3D Reconstruction from Depth Images using Machine Learning

Fahd T. Alhamazani

Cardiff University

School of Computer Science and Informatics

A thesis submitted in partial fulfilment

of the requirement for the degree of Doctor of Philosophy

October 2023

Abstract

This thesis investigates 3D reconstruction from depth images, focusing on three related tasks. First, we present our work on reconstructing complete volumetric shapes from a single depth image. Our model proposes to incorporate a dynamic latent code, allowing the model to determine the appropriate code for the estimation. We further develop a multi-stage approach to iteratively improve completion, and employ a classifier as an auxiliary task to enhance estimation. Second, we advance the quality metric for 3D shapes by leveraging rendering them using various styles and from different views. We also improve the SSIM metric by introducing a mask to ensure it is stable with different rendering canvas sizes. Subsequently, we develop a neural network to mimic human visual judgment. Lastly, traditional reconstructions primarily target rigid bodies due to the straightforwardness of their shape formation. So we further develop a method for predicting canonical form, i.e., returning shapes to their original pose, which can significantly simplify shape completion for deformable objects.

Contents

List of Publications

The work introduced in this thesis is based on the following publications. Specifically, first paper coming from chapter 3, while second paper coming from chapter 4.

- Alhamazani, F., Lai, Y.K. and Rosin, P.L., 2023. 3DCascade-GAN: Shape completion from single-view depth images. Computers & Graphics.
  doi: https://doi.org/10.1016/j.cag.2023.07.033

- Fahd Alhamazani, Paul L. Rosin, and Yu-Kun Lai. An Image-based Model for 3D Shape Quality Measure. In Peter Vangorp and David Hunter, editors, Computer Graphics and Visual Computing (CGVC). The Eurographics Association, 2023.

List of Figures

List of Tables

List of Algorithms

—«وَفَوْقَ كُلِّ ذِي عِلْمٍ عَلِيمٌ»—

–«above every possessor of knowledge, there is a Knower»–

Acknowledgements

To my supervisors, Professor Yu-Kun Lai and Professor Paul Rosin: Without your guidance, I would not be the person I am today. Every meeting, every point you mentioned, has shaped the person I've become. I eagerly looked forward to our discussions; they were enlightening, providing me with clear direction. During our first year of scientific discussions, I struggled to grasp even the basic scientific vocabulary. Yet, before I realized it, I found myself able to actively participate and contribute to our conversations.

To my parents, who spent precious days and nights ensuring I become the person I am today. To my mother, who was patient with me and recognized the importance of education. I hope one day to return a fraction of what you have given me. To my father, whose sternness was a testament to his desire to see me succeed. To my brothers and sisters, who were like second mothers and fathers to me.

To my Wife, my companion, a gifted light in a dark journey.

To my son Omar (caustic).

To those with whom I have shared this journey: Asmail Muftah, Osama Al-murshed, Turky Al-lelah, Fahad Alodhyani, Hongjin Lyu, Tao Wang.

To the anonymous soldiers: I may forget your face, I may forget your name, yet I believe in your existence. I can't count how many times you have helped me, but please know that you shall always be remembered.

«I wish you all the best»

Chapter 1

Introduction

1.1 Overview

Three-dimensional (3D) reconstruction refers to the reconstruction of 3D shapes from lower dimensional inputs, such as 2D images (either single- or multi-view), or 2.5D, i.e. depth images. The challenges in this process often arise from situations such as ill-posed shapes where multiple solutions are possible due to heavy (self-)occlusion. Although the world consists of 3D shapes, 2D images and relevant processing techniques have been developed to more conveniently capture and analyse scenes. However, recent research indicates a multitude of applications for 3D reconstruction, ranging from object grasping [35, 38, 156] to depth estimation [108, 96, 121], while other studies focus on the overall reconstruction process itself [49, 42]. Compared to 2D images, 3D objects are more closely related to real-world tasks. Another major challenge faced in 3D research is the representation of 3D shapes. There is no single method for representing a 3D shape, which can be characterised by point clouds, voxels or mesh forms, which, in turn, influence and divide research efforts.

In this chapter, we present an overview of the work completed in this thesis. Section 1.2.2 delineates the goals of the thesis, while Section 1.2.1 discusses the underlying motivations. The structure of the thesis is detailed in Section 1.2.3. Finally, Section 1.3 summarises our contributions throughout the manuscript.

1.2 Thesis Summary

1.2.1 Motivation

Modern technological operations, like robotics and obstacle avoidance, heavily depend on 3D reconstruction. Depth images are a primary data source for this. Capturing depth details was once a significant hurdle, but affordable depth cameras have changed this, making data collection easier. This has paved the way for novel uses, such as virtual reality (VR) [76], with supporting datasets [63]. However, reconstructing a full 3D form from a single depth image, which represents just one perspective, is still challenging. Depth images do not fully depict a shape due to inherent self-occlusion, causing incomplete reconstructions with gaps and inaccuracies. The ideal solution should handle these challenging views since obtaining complete 3D information is often unrealistic in real-world scenarios because of high costs and time. For example, fully capturing indoor furniture would be tough due to significant blockages.

In this thesis, we focus on the reconstruction of 3D objects from single depth images, proposing a novel cascading model to tackle the challenges (like estimating full shape for heavily occluded shape) inherent to this process. Additionally, we introduce an approach to selecting latent codes; previous methods utilised the entire set of latent codes, potentially diminishing the ability to reconstruct a complete shape. We further incorporate a self-attention layer to concentrate on regions of interest. Finally, we introduce a classifier for reconstructed shapes as an auxiliary task, which helps enhance the reconstruction task, as well reconstructed shapes are more recognisable.

We also present a new evaluation metric for assessing 3D shapes. Current metrics tend to be representation-specific, but converting an objects into a unified representation can cause corruption and noise. Moreover, metrics that directly measure 3D shape differences are often inconsistent with human perception, as some small changes of geometry can be highly visible, whereas larger but smoother geometry changes can be hardly noticeable. Therefore, our proposed method is based on rendering 3D shapes

to 2D images, along with 2D-image based measures, which creates an unbiased model across all representations. We also use a systematic approach to rendering images that covers shapes with equally distributed views and a diverse range of rendering styles. We introduce Mask-SSIM (Structural Similarity Index Measure), an extended version of SSIM in which the foreground is separated from the background, making it insensitive to the rendering canvas size, which is helpful in our reconstruction task.

Finally, we tackle non-rigid shapes, which pose a greater challenge than rigid shapes due to the potentially large deformation space. We have successfully reached the goal of achieving a canonical form for non-rigid shapes by carefully isolating the underlying deformations through a deep neural network. The benefit of attaining this canonical form is that it significantly enhances our ability to reconstruct and analyse these non-rigid shapes in a more generalised and systematic way.

1.2.2 Goals

The aim of this thesis is to address 3D challenges and identify viable solutions to 3D reconstruction and related techniques that bridge gaps in the research field. Our contributions specifically address three primary tasks: The first task considers volumetric shape reconstructions from single-view depth images. Furthermore, We identify the need for a unified, more perceptually meaningful metric for 3D geometry, which can cope with variations in the 3D representations, where comparisons can assist in recognising similarities in real-world scenarios. We demonstrate that recent contributions are ineffective in this regard. Moreover, we find that existing 3D reconstruction research predominantly focuses on the reconstruction of rigid shapes due to difficulties in handling non-rigid deformations. Consequently, the last contribution aims to investigate canonical forms, i.e., bringing shapes back to their standardised poses.

In this thesis, we present novel contributions across three related tasks. First, we focus on recovering volumetric shapes from single depth images, employing a cascaded ap-

proach that enables the model to outperform state-of-the-art (SOTA) models. Second, we propose a unified evaluation metric function, using a 2D metric that combines rendering of 3D shapes from different views with different styles to evaluate 3D geometry. We demonstrate that this model is capable of identifying the differences between objects. Along with a neural network based learning approach, our 2D image based metric achieves better correlation with human perception in terms of 3D shape quality. Finally, we convert a depth image of an object into its canonical form, normalising its pose and deformation to facilitate the reconstruction of a complete 3D shape.

1.2.3 Organisation

This thesis consists of six chapters, commencing with the Introduction (Chapter 1), which provides a general overview. Chapter 2 summarises the SOTA models. We explore diverse approaches to 3D shape reconstruction, spanning methods that leverage a single image as input, techniques focusing on 2.5D shape recovery, and point cloud completion strategies. We also delve into the signed distance field (SDF) and recent contributions to evaluation metric enhancement, elucidating the relationship between representation and the presented metrics. Moreover, we explore 3D reconstruction papers that employ 2D supervision, showcasing the most recent contributions in this domain, wherein the model is optimised through reconstructions with lower-dimensional inputs. Finally, we review contemporary work on canonical forms.

Chapters 3, 4 and 5 introduce our three primary contributions, all of which focus on solving and investigating 3D reconstruction and related tasks, specifically in the areas of reconstruction, evaluation, and deformation. We strive in this thesis to reconstruct plausible shapes. Starting with Chapter 3, we propose a novel cascade model for reconstructing shapes in three stages, beginning with the depth image and progressively enhancing the reconstruction. In Chapter 4, to address the challenges associated with evaluating geometries, we introduce an innovative model, which streamlines comparisons, by adopting a systematic method by capturing images to disentangle the shapes

representation. Moreover, we utilise diverse styles and shader types to ensure a more generalised evaluation process. In addition, we employ a deep learning model to accurately determine the score of these representations. Chapter 5 delves into the task of finding canonical forms, exploring the deformation of non-rigid shapes. With a depth image as input, this approach leads to a canonical pose, and it can be further extended to enable the reconstruction of a complete shape from this pose.

Chapter 6 concludes our research, summarising the efforts undertaken in this thesis and outlining potential avenues for future work to expand upon this research.

1.3 Contributions

In this thesis, we endeavoured to reconstruct shapes from single depth images, successfully achieving our first contribution. To assess these shapes accurately, it is imperative to employ an evaluation metric that is congruent with human perception and capable of unifying diverse shape representations; this necessity led to the introduction of our second contribution. Furthermore, depth images may encompass non-rigid shapes, introducing additional complexity. To address this and estimate the complete shape for deformable objects, we introduced our third contribution that learns to turn (incomplete) deformable objects to their canonical form, thereby enhancing the robustness and applicability of our methodology.

In this section, we summarise our proposed solutions to problems posed by 3D reconstruction. The contributions are summarised below.

3D Reconstruction from Single Depth Images

We introduce a novel model for reconstructing a volumetric shape from a voxelised depth image. The key contributions of this chapter are as follows:

- We propose a cascade architecture consisting of multiple encoder-decoder blocks with additional skip links, which provides better 3D reconstruction than a single encoder-decoder.
- We incorporate a self-attention layer to refine the 3D shapes, mimicking the human ability to focus on a region of interest in volumetric space.
- We introduce a novel dynamic latent space in which the model has the ability to select only relevant latent codes to estimate 3D shapes. This provides a strong approach to sparse regularisation that enhances the robustness of the network.
- A classifier network is introduced as an auxiliary task to provide additional guidance to the reconstruction model.

Rendering based 3D Shape Evaluation

The contributions of this chapter are as follows:

- We propose an image-based method of measuring the shape distortion of 3D shapes. We further combine a variety of rendering styles and 2D image quality measures along with a neural network based learning approach for improved subjective 3D quality prediction.
- To ensure more stable performance when shapes are rendered to different canvas sizes and accurately detect the similarities despite image resolution, we extend SSIM to only focus on the foreground region. This method is referred to as Mask-SSIM, which is proved effective for our task.
- Experiments on public datasets demonstrate that our method achieves reliable predictions of subjective quality scores, outperforming existing techniques.

Learning to Generate Canonical Forms for Single Depth Images

The contributions of this work are as follows:

- We propose Canonical pose model, an end-to-end 2D network designed for the canonical pose prediction of a single depth image. It comprises three components, Local Features Extractor (LFE), Multi-Scale Features Extractor (MSFE) and Deformation prediction component.
- We propose parallel encoders and a single decoder block that extract features at different scales and use a fusing decoder to decode multi-scale, high-dimensional features.
- The extensive experimental results on TOSCA [19] and human [113] datasets demonstrate that our model outperforms the existing state-of-the-art methods and has competitive inference time. Moreover, our model is also capable of preserving high quality geometric information while deforming shapes across different types of objects, such as humans and animals.

1.4 Research Question

The central inquiry of this thesis revolves around the challenge of reconstructing and analysing 3D shapes from single depth images. This investigation is motivated by the need to improve the accuracy and utility of depth images in various applications, including computer vision, augmented reality, and robotics. The general research question that guides this thesis is:

How can we effectively reconstruct, evaluate, and deform three-dimensional shapes from single depth images to achieve a comprehensive understanding and manipulation of the captured objects in digital form?

This thesis contributes to the field through three distinct but interconnected avenues:

Contribution 1: 3D Reconstruction from Single Depth Images. The first contribution addresses the challenge of reconstructing a complete and accurate 3D shape from a single voxelised depth image.

Contribution 2: Rendering based 3D Shape Evaluation. The second contribution focuses on assessing the fidelity of the reconstructed shapes by evaluating the distortion introduced during the reconstruction process. This involves developing metrics and methods to quantify and analyse the deviation of the reconstructed shape from its original form.

Contribution 3: Learning to Generate Canonical Forms for Single Depth Images. The final contribution explores the depth images of deformable shapes, by reconstructing the default pose depth images.

Each of these contributions addresses a critical aspect of the problem space, collectively advancing our ability to manipulate and understand 3D shapes.

1.5 Research Aim and Objectives

The general aim of this research is to enhance the methodologies for reconstructing, evaluating, and deforming 3D shapes from single depth images. This aim is pursued with the intent to address existing limitations in shape analysis and manipulation, particularly for applications that require high fidelity and functional flexibility in the rep-

resentation of 3D objects. To achieve this aim, the research is structured around three key objectives, each corresponding to the thesis contributions previously outlined.

1.5.1 Objective 1: To Develop a Model for Complete Shape Reconstruction

- Innovate a voxelisation-based approach to transform single depth images into comprehensive 3D shapes, capturing the complete geometry of objects with high accuracy.
- Implement models that optimise the reconstruction process, minimising data loss and maximising geometric fidelity.

1.5.2 Objective 2: To Evaluate and Quantify Shape Distortion

- Develop a framework for quantitatively assessing the distortion in shapes, comparing them against their referenced shapes.
- Create metrics and tools that facilitate the objective evaluation of shape fidelity.

1.5.3 Objective 3: To Achieve Realistic Shape Deformation for Non-Rigid Bodies

- Explore methodologies for deforming shapes depth image of non-rigid bodies, aiming to represent their default and transformed poses with high realism.
- Develop models that allow for the efficient reconstruction of non-rigid shapes.

1.6 Challenges

The pursuit of reconstructing 3D shapes from single depth images, evaluating distortion shapes, and deforming depth images encompasses a range of technical and theoretical challenges. These obstacles are intrinsic to the complexity of the tasks at hand and the current limitations of existing methodologies. Addressing these challenges is fundamental to advancing the field and achieving the objectives outlined in this thesis. The primary challenges encountered in this research are detailed below.

1.6.1 Challenge 1: Incomplete Data and Occlusions

- Single depth images provide a limited view of an object, leading to incomplete data capture and occlusions. Overcoming this challenge requires innovative approaches to infer the missing information and accurately reconstruct the full 3D shape of the object.

1.6.2 Challenge 2: High Computational Complexity

- The processes of voxelisation, shape evaluation, and depth images deformation, demand significant computational resources. Optimising models to balance accuracy and computational efficiency is a critical challenge.

1.6.3 Challenge 3: Realistic Deformation of Non-Rigid Bodies

- Reconstructing realistic deformation for non-rigid bodies from single depth images poses significant difficulties, particularly in maintaining depth distance consistency.

1.7 Summary

In this thesis, We have initiated an academic study focused on the field of 3D reconstruction, a field that is rich with both opportunities for advancement and inherent challenges to overcome. Across six chapters, we present our innovative approaches to the key problem, shedding new light on the recovery of 3D shapes from lower-dimensional sources. Our work includes the development of a cascaded model to refine reconstruction, the introduction of a unified evaluation metric and techniques for handling non-rigid shape deformations. From outlining the current SOTA models to delving into recent contributions and detailing our novel solutions, our study advances the field of 3D reconstruction and sets the stage for further research. The conclusion summarises our achievements and discusses future prospects in this field of study.

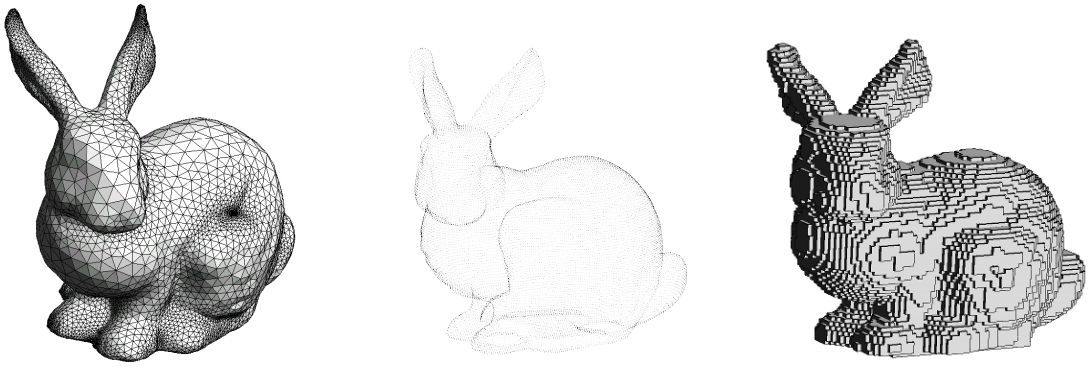
Chapter 2

Literature Review

2.1 Overview

Deep Learning (DL), a specialised branch of the broader machine learning discipline, is fundamentally involved in crafting algorithms known as artificial neural networks. These algorithms are inspired by the complex structure and functions of the human brain and have the capability to model high-level abstractions of data. DL has found applications across a diverse array of fields. In the medical domain, it is instrumental in imaging techniques for detecting and classifying various cancers and diseases [94, 152], while in the automotive industry, it is employed for obstacle detection in self-driving cars [116] [114]. The technology extends to natural language processing, where DL-based neural machine translation enables instantaneous translation between various languages and English [43], and even speech recognition via models like DeepSpeech, allowing recognition of languages such as Slovak, English, and Mandarin [115].

