstruction task decoded objects as voxels (volumetric elements) in a 3D grid. This shape representation was used due to easy incorporation into neural network structure; many further studies used this method. However, voxel 3D object representation and decoding process is inherently inefficient, expensive to compute and requires additional post-processing. This review explores alternative decoding strategies and object representations, such as mesh, point clouds, implicit or parametric, and discusses their inherent qualities, performance and quality of the results.

Date: Friday 22$^{\text{nd}}$ January, 2021

Supervisor: [redacted]

1 Introduction

1.1 Introduction to the review

3D object reconstruction from single-image is an increasingly researched problem with a goal of automating 3D object modelling. Research within this area increased substantially with the rise in the use of deep learning models. Voxelized object representation, which describes an object as a collection of blocks in a 3D grid, was widely applied for first deep learning solutions to decode an object due to ease of integration into a neural network. However, the voxel decoding strategy produces inaccurate 3D models, is computationally inefficient and requires additional post-processing. This review aims to discuss alternative object decoding methods for deep learning single-image reconstruction and compare their capabilities. This review will discuss (1) why voxel object output methods to reconstruct 3D objects from a single image were commonly used and are unsatisfactory, (2) what are the alternatives to the voxel decoding method, (3) how the reconstruction quality of these alternatives compares quantitatively and qualitatively and how easy are they to integrate into deep learning model, (4) what are the current trends in this research and what solutions are the most promising regarding future research. This review does not aim to discuss deep learning models generally for mentioned reconstruction problem or their encoding methods, as well as problems relating to the data used to train such models. The reader is expected to have an understanding of deep learning and convolutional neural networks. Knowledge in computer vision, computational geometry and 3D object reconstruction problems is useful but not necessary.

1.2 Single-image object reconstruction

3D object reconstruction is an abstract and important task of computer vision. This task can be separated into single-image and multi-image 3D object reconstruction. Solutions for single-image reconstruction problem can be applied to solve more specific problems, such as automating modelling of 3D meshes for computer graphics, reconstruction of more constant shapes, such as 3D human body shape [1, 2], or elsewhere. Single-image reconstruction requires a much smaller amount of data to reconstruct a 3D object and can be applied to a substantial amount of existing data, like real-life photographs or target objects' sketches.

Prior to an increased popularity in deep learning, the research in single-image 3D object reconstruction was scarce. Initial methods were mostly limited to reconstructing objects of only specific categories with a stable visual structure, such as faces [3, 4], or objects containing some specific features, such as repeating texture patterns [5]. Reconstruction algorithms for more general shapes were capable of reconstructing only very abstract visual forms, such as line drawings [6], or were based on similar prior 2D image and 3D shape matches [7].

Deep learning solutions for this problem enabled the reconstruction of a much broader spectrum of objects with more complex shapes and visual structures and provided better output quality. It also introduced new challenges, such as reconstructing detailed objects, increasing memory usage and computational complexity, learning to reconstruct rather than recognize objects [8], inferring invisible or obstructed parts of objects, and how to efficiently output reconstructed objects. Initial deep learning methods used convolutional neural networks for image processing, and voxel (volumetric element) grid as a representation of a decoded object [9, 10] due to easiness of incorporating it into a neural network structure. Despite many disadvantages (section 1.3), voxel decoding solution evolved into a trend, and further research used this strategy, while even further research focused on offering alternative shape representation methods (section 2).

1.3 Related work and contribution

Previous reviews of this subject encompassed a more general reconstruction topic, such as mainly non-deep-learning-based 3D shape reconstruction from single or multi-image input or videos [11], or other review focusing on mainly deep learning approaches for also single and multi-image reconstruction [12]. Fu et al. [13] reviewed single-image deep learning solutions generally, by discussing their encoder and decoder types, as well as training and testing details with a comparison of the results.

In this review, the decoding structures and shape representations used in single-image 3D reconstruction deep learning models will be explored more deeply, by progressively building a framework of essential ideas and innovations. The fundamental shortcomings, advantages, implementation challenges and final results of the decoding methods will be discussed. The goal of this review is to explain fundamental inferiority of voxel decoding strategy and guide the reader towards potentially better alternatives for further research within single-image 3D reconstruction or incorporation of such models in some domain.

2 Voxel decoding method and its shortcomings

3D object representation using a grid of voxels was used in initial single-image reconstruction deep learning methods [9, 10]. Voxels (volumetric elements) allow representing 3D shape in a 3D grid of blocks, similar to how an image is expressed in a 2D grid of pixels. Voxels have a binary state- they can either be filled and express a part of an object or be void. Such a representation layer can be easily attached to convolutional neural networks to output a 3D structure because of being a fixed-size grid. However, this structure has some significant disadvantages. Firstly, such structure requires post-processing to convert it to a much more usable type, such as 3D mesh. Due to a grid structure of voxels, they need to interpolation to be converted to mesh, which induces loss of information, lower quality and additional computation. To compensate for reduced quality, this leads to a second problem- high memory and computational costs. A larger amount of grid cells is sought to obtain a more detailed object, which in practice can even reach amounts of 512^3 [14]. Even for smaller sizes, this presents a very high output space due to cubed structure space which requires a lot of resources, especially GPU or CPU memory. Also, a large neural network must be trained on equally high-detail and computationally expensive data samples to be capable of decoding small features of objects.

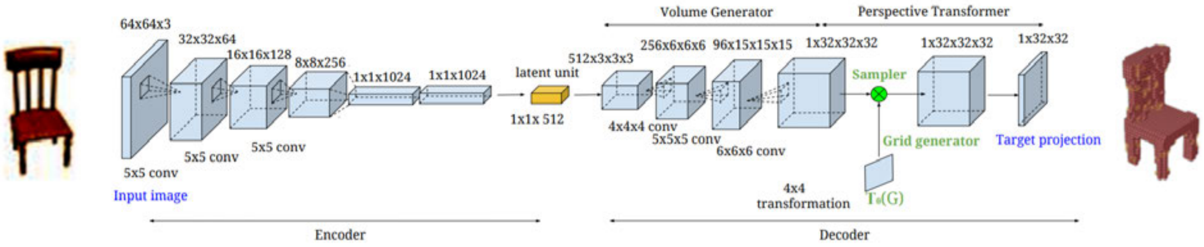

Figure 1: A Neural Network structure to decode voxel shape used by Yan et al. [10] with an example of input shape with it's reconstruction in voxels. (Original images from [10])

After the introduction of voxel grid output method to deep learning models [15] and their early incorporation into single-image reconstruction models [9, 10, 16], much further research has been devoted for improvement of models using this shape representation [17, 18, 19, 20, 21, 22]. Also, plenty of research tried to improve voxel representation method to mitigate described challenges, mostly by using octree [14] or other hierarchical structures [23, 24] to lower computational demands and increase recovered detail, or using some intermediate representations, such as 2.5D sketches [25] or other [26, 27, 28] to also increase quality. While some of these methods provide competitive results or creative solutions and the amount of research conducted for this decoding method is much higher compared to others, the previously mentioned fundamental problems of decoding voxel shapes are not mitigated.