The field of computer vision also harnesses DL, utilising the Convolution Neural Network (CNN) for tasks like facial recognition, image segmentation, and processing both online and offline videos, establishing CNN as a pivotal component in vision-based models [73]. Furthermore, pioneers like Dan Claudiu Ciresan have leveraged GPUs, such as NVIDIA GTX 280, for training learning-based models, leading to innovations like nine-layer CNNs [33].



(a) Mesh (b) Point Cloud (c) Voxel

Figure 2.1: Bunny in different representations.

When it comes to 3D object reconstruction, the scientific community grapples with intricate open problems. Current practices often rely on datasets derived from easily accessible 2D images, such as those captured by camera phones. However, since our physical world is inherently three-dimensional, this reliance on 2D representation can lead to significant loss of critical information. Unlike 2D, 3D shapes possess multiple representations, contributing to a unique research challenge, as no single function can process these varied forms. The same shape might have disparate representations, such as point clouds, triangular meshes, parametric surfaces, or voxels, as depicted in Figure 2.1. This diversity in representation leads to complications when assessing accuracy and conducting comparisons.

In this literature review, we have organised our exploration based on the input types to reconstruction 3D shapes, starting with 2D and 2.5D representations and progressing to complete 3D models. Our examination will provide insights into the methods used to convert these varying inputs into coherent 3D shapes. Following this, we will discuss the essential evaluation metrics that assess the quality and accuracy of these reconstructions. Lastly, we will shed light on the concept of canonical form, a specific aspect of 3D modelling. Our aim is to offer a clear and accessible overview of the current research and practices in 3D reconstruction, sorting the literature based on input, to guide both researchers and practitioners in the field.

2.2 3D Reconstruction and Completion

2.2.1 2D Image as Input

In recent literature, several methods have been developed in the filed of 3D object reconstruction from 2D images. Hane et al. [53] presented the hierarchical surface prediction (HSP) approach. While it performed on par with uniform prediction baselines at coarse resolutions, its edges were noticeable at finer resolutions, offering slightly more accurate and detailed results. A common approach in this area is the employment of deep learning architectures to disentangle shape from pose and lighting from single images, as detailed in [55, 97, 98]. While these methods showed improvements over 2D-supervised methods, especially when making use of shading cues, they exhibited consistent limitations when models were restricted to silhouettes alone. The role of silhouette losses in 3D reconstruction was a central objective in [109]. Though silhouette losses provided variable results, improving outputs in certain scenarios and deteriorating them in others, leveraging shading cues in a standard white light setup yielded an advantage. The work also pointed out the benefit of using multiple views for each object but did not explore depth information. Further elaborating on the variability of silhouette losses, the work [150] acknowledged both the potential improvements and challenges that can emerge when employing them. Shading cues continued to play an important role in refining outputs, especially under varied lighting conditions. Comparatively analysing learning mechanisms, Yan et al. [153] discussed three methods. The research suggested that projection loss had a slight edge over other methods in terms of generalisation, particularly when multiple categories were incorporated into the learning process. The findings of [155] echoed the consistent observation of silhouette losses' dual nature. Their method, while adept at detailing, showed limitations when silhouettes were the sole input. In essence, the previous literature points to a modest progression in 3D object reconstruction from 2D images. Shading cues and deep learning architectures emerge as common tools, with silhouette losses presenting

both opportunities and challenges. Also Zhang et al. [168] used 2.5D sketches for reconstructing 3D objects in the Generalisable Reconstruction (GenRe) model, which has three learnable steps. The first component of the model is a depth estimator, in which the input is a single image to an encoder-decoder model where both have four convolution layers (ResNet-18 [54]). In addition, for each layer between the encoder-decoder, there is a skip connection (so with a U-Net [120]). The local features extracted in the encoder layer are stacked in the decoder layer, and the output is a depth image. The depth image is projected as voxels (used at the end in the model) and as a partial spherical map. Second, a spherical map in-painting network uses the partial spherical map to output the projected voxels. The model also has the same structure as the first network, yet the decoder has 3D convolution layers. Finally, the voxel refinement network also consists of an encoder-decoder network; however, the input channels are $128 \times 128 \times 128 \times 2$, with the last dimension representing the number of inputs (i.e. both the projected voxels from the depth image and in-painting spherical map). In another approach using 2.5D, as proposed by Wu et al. [146], ShapeHD consists of two cascade models capable of recovering realistic-looking 3D shapes in which the output is penalised regarding the appearance only (unrealistic or realistic). First, in the encoder-decoder based 2.5D estimator model, both networks (encoder and decoder) consist of four layers: for the encoder, convolution layers are used, and for the decoder, transposed convolution layers are used. The model takes a single image and outputs silhouette, depth and surface normal images. After that, the shape is recovered by using the encoder-decoder model where 2.5D images are used as the input, but the encoder is a 2D convolution layer consisting of four layers, and the decoder consists of 3D convolution layers, mapping a 200-D latent code vector to a $128 \times 128 \times 128$ 3D shape (voxel grid). Finally, an adversarial model is used (Generative Adversarial Network, or GAN [45]) in which the generator is first trained to estimate a 3D object, and the discriminator is trained on distinguishing real ones from fake (synthesised ones), enabling the discriminator to learn to identify real appearance, which can be used as "naturalness loss" in the final stage.

Other works [36, 46] offer methods that aim at converting 2D images into 3D models. The importance of these approaches is to leverage information from a collection of 2D images during training, utilising point features to refine and maintain the topology of the original shape. These papers stress on the complexities emerging from the absence of direct 3D supervision where they utilised differential renderers.

Taking it a step further, Hu et al. [57] also focused on the idea of recovering 3D shapes using just a single image. The novelty lies in how consistency in the methods enhances the local parts' reconstruction quality. However, [60] contributes a topologically-aware deformation field approach, built upon the SDF-SRN network (Signed Distance Function-Scene Representation Network) architecture introduced by [81], which tends to surpass a modified version of SDF-SRN, even though, with the caveat of requiring more than a single image.

There has been an evident push towards minimising the dependency on multi-view images for 3D reconstruction, as indicated by [64] and [170]. Lin et al.'s work [81] stands out for its singular employment of an implicit SDF (Signed Distance Function) representation under single-view supervision. Furthermore, Huang et al. [60] leveraged the single-view, multi-category learning of an implicit shape representation using SDF, shedding light on the versatility of the approach across various categories. Lastly, Zheng et al. [172] offer a method entrenched in a generalised deep implicit surface network, showcasing the method's adeptness across 240 diverse shapes per category. The approach's expansive potential is evident in its ability to convert images into 3D shapes. Nonetheless, challenges persist, specifically those rooted in training instability and the inherent multi-modal nature of certain categories.

Recent advances in 3D shape reconstruction have illuminated the path towards capturing intricate structural details, albeit with certain limitations. Li et al. [74] developed a method leveraging a network to predict a combined Signed Distance Function (SDF), primarily aimed at minimising reconstruction loss. Their approach is particularly adept at discerning complex structural details, such as chair slats. However, it presupposes

that surface details predominantly extend over largely flat surfaces, a limitation further exacerbated by the exclusive application of Laplacian loss to the frontal surface, potentially restricting its broader applicability.

Following Li et al.'s approach, Lin et al. [81] introduced a method that similarly grapples with the challenges of detailing surface nuances, albeit mitigated slightly by a weighted sampling technique intended to enhance the recovery of slender structures. Yet, this method shares the fundamental challenge of adequately recovering details on heavily curved surfaces. Patino et al. [106] and Remelli et al. [118] both advocate for weighted sampling strategies to mitigate some of these challenges, suggesting the integration of view parameter inference and symmetry priors as innovative solutions to prevailing issues in the field.

In contrast, Shan et al. [126] critique the inherent limitations of current 3D reconstruction techniques, particularly in accurately capturing geometric details in areas with "overhangs", offering a counterpoint that underscores the potential of alternative contemporary methods that perform well in 3D mesh and keypoint reconstruction.

Xu et al. [151] propose a novel approach through a bi-level optimisation framework that optimises object pose and shape concurrently, demonstrating superior accuracy and a reduction in artefacts, such as self-intersections. This underscores the potential benefits of leveraging features from neighbouring vertices in 3D reconstruction tasks, setting a new benchmark for future research in the domain.

2.2.1.1 Transformer-based Methods

Other methods suggest using the transformer layer to tackle 3D reconstruction, thanks to its strong learning capabilities exploiting attention mechanisms. For example, Shi et al. [127] proposed a new method called 3D-RETR for end-to-end 3D reconstruction using Transformers. The method involves using a pretrained Transformer to extract visual features from 2D input images, followed by another Transformer network to

extract voxel features, and finally a CNN (Convolutional Neural Network) Decoder to reconstruct objects. The model is capable of recovering 3D reconstruction from a single view or multiple views.

Recent advancements in 3D shape reconstruction have increasingly leveraged transformer models due to their proficiency in capturing long-range dependencies. Li et al. [77] introduced a novel approach employing a Vision Transformer (ViT) encoder to extract regional features from 2D images, coupled with a voxel decoder for generating 3D voxels. This method has shown effectiveness in processing both synthetic and real-world images, underscoring the model's versatility and its ability to learn nuanced feature representations. Similarly, Maxim et al. [91] utilised a ViT for volumetric shape reconstruction from a single image, focusing on estimating occupancy probabilities, which signifies a converging interest in transformer-based solutions for 3D reconstruction tasks.

Peng and his team [107] proposed a hybrid model that combines a transformer encoder with a 3D CNN to exploit both long-range and spatial interactions effectively, introducing a 3D feature fusion mechanism to integrate data comprehensively. This approach indicates a promising direction for enhancing model performance through architectural synergy.

On a different note, Mazur et al. [92] explored the dimensionality reduction of high-dimensional point clouds before processing with a CNN, followed by an inverse projection to recover the structure. This method offers an innovative perspective on managing computational complexity and data representation.

While these approaches demonstrate significant potential, the reliance on transformers raises questions about computational efficiency and the applicability to real-time systems. Furthermore, the varying methodologies underscore the need for standardised benchmarks to facilitate direct comparisons across models. Future research should consider these aspects, aiming to refine the balance between accuracy, efficiency, and generalisability in 3D shape reconstruction technologies.

Kurenkov et al. [68] proposed DeformNet, which investigates reconstruction through deformation. The authors first suggest retrieving the closest shape from the dataset to the given input image. Furthermore, both the image and the retrieved shape are used as input for the encoder-decoder network. The image is then inserted into a 2D encoder, and the 3D shape is inserted into a 3D encoder, where the latent codes for both are stacked together as the input given to the decoder. The output is a vector containing the offsets of the control points in the Free-Form Deformation (FFD) representation [124]. In the last stage, the shape points are sampled on the surface of the retrieved shape to generate a point cloud representation, and coordinates are applied to the shape by using FFD, so the final output shape is deformed and matches the query image (Fig. 2.3).

2.2.2 2.5D as Input

Wu et al. [145] proposed adding 2.5D in the pipeline before recovering the 3D shape. The model (MarrNet) consists of three phases. The first phase, estimating 2.5D by using 2D as input (256×256 resolution), uses an encoder and a decoder architecture. The encoder contains four convolution hidden layers and a fully connected layer, and the decoder contains four convolution layers; the output contains silhouette map images, surface normal maps and depth image. After that, because the previous phase only outputs silhouette, normal and depth images, the model is trained on synthetic data to ignore unwanted features like textures and lighting. The model also leverage an encoder-decoder network where the encoder consists of five convolution layers and a two-layer MLP (Multi-Layer Perceptron), and the output is a 200-D latent vector. The decoder consists of five 3D convolution layers that output a $128 \times 128 \times 128$ voxel-based occupancy map. In the third and final phase (namely reprojection consistency),the authors introduce two loss functions: depth reprojection loss and surface normal reprojection loss. Those functions help refine the 3D shape compared to the input (2.5D input). They first trained the model on synthetic data, and after that they fine tuned the model on realistic appearance shapes (compared to 2.5D); however, the

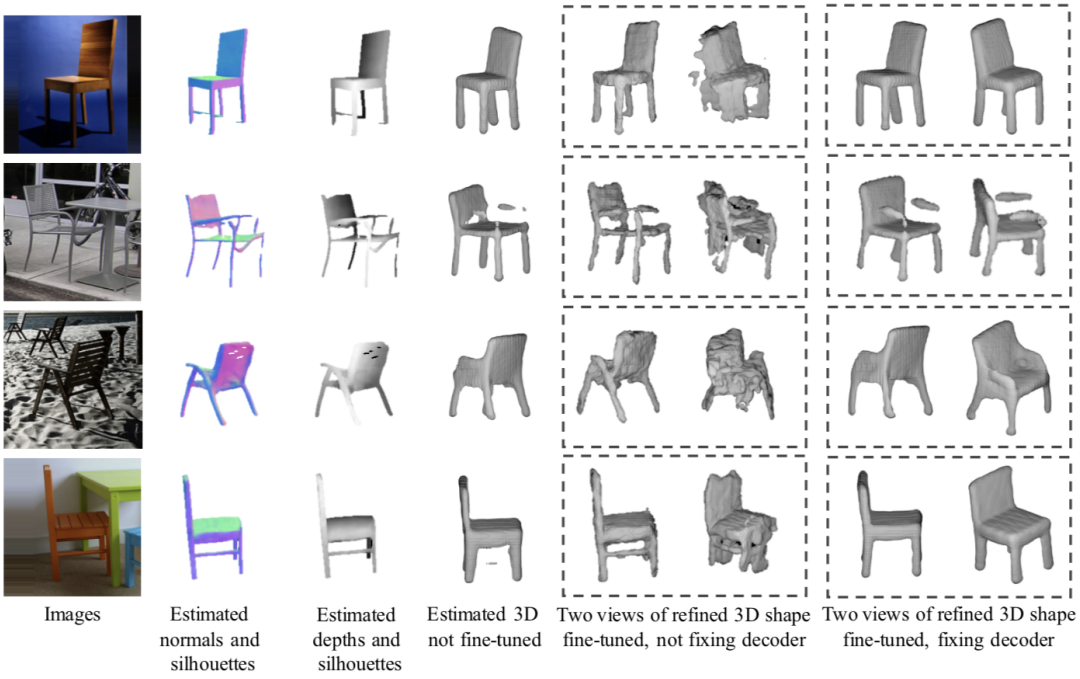


Figure 2.2: Column 1 shows the input images. The columns 2 and 3 present the estimated normals and depth information. Unrealistic output results are obtained without fine-tuning (column 4), whereas columns 5 and 6 present results after fine-tuning (enforcing the output to be closer to the input image), but the reconstruction quality is limited. Columns 7 and 8 are the results after fine-tuning with the decoder fixed. These results are similar to the synthetic appearance more than the input [145].

output may overfit the images, leading to poor reconstruction, so the authors proposed to fix the decoder. Although fixing the decoder makes the output maintain the shape, the limitation in this model is similar to the scenario before fine-tuning, namely the model predicts shapes similar to those in the synthetic dataset and fail to match the detailed realistic appearances as specified by the input images, as shown in Fig. 2.2.

Yang et al. [154] reconstructed 3D objects using 2.5D input. The model contains an encoder-decoder network in which the encoder accepts a 2.5D image and compresses the features to a 2000-D latent vector, and the decoder takes the latent code and reconstructs the 3D shape into a $256 \times 256 \times 256$ upsampled usin a U-Net. [120] ar-

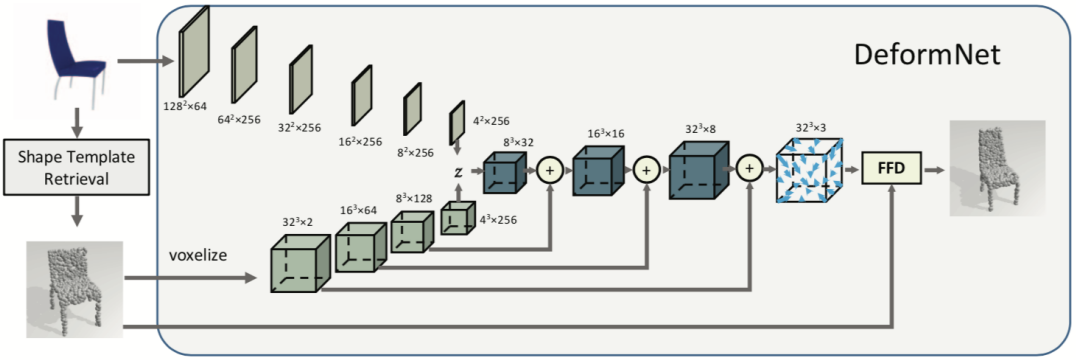


Figure 2.3: Overview of the DeformNet model [68].

chitecture. After that, the original input is concatenated with the reconstructed shape and the ground truth shape, and both shapes are then introduced to a discriminator to generate a loss value. To show an acceptable result, a threshold is used to eliminate unwanted voxels in the volume space; furthermore, the threshold varies for each class. The model could still be unstable during training with the discriminator collapsed.