3 Alternative decoding methods of 3D objects

3.1 Polygon meshes

Polygon meshes are 3D object representations described using a collection of faces, edges and vertices. This object representation is very commonly used for various applications. Therefore it is appealing to decode an object in this format directly. It is also easy to further expand this representation by recovering textures or other features in the same model. However, it is challenging to create an efficient mesh decoding method applied to neural network architectures. This challenge arises due to (1) mesh being a more complex structure with its structural elements (vertices, faces, edges) being connected, opposite to independence among voxel cells in a voxel representation, (2) varying amounts of detail in a single model requiring differing amounts of density of vertices and faces in different places of an object, (3) it is challenging to reconstruct more complex topological features (e.g. holes) and (4) reconstructed mesh faces are prone to colliding with each other.

Kato et al. [29] criticizes voxel and point could decoding methods and offer an early alternative by directly decoding mesh shape using a convolutional neural network. Due to complex structure of meshes, which are not easily compatible with neural networks, authors taught a neural network to decode a mesh object by offsetting vertices of a pre-set mesh shape with a fixed amount of vertices, rather than to decode the mesh from scratch. For their experiments, authors use a pre-set isotropic sphere mesh with 642 vertices, which is a much smaller amount of structural elements compared with even very low quality of voxel spaces used in practice to decode object (e.g. 32^3 [9]). Authors also offer a creative way to calculate neural network loss for reconstructed mesh structures. Due to difficulty of calculating loss by comparing generated mesh structure with the ground truth, Kato et al. render generated shape to the 2D silhouette and compare its overlap with an object silhouette in an input 2D image. To allow back-propagation, they present a differentiable renderer, which allows gradient flow from silhouette overlap comparison to the mesh output stage of the neural network. This solution also allows recovering the texture of an object by rendering a mesh shape with a texture and comparing it with an input image itself. Another early solution by Pontes et al. [30] offers an entirely different method. The proposed model encodes an input image into an embedding vector which is later used to find the most related 3D shape from a large set of already modelled 3D objects. This pre-defined model is later additionally transformed. The method produces accurate results but is limited in terms of classes or abstract shapes of objects and requires saving many object mesh models instead of learning the reconstruction task.

Chen et al. [31] further expanded Kato et al.'s solution [29] to incorporate lightning into the differentiable renderer and calculated loss between an original image and generated rendered object with texture. Liu et al. [32] offers a renderer which rasterizes objects in a probabilistic manner and allows better gradient flow. Kanazawa et al. [33] provides a solution which can learn 3D reconstruction and texture without having a 3D mesh annotation. This solution also initializes the shape of a template mesh as a mean of meshes of an input object's class, further improving the results when offsetting vertices of this mesh to generate an output. However, this weakly supervised method requires other types of annotations- object masks and key-points, which are not easy to attain. Wang et al. [34] obtains state-of-the-art by iteratively up-sampling a generated 3D mesh, which is initially regressed from an ellipsoid. Authors also obtain benefit by using graph-based convolutional neural networks [35] to decode mesh structure and an improved loss calculation strategy. Pan et al. [36] follows a similar path and also claims to have achieved state-of-the-art. Authors also use an iterative refinement of the structure with more advanced neural network architecture. Most importantly, authors modify their model to design shapes with complex topological features, such as creating holes in their shapes. During training, it is estimated how much each vertex is distanced from ground truth, and dense patches of such vertices are progressively erased from the mesh. Finally, authors introduce a loss calculation using Chamfer distance from a sampled point set in a generated and ground truth 3D shapes, opposing to the rendered 2D shape comparisons as in previous research. This combination of improvements provides impressive quantitative and especially qualitative results.

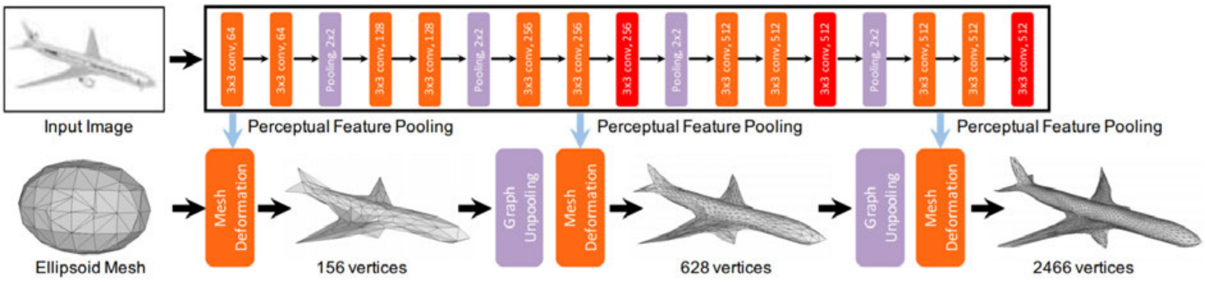

Figure 2: An image from Wang et al. [34] displays their proposed iterative decoding structure. A 3D object is reconstructed by offsetting vertices from a pre-set ellipsoid shape which is iteratively up-sampled.

However, the research direction introduced by Pontes et al. [30] where a pre-defined shape of an input image object's class is deformed by a neural network is less popular. Some research [37, 38] was conducted in this direction, but the solutions were less innovative, and the fundamental issues of this method still remain. Another direction of the approach taken by Gkioxari et al. [39] offers a solution which is capable of extracting target objects from a scene with a modified Mask R-CNN [40] object detectors. A detected object in an input image is converted into an intermediate voxel representation to avoid mesh topological limitations by allowing to form holes and other complex features. The voxel shape is then converted to a mesh and refined using graph convolutions. Also, the usage of an object detector allows the algorithm to reconstruct multiple objects from the scene and adjust their rotation according to their original position in a 2D input. Despite innovations and topological capabilities, the solution is limited in terms of the quality and smoothness of a decoded mesh shape.

3.2 Point clouds

3D point clouds are simply collections of points in a 3D space. Due to this simplicity, it is tempting to reconstruct objects in this format. This object representation allows a much simpler object decoding, as points in point clouds, as do voxels, do not have connectivity with each other, opposite to mesh vertices. Decoding objects as points also allows potentially more complex topological features due to the absence of faces and edges. Most importantly, by retaining some important advantages of voxel grids, point clouds are represented in a continuous space, opposite to a discrete voxel space, requiring a much smaller amount of output parameters for a neural network as all of these parameters are being used only to represent surface instead of volume. However, point clouds are not as commonly used format as other formats, such as meshes, and they are not easy to display visually due to the absence of surface in their structural representation. The conversion process from point cloud to mesh or other type is non-intuitive, leading to loss of important information, such as holes or small detail, or inaccurate reformatting, a problem also related with voxel objects. Usually, in the discussed solutions, the point cloud is converted to the mesh using a non-deep learning algorithm, such as Marching Cubes [41].

Fan et al. [42] offers the first and an important study of an application of point cloud decoding to single-image reconstruction model. To represent a 3D shape in their experiments, authors use a neural network to generate a point cloud output of 1024 unordered points with three variables, each representing coordinates in (x, y, z) space. During training, the generated points are compared with the set of points from ground truth 3D mesh using a combination of Chamfer distance and Earth Mover's distance losses. Authors claim to have surpassed the state-of-the-art of previous best single-image reconstruction model and argue that point cloud shape representation is easier to learn due to the absence of combinatorial connectivity patterns, opposite to, e.g. mesh structure. Besides the novel approach in shape representation, authors improve their neural network structure to work in a generative manner. During training and inference, the neural network is given a random string of fixed length, additionally to its image input. This allows the neural network to generate various interpretations of an invisible side of an object by forming a distribution of plausible reconstructions. Despite these innovations and great quantitative results during the publication of this study, the resulting qualitative results are poor when the point cloud is transformed into a mesh, generating rough and inaccurate surfaces.

Another solution by Lin et al. [43] criticizes volumetric grid decoding as having inherent drawbacks and offers their alternative for decoding object as a set of points. Different from Fan et al. [42], Lin et al. generate the final 3D structure by initially generating this structure from multiple viewpoints and all generated point sets are fused into one. Authors also take a different approach to loss calculation. Instead of comparing their shape to 3D ground truth object, they use a pseudo-renderer to render generated and ground truth objects into depth images which are compared pixel-wise, an approach which is similar to renderers for mesh comparison for loss calculation [31, 29, 32]. Similarly to Fan et al. [42], authors claim to have surpassed the state-of-the-art solution, however, this solution also does not gradually improve results from a qualitative standpoint, while using significantly more points to reconstruct a shape.

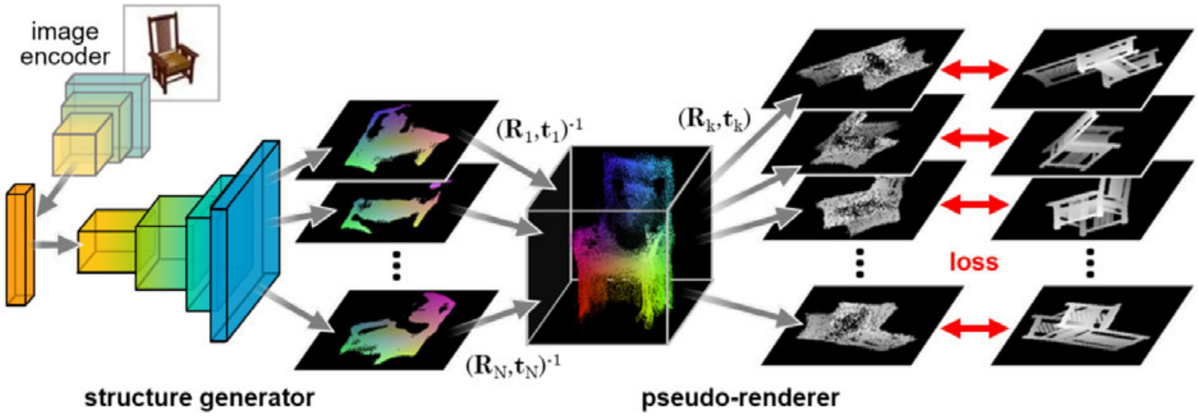
image encoder
$(\mathbf{R}_1,\mathbf{t}_1)^{-1}$
$(\mathbf{R}_k,\mathbf{t}_k)$
loss
$(\mathbf{R}_N,\mathbf{t}_N)^{-1}$
structure generator
pseudo-renderer

Figure 3: An image from Lin et al. [43] shows the model structure. The main point cloud is inferred from multiple point outputs. Multiple depth map projections are compared with ground truth to calculate loss.

This idea is further improved by Insafutdinov and Dosovitskiy [44] to allow for unsupervised learning. The differential renderer, similarly to Kanazawa et al. [33] with mesh reconstruction, is improved to remove the need for 3D annotation during loss calculation, as well as to allow texture reconstruction by comparing input image with a rendered image, rather than comparing their silhouettes. In addition to point cloud output, authors also estimate the pose of an object by learning camera position, which is used to generate an ensemble of multiple rasterizations from slightly varying directions. Rasterizations from differing angles are then compared with the input image, and the loss of the best-matching rasterization is back-propagated through the network. These improvements not only allow to infer texture of a shape, significantly increase training data but to also learn to generate more accurate shapes qualitatively, opposite to previous research. Finally, improvements by Mandikal et al. [45] display promising capabilities of this decoding method. Researchers gain further improvement by implementing embedding vector loss of trained network and a prior trained auto-encoder. Despite the absence of innovations in decoding side, authors are capable of reconstructing high-quality point clouds from only 2048 points, which is thousands of times lower compared to the output parameter spaces required to compute high-quality voxel objects.