Another depth-based approach was proposed by Malik et al. [90] where they present a model to estimate a 3D hand from a single depth image. The input is voxelised to $88 \times 88 \times 88$ resolution. The model consists of three steps. First, a network is trained to predict the hand pose (represented as a "joints heatmap") at lower resolution $44 \times 44 \times 44$ grid, and then concatenate the input and output as input for the estimation network. Second, the model uses a CNN to reconstruct the voxelised shape of size $64 \times 64 \times 64$. However, as described in the paper, the output cannot preserve the hand topology, so as a result, another network is further trained to predict the hand surface. Finally, the outputs (mesh shape and voxelised shape) are combined as input for the final network to register the shape which consist of Fully convolution layers. The output resolution is $64 \times 64 \times 64$. The model operates on weak supervision, where another network is used to generate depth images. The model achieves an improvement of 10% compared with existing methods, but the number of parameters is still high. On the other hand, Yao et al. [157] propose another method that initially predicts a 2.5D depth image, followed by a symmetric reflection, and finally a back view of the shape. The

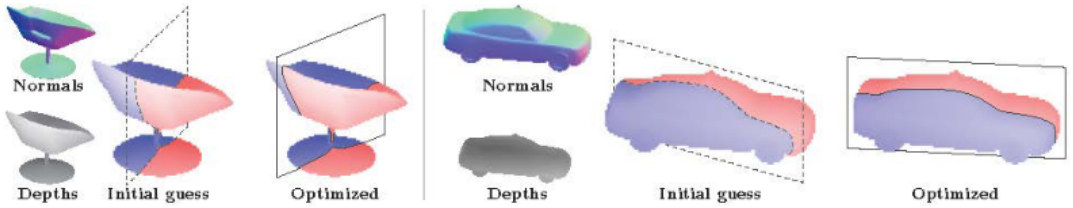


Figure 2.4: Some examples of predicted symmetric planes [157].

model employs an encoder-decoder with skip links to recover the 3D surface. Where the novelty is on leveraging symmetric information. Specifically, Symmetry detection is trained separately. The model is trained on the ShapeNet dataset [24]. Although many shapes are symmetrical, this does not apply to all shapes. Consequently, the model forces a plane of symmetry, treating the shape from that perspective, which may introduce distortions (see Fig. 2.4).

Chen et al. [26] suggest a new approach to 3D reconstruction called Multiresolution Deep Implicit Functions (MDIF), which is a hierarchical representation that can capture fine geometry details while also being able to perform global operations such as shape completion. MDIF is unique in that it can represent different levels of detail by leveraging both autoencoder and a single decoder with latent parameterised. Furthermore, The authors propose a dropout layer for latent code to enhance the reconstruction details.

2.2.2.1 Transformer-based Methods

In their study, Chen et al. [27] put forward a method based on transformers for generating 3D point clouds. They employ a pre-learned canonical space to break down point clouds of a particular category into sequences that align semantically. These sequences are subsequently quantised. This data then serves to develop a context-sensitive codebook composition, facilitating point cloud generation and completion from a depth image.

In this thesis, a new method for robustly reconstructing 3D shapes from a single depth image is developed, where various techniques such as selective latent code, self attention, cascaded architecture and multi-task learning are exploited to improve the model (see Chapter 3 for more details).

2.2.3 Point Cloud Completion

As directly reconstructed 3D point clouds from captured data are often incomplete due to occlusion and sensor range limit, some research works aim to produce complete 3D shapes given partial point clouds as input.

Zhang et al. [164] proposed to generate full 3D shapes in an unsupervised manner. Given a pre-trained GAN for complete shape generation, the method tries to optimise the latent code for the GAN such that it produces a complete shape that matches the partial input. To achieve this, the generated complete shape goes through a degradation function to retain partial points that match the input based on k-nearest neighbours, and both Chamfer Distance and Feature Distance are used to measure the differences between the degraded and the input shapes, which in turn optimises the latent code through gradient descent. The method can achieve similar performance as supervised approaches. Hu et al. [56] leveraged a generator to complete shapes where the model renders multi-view depth images and pools across all outputs. Wang et al. [135] proposed to use a GAN model to reconstruct coarse shapes, followed by refinement to match the ground truth while Huang et al. [59] completed shapes implicitly by generating latent vectors of depth shapes. However, both Wang et al. [135], Huang et al. [59]suffer from geometric inconsistency. Wen et al. [140] addressed the issue by adding folding-block and skip attention where the features' locations are matched against the input.

In the work [104] they implemented parallel models for complete and incomplete shapes where the models share weights during training to preserve geometric con-

sistency. However, the models may not work well for unseen objects. ForkNet [137] addresses this issue, and the model consists of three parallel generators with shared latent features. Two branches reconstruct the SDF (Signed Distance Field) representation and complete the surface respectively, while the third branch concatenates features from both previous reconstruction branches to semantically complete the volume scene. Alliegro et al. [15] develop a contrastive model, where they utilise pretrained encoders to capture semantic information and geometry features. The model naturally completes the missing parts.

Zhang et al. [169] claim existing point completion techniques cannot reconstruct fine details on objects due to the heavy task on a single model. They divide the task to three stages starting from completing the shape with a dense point cloud yet with less accurate locations.

They then trim and refine the point cloud in other stages based on the symmetric information. However, the model only works well when the shapes are symmetric which is not always the case. on the other hand, Wen et al. [139] suggest another refinement model in a cycle manner, where the learning network learns to complete the incomplete input, while also trying to make the target shape incomplete. They believe learning the other direction at the same time yields important features. Wen et al. [141] proposed to deform the point cloud points gradually, where they use gated recurrent units (GRUs) to capture previous point mapping and suggest new deformation.

Another work done by Cai et al. [22] proposed a model that aims to enhance point cloud completion in an unsupervised manner by learning a latent code for both partial and complete shapes. This is achieved by mapping the latent code for the occluded shape to the complete shape latent code. The model applies constraints to regularise the latent space. However, the method still has limitations for reconstructing fine-grained structures of objects such as those with complex geometric textures.

Yin et al. [159] also utilised latent code for completion; however, their work focuses on concatenating multi-scale shape features. The translator network in their method is

a generative adversarial network (GAN), which enables cross-domain translation and preserves the shape features for a natural shape. The network is trained in two steps: training the autoencoder to produce over-complete latent codes for the input shapes, followed by training the translator using the over-complete latent codes to perform cross-domain shape transformation.

Park et al. [105] also leveraged latent code. They suggested using an SDF (Signed Distance Function), where the input is a latent code concatenated with 3D point locations to elevate a high dimensional representation. At first, the model optimises the weights and the latent code to generate plausible SDF values while during inference, the model optimises latent code to generate an appropriate SDF. Instead of representing a shape's surface as a mere boundary or occupancy field, DeepSDF depicts it as a continuous volumetric field. In this representation, the value at any given point indicates its distance to the closest surface point, while its sign determines whether the point lies inside or outside the shape. A standout feature of DeepSDF compared to previous models is its ability to represent a wide range of shapes while maintaining a compact model size. It also excels at handling complex structures, fully enclosed surfaces, and provides accurate surface normals for a given shape. Chen et al. [25] also presented an innovative method called the 3-Pole Signed Distance Function (3PSDF) for learning surfaces that have varying topologies. The paper demonstrates that while the 3PSDF offers more robust results than the 2-way counterpart, it entails a more complex learning curve necessitating richer feature inputs and an extended training duration. Nonetheless, a potential challenge with the 3PSDF is its reliance on the synchronisation of results from two branches; any misalignment could lead to undesired artefacts or holes in the surface.

The studies [32, 88] delve into the comparison of two prevalent learning approaches tailored for 3D shape reconstruction. Both papers emphasise the inherent limitations faced by the current implicit methodologies, especially when dealing with entities that defy simple classification, such as the human form with its varied shapes and articula-

tions. Another significant observation shared by these papers pertains to the drawbacks of using a vectorised latent representation. This format lacks the nuances of 3D structure, culminating in results that seem more skewed towards prototype classifications than offering a smooth regression.

On a somewhat divergent note, Mittal et al. [95] introduced a method that focuses on generating 3D shapes from textual descriptions. Specifically, the model exhibits the capability to generate shapes that not only resonate with the input descriptions but also maintain a consistent global structure even when the details provided are fragmentary. Furthermore, in the field of 3D super-resolution tasks, the proposed methodology stands out by consistently reconstructing intricate details and outpacing other established methods.

Lastly, Stutz et al. [129] offer insights into 3D shape reconstruction and generation, specifically using implicit functions in the feature space. The findings from this paper bear some resemblance to [95]. Both models demonstrate a promising ability to generate realistic 3D shapes that align with the given descriptions, ensuring consistency and coherence even with partial inputs. In benchmark tests, the methodology exhibits its superiority over other learning-based baselines by offering detailed reconstructions. But, as with most models, its efficacy diminishes when it encounters shapes that are significant outliers from its training set.

2.2.3.1 Transformer-based Methods

Recent methods for 3D shape reconstruction and point cloud completion have leveraged transformer-based models to enhance accuracy and detail in generated shapes. Liu et al. [89] introduced a novel approach combining transformers with a strategy that fuses neighbourhood features, employing both global feature selection and local k-nearest neighbour techniques. Their method is distinguished by a genetic hierarchical point generation module that iteratively refines point structures through a dynamic transformer, emphasising the inheritance of shape features.

Lin et al. [82] proposed a framework utilising a *Point Cloud Transformer*, incorporating an encoder-decoder architecture to first learn spatial features from partial point clouds and then reconstruct the full 3D shape. Their model, enhanced with a multi-head attention mechanism, excels in generating detailed, high-resolution shapes, showcasing the potential of transformers in managing complex spatial data.

Wen et al. [142] presented the *PMP-Net++*, a model aimed at not only completing shapes but also improving point cloud resolution. Inspired by the Earth Mover Distance concept, their network predicts a distinct moving path for each point, optimising the overall structure through a feature-enhancing transformer that includes an innovative Recurrent Path Aggregation (RPA) component for merging current and past feature data, ensuring high-quality shape completion.

Lastly, Zhang et al. [166] advanced the field with their *Skeleton-Detail Transformer network*, adept at leveraging both local and global features. Their approach integrates cross-attention and self-attention layers to effectively correlate local patterns with the overall shape structure, alongside a selective attention mechanism designed to balance memory efficiency with computational performance. This model exemplifies the cutting-edge techniques in utilising attention mechanisms to refine point cloud completion.

Collectively, these studies underscore a paradigm shift towards integrating transformer models with traditional geometric processing techniques, aiming to reconcile local detail enhancement with global shape comprehension. The progressive evolution from [89]'s genetic hierarchical approach to [166]'s selective attention mechanism reflects a concerted effort to optimise transformer-based models for the nuanced demands of 3D shape reconstruction, marking a significant stride in the pursuit of more accurate and detailed digital representations of complex shapes.

Our work focuses on 3D reconstruction from depth images as input, which unavoidably involves completing a large part of the surface due to (self-)occlusion. Although the problem setting is different from general point cloud completion, some ideas of these

methods can be useful.

2.2.4 Neural Radiance Field based Methods

In recent literature, there has been a concerted effort toward enhancing 3D computer vision capabilities, particularly in the manipulation, reconstruction, and synthesis of 3D objects from limited data inputs. A review of the emerging research indicates a strategic move toward the use of neural rendering techniques, especially Neural Radiance Fields (NeRFs).

The framework introduced in [87] represents a step forward in enabling the flexible change of camera viewpoints from a single RGB image. By employing a conditional diffusion model that utilises geometric priors and is trained on a synthetic dataset, this approach facilitates the generation of new images under specified camera transformations. Remarkably, despite its reliance on synthetic data for training, the model exhibits strong generalisation capabilities across both out-of-distribution datasets and real-world images.

In addressing the challenge of generating novel viewpoints from sparse inputs, the work [99] proposes a novel optimisation process for neural radiance fields. This method focuses on regularising the geometry and appearance of rendered scenes, demonstrating an improvement over existing techniques. The approach is particularly noteworthy for its methodological clarity in addressing sparse input scenarios, presenting a viable solution for enhancing scene geometry and appearance.

Building on the concept of neural scene representation from a single image, the work [117] presents a method that leverages a geometric scaffold to guide the reconstruction of the radiance field. This approach distinguishes itself by its ability to disentangle shape and appearance effectively, enabling the rendering of new views with geometric consistency. This technique demonstrates adaptability to images beyond the training domain, including realistic renderings and actual photographs.

Further exploration into object generation as locally defined NeRFs is presented in [131]. This study introduces the concept of augmenting objects with affine transformations to facilitate part-based editing operations. By enforcing a hard assignment of rays to parts, the model ensures that modifications to one part do not impact the appearance of others, enhancing the editability of 3D objects without compromising fidelity.

Lastly, Yuan et al. [160] delved into NeRF geometry editing through a method that extracts a triangle mesh representation, which can be modified using traditional 3D deformation algorithms. This approach extends edits from the mesh to the volume, maintaining a mapping between ray queries in the deformed and original NeRF. The introduction of box abstractions and semantic labels further refines the editing process, providing users with intuitive and meaningful interaction mechanisms.

2.3 Evaluation Metrics

The task of assessing the visual quality of shapes has grown in importance as 3D models find use in diverse contexts. The process of reconstruction or completion of shapes can affect the quality, making it crucial to quantify this effect. Traditionally, subjective assessments by human observers have been the benchmark. However, this approach can be resource-intensive and time-consuming. Consequently, objective visual quality assessments have emerged, employing automated metrics designed to simulate a human visual judgement. These approaches can be classified into two categories: Full-Reference (FR) methods where a reference object is available as the ground truth and No-Reference (NR) methods where there are no object to compare to.

2.3.1 Full-Reference Methods

As mentioned before, most evaluation metrics are designed for specific representations, such as point clouds, meshes, etc. Evaluation metrics for point cloud completion tasks

involve both the ground truth complete point cloud and the generated point cloud to measure their similarity. This typically involves measuring the distance between each point in the two point clouds, and often the accuracy of the completion is judged on the amount of points shared in the two point clouds.

Earth-mover distance (EMD) is the measure of the dissimilarity between two sets of data. It can be used to compare images, text documents, and even high-dimensional datasets. The metric measures the minimum amount of work needed to convert one set to the other set, where work is defined as the amount of "earth" that needs to be moved. This concept is flexible, with applications in areas like assessing point cloud similarity. While determining the EMD between two point clouds requires matching based on the Euclidean distance between points, this condition for a strict one-to-one match tends to be computationally taxing. Nevertheless, EMD consistently detects shifts in the distribution. As a result, the optimal solution for the transportation issue mainly relies on the overall pattern, ignoring local details [40].

The Chamfer Distance (CD) is a measurement used for assessing the similarity between two point clouds. It evaluates the degree of dissimilarity between two point sets by taking into account the distance between point in one set and its nearest neighbour in the other set. CD has been employed in numerous applications, including 3D point cloud registration [16, 75] and object recognition [58, 41, 86]. In addition, it requires low computation time and is highly robust to noise and outliers. However, as it is not invariant to scale, rotation, or translation, it is often computed after applying certain transformation methods. The Chamfer Distance offers a reliable means of quantifying the degree of difference between two point clouds.

When applied to 3D reconstruction, cross entropy is a popular evaluation metric for voxel-based representations. It measures the difference between two probability distributions, and is often used with models that predict occupancy in a voxel representation. Cross entropy is useful for comparing the accuracy of different deep learning models trained using different datasets and architectures for 3D reconstruction tasks. This

metric can also be used to evaluate the performance of 3D vision algorithms such as semantic segmentation or 3D object detection.

Lavoué et al. [71] developed a method to compare the quality of two 3D meshes, even if they are structured differently. They based their approach on a 2D image quality method called SSIM (Structural Similarity) and looked at curvature differences in local areas of the two meshes being compared. To make the method more efficient and reliable, they looked at these differences at multiple scales. Each scale focuses on a specific neighbourhood size used to calculate curvature. Wang et al. [134] also proposed a new objective metric for 3D shape quality assessment. The metric predicts the difference between a reference shape and a noisy mesh using a local roughness measure derived from Gaussian curvature. It accounts for visual masking and psychometric saturation effects. The global roughness is ascertained by taking normalised surface integrals of this local measure. The perceptual gap between the two meshes is subsequently assessed by determining the difference between these integrals.

Bian et al. [17] suggested a way to measure the minor visual changes between 3D mesh models that have a similar structure. It is based on the theory of strain field energy, which is used to describe the deformation of elastomer objects. The perceptual distance is defined as the weighted average strain energy (ASE) over all triangles, normalised by the total area of the triangular faces. This distance should be independent of both the size of the model and the number of triangles.

Abouelaziz et al. [12] proposed a metric for full reference mesh visual quality assessment. The proposed objective metric utilises the Kullback-Leibler (KL) divergence of dihedral angles extracted from a given statistical distribution to estimate perceptual distances between the reference and noisy meshes.

Nouri et al. [101] suggested a full-reference evalution metric for the quality assessment of 3D meshes which is viewpoint-independent. It relies on utilising a multi-scale visual saliency map to extract features of a 3D mesh. A roughness map is also used to capture the visual masking effect, and four comparison functions are used to capture

the structure's differences.

Chetouani et al. [28] proposed a 3D mesh quality measure based on the fusion of selected features, which are extracted from the original mesh and its distorted one. The values are used as inputs to a regression tool via a support vector machine for regression. The objective is to find a function that fits the target with a certain deviation and kernel function.

Nouri et al. [103] introduced a quality measurement method for 3D meshes based on visual importance. Each point on a 3D mesh is given a significance level. The method looks at structural attributes of the original and altered meshes. The key idea is that changing the visual importance of a 3D mesh reduces its visual quality. The method uses multiple levels of visual importance maps and a texture map to understand visual effects. To gather structural details of a 3D mesh, the average, variability, and relationship of importance levels are calculated.

Yildiz et al. [158] suggested using machine learning to assess the visual quality of 3D grids. They create a 28-element feature vector from geometric properties, which includes the average, spread, peakness, and asymmetry of each vertex-based property distribution. These properties encompass curvature, shape indicators, bend intensity, and surface unevenness. Finally, a modified Euclidean distance is employed as a metric, aiming to optimise the probability.

Ilyass et al. [62] introduced a measurable quality standard for assessing the visible quality of 3D grids. It utilises a pre-trained convolutional neural network (VGG-16 [128]) to derive features from the altered mesh and its original version. Indices from recognised mesh visual quality metrics are combined with these features, producing a comprehensive feature vector. This vector is then utilised in a support vector regression (SVR) to determine the final quality rating. The 3D structure is depicted as 2D snapshots, broken down into smaller sections which are standardised and provided to the VGG-16 [128] system.