3.3 Shapes as equations

3.3.1 Implicit shapes

Implicit shapes are described using equations, rather than a set of points, vertices or other elements, which is an emerging area of study for shape reconstruction problem. An object described by a continuous function with a formal format of $F(x, y, z) = 0$ can be much more accurate than in some more discrete form, such as polygon mesh. Such implicit shape is unlimited by the resolution and can potentially take up less memory than voxel shapes, making it appealing for reconstruction. Since the neural networks themselves can very accurately approximate various complex functions [46], the decoder itself can be used to learn to represent implicit shapes, and its outputs used in some form to discretize the shape into an interpretable format. Such decoding strategy can have some very convenient advantages: (1) as the implicit shape is a continuous function, its conversion to a more easily interpretable format, such as mesh, can have varying, adjustable and infinite amount of detail and (2) decoders can learn the shape forms themselves, rather than some manipulation operations of a discrete shape, such as described in mesh reconstruction, allowing a more easily learnable task.

A study by Chen and Zhang [47] was the first to offer decoding of a shape by learning implicit fields. The proposed novel way of decoding a shape used a decoder to output whether a specified 3D point is inside or outside an object, given a latent vector of a shape as an input. A shape can be decoded by consulting a decoder with many point coordinates and interpolating a boundary between inside and outside of a shape, as a point can be anywhere in 3D space (opposite to slots in voxel grids). Since a neural network can learn to approximate functions, the decoder, in this case, can be interpreted as an approximator of an implicit shape constructed out of equations, and a boolean point output strategy allows a way to decode it into a discrete and usable format. Here, the only theoretical limitation to object's detail is the decoder's ability to approximate this object, where in other cases it is also the number of allocated shape units, such as grid size for voxels, points for point clouds and more. Mescheder et al. [48] explores a similar idea and provides a more advanced shape recovery, which clearly demonstrates an advantage of such implicit reconstruction by providing state-of-the-art results. Authors use a network to provide a probability for whether a point is inside or outside of a shape, rather than a boolean.

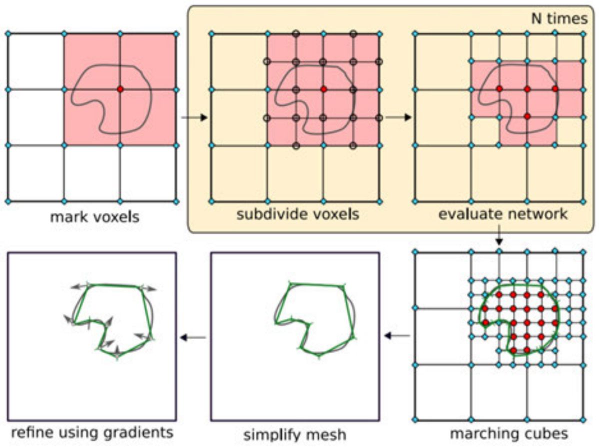

Figure 4: Image from Mescheder et al. [48] describes surface decoding from an implicit decoder. Initial points are interpolated depending on occupancy to increase detail, then the mesh is reconstructed and refined.

To recover a shape, a low-resolution grid of points is defined, passed though a network and recovered probabilities are thresholded with a goal of extracting iso-surface from an implicit field. The points inside or near a shape are interpolated iteratively by increasing detail in a similar approach to an octree. Both Chen and Zhang [47] and Mescheder et al. [48] reconstruct a mesh from obtained points using Marching Cubes algorithm [41], where the latter approach simplifies a shape with Fast-Quadric-Mesh-Simplification [49] algorithm and refines using gradients from a neural network. Both studies use 3D-annotation based loss for training their network. Niemeyer et al. [50] implements an idea of differential rendering, which is circulating in mesh or point cloud reconstruction. It allows to recover implicit shape and texture and abolishes the need for 3D annotation for training. Wu et al. [51] provide an interesting development, first of such kind in shape reconstruction, by creating model able to reconstruct a shape from separate parts as a sequence. A Bidirectional Recurrent Neural Network [52] sequentially reconstructs a shape by separately reconstructing major surface parts of this shape, which are learned implicitly, and attaching them together to form a full shape.

The usefulness of implicit shape representation was recognized and further studies aimed for improvements using this idea. Park et al. [54] were the first to offer to learn a representation of a shape as a Signed Distance Function (SDF). An SDF is a continuous function which for any given point in space returns a distance from that point to the closest surface of a shape, where a given result is positive if a point is outside of a shape and negative if it is inside. An actual shape is represented by an iso-surface where given values of SDF are equal to 0. A network for decoding a shape is trained using L_1 loss by predicting SDF values of randomly sampled points from a training 3D shape. However, authors indicate a comparably large value of memory needed to reconstruct a mesh in their experiments (7.4 MB for each shape), being

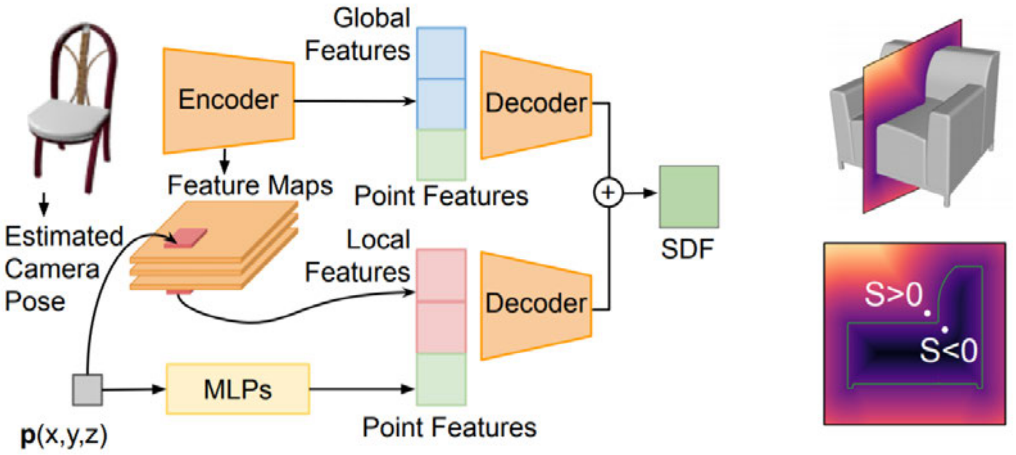

Figure 5: Illustrations from Wang et al. [53] display their model architecture and an example of an SDF.

just half of the memory needed to reconstruct a voxel grid shape with a size of 512^3. Wang et al. [53] further advance an idea of learning a shape as an SDF, but focus on recovering more detailed and local features of a shape, also resulting in a state-of-the-art performance. During encoding, the proposed model recovers an embedding vector together with local feature maps from intermediate stages of a neural network, as well as camera pose. Both local and global features are decoded separately and later added together, mixing both levels of detail. This results in an ability to generate more thin shapes, holes and more detailed forms where needed. However, the recovered shape using this method can be overly rough, sometimes resulting in worse quality than its predecessor by Park et al. [54]. Both proposals by Park et al. [54] and Wang et al. [53] retrieve a shape from a model by pre-defining a dense grid of points, inferring their SDF values and reconstructing the results using Marching Cubes algorithm. Opposite to previous binary implicit shape retrieving methods [47, 48], SDF shape networks do not need multiple iterations of shape inference to interpolate the shape boundary, as SDF values of points with close proximity to the boundary directly indicate the distance to it. Finally, Jiang et al. [55] incorporate an SDF-based differential renderer allowing unsupervised learning. All provided solutions to decode implicit shapes demonstrate impressive evaluation metrics, reconstructed shape quality and topological features.

3.3.2 Parametric shapes

Similarly to implicit surfaces, parametric surfaces are also described using equations, which in this case are describing transformations of points from 2D space to 3D space. More formally, these shapes can be understood as parameters $\overrightarrow{r}$ and u, v coordinates, returning a mapping to x, y, z space, where $\overrightarrow{r}$ is a collection of parametric equations $f(u, v)$, $g(u, v)$, $h(u, v)$ returning coordinates in corresponding dimensions. Learning such shape representations has similar advantages to learning implicit shapes.

Sinha et al. [56] provide the first solution of parametric shape reconstruction for single-image input. Authors use three separate networks with each being responsible for a specified parametric mapping function $f(u, v)$, $g(u, v)$, $h(u, v) \rightarrow x, y, z$. To reconstruct a shape, a $64 \times 64 \times 3$ geometry image with x, y, z coordinates is generated to infer a shape. However, the approach is limited topologically, which is solved by Groueix et al. [57], where a different approach to recover a parametric surface is used. Authors firstly sample multiple points from a 2D space and transforming them using a Multi-Layer Perceptron, which acts as a parametric function, transforming a point into a 3D space. This neural network is used to generate just a small patch of a shape, needing multiple neural networks to generate all patches of the shape, which are later ”stitched” into a final shape. Despite the recovered shape patches being accurate in their form, the final shape reconstructed using these patches lacks visual quality due to intersections of multiple parts being clearly visible in odd places.

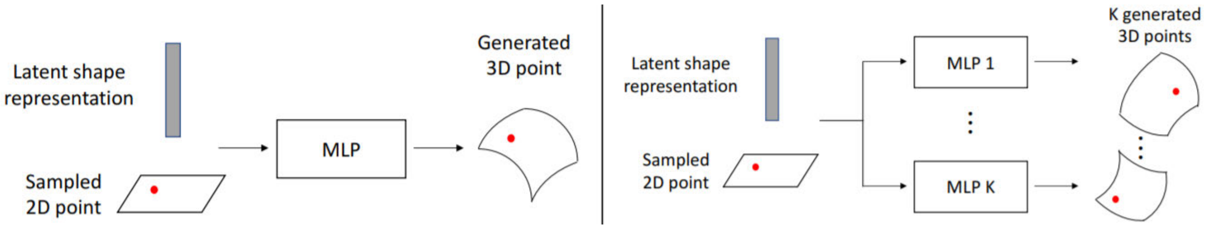

Figure 6: Illustrations from Groueix et al. [57] display simple pipeline of reconstruction. The embedded shape (latent shape representation), together with point in u, v space is passed through multiple Multi-Layer Perceptrons to transform its position in multiple patches in x, y, z space.

4 Comparison

4.1 Quantitative comparison

Table 1 depicts reconstruction quality evaluation of various models when reconstructing common classes of 3D objects. These reconstruction models are either best in their domain, claim to surpass state-of-the-art or their idea was key to further research. Reconstruction is evaluated on ShapeNet [58] dataset with 3D annotations of various common but differing classes of objects to evaluate performance in various circumstances. The most common and selected metrics are IoU (Intersection over Union), CD (Chamfer Distance) [42], EMD (Earth Mover's Distance) [42]]. IoU is simply a percentage of overlapping volume between total volume of aligned output and ground truth shapes, well describing whether reconstructed shape does not have excessive parts or big gaps of volume. CD and EMD both are distance functions between two sets of points sampled or inferred from ground truth and reconstructed shapes. Higher CD values indicate more rough or poor quality shapes (such as missing holes) with high point deviations from their ground truth values. EMD meanwhile, similarly to IoU, is more sensitive to more abstract aspects of a shape, however is less sensitive when comparing very thin shapes. Both CD and EMD are from 2048 point samples, multiplied by 10^3 and 10^{-2} respectively.

The results indicate that implicit methods provide the best trade-off between various shape types and works well on more detailed shapes, especially DISN [53]. 3DN [38] demonstrates even better performance on detailed or thin shapes as the model can quickly find a very similar pre-defined mesh structure matched to an input. Pixel2Mesh provides better performance and