Chetouani et al. [29] suggested a deep learning technique for estimating the quality of altered point clouds. This method involves extracting patches, calculating patch-based distances, and predicting patch quality using a CNN model. Features are derived from randomly chosen patches from both the original and modified point clouds and are used to train a CNN model. Patch Quality Indexes (PQIs) are determined from both perspectives: from the original to the altered and the other way around. These are then averaged or combined to produce the Global Quality Index (GQI).

Lin et al. [85] introduced an approach for objective quality evaluation of 3D meshes by leveraging curvature characteristics to gauge the visual discrepancies between the original and altered meshes. Both meshes have their Gaussian and mean curvatures computed, which are then connected using correlation methods to determine the correlation coefficient. A Support Vector Regression model unites these two features to produce a final quality rating.

The field of 3D shape quality assessment has seen various methodologies being proposed over the years. The work [12] advocated for a statistical model-based approach, emphasising its correlation capabilities. For similar objectives, leveraging the strain field theory, was also pursued by others like [17, 28]. While these studies have demonstrated promising correlations with human judgements, constraints in experimental models, especially around pronounced edges, were flagged as areas of caution [28, 101]. Comparative evaluations, as carried out by[62], further highlighted the need for comprehensive datasets and methodological clarity. Interestingly, works by [134, 158] both accentuated model limitations, urging careful consideration during assessments [134, 158].

Majority of existing methods are designed for specific shape representations. Although it is possible to convert other representations to the desired one, the conversion process could introduce additional loss of information. In this thesis, we developed a new, representation neutral full-reference method by rendering 3D shapes (both reference and distorted/reconstructed ones) to 2D images using various views and rendering styles,

motivated by the fact that perceptual quality of 3D shapes is largely based on when they are viewed by human subjects. We further develop a neural network based approach to combine image-based metrics to produce a final score more closely related to human perceptual quality (see Chapter 4 for more details).

2.3.2 No-Reference Methods

Some other methods consider the case where no reference is available.

Abouelaziz et al. [6] introduce a no-reference objective approach for assessing the visual quality of 3D meshes, working exclusively with noisy meshes. This method fuses features from a pre-trained convolutional neural network (specifically, VGG) with manually crafted features drawn from the 3D mesh, including curvature and the dihedral angle. These features are then depicted using Gamma statistical distributions. In the concluding step, a General Regression Neural Network (GRNN) is deployed to forecast the quality score.

In their work, Abouelaziz et al. [8] developed a no-reference technique that utilises a general regression neural network (GRNN) trained on mean curvature - a crucial perceptual attribute for depicting the visual of a 3D mesh. Notably, the method only operates on the altered mesh, bypassing the need for a reference mesh. Initially, the system isolates the curvature attribute for its perceptual relevance. Subsequently, it employs the GRNN to learn from this feature, aiming to accurately estimate the objective quality score in the latter phase.

Abouelaziz et al. [11] put forward a model tailored for blind 3D mesh visual quality evaluation. The system leans heavily on feature learning, specifically focusing on statistics related to dihedral angles. Additionally, support vector regression (SVR) is used in tandem with key functionalities of the human visual system (HVS), emphasising aspects like visual masking and saturation effects. Drawing from these extracted features, the proposed approach is trained and subsequently forecasts the quality score.

Nouri et al. [102] proposed a view-independent 3D Blind Mesh Quality Assessment Index (BMQI) to assess the visual quality of 3D distorted meshes without the need of a reference content. To do this, both visual saliency and roughness maps are used to quantify the structural deformation. A feature vector is constructed for each superfacet with 4 attributes: saliency and roughness at each vertex, and the sum of saliency and roughness inside each superfacet.

Zhang et al. [171] introduced a no-reference quality evaluation metric tailored for colored 3D models, which can be represented as either point clouds or meshes. The approach leans on 3D natural scene statistics (3D-NSS) and entropy to isolate features sensitive to quality. These identified features are subsequently translated into visual quality ratings via support vector regression (SVR).

Abouelaziz et al. [9] proposed a no-reference method to predict the perceived mesh quality without reference or knowledge of distortion type. The method involves extracting dihedral angles as surface roughness indexes, applying visual masking modulation based on characteristics of human visual system, fitting dihedral angles using the Gamma model, and using support vector regression to predict the quality score.

In their study, Abouelaziz and associates [7] presented a method tailored for assessing the visual quality of meshes without needing a reference, harnessing the power of deep learning. The initial step involves computing mesh saliency, followed by rendering views from the 3D model. These views are segmented into patches, which are then processed based on a saliency threshold. For feature extraction, the method taps into three renowned pre-trained deep convolutional neural networks: VGG [128], AlexNet [67], and ResNet [54]. The features derived from these networks are integrated into a comprehensive feature depiction through Compact Multi-linear Pooling (CMP). The concluding phase employs a regression component to determine the quality score.

In their work, Abouelaziz et al. [5] introduce a no-reference convolutional neural network (CNN) framework, dubbed SCNN-BMQA, tailored for blind mesh quality assessment. This approach harnesses a CNN in tandem with 3D visual saliency to gauge

the visual quality of altered 3D meshes. The procedure involves generating 2D projections from the 3D mesh and its allied 3D saliency map. Subsequently, these projections are segmented into petite patches, which are sifted based on their saliency magnitude. The curated patches then serve as the input for the CNN model, which predicts a similarity rating.

Abouelaziz et al. [10] put forth a approach for blind mesh visual quality evaluation leveraging deep learning. Key features such as mean curvature and dihedral angles are derived from the deformed meshes. These features are then transformed into 2D patches and channeled to a convolutional neural network (CNN) for learning. Subsequent to this, a multilayer perceptron (MLP) is introduced to condense the derived representation into a singular output node. This structure enables the prediction of the quality score without the necessity of a reference mesh.

Abouelaziz et al. [3] proposed a CNN framework to predict the quality of 3D meshes without having access to the reference. The 3D mesh saliency is used to obtain 2D projections, which are split into small patches. The relevant ones are selected with a fixed threshold. These selected patches are then fed to the network and the quality score is given by averaging the scores over the patches.

Abouelaziz et al. [2] put forward an approach for blind mesh visual quality evaluation harnessing a graph convolutional network. In this approach, 3D mesh data is transmuted into a graph form, utilising the adjacency matrix and manually crafted features as inputs for the network. The culmination of the network's process, the max-pooling layer, delivers the definitive feature representation. This is then fed to a Softmax layer, which discerns and predicts the quality score category, all without necessitating a reference mesh.

Lin et al. [84] presented a new approach for no-reference 3D mesh quality assessment that analyses concave, convex and structural features. Shape index, curvedness, vertex scatter and the distribution of topology area are extracted to construct a feature vector. Random forest regression is then used to estimate a quality from the feature space to

quality space.

Lin et al.[83] introduced a new Blind Mesh Quality Assessment technique, leveraging both Graph Spectral Entropy and Spatial attributes. The signal from Gaussian curvature is transformed into the graph spectral domain. Within this domain, features indicative of smoothness and information entropy are extracted to assess distortion levels. Additionally, the method gleans both convex/concave and structural attributes. These extracted features are then amalgamated and trained using random forest regression, culminating in the creation of a model capable of predicting quality.

In [30], a two-step procedure is used to evaluate point cloud (PC) quality without a reference. Local patches are first extracted with geometric distance, local curvature and luminance values, and then a deep neural network learns a mapping to the extracted features from ground truth. The network characterises the PC through attributes like mean curvature, geometric distance and grey-level. Stacked patches form a new patch which is used by the CNN model to estimate the quality of the distorted PC.

Alcouffe et al. [13] presented blind mesh quality measures which can be used to assess the quality of a 3D reconstructed model. The metrics include a local roughness measure, which is the distance between a point and its best fitting plane, and a mean curvature measure, which is the least squares fitting of a quadric equation to a vertex's k-ring neighbourhood.

Abouelaziz et al. [4] introduced an objective, blind technique to evaluate the visual quality of 3D meshes. Central to this method is the use of pre-trained deep convolutional neural networks, with the quality assessment relying solely on the information extracted from the altered mesh. To gather this data, 2D visualisations of the 3D mesh and its aligned saliency map are produced. These renderings are then segmented into uniformly-sized, salient patches. These curated patches serve as inputs to three distinct pre-trained deep convolutional neural networks, namely VGG [128], AlexNet [67], and ResNet [54]. Following fine-tuning, each network independently computes a quality score. These individual scores are then amalgamated through a weighted sum, result-

ing in the final comprehensive quality score.

Abouelaziz et al. [1] introduced a deep learning approach grounded in graph theory for assessing the visual quality of meshes. The given distorted mesh undergoes a transformation into a graph, characterised by its adjacency matrix. From this representation, a compilation of geometric and perceptual attributes is derived and catalogued within a feature matrix. This matrix is then channelled into a graph convolutional network (GCN) structured with two convolutional strata and a max-pooling layer. Leveraging a softmax-based classifier, the system predicts the quality based on a node classification challenge. Five predetermined categories, mirroring the validated ground truth scores, are utilised for classification: very bad, bad, medium, good, and excellent quality.

No-reference methods are generally more challenging for reliable quality measure compared with full-reference scenarios. Since our thesis aims to use objective quality measure for evaluating 3D reconstruction where ground truth is available, we focus on full-reference methods.

2.4 Canonical forms

Many tasks including 3D reconstruction and shape retrieval benefit from putting deformable shapes into some standardised poses (such as T-pose for human bodies), which are referred to as canonical forms. For example, shape retrieval is an important task that aims to find similar shapes to the query. Many methods work well on rigid bodies where all shapes have fixed pose. However, these methods may work poorly on non-rigid shapes, where the same shape can have different poses. Without a standardised pose (canonical form), determining correspondence between points on two non-rigid shapes can be ambiguous, as the geometric distances caused by pose difference are often much larger than those of different instances. Also, Machine learning algorithms, especially those based on deep learning, require consistent data representation for effective training, such as learning-based 3D reconstruction. Different poses

can be seen as "noise" or "variations" that can affect the learning process if not standardised through canonical forms. To solve that, a canonical forms standardises the shapes to a fixed pose. In this section we will review canonical form for non-rigid shapes techniques.

Lian et al. [80] presented a feature-preserved canonical form for non-rigid 3D watertight meshes. The idea is to naturally deform the original models against corresponding initial canonical forms calculated by Multidimensional Scaling (MDS). Objects are segmented into near-rigid subparts, and then, original subparts are transformed via rotations and translations to poses corresponding to their MDS canonical forms. Optimal alignments and boundaries between subparts are obtained by solving nonlinear minimisation problems.

Pickup et al. [111] introduced a method to compute a canonical form with linear time complexity. This technique leverages Euclidean distances between pairs from a select subset of vertices. Notably, its accuracy parallels methods using global geodesic distances, yet it operates faster, facilitating the processing of higher-resolution meshes or more meshes within a specified time frame. The vertex subset selection hinges on their conformal factors, which amplify along the mesh protrusions. While determining the distances, the method aims to maximise the distances between the selected vertices and concurrently endeavors to maintain the original edge lengths of the mesh.

Lian et al. [79] presented an image-based method to address the 3D shape matching problem. A canonical form is calculated for each object using MDS (multi-dimensional scaling) and PCA (principal component analysis), and represented by 66 depth-buffer images captured on the vertices of a unit geodesic sphere. Each image is described as a word histogram obtained by vector quantisation of the image's salient local features, and a multi-view shape matching scheme is used to measure the dissimilarity between two models.

In [136], an intrinsic embedding technique, called the contour canonical form, is presented to express an isometry-invariant shape representation. Feature points are located

on the shape surface and their canonical mapping positions are calculated, which are globally optimised under geodesic constraints. Geodesic contours around each feature point are decomposed and placed at new positions, resulting in the contour canonical form.

Pickup et al. [112] discussed a method for estimating the canonical pose of an object. Starting with the extraction of the mesh's skeleton, the method is about contracting the mesh to a zero-volume skeletal shape using Laplacian smoothing, converting it to a 1D curve skeleton, refining it by merging junctions and repositioning the joints for better centering. Next, they compute geodesic distances between all the joints and then perform multidimensional scaling. Finally, they deform the mesh so that it matches the transformation of its assigned bone from the original to the canonical skeletons, while preserving the mesh's connectivity.

Zeng et al. [162] This paper proposes a novel multi-feature fusion method for non-rigid 3D model retrieval. It begins with computing the canonical form and generating projective depth images. Multiple pooling fusion methods are then used in the multi-view convolutional neural network to reduce information loss and extract more effective multi-view features, while wave kernel signature is computed to construct the multi-energy shape distribution and 3D shape feature. Finally, kernel canonical correlation analysis is used to fuse the multi-view feature and 3D shape feature.

In the work by Jribi et al. [65], the paper presented a novel approach for the extraction of a canonical form of 3D objects with different, non-rigid inelastic deformations. This is accomplished by defining each point on the two dimensional differential manifold by the length of the geodesic curves between it and three reference points. The corresponding novel points are then defined to have the same Euclidean distances as the original geodesic distances. The extracted canonical form can then be used for comparison and recognition purposes.

Haj et al. [51] presented a 3D non-rigid shape retrieval method based on canonical shape analysis. It transforms the problem of non-rigid shape retrieval into a rigid one

using multi-dimensional scaling and random walks on graphs. The local commute time distance is used to preserve shape details by segmenting the non-rigid shape into local partitions. A global constrained problem is formulated with biharmonic functions between local salient features. This produces canonical forms that are invariant to shape poses, which can then be treated as rigid shapes and used for non-rigid object retrieval.

Haj et al. [50] proposed to leverage the feature space for acquiring a condensed representation of points within a limited-dimensional Euclidean domain. The deformation of the mesh is directed by the local weighted commute time. For crafting canonical forms that remain unchanged to the pose of 3D figures, the mesh is divided into localised zones through a Voronoi diagram. Following this, geodesic distances are determined using the heat methodology. To culminate, a stipulation is interposed between varying partitions, facilitating the merger of local canonical forms to yield the ultimate canonical configuration.

To summarise, canonical forms can be an important component to handle non-rigid shapes in various tasks. Existing methods are largely based on hand-crafted features and traditional methods, which have limited capabilities. In this thesis, we consider developing a deep learning method to standadise 3D deformable shapes (see Chapter 5, which provides a basis for allowing 3D reconstruction methods to be generalised to effectively process deformable objects.

Chapter 3

3D Reconstruction from Single Depth Images

3.1 Introduction

Many tasks of modern technology, such as robotic vision and obstacle avoidance, rely heavily on 3D reconstruction for which depth images are a common source of data. Until recently, capturing depth information was challenging, but with the availability of low-cost depth cameras, depth images can now be quite easily obtained, allowing datasets to be created [63] that make possible novel applications such as virtual reality (VR) [76]. However, estimating the full 3D shape from a depth image, which only represents one viewpoint, is still challenging. Since a depth image only contains partial information about the shape due to unavoidable self-occlusion, a single depth image may not be sufficiently descriptive to fully reconstruct a shape, causing holes and spurious surfaces in the reconstruction. Ideally a system should be able to cope with such difficult or unusual viewpoints. The alternative, capturing sufficient depth maps to form complete 3D data, is not feasible for many real-world applications due to the increase in cost and time. For example, in indoor scene modelling, capturing complete furniture would be near-impossible due to substantial occlusion, and even capturing multiple depth images adds complexity to the problem requiring registration of depth images.

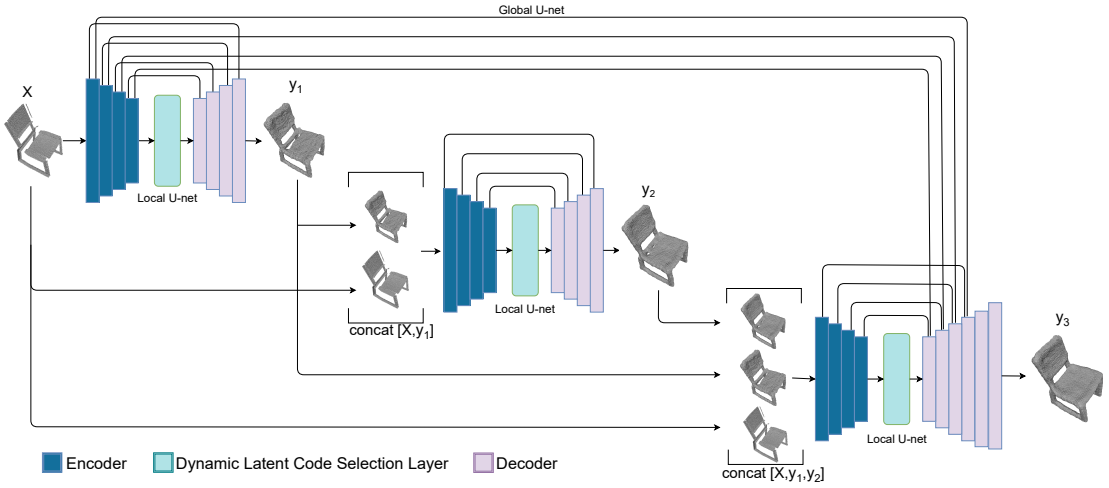


Figure 3.1: The generator turns an input volume from a depth image to a high-resolution 3D volumetric output.

Our work in this chapter focuses on reconstructing a 3D shape from a single depth image using a 3D convolution neural network (CNN). The CNN approach shows impressive results compared to other non-learning-based models [133, 31, 100] where the bounding ray cone or voxel hashing are used. Non-learning models usually require multiple viewpoints of the shape, while the learning models can learn from existing full shapes to reconstruct complete shapes from single depth images [52, 154], or single RGB images [143, 167, 144].