Metric	Model	Yr.	Type	plane	car	chair	lamp	rifle	sofa	table
IoU	AtlasNet [57]	'18	Param.	39.2	22.0	25.7	21.3	45.3	27.9	23.3
	Pixel2Mesh [34]	'18	Mesh	51.5	50.1	40.2	29.1	50.9	60.0	31.2
	IM-NET [47]	'19	Impl.	55.4	74.5	52.2	29.6	52.3	64.1	45.0
	OccNet [48]	'19	Impl.	54.7	73.1	50.2	37.0	45.8	67.1	50.6
	DISN [53]	'19	Impl.	57.5	74.3	54.3	34.7	59.2	65.9	47.9
	3DN [38]	'19	Mesh	54.3	59.4	34.4	35.4	57.6	60.7	31.3
	PQ-NET [51]	'20	Impl.	-	-	67.3	39.6	-	-	47.4
	P. S. Gen. [42]	'17	Point cl.	60.1	83.1	54.4	46.2	60.4	70.8	60.6
	Pix2Vox [17]	'19	Voxels	68.4	85.4	56.7	44.3	61.5	70.9	60.1
CD	AtlasNet [57]	'18	Param.	5.98	17.24	13.21	38.21	4.59	8.29	18.08
	Pixel2Mesh [34]	'18	Mesh	6.10	13.45	11.13	31.41	4.51	6.54	15.61
	IM-NET [47]	'19	Impl.	12.65	8.86	11.27	63.84	8.73	10.30	17.82
	DISN [53]	'19	Impl.	9.96	5.39	7.71	25.76	5.58	9.16	13.59
	3DN [38]	'19	Mesh	6.75	7.09	17.53	12.79	3.26	8.27	14.05
EMD	AtlasNet [57]	'18	Param.	3.39	3.72	3.86	5.29	3.35	3.14	3.98
	Pixel2Mesh [34]	'18	Mesh	2.98	3.43	3.52	5.15	3.04	2.70	3.52
	IM-NET [47]	'19	Impl.	2.90	2.73	3.01	5.85	2.65	2.71	3.39
	DISN [53]	'19	Impl.	2.67	2.67	2.67	4.38	2.30	2.62	3.11
	3DN [38]	'19	Mesh	3.30	3.28	4.45	3.99	2.78	3.31	3.94

Table 1: Models' output shape quality of various object classes from ShapeNet dataset is compared using 3 evaluation metrics. Metrics: IoU- Intersection over Union (higher- better); CD- Chamfer's Distance (lower-better); EMD- Earth Mover's Distance (lower-better). Data is either from DISN [53] or from original papers. Representations of shapes during evaluation may differ (e.g. Voxel shape IoU for Pix2Vox [17] instead of mesh).

smooth shapes when processing more simple and abstract shapes, such as a sofa or an airplane. Both voxel and point cloud methods also provide great performance on most of the shapes. However, their evaluation data on the provided metrics are limited. Pix2Vox [17] evaluation is done on voxel shapes, not point sets or meshes, therefore potentially altering IoU score.

4.2 Qualitative comparison

Quantitative results alone cannot fully represent the quality of the reconstructed shape. This section compares visual results. Figure 7 depicts examples of chairs reconstructed by advanced models. The best visual results are obtained when reconstructing implicit shapes or mesh shapes with iterative up-sampling and advanced topological capabilities. These models provide superior detail in the reconstructed shape, return smooth and realistic surfaces, as well as realistic topological features. Despite good quantitative results, point cloud models return inaccurate, noisy and rough shapes. Parametric models return good abstract forms, but places, where different parametric surfaces were "stitched", are clearly visible, reducing visual quality. Initial mesh models, however, return poor quality due to their limited topological capabilities, such as the inability to produce holes. The best mesh or implicit reconstruction models arguably can provide equal or better visual performance than the best voxel or octree methods.

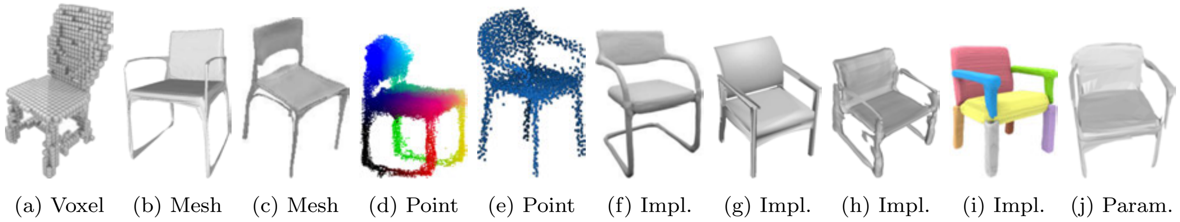
(a) Voxel (b) Mesh (c) Mesh (d) Point (e) Point (f) Impl. (g) Impl. (h) Impl. (i) Impl. (j) Param.

Figure 7: Examples of reconstructed chair using various types of representation. Images from: (a) - Pix2Vox [17]; (b) - Topology Modification Networks [36]; (c) - 3DN [38]; (d) - Eff. point cloud gen. [43]; (e) - 3D-LMNet [45]; (f) - OccNet [48]; (g) - DeepSDF [54]; (h) - DISN [53]; (i) - PQ-NET [51]; (j) - AtlasNet [57]

5 Summary & Conclusion

The research community has explored many ways to decode a 3D object from a single image using deep learning since the inception of first methods and the adoption of a voxel output strategy. Some of the applied methods have significant inherent advantages over voxel decoding and provide a great alternative. Mesh reconstruction methods are allowing to reconstruct a shape without a need for additional post-processing, the mesh structure itself is visualizable and widely used. Such methods are expanded to up-sample the shape iteratively until the desired quality is reached. Another successful direction of research is implicit shape reconstruction. Such methods allow unlimited resolution and recovery of complex topological features with superior reconstruction quality. Other types of output methods, such as point clouds, can be applied to some specialized domains due to other inherent advantages, such as quick reconstruction time or low memory requirements. The reviewed methods of shape decoding eliminate some or all voxel shape reconstruction drawbacks, providing either great alternatives for more specialized domains or already producing a more superior quality of reconstruction. Regarding discussed evidence, this review holds an optimistic view that further research will focus more on non-voxel output strategies of single-view 3D deep learning methods and through further innovations will provide substantial qualitative and quantitative reconstruction gains.

References

[1] Zerong Zheng, Tao Yu, Yixuan Wei, Qionghai Dai, and Yebin Liu. Deephuman: 3d human reconstruction from a single image. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, October 2019.

[2] Nikos Kolotouros, Georgios Pavlakos, and Kostas Daniilidis. Convolutional mesh regression for single-image human shape reconstruction. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2019.

[3] Roman Dovgard and Ronen Basri. Statistical symmetric shape from shading for 3d structure recovery of faces. In *European Conference on Computer Vision*, pages 99–113. Springer, 2004.

[4] Volker Blanz and Thomas Vetter. A morphable model for the synthesis of 3d faces. In *Proceedings of the 26th annual conference on Computer graphics and interactive techniques*, pages 187–194, 1999.

[5] Angeline M Loh, Richard I Hartley, et al. Shape from non-homogeneous, non-stationary, anisotropic, perspective texture. In *BMVC*, volume 5, pages 69–78. Citeseer, 2005.

[6] Hod Lipson and Moshe Shpitalni. Optimization-based reconstruction of a 3d object from a single freehand line drawing. *Computer-Aided Design*, 28(8):651–663, 1996.

[7] Abhishek Kar, 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 (CVPR)*, June 2015.

[8] Maxim Tatarchenko, Stephan R Richter, René Ranftl, Zhuwen Li, Vladlen Koltun, and Thomas Brox. What do single-view 3d reconstruction networks learn? In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 3405–3414, 2019.

[9] Christopher B. Choy, Danfei Xu, JunYoung Gwak, Kevin Chen, and Silvio Savarese. 3d-r2n2: A unified approach for single and multi-view 3d object reconstruction. In Bastian Leibe, Jiri Matas, Nicu Sebe, and Max Welling, editors, *Computer Vision – ECCV 2016*, pages 628–644, Cham, 2016. Springer International Publishing.

[10] Xinchen Yan, Jimei Yang, Ersin Yumer, Yijie Guo, and Honglak Lee. Perspective transformer nets: Learning single-view 3d object reconstruction without 3d supervision. In D. Lee, M. Sugiyama, U. Luxburg, I. Guyon, and R. Garnett, editors, *Advances in Neural Information Processing Systems*, volume 29, pages 1696–1704. Curran Associates, Inc., 2016.

[11] Julian;Hendra Hendra Ham, Hanry;Wesley, Hanry Ham, Julian Wesley, and Hendra Hendra. Computer vision based 3d reconstruction : A review. *International Journal of Electrical and Computer Engineering (IJECE)*, 9(4):2394, 2019.

[12] A. Yuniarti and N. Suciati. A review of deep learning techniques for 3d reconstruction of 2d images. In *2019 12th International Conference on Information Communication Technology and System (ICTS)*, pages 327–331, 2019.

[13] Jiansheng;He Qiwen;Zhang Hanxiao Fu, Kui;Peng. Single image 3d object reconstruction based on deep learning: A review. In *Multimedia Tools and Applications*, volume 80, pages 463–498, 2021.

[14] Maxim Tatarchenko, Alexey Dosovitskiy, and Thomas Brox. Octree generating networks: Efficient convolutional architectures for high-resolution 3d outputs. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 2088–2096, 2017.

[15] Zhirong Wu, Shuran Song, Aditya Khosla, Fisher Yu, Linguang Zhang, Xiaoou Tang, and Jianxiong Xiao. 3d shapenets: A deep representation for volumetric shapes. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2015.

[16] Rohit Girdhar, David F. Fouhey, Mikel Rodriguez, and Abhinav Gupta. Learning a predictable and generative vector representation for objects. In Bastian Leibe, Jiri Matas, Nicu Sebe, and Max Welling, editors, *Computer Vision – ECCV 2016*, pages 484–499, Cham, 2016. Springer International Publishing.

[17] Haozhe Xie, Hongxun Yao, Xiaoshuai Sun, Shangchen Zhou, and Shengping Zhang. Pix2vox: Context-aware 3d reconstruction from single and multi-view images. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 2690–2698, 2019.

[18] Danilo Jimenez Rezende, S. M. Ali Eslami, Shakir Mohamed, Peter Battaglia, Max Jaderberg, and Nicolas Heess. Unsupervised learning of 3d structure from images. In D. Lee, M. Sugiyama, U. Luxburg, I. Guyon, and R. Garnett, editors, *Advances in Neural Information Processing Systems*, volume 29, pages 4996–5004. Curran Associates, Inc., 2016.

[19] M. Gadelha, S. Maji, and R. Wang. 3d shape induction from 2d views of multiple objects. In *2017 International Conference on 3D Vision (3DV)*, pages 402–411, 2017.

[20] Chengjie Niu, Jun Li, and Kai Xu. Im2struct: Recovering 3d shape structure from a single rgb image. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 4521–4529, 2018.

[21] Weichao Shen, Yunde Jia, and Yuwei Wu. 3d shape reconstruction from images in the frequency domain. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 4471–4479, 2019.

[22] Rui Zhu, Hamed Kiani Galoogahi, Chaoyang Wang, and Simon Lucey. Rethinking reprojection: Closing the loop for pose-aware shape reconstruction from a single image. In *Proceedings of the IEEE International Conference on Computer Vision (ICCV)*, Oct 2017.

[23] Christian Häne, Shubham Tulsiani, and Jitendra Malik. Hierarchical surface prediction for 3d object reconstruction. In *2017 International Conference on 3D Vision (3DV)*, pages 412–420. IEEE, 2017.

[24] Edward Smith, Scott Fujimoto, and David Meger. Multi-view silhouette and depth decomposition for high resolution 3d object representation. In *Advances in Neural Information Processing Systems*, pages 6478–6488, 2018.

[25] Jiajun Wu, Yifan Wang, Tianfan Xue, Xingyuan Sun, Bill Freeman, and Josh Tenenbaum. Marrnet: 3d shape reconstruction via 2.5d sketches. In I. Guyon, U. V. Luxburg, S. Bengio, H. Wallach, R. Fergus, S. Vishwanathan, and R. Garnett, editors, *Advances in Neural Information Processing Systems*, volume 30, pages 540–550. Curran Associates, Inc., 2017.

[26] Stephan R. Richter and Stefan Roth. Matryoshka networks: Predicting 3d geometry via nested shape layers. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2018.

[27] Edward Smith, Scott Fujimoto, and David Meger. Multi-view silhouette and depth decomposition for high resolution 3d object representation. In S. Bengio, H. Wallach, H. Larochelle, K. Grauman, N. Cesa-Bianchi, and R. Garnett, editors, *Advances in Neural Information Processing Systems*, volume 31, pages 6478–6488. Curran Associates, Inc., 2018.

[28] Jiajun Wu, Chengkai Zhang, Tianfan Xue, Bill Freeman, and Josh Tenenbaum. Learning a probabilistic latent space of object shapes via 3d generative-adversarial modeling. In D. Lee, M. Sugiyama, U. Luxburg, I. Guyon, and R. Garnett, editors, *Advances in Neural Information Processing Systems*, volume 29, pages 82–90. Curran Associates, Inc., 2016.