In this chapter, we present a model capable of producing a complete shape from a single depth image. Given a 2.5D depth image as input, the model can learn to reconstruct a high resolution shape. As shown in Figures 3.1 and 3.2, an end-to-end learning model containing a sequence of multiple encoder-decoders with global and local skip links is trained to complete the volumetric shape, where the later stages take both the input and outputs from previous stages to further improve completion. We also introduce a self-attention layer that helps refine the 3D shapes, mimicking the human ability to focus on a region of interest in the volumetric space. In addition, if a 3D shape is missing certain features (e.g., due to occlusion), self-attention aids in improving its details by exploiting clues from non-local regions. Such non-local information is

useful as only partial single-view depth is given. For example, the geometry of one table leg gives a useful clue for reconstructing the other table legs. We further introduce a dynamic latent space where the model has the ability to select only relevant codes to estimate 3D shapes. As we will later demonstrate, this strategy provides a strong sparse regularisation that improves the robustness. Furthermore, we extend the shape completion to a multi-task setting, where the generated shape is further classified into one of the object categories, as shown in Figure 3.3. As properly completed shapes are easier to classify, these two tasks help with each other, contributing to improved shape completion results.

Our contributions are:

- We propose a cascade architecture consisting of multiple encoder-decoder blocks with additional skip links, which provides better 3D reconstruction than a single encoder-decoder.
- We incorporate a self-attention layer to refine the 3D shapes, mimicking human ability to focus on a region of interest in the volumetric space.
- We introduce a dynamic latent space where the model has the ability to select only relevant latent codes to estimate 3D shape. This provides a strong sparse regularisation that enhances the robustness of the network.
- A classifier network is introduced as an auxiliary task to provide additional guidance to the reconstruction model.

Extensive experiments show that our method outperforms state-of-the-art methods.

3.2 Methodology

The model addresses the problem of reconstructing a 3D shape from a single depth image where the 3D space is voxelised. The voxel representation provides flexibility for topological change, which is required when turning the depth image into a complete

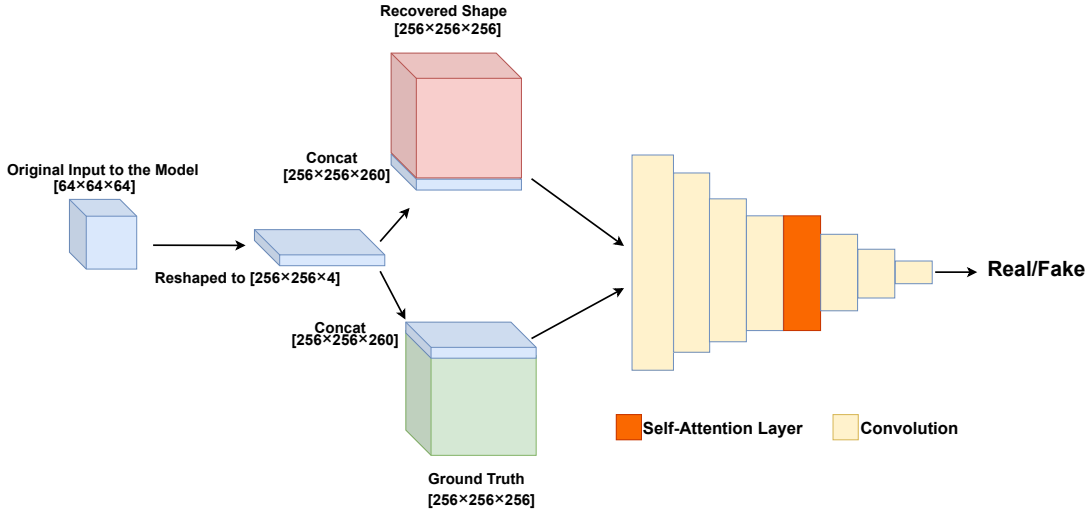


Figure 3.2: The discriminator takes the concatenation of the original single view volume and either the ground truth or the reconstructed shape as its input. We also introduce a 3D self-attention layer to the discriminator to improve the generated shape.

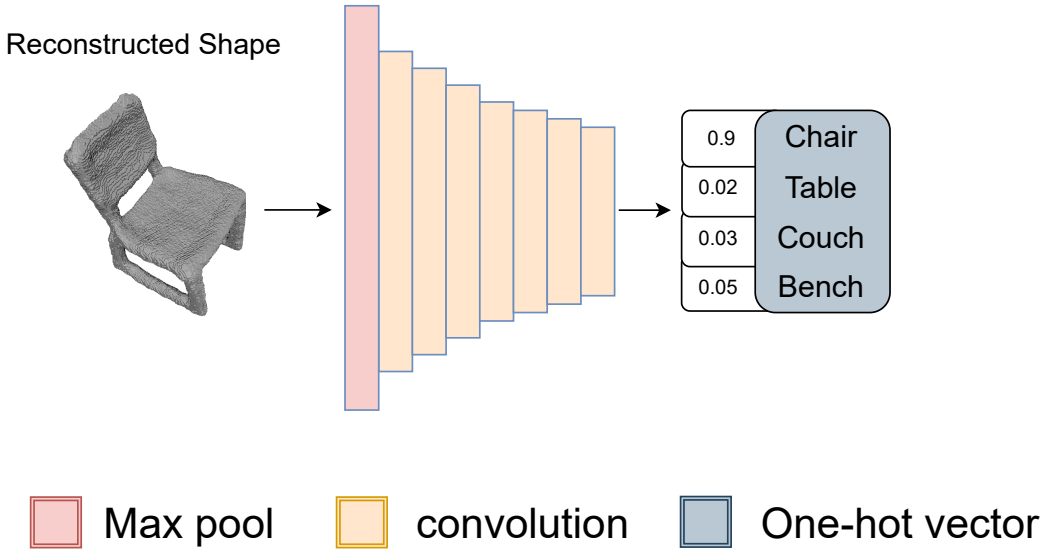


Figure 3.3: The classifier that classifies the type of the shape helps the generator to produce shapes with proper structure and details to improve the chance of correct classification.

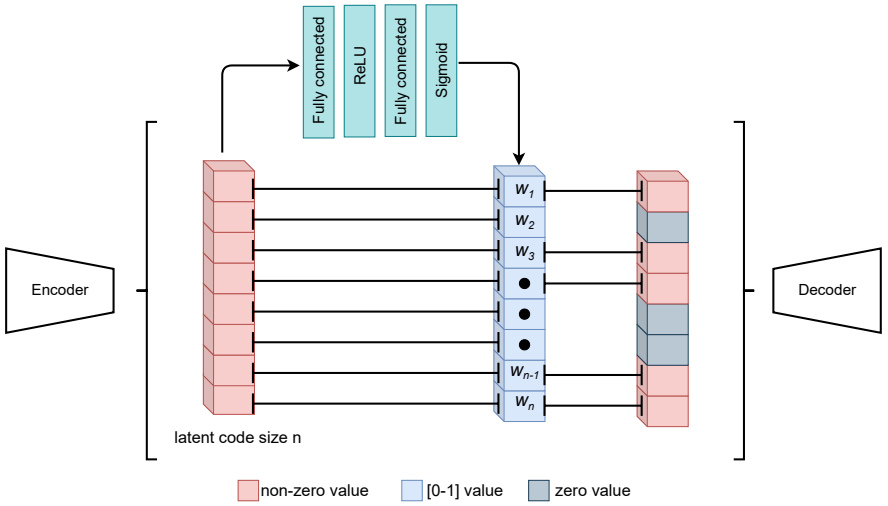


Figure 3.4: The n-dimensional latent code is first processed by two fully connected layers to predict an n-dimensional weight vector. Then the top K codes are selected according to the weight vector and values in the remaining dimensions are set to zero, leading to a sparsified latent space.

3D shape. A cascade approach was adopted in which shape estimation was enhanced at each stage of the model. In addition, instead of passing the entire latent vector, we suggest a selection process to dynamically select appropriate latent codes. Furthermore, self-attention has the ability to find links between features; the self-attention layer works globally on the whole space while convolution works on the local region with the volume occupancy represented by 1 for occupied and 0 for unoccupied.

Our model takes 64^3 voxels representing the input depth image and reconstructs the 3D shape sampled to 256^3 voxels to retain more details.

3.2.1 Dynamic Latent Code Selection

In a typical encoder-decoder architecture, the latent space is fixed $l \in \mathbb{R}^n$, where n is the latent dimension. However, for a given shape, not all the latent dimensions are relevant. Responses from such irrelevant dimensions may have negative impact

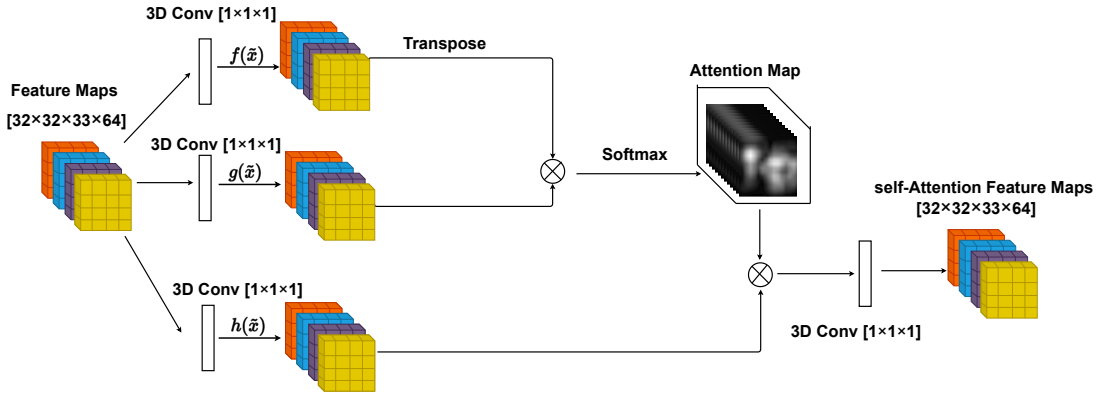


Figure 3.5: Network architecture of our 3D self-attention layer.

on the reconstruction quality. To address this, as shown in Figure 3.4, we introduce a selection process such that only selected latent dimensions are retained, with the remaining components in the latent code set to zero. Specifically, the model first learns to predict the weight for each latent dimension, collectively as a latent weight vector $w \in (0,1)^n$, denoted as $w = \omega(l)$, where $\omega(\cdot)$ is the weight prediction network, and in practice, it is achieved by passing the latent code l through two fully connected (FC) layers each with n units, and ReLU and sigmoid activation functions are used after the two FC layers respectively. This makes the output w to be in the range $(0,1)$ for each dimension. Then, we use the predicted weights to determine which latent components should be retained, namely, only those with the weights in the top K weights (where K is a hyper-parameter) are kept. Then the i-th component of the output latent code $\tilde{l}$ satisfies:

$$\tilde{l}_i = l_i \cdot \mathbf{1}(w_i \in W_K), \tag{3.1}$$

where $\mathbf{1}(\cdot)$ is 1 if the predicate is true, and 0 otherwise. W_K is the set containing the top K weights. This approach achieves two effects. On the one hand, by suppressing low-weight (i.e., recognised as unimportant) components, this avoids their negative impacts. On the other hand, the network strives to reconstruct high-quality complete 3D shapes with at most K latent components, essentially serving as a strong sparse regularisation, that helps improve the robustness of the network. Note that while selecting K latent components, we maintain their positions in the latent space, rather

than removing zero components. This makes the follow-up FC layers more efficient to learn.

3.2.2 3D Self-Attention Layer

Self-attention has been shown to be effective in the GAN framework for improving *image* generation [163] and due to the nature of the input (depth image), the shapes are missing significant information. A limitation of convolution is that it can capture only local features, and so convolution tends to distort the shapes while attempting to recover non-local features. From this prospective, we introduced a self-attention layer. The self-attention mechanism focuses attention on the most important global features, which helps to reduce distortion in the reconstruction. In the context of 3D shape reconstruction from depth images, the self-attention mechanism can analyse the input depth image as a set of local regions. By doing so, it identifies which features are most relevant for accurately reconstructing the 3D shape. This process involves computing attention scores that reflect the importance of each part of the depth image in relation to the rest.The self-attention GAN (SAGAN) incorporates a self-attention mechanism for both the generator and the discriminator. However, in our 3D reconstruction setting, self-attention can only be applied to feature maps with relatively lower resolution (e.g. around 16^3) since the relationships between any pair of locations need to be considered. As we will later show, incorporating such a 3D self-attention (3DSA) layer in the generator is unable to capture meaningful non-local relationships and actually leads to worse performance. We therefore only consider incorporating the 3DSA layer in the discriminator network.

$$\beta_{j,i} = \frac{\exp f(\tilde{x}_i)^T g(\tilde{x}_j)}{\sum_{i=1}^{\tilde{N}} \exp f(\tilde{x}_i)^T g(\tilde{x}_j)}, \tag{3.2}$$

which shows the contribution of the jth location from the feature map at the ith location, where $f(\tilde{x})$ and $g(\tilde{x})$ are two different $1 \times 1 \times 1$ convolutions. β is then used as

weights to combine feature maps $h(\tilde{x})$, also obtained through $1 \times 1 \times 1$ convolution, and then the final output of the 3DSA layer is obtained through another $1 \times 1 \times 1$ convolution $v(\cdot)$.

$$o_i = v\left(\sum_{i=1}^{N} \beta_{j,i} h(x_i)\right), h(x_i) = W_h x_i, v(x_i) = W_v x_i \tag{3.3}$$

Where o is the ouput of attention layer , $o = (o_1, o_2, ..., o_n)$ and γ is a hyper-parameter.

$$y_i = \gamma o_i + x_i \tag{3.4}$$

3.2.3 Network Architecture

3DCascade-GAN consists of two components: the generator and discriminator. Figures 3.1, 3.2 and 3.3 show the complete network architecture where Figure 3.1 is the multistage encoder-decoder (generator), Figure 3.3 is the classifier and Figure 3.2 is the discriminator.

Generator. The generator is multistage (three stages), and each stage is an identical encoder-decoder-like network (except the last stage where we add two up-sampling layers). The encoder contains four 3D CNN layers starting with an input that is 64^3 in size (the depth view of the shape); the kernel size for each layer of $4 \times 4 \times 4$, and $1 \times 1 \times 1$ strides. Each layer uses a leaky ReLU activation function, and after each convolution layer, a max pooling layer with a kernel size of $2 \times 2 \times 2$ follows $2 \times 2 \times 2$ strides; the size of the feature maps for each layer is 64, 128, 256 and 512, respectively, followed by a fully connected layer to map the higher abstraction of the shape and generate a 1000-dimensional latent code. Before the decoder runs, a selector layer processes the latent vector to select the top K codes, where K is set to 100 (for different K values, see the Dynamic Latent Code and the ablation sections). Another fully connected layer is then introduced which generates a 512-dimensional

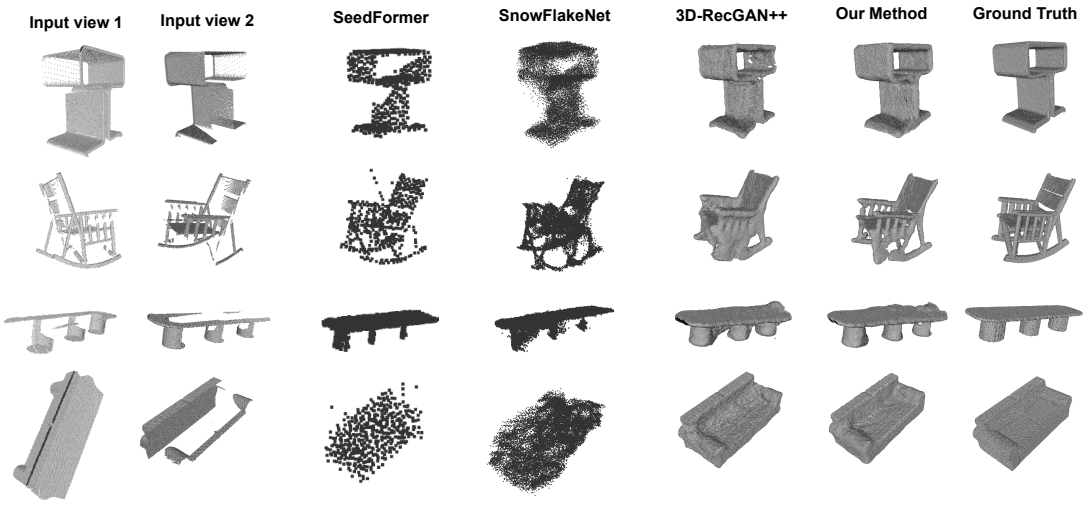


Figure 3.6: Visual comparison of completed single categories on same view samples.

feature map. The decoder consists of four layers of transpose convolution with each layer followed by a ReLU. Skip links are used between the encoder and decoder where feature maps are concatenated; skip links enhance the shape details, as the latent code appears to preserve the general structure of shape without any fine details. No max pooling is used in the decoder; however, a kernel size of $4 \times 4 \times 4$ and $2 \times 2 \times 2$ strides is used, and each layer is followed by a ReLU except for the last layer where we used sigmoid. Note, the third stage has extra up-sampling layers so as to reconstruct to 64^3.

We concatenate both the output and the original input at the feature channel to form $64^3 \times 2$, which will be the input for stage two. The process is also repeated for stage three, where the input is a concatenation of stage one and stage two and the input size is $64^3 \times 3$. We found that the model tends to rely heavily on stage three and two, and consequently the output at stage one was very fragmented and not useful. To address this issue, we added global skip links between the encoder in stage one and the decoder in stage three.

Discriminator. The discriminator is useful to ensure the completion of the partial input

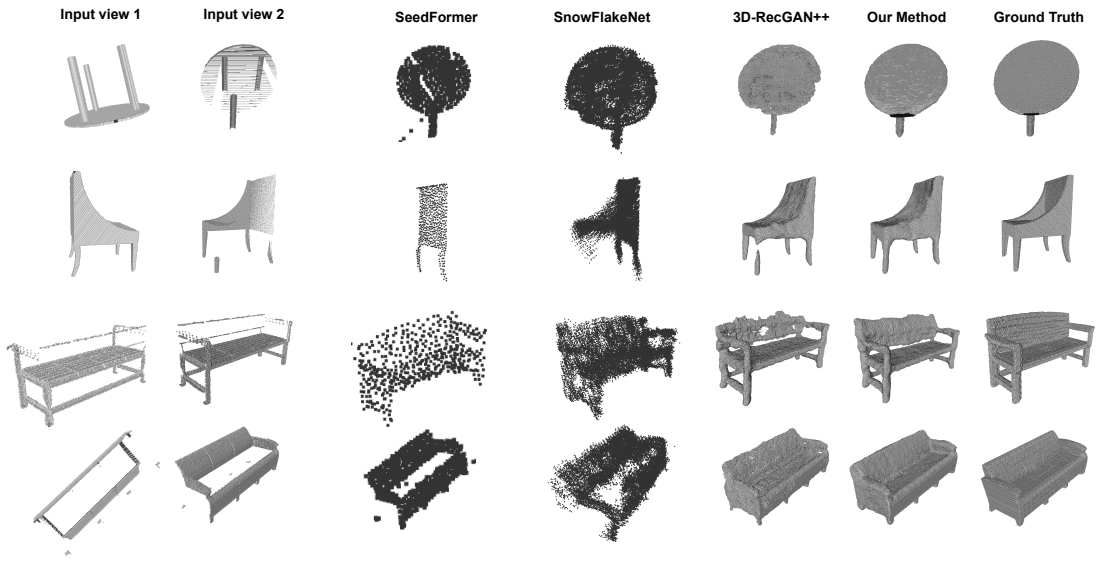


Figure 3.7: Visual comparison of completed Multi categories on same view samples.

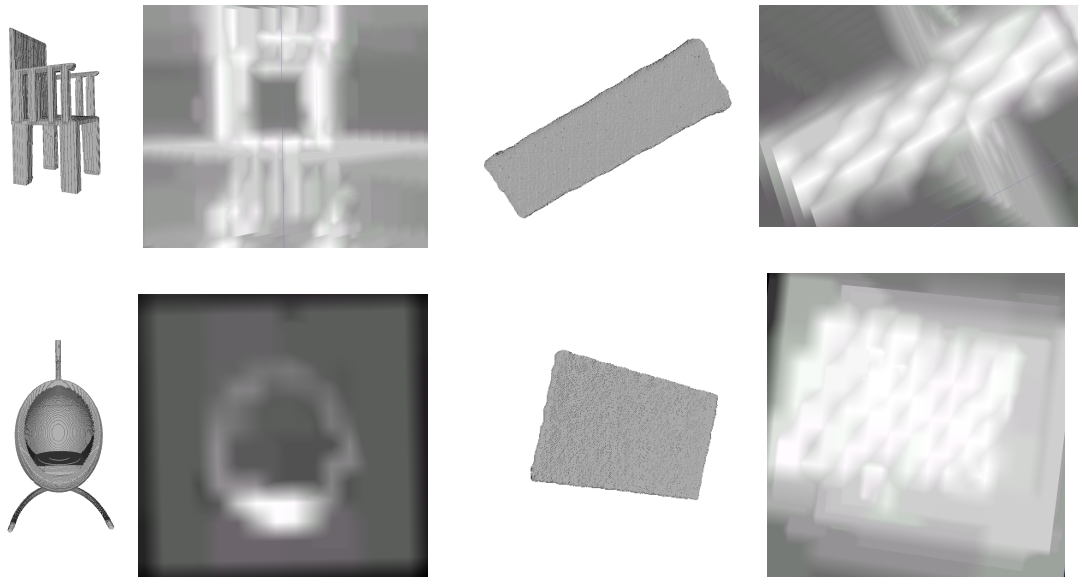


Figure 3.8: Visualisation of self-attention maps where the layer attends to features relating to shapes.

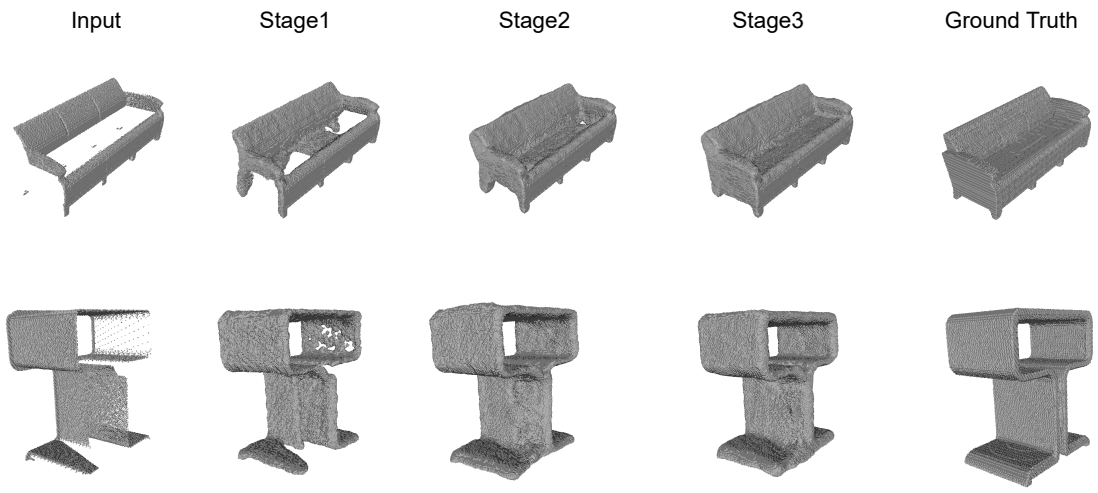


Figure 3.9: Visualisation of cascade stages.

shape. The input for the discriminator is either a fake pair (2.5D and the recovered shape) or a real pair (2.5D and ground truth). Again, the component contains seven 3D convolution layers. Each layer has a kernel size of $4 \times 4 \times 4$ and strides of $2 \times 2 \times 2$. At the end of each layer, a ReLU activation function is used; however, the last layer consists of a sigmoid to generate a semantic representation of the shapes. Finally, we applied the strategy of [154] by outputting the mean of a vector feature rather than a scalar in order to stabilise training because the discriminator cannot discriminate high dimension data (the input concatenated with either ground truth or the reconstructed shape) and the model usually collapses at an early stage. Our 3DSA layer is introduced to capture non-local relationships.

Classifier. The classifier network consists of 7 CNN layers each with kernel size of $4 \times 4 \times 4$ and $1 \times 1 \times 1$ strides. Each layer is followed by max pooling layers with kernel size of $2 \times 2 \times 2$ follows $2 \times 2 \times 2$. For the activation function, we use Leaky ReLU. The resulting output is reshaped to form a 4 element vector representing the categories {chair, bench, table, couch}, followed by a softmax layer to reconstruct the one-hot vector. It was not necessary to use the full 256^3 resolution as input to the classifier, and so we applied max pooling to reduce the input dimensions to 64^3.

3.2.4 Loss Function

The model has three loss functions: reconstruction loss, GAN loss and classifier loss, and the GAN has generator and discriminator losses.

Reconstruction Loss. As in [154], modified binary cross entropy (BCE) [18] is used rather than mean square error (MSE), to avoid a non-convex problem:

$$L_{BCE} = -\frac{1}{N}\sum_{i=1}^{N}[-\bar{y}_i \log(y_i) - \alpha(1-\bar{y}_i)\log(1-y_i)]. \tag{3.5}$$

When using the standard BCE equation the empty space will dominate the generated volume, which encourages the model to classify occupied grid cells as empty voxels, resulting in estimation errors. Thus, α is introduced in Eq. 3.5 to represent the cost weight of the terms. $\bar{y}_i$ represents the ith voxel in the ground truth and y_i represents the ith voxel in the reconstructed shape where N is the number of voxels in the space.

GAN Loss. L_G (Eq. 3.6) is the loss for generating fake shapes, while L_D (Eq. 3.7) is the discriminator loss used by WGAN-GP [47]. y represents the generated shape from input x (2.5D) and $\bar{y}$ is the ground truth for the complete shape. In order to tackle the vanishing gradient problem, WGAN-GP adds a penalty term (with weight λ) to encourage the gradient norm of the discriminator to be close to 1; $\hat{y}$ is a perturbed version of y.

$$L_G = -E[D(y|x)]. \tag{3.6}$$

$$L_D = E[D(y|x)] - E[D(\bar{y}|x)] + \lambda E[(\|\nabla_{\hat{y}} D(\hat{y}|x)\|_2 - 1)^2].$$

Classifier Loss. We use log loss. M represents the number of classes. y is a binary indicator for whether class label c is the correct classification for observation o. p is the predicted probability that observation o is of class c.

$$L_{Classifier} = -\sum_{c=1}^{M}[y_{o,c}\log(p_{o_{,c}})]. \tag{3.7}$$

Combined generator loss. As the generator has two objectives, a weight is applied to balance both losses during optimisation as follows:

$$L_{weighted} = \gamma L_{BCE} + (1-\gamma)L_G + \zeta L_{Classifier}. \tag{3.8}$$

$L_{weighted}$ is minimised when training the generator, and L_D is minimised when training the discriminator.

3.2.5 Experiments

3.2.5.1 Training Details

The model was trained for 20 epochs with a batch size of 3. [47] suggested a learning rate of 0.0001 for the generator and 0.00005 for the discriminator, but we increased the learning rate to 0.0001 for the discriminator, as the model showed better stability with our dynamic latent code and self-attention. For the optimizer, Adam [66] was used with $\beta_1 = 0.9$, $\beta_2 = 0.999$, and $\epsilon = 10^{-8}$. We set the WGAN-GP gradient penalty to $\lambda = 10$ and $\alpha = 0.35$ for modified binary cross entropy. Finally, we set the weighted loss parameter $\gamma = 0.8$ and $\zeta = 0.01$. The networks were trained on Nvidia GTX 1080ti and Nvidia P100, and it took on average 4.5 days to train a model.

3.2.5.2 Dataset

In our experiments, we used datasets provided by [154], for which the author had generated depth views from ShapeNet datasets. In total, 272 CAD models were used. For training 220 CAD used, testing 40 CAD used and validation 12 CAD used. All models in the dataset were voxelised to a 256^3 grid. Datasets were split into two sets: same view (all input depth images captured in one direction, 125 different views) and cross

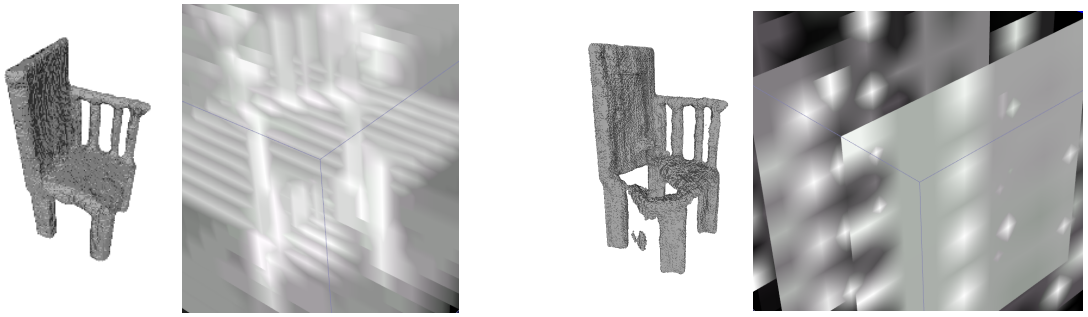


Figure 3.10: Comparison of applying self-attention to the discriminator (left) and generator (right). A more meaningful self-attention map and shape are obtained when incorporating self-attention in the discriminator.

view (depth images from multiple views, 216 different views).For training, only the same view depth images were generated,while for testing and validation both same view and cross view sets generated. In total, there are 26000 training samples. The same view test consists of 4500 samples and 8000 cross view test samples. The validation set contains 1500 samples for same view and 2500 for cross view. Four categories have training sets (chair, table, bench, couch) while the rest are used for testing as unseen objects (plane, car, monitor, faucet, guitar, firearm). All samples have been voxlized.

3.3 Evaluation

To compare our work with other state-of-the-art methods, we evaluated our model using intersection over union (IoU). IoU was applied on a per voxel basis to the ground truth and recovered shape. The second evaluation metric was mean value cross-entropy (CE). As discussed in [154], Chamfer distance and earth mover distance are infeasible for high-resolution voxel sets due to the high computational cost.

Comparison to prior work. To evaluate the performance of the model in reconstructing a 3D shape from a single-depth view, we compared it to three recent works on

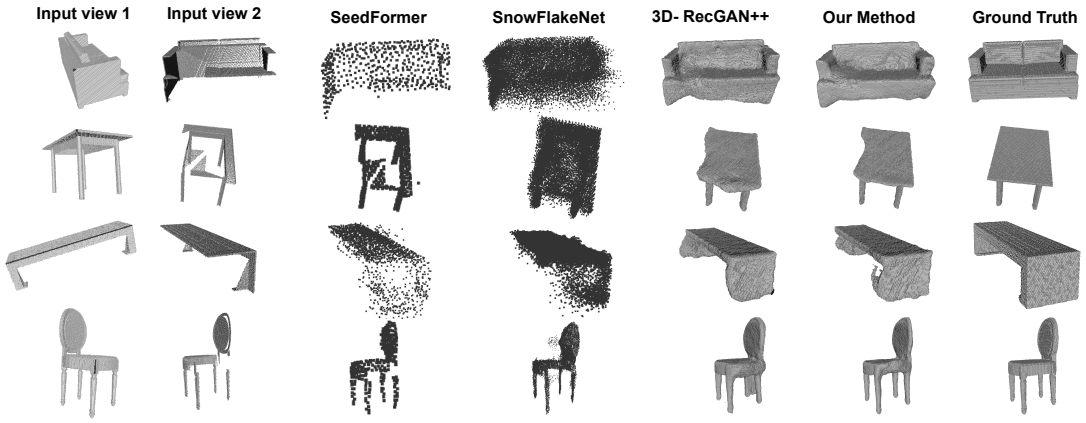


Figure 3.11: Qualitative results of single category reconstruction on testing datasets with cross viewing angles.

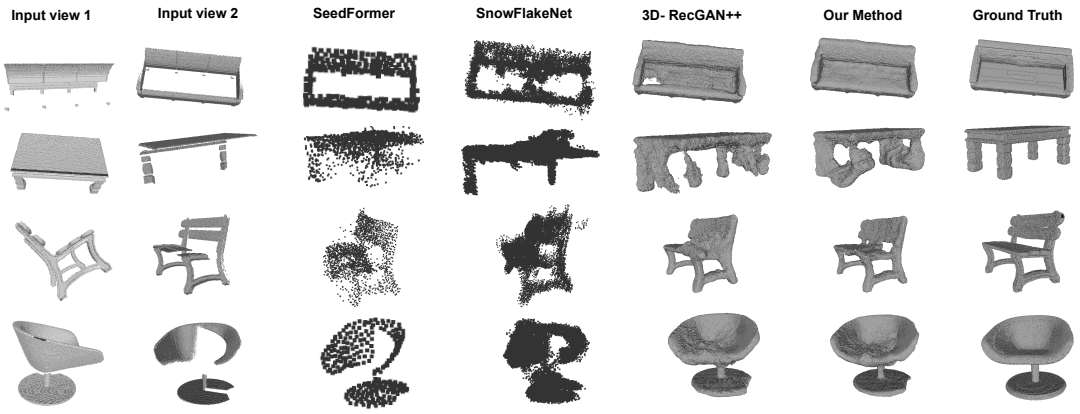


Figure 3.12: Qualitative results of Multi-categories reconstruction on testing datasets with cross viewing angles.

reconstructing a 3D shape from a single-depth image. (1) The 3D-EPN model presented by [34] completed the shape by leveraging semantic features; the resolution of the reconstructed shape was 32^3. The model then used a retrieval approach to collect similar shapes for shape reconstruction.

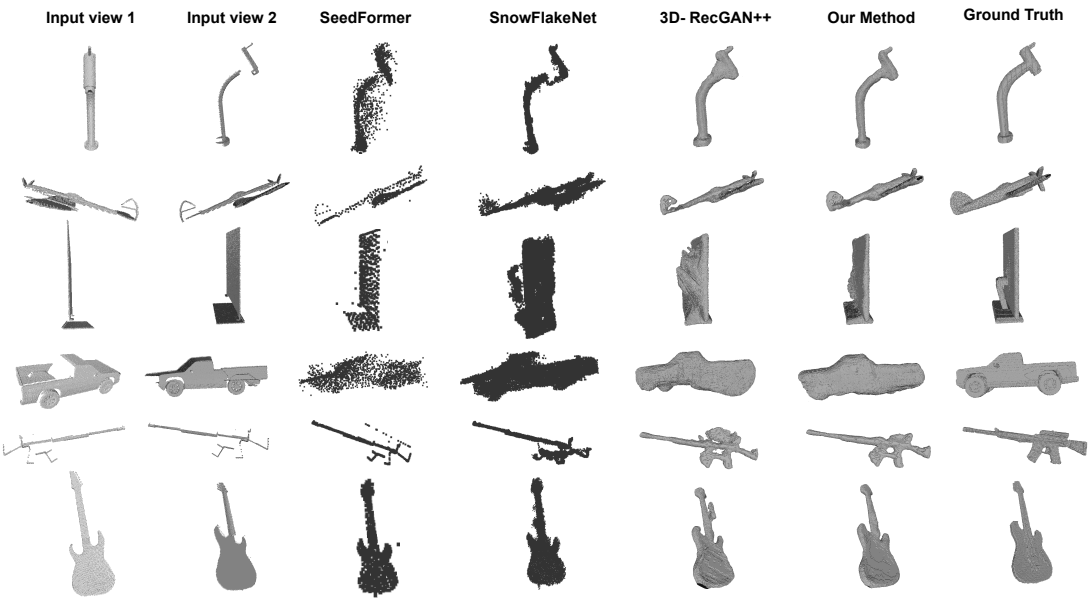


Figure 3.13: Qualitative results of Multi-categories reconstruction on testing datasets with same viewing angles.

(2) Varley [132] addressed the issue of robot grasp planning; the model reconstructed a 3D shape from 2.5D images that were captured using a depth camera. The model resolution was 40^3 voxels. (3) SnowflakesNet [149] processes a point cloud representation, and the model predicts a complete shape from an incomplete point cloud. We process the output by voxelising the output points to 256^3 resolution for quantitative comparison. (4) SeedFormer [173] also uses a point cloud representation where the input is an incomplete point cloud and the prediction is a complete shape. We process the output by voxelising the output points to 256^3 resolution for quantitative comparison. (5) 3D RecGAN++ [154] reconstructed a 3D shape from a 2.5D image with a resolution of 64^3 and up sampled to 256^3. For methods based on implicit representations, neither [105] or [44] provided the code for 3D completion, so we trained the model of [93] on our datasets, but it failed to learn the representation.

For the qualitative comparison, we show results of 3D RecGAN++ [154], SnowFlakeNet [148] and SeedFormer [173], as these models are state-of-the-art and have the same re-

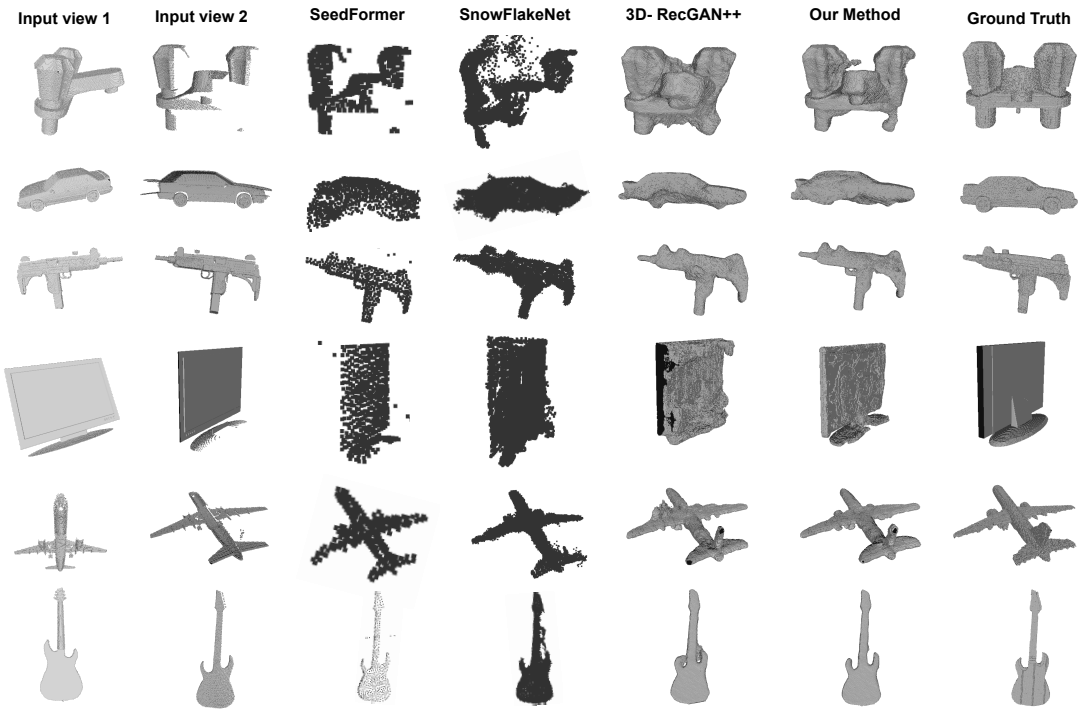


Figure 3.14: Qualitative results of Multi-categories reconstruction on testing datasets with cross viewing angles.

covered shape resolution as our model. Note, in the qualitative results for [149] and [173] we show point cloud representations to avoid the potential distortions caused by discretisation.

3.3.1 Results

Seen shape category experimental results. The model was trained on 4 different datasets (chair, table, bench, and couch). A single category means each one was trained separately with the same settings as mentioned. On the other hand, Multi-categories means the model was trained on all the 4 datasets (chair, table, bench, and couch). The IoU and CE results for single categories, same view are displayed in Table 3.1. Table 3.2 show IoU and CE results for Multi categories same view. Table 3.3 present

Table 3.1: IoU and Cross entropy evaluation metric for Single categories, same view, comparing 3D-EPN [34], Varley [132], SnowFlakeNet [149] , Seed-Former [173], 3D-RecGAN++ [154] (denoted as Yang in the table) and our 3DCascade-GAN.

IoU	Bench	Chair	Couch	Table
3D-EPN	0.423	0.488	0.631	0.508
Varley	0.227	0.317	0.544	0.233
SnowFlakeNet	0.562	0.631	0.745	0.659
SeedFormer	0.553	0.618	0.740	0.656
Yang	0.580	0.647	0.753	0.679
Ours	**0.641**	**0.701**	**0.809**	**0.698**
CE	Bench	Chair	Couch	Table
3D-EPN	0.087	0.105	0.144	0.101
Varley	0.111	0.157	0.195	0.191
SnowFlakeNet	0.037	0.063	0.068	0.043
SeedFormer	0.038	0.065	0.069	0.044
Yang	0.034	0.060	0.066	0.040
Ours	**0.030**	**0.053**	**0.063**	**0.038**

single categories cross view using IoU and CE respectively and Table 3.4 cross view for Multi categories. After training, we find the best threshold between [0.1, 0.9] with a step of 0.05 on a validation dataset using only the IoU criterion. After finding the best threshold to represent the model, we applied it on the test dataset as suggested by [154]. In the quantitative results, both IoU and CE demonstrated that our model outperformed the state-of-the-art model, and qualitatively it can be seen that our method recovered 3D shapes at high resolution with accurate details. For the qualitative results for single categories in same view testing datasets, see Figure 3.6, where artifacts appear in the results of 3D RecGAN++ such as incorrect structure/geometry and Multi

Table 3.2: IoU and Cross entropy evaluation metric for Multi categories, same view.

IoU	Bench	Chair	Couch	Table
3D-EPN	0.428	0.484	0.634	0.506
Varley [132]	0.234	0.317	0.543	0.236
SnowFlakeNet	0.548	0.624	0.736	0.633
SeedFormer	0.542	0.613	0.727	0.628
3D-RecGAN++	0.581	0.640	0.745	0.667
3DCascade-GAN	**0.624**	**0.669**	**0.773**	**0.682**
CE	Bench	Chair	Couch	Table
3D-EPN	0.087	0.107	0.138	0.102
Varley [132]	0.103	0.132	0.197	0.170
SnowFlakeNet	0.035	0.053	0.064	0.043
SeedFormer	0.036	0.054	0.066	0.045
3D-RecGAN++	0.030	0.051	0.063	0.039
3DCascade-GAN	**0.028**	**0.049**	**0.060**	**0.037**

categorises also in same view datasets in Figure 3.7. For single categories and cross view, Figures3.11. 3.12 show multi- category results in cross view datasets. Figure 3.8 visualises self-attention maps when completing some shapes, which clearly capture global structures. The intermediate results after each of the three stages are shown in Figure 3.9.

Unseen shape category experimental results. Lastly, we conduct experiments on six more categories where the model is trained on chair, bench, couch, table and then tested on car, faucet, firearm, guitar, monitor, plane for both same view and cross view datasets. The IoU and CE results for cross-view results are shown in Table 3.5 and same view results in Table 3.6. Figure 3.13 shows visualisation for the same view dataset and figure 3.14 shows cross view visualisation. Our method performs consistently better

Table 3.3: IoU and Cross entropy evaluation metric for Single categories, cross view .

IoU	Bench	Chair	Couch	Table
3D-EPN	0.408	0.446	0.572	0.482
Varley [132]	0.185	0.278	0.475	0.187
SnowFlakeNet	0.508	0.578	0.628	0.603
SeedFormer	0.503	0.563	0.627	0.601
3D-RecGAN++	0.531	0.594	0.646	0.618
3DCascade-GAN	**0.585**	**0.628**	**0.680**	**0.647**
CE	Bench	Chair	Couch	Table
3D-EPN	0.086	0.112	0.163	0.103
Varley [132]	0.108	0.171	0.210	0.186
SnowFlakeNet	0.045	0.079	0.118	0.055
SeedFormer	0.046	0.080	0.120	0.056
3D-RecGAN++	0.041	0.074	0.111	0.053
3DCascade-GAN	**0.038**	**0.070**	**0.109**	**0.051**

than state-of-the-art methods in all categories, and both same-view and cross-view cases.

3.3.2 Ablation Studies

In this section, we describe three ablation studies: dynamic latent code, second self-attention layer and classifier. For comparison, we choose the chair datasets for our ablation experiments as these samples show more complex structure compared to bench, table and couch.

Dynamic latent code. We conducted an experiment where the dynamic layer was disabled and a fixed 2000 code size was used; the result was worse compared to the

Table 3.4: IoU evaluation metric for Multi categories, cross view.

IoU	Bench	Chair	Couch	Table
3D-EPN	0.415	0.452	0.531	0.477
Varley [132]	0.201	0.283	0.480	0.199
SnowFlakeNet	0.534	0.586	0.631	0.612
SeedFormer	0.532	0.583	0.629	0.609
3D-RecGAN++	0.540	0.594	0.643	0.621
3DCascade-GAN	**0.574**	**0.620**	**0.673**	**0.633**
CE	Bench	Chair	Couch	Table
3D-EPN	0.091	0.115	0.147	0.111
Varley [132]	0.105	0.143	0.207	0.174
SnowFlakeNet	0.039	0.068	0.095	0.050
SeedFormer	0.040	0.069	0.097	0.052
3D-RecGAN++	0.038	0.061	0.091	0.048
3DCascade-GAN	**0.036**	**0.058**	**0.089**	**0.047**

dynamic layer, as shown in Table 3.7. Also, three different experiments with three different K values: 50, 100 and 150 conducted. We found that the result was worse when $K = 50$; however, performance with both $K = 100$ and 150 had the same result. We also observe the model behavior when k approaches n ($K = 600$, $K = 900$), and the results show the performance drops gradually. Using the dynamic latent code encoder tends to optimize the latent codes where most values are set to zero, and these codes vary based on input shape. Furthermore, to show effectiveness of dynamic latent code, we trained the model with/without each components, the results shown in Table 3.8.

Self-attention. We tried using self-attention in both the networks (i.e. the encoder-decoder and discriminator), as shown in Figure 3.10, and tried using it on different layers to achieve the optimum results. The trials revealed that adding self-attention

Table 3.5: IoU and cross entropy evaluation metric for multi-category training and applied to unseen object categories, cross view, comparing 3D-EPN, Varley [132], SnowFlakeNet [149] (denoted Snow) , SeedFormer [173] (denoted Seed), 3D-RecGAN++ (denoted Yang) and our 3DCascade-GAN.

IoU	car	faucet	firearm	guitar	monitor	plane
3D-EPN	0.446	0.439	0.324	0.359	0.448	0.309
Varley	0.489	0.260	0.274	0.255	0.334	0.283
Snow	0.534	0.510	0.409	0.437	0.549	0.384
Seed	0.527	0.507	0.407	0.435	0.546	0.383
Yang	0.553	0.529	0.416	0.449	0.555	0.390
Ours	**0.564**	**0.537**	**0.425**	**0.455**	**0.560**	**0.394**
CE	car	faucet	firearm	guitar	monitor	plane
3D-EPN	0.160	0.086	0.033	0.036	0.127	0.065
Varley	0.171	0.123	0.028	0.030	0.136	0.043
Snow	0.103	0.060	0.018	0.016	0.078	0.033
Seed	0.105	0.061	0.018	0.017	0.079	0.034
Yang	0.100	0.055	0.014	0.015	**0.074**	**0.031**
Ours	**0.098**	**0.054**	**0.013**	**0.013**	**0.074**	**0.031**

to the encoder-decoder did not improve the results; in fact, the self-attention maps obtained when adding the self-attention layer to the generator network did not capture global structures well, and lead to poor reconstruction results. On the other hand, adding our self-attention layer to the discriminator effectively increased its capability to differentiate between real and fake 3D shapes, and eventually helped improve the capability of the generator to produce improved reconstruction.

Classifier. For the classifier, we compared the full version of the model (including cascade, dynamic latent code, self-attention and classifier) against a model without a classifier. As shown in Table 3.9, there are slight differences in that the classifier

Table 3.6: IoU and cross entropy evaluation metric for multi-category training and applied to unseen object categories, same view, comparing 3D-EPN, Varley [132], SnowFlakeNet [149] (denoted Snow) , SeedFormer [173] (denoted Seed), 3D-RecGAN++ and our 3DCascade-GAN.

IoU	car	faucet	firearm	guitar	monitor	plane
3D-EPN	0.450	0.442	0.339	0.351	0.444	0.314
Varley	0.484	0.260	0.280	0.255	0.341	0.295
Snow	0.548	0.526	0.412	0.438	0.554	0.371
Seed	0.545	0.524	0.409	0.435	0.553	0.367
Yang	0.555	0.536	0.426	0.442	0.562	0.394
Ours	**0.559**	**0.541**	**0.430**	**0.455**	**0.569**	**0.395**
CE	car	faucet	firearm	guitar	monitor	plane
3D-EPN	0.170	0.088	0.036	0.036	0.123	0.066
Varley	0.173	0.122	0.029	0.030	0.130	0.042
Snow	0.104	0.056	0.018	0.017	0.069	0.033
Seed	0.105	0.058	0.019	0.018	0.068	0.034
Yang	0.102	**0.053**	**0.016**	0.014	0.067	**0.031**
Ours	**0.101**	**0.053**	**0.016**	**0.013**	**0.065**	**0.031**

enhances the shapes, and this improvement is consistent.

3.4 Conclusion

In this chapter, we proposed an end-to-end model for 3D reconstruction from a single depth image. We introduced a 3D self-attention layer to attend to the non-local features, helping to connect the recovered views with the known view of the 3D shape. We also demonstrate introducing a dynamic latent code as an aid to optimizing the encoder, reducing the effective size of the latent space which enhanced the results.

Table 3.7: Ablation study on Dynamic latent code, we compare fixed latent code with different variation of dynamic code.

	Chair-IoU	Chair-CE
Fixed latent code: 2000	0.649	0.059
$n = 1000$, $K = 50$	0.645	0.061
$n = 1000$, $K = 100$	0.701	0.053
$n = 1000$, $K = 150$	0.700	0.053
$n = 1000$, $K = 600$	0.698	0.057
$n = 1000$, $K = 900$	0.656	0.059

Table 3.8: Ablation study on Dynamic latent code and self-attention.

	Chair-IoU	Chair-CE
3D-Cascade-GAN	0.701	0.053
without Dynamic layer	0.663	0.054
without self-attention	0.692	0.053
without self-attention & dynamic	0.654	0.054

These additions helped stabilise adversarial learning which leads to better estimation as demonstrated on different shape categories, both qualitatively and quantitatively. We further added multi-stage networks to sequentially refine 3D shapes. Furthermore, incorporating the classifier network showed improvement to the reconstructed shapes. Our method produces shapes with improved structure/geometry, outperforming state-of-the-art methods.

Table 3.9: Ablation study on Classifier.

	Bench	Chair	Couch	Table
with classifier	0.624	0.669	0.773	0.682
without classifier	0.622	0.667	0.771	0.681

3.5 Limitations

The proposed model consist of the stages which require higher power consumption.The model introduces a dynamic selection layer to eliminate unwanted codes. However, it still requires a set of latent codes to choose from, which in turn demands space.

3.6 Summary

In this work, we presented a 3D reconstruction model from a single depth image. The model is based on a cascade architecture that utilises voxelisation for flexibility and a dynamic latent code selection process for selecting appropriate latent codes. Furthermore, we incorporated a 3D self-attention layer to capture global information. For evaluation, we tested the model on a variety of 3D shapes and showed that it can generate detailed 3D shapes from depth images with good accuracy.

Chapter 4

Rendering based 3D Shape Evaluation

4.1 Introduction

Our living world is 3D, and so analysis and processing of 3D shapes are fundamental techniques for a variety of application domains, ranging from design and manufacturing, robotic navigation to virtual and augmented reality (VR/AR). 3D shapes have a wide range of applications, from grasping [35, 38, 156] to reconstruction [49, 42]. In many applications, measuring the distortion of 3D shapes is required. For example, when 3D objects are manufactured, the produced shapes have unavoidable deviations compared with the original designs, and it is therefore useful to quantify the deviations based on users' subjective perception. Another example is when 3D data is streamed in VR/AR applications, distortions could be introduced due to data compression with limited bandwidth, and measuring the distortion of the shapes is not only useful for distortion control, but can also help guide how to better allocate the limited bandwidth. Furthermore, bokeh (the effect of an out-of-focus background when shooting an object) is another example of a depth image evaluation application. For good results, bokeh requires accurate depth image segmentation, and due to the blurring-like effect, depth image evaluation is necessary [161]. Within the context of the thesis, it is important to evaluate the distortion of 3D reconstruction, ideally in consistency with human perception, where human perception refers to the process by which humans interpret and make sense of sensory information received from the environment.

As most 3D reconstruction test sets contain ground truth shapes, in this work, we focus on full-reference shape distortion measures, which tend to be more reliable. Given a pair of 3D shapes, one original and one distorted, the task we address in this chapter is to predict a similarity score, that is ideally close to human subjective judgement. Traditional methods tend to directly measure errors on 3D shapes. However, such measures are often inconsistent with human judgement. For example, a distorted shape might represent only a minor geometric change, yet this alteration can be perceptually significant. Observing that human eyes essentially perceive 2D views of 3D shapes, whether in the real world or in the virtual settings,

We propose to measure 3D shape distortion based on their 2D renderings. To give a sufficient coverage, we start by rendering the shapes to multiple views. We specifically choose centres of dodecahedron faces as camera locations for capturing the shape views and placing a directional light. Perceptual quality of 2D views can also be influenced by rendering styles, as different aspects of 3D shapes would be emphasised with different renderings. For example, even for a relatively small dent on a surface, the local geometric normal may change significantly. Rendering with metal styles can highlight subtle changes on shape areas that cause specular highlights to look different, whereas rim type of rendering is more sensitive to edges (see Figure 4.3 for some examples of rendering styles.

We therefore propose to use combinations of different rendering styles (different shading and material properties) to better capture the visual distortion of 3D shapes. Next, we extract distortion measures using a 2D method, such as structural similarity index measure (SSIM) [138] and mean squared error. To avoid the influence of empty space for image distortion measure when rendering shapes, we further propose a modified SSIM that only accounts for the foreground regions (called Mask-SSIM). These features are combined using a neural network based approach to predict the subjective distortion measure.

The contributions of this work are as follows:

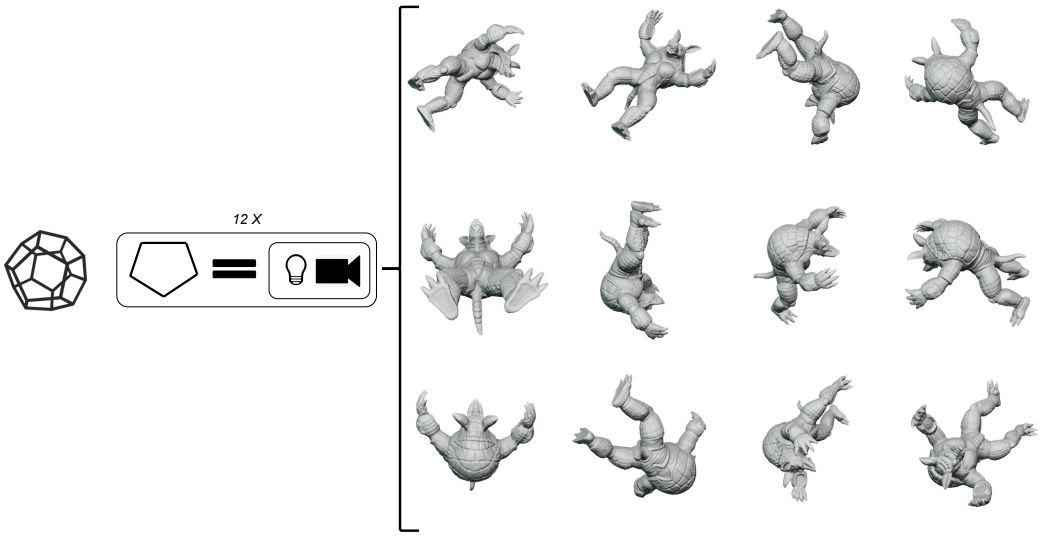


Figure 4.1: A list of rendered views of Armadillo using dodecahedron faces, each face consisting of a directional light and a camera both pointing to the centre of the dodecahedron. .

- We propose an image-based method to measure perceptual distortion of 3D shapes. We further combine a variety of rendering styles and 2D image distortion measures, along with a neural network based learning approach for improved 3D subjective distortion prediction.

- In order to ensure more stable performance when shapes are rendered to different canvas sizes, we extend SSIM to only focus on the foreground region, referred to Mask-SSIM, which is effective for our task.

- Experiments on public datasets demonstrate that our method achieves good prediction for subjective distortion scores, outperforming existing techniques.

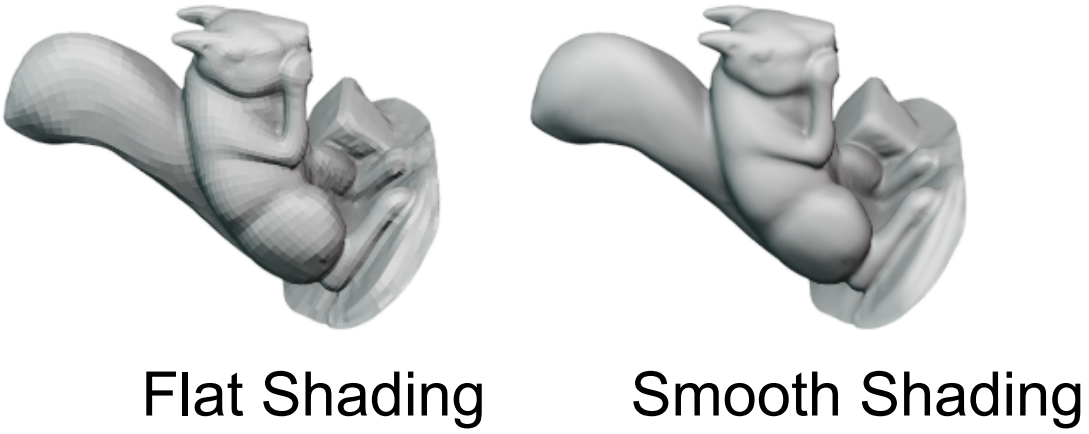


Figure 4.2: We use two types of shading (i.e., flat and smooth) for more generalised distortion measure.

4.2 Methodology

We first describe how the 3D shapes are rendered using different views, styles and shaders. Next, we will discuss Mask-SSIM, which is a metric that is agnostic to which 3D representation is used. Finally, we list a collection of different 2D measures used as the basis to build the machine learning based model for final distortion prediction.

4.2.1 Rendering Setup

We use centres of the faces of a regular dodecahedron as camera locations for shape rendering. The dodecahedron is one of the five Platonic solids; it ensures that the cameras are equally spaced, providing a good coverage for the entire shape. The target shape is placed in the centre of the dodecahedron with all 12 cameras facing it. For simplicity, we normalised all shape sizes during the rendering process. Each camera is paired with a directional light, as shown in Figure 4.1. Our rendering setup was built in Blender as it offers automation through the Python binding library; however, a similar environment could be built in any appropriate application.

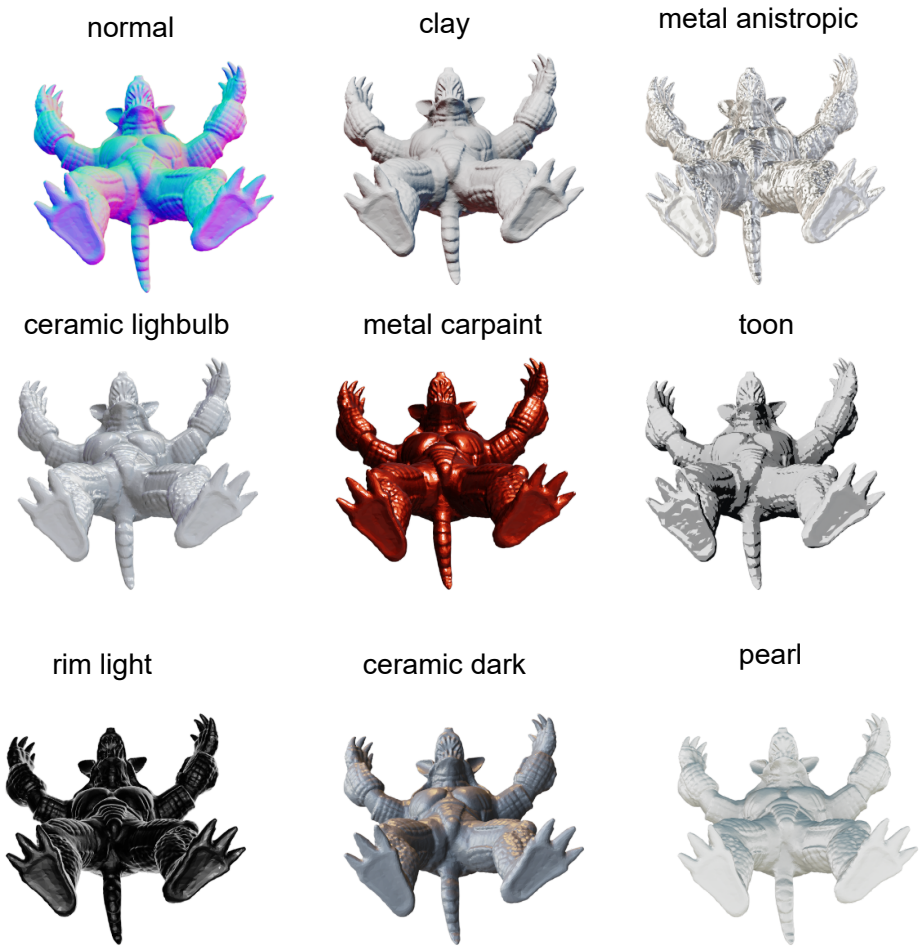


Figure 4.3: Some of the styles used in the experiments. As can be seen, different rendering styles tend to highlight different aspects of the shape characteristics..

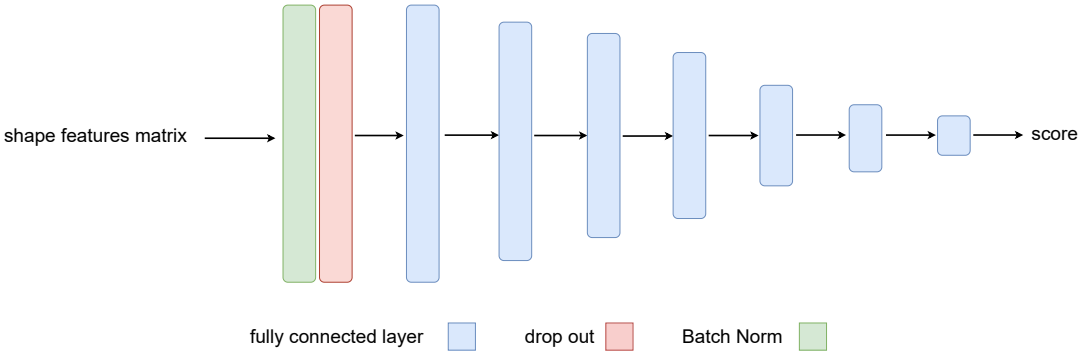


Figure 4.4: Our network architecture for learning to predict shape distortion score. It consists of batch normalisation and dropout and fully connected layers, and we utilise residual blocks between layers..

4.2.2 Rendering Styles

We utilise two different kinds of shading for each different style. To begin with, a flat shading was utilised; this uses the normal of each triangle to render the triangle, so the triangulation details are clearly visible. In addition to that, we utilise smooth shading, which effectively blurs the boundary between triangular faces so that they appear smooth. See Figure 4.2 for an example. We make use of the predefined Blender textures and colours in addition to importing some additional ones, which brings in a total number of 30 styles: flat & clay muddy, smooth & metal lead, basic 1, basic 2, basic dark, basic side, brown, ceramic dark, ceramic lightbulb, normal+y, check rim dark, check rim light, clay brown, metal shiny, orange-blue, pearl, reflection check horizontal, reflection check vertical, resin, skin, toon, clay muddy, clay studio, dark grey, jade, matt blue, matt brown, metal anisotropic, metal car paint, metal lead. An example of the generated samples can be seen in Figure 4.3. Note those styles were predefined in blender or are widely used.

The process proceeds as follows: first, we identify the shape and place it within the geometry of the dodecahedron. Second, we decide which kind of shading to use (flat or smooth). We render each style from a total of twelve distinct angles (the geometric faces of a dodecahedron), starting with the shape that serves as a reference and moving on to the distorted versions of that shape. The datasets have been normalised so that all of the shapes are the same size [0-1].

4.2.3 Mask-SSIM

SSIM can identify the differences between the targets, by measuring luminance, contrast and structure. It is later extended in a multi-scale manner, such as MSSIM and DSSIM. However, the function does not provide consistent results for targets when the same 3D shape is rendered to canvas of different resolutions. This is because background left blank is always consistent with background of another shape, leading to

high SSIM scores. Therefore, the similarity scores varies according to image resolution of the rendering canvas (see Table 4.5).

In our approach, we first separate the target shape from the background. Assume we have two images, A and B. To create the mask, we match each pixel inside image A to the pixel at the same location in image B. If both pixels are inside the target shape, a value of 1 is given for the position in the mask; otherwise, the mask is set to 0. Figure 4.5 shows the proposed algorithm.

The operation by definition is pixel-wise. However, since the SSIM calculation is based on windows, we also implement Mask-SSIM Window, which only considers a pixel to be included if the entire neighbourhood window is included. These models (Mask-SSIM and Mask-SSIM window) produce robust results with varying canvas resolutions.

The quantitative comparisons are shown in Table 4.5. Specifically, Mask-SSIM is implemented on top of SSIM when generating the SSIM map as follows:

$$C = SSIM(A, B) \tag{4.1}$$

where $SSIM(\cdot)$ returns the SSIM map of A and B, denoted as C. Let A_{mask} and B_{mask} be the foreground masks of images A and B:

$$Mask_{AB} = A_{mask}B_{mask} \tag{4.2}$$

$Mask_{AB}$ identifies the agreement between the two masks, where 1 means pixels at a position in both images are foreground, and 0 otherwise.

$$MaskSSIM(A, B) = \frac{1}{N}\sum_{i=1}^{n} C_i Mask_{AB,i} \tag{4.3}$$

$MaskSSIM$ describes how Mask-SSIM works. C_i represents pixels at the ith location. We multiply pixel C_i by $Mask_{ab,i}$ to limit the operation w.r.t. A_i and B_i where $Mask_{AB,i}$ is a binary pixel value $[0, 1]$.

n is the number of pixels in the image, and $N = \sum_{i=1}^{n} Mask_{AB,i}$ is the number of selected pixels.

4.2.4 Selected Features

We select various 2D/3D distortion measures and features. For 2D, we use SSIM [138] where SSIM is a metric used to assess the perceptual quality of digital images and videos by comparing their structural information, luminance, and contrast. FSIM [165] is a metric for evaluating image quality by comparing the similarity of local features between a reference and a test image. Root mean square error (RMSE) measures the difference between the distorted and original shapes. Canny edge detector [23] is an algorithm used to identify the edges in images by detecting areas with strong gradients, using a multi-stage process involving noise reduction, gradient calculation, non-maximum suppression, and edge tracking by hysteresis. SRE [69] is a metric used to evaluate the quality of reconstructed signals or images by comparing the original signal with the error introduced during the reconstruction process. Our modified Mask-SSIM or its variants is also included. Different variants of Mask-SSIM are explained in Section 4.3.2. We also implemented 3D measures, Chamfer distance and Hausdorff distance [61] for comparison.

4.2.5 Network Architecture

The learning task is to find a relationship between the results of the individual measures and map them to the final distortion prediction, which can be thought of as bridging the gap between evaluation metrics and human judgements. After calculating the values of the Mask-SSIM, we combine these values with the scores from other algorithms to create the input matrix for our neural network model.

We start building using bottom-up strategies, where we add layer and evaluate until, this process becomes repetitive and we found the accuracy not increasing.

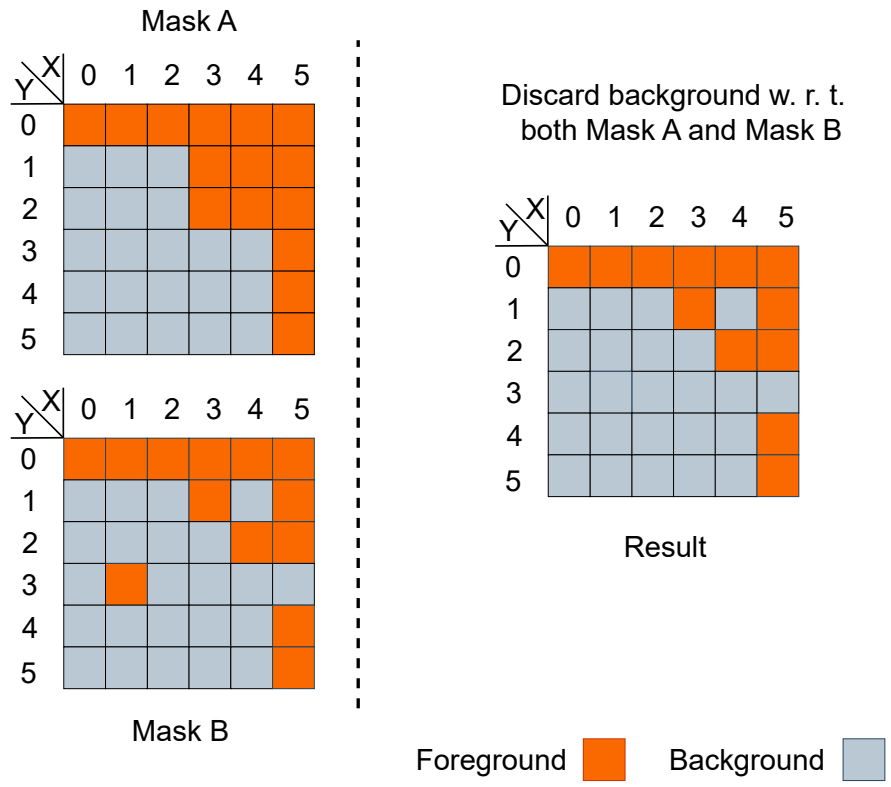


Figure 4.5: Illustration of Mask-SSIM operation. We classify background/foreground pixel w.r.t. to both A and B. Left shows both original masks A and B before classification. Right shows the output after classification w.r.t. both A and B. For example, looking to the resulted mask we see pixel at *Result[4,1]* is classified as background however in Mask A is not, in this situation clearly there is some distortion between the two shapes in the 3D space.

The model starts with batch normalisation and a dropout layer, followed by fully connected layers. Each layer has a ReLU activation function, except for the final layer which has a sigmoid activation function. We utilise the mean squared error as the loss function (see the model shown in Figure 4.4).

To compare different distortion metrics, since they are often in different ranges like SSIM range is [0,1] while MSE is zero to infinity, instead of directly comparing values of user voting and prediction, we use Pearson correlation to fit the functions into a [-1,1] domain, the Pearson correlation coefficient ranges from -1 to +1. A value close

to -1 indicates a strong negative correlation, meaning that as one variable increases, the other decreases. Conversely, a value close to +1 signifies a strong positive correlation, where both variables move in the same direction. In addition, the number of features is still high and some may not be helpful in learning the model. So