[29] Hiroharu Kato, Yoshitaka Ushiku, and Tatsuya Harada. Neural 3d mesh renderer. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 3907–3916, 2018.

[30] Jhony K Pontes, Chen Kong, Sridha Sridharan, Simon Lucey, Anders Eriksson, and Clinton Fookes. Image2mesh: A learning framework for single image 3d reconstruction. In *Asian Conference on Computer Vision*, pages 365–381. Springer, 2018.

[31] Wenzheng Chen, Huan Ling, Jun Gao, Edward Smith, Jaakko Lehtinen, Alec Jacobson, and Sanja Fidler. Learning to predict 3d objects with an interpolation-based differentiable renderer. In *Advances in Neural Information Processing Systems*, pages 9609–9619, 2019.

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

[33] Angjoo Kanazawa, Shubham Tulsiani, Alexei A Efros, and Jitendra Malik. Learning category-specific mesh reconstruction from image collections. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pages 371–386, 2018.

[34] Nanyang Wang, 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)*, pages 52–67, 2018.

[35] Michaël Defferrard, Xavier Bresson, and Pierre Vandergheynst. Convolutional neural networks on graphs with fast localized spectral filtering. In D. Lee, M. Sugiyama, U. Luxburg, I. Guyon, and R. Garnett, editors, *Advances in Neural Information Processing Systems*, volume 29, pages 3844–3852. Curran Associates, Inc., 2016.

[36] Junyi Pan, Xiaoguang Han, Weikai Chen, Jiapeng Tang, and Kui Jia. Deep mesh reconstruction from single rgb images via topology modification networks. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, October 2019.

[37] Dominic Jack, Jhony K Pontes, Sridha Sridharan, Clinton Fookes, Sareh Shirazi, Frederic Maire, and Anders Eriksson. Learning free-form deformations for 3d object reconstruction. In *Asian Conference on Computer Vision*, pages 317–333. Springer, 2018.

[38] Weiyue Wang, Duygu Ceylan, Radomir Mech, and Ulrich Neumann. 3dn: 3d deformation network. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 1038–1046, 2019.

[39] Georgia Gkioxari, Jitendra Malik, and Justin Johnson. Mesh r-cnn. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 9785–9795, 2019.

[40] Kaiming He, Georgia Gkioxari, Piotr Dollar, and Ross Girshick. Mask r-cnn. In *Proceedings of the IEEE International Conference on Computer Vision (ICCV)*, Oct 2017.

[41] William E. Lorensen and Harvey E. Cline. Marching cubes: A high resolution 3d surface construction algorithm. *SIGGRAPH Comput. Graph.*, 21(4):163–169, August 1987.

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

[43] Chen-Hsuan Lin, Chen Kong, and Simon Lucey. Learning efficient point cloud generation for dense 3d object reconstruction. In *proceedings of the AAAI Conference on Artificial Intelligence*, volume 32, 2018.

[44] Eldar Insafutdinov and Alexey Dosovitskiy. Unsupervised learning of shape and pose with differentiable point clouds. In *Advances in neural information processing systems*, pages 2802–2812, 2018.

[45] Priyanka Mandikal, KL Navaneet, Mayank Agarwal, and R Venkatesh Babu. 3d-lmnet: Latent embedding matching for accurate and diverse 3d point cloud reconstruction from a single image. *arXiv preprint arXiv:1807.07796*, 2018.

[46] Kurt Hornik, Maxwell Stinchcombe, Halbert White, et al. Multilayer feedforward networks are universal approximators. *Neural networks*, 2(5):359–366, 1989.

[47] Zhiqin Chen and Hao Zhang. Learning implicit fields for generative shape modeling. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 5939–5948, 2019.

[48] Lars Mescheder, Michael Oechsle, Michael Niemeyer, Sebastian Nowozin, and Andreas Geiger. Occupancy networks: Learning 3d reconstruction in function space. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 4460–4470, 2019.

[49] M. Garland and P. S. Heckbert. Simplifying surfaces with color and texture using quadric error metrics. In *Proceedings Visualization '98 (Cat. No.98CB36276)*, pages 263–269, 1998.

[50] Michael Niemeyer, Lars Mescheder, Michael Oechsle, and Andreas Geiger. Differentiable volumetric rendering: Learning implicit 3d representations without 3d supervision. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 3504–3515, 2020.

[51] Rundi Wu, Yixin Zhuang, Kai Xu, Hao Zhang, and Baoquan Chen. Pq-net: A generative part seq2seq network for 3d shapes. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 829–838, 2020.

[52] M. Schuster and K. K. Paliwal. Bidirectional recurrent neural networks. *IEEE Transactions on Signal Processing*, 45(11):2673–2681, 1997.

[53] Qiangeng Xu, Weiyue Wang, Duygu Ceylan, Radomir Mech, and Ulrich Neumann. Disn: Deep implicit surface network for high-quality single-view 3d reconstruction. In *Advances in Neural Information Processing Systems*, pages 492–502, 2019.

[54] Jeong Joon Park, Peter Florence, Julian Straub, Richard Newcombe, and Steven Lovegrove. Deepsdf: Learning continuous signed distance functions for shape representation. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 165–174, 2019.

[55] Yue Jiang, Dantong Ji, Zhizhong Han, and Matthias Zwicker. Sdfdiff: Differentiable rendering of signed distance fields for 3d shape optimization. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 1251–1261, 2020.

[56] Ayan Sinha, Asim Unmesh, Qixing Huang, and Karthik Ramani. Surfnet: Generating 3d shape surfaces using deep residual networks. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 6040–6049, 2017.

[57] Thibault Groueix, Matthew Fisher, Vladimir G Kim, Bryan C Russell, and Mathieu Aubry. A papier-mâché approach to learning 3d surface generation. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 216–224, 2018.

[58] Angel X. Chang, Thomas A. Funkhouser, Leonidas J. Guibas, Pat Hanrahan, Qi-Xing Huang, Zimo Li, Silvio Savarese, Manolis Savva, Shuran Song, Hao Su, Jianxiong Xiao, Li Yi, and Fisher Yu. Shapenet: An information-rich 3d model repository. *CoRR*, abs/1512.03012, 2015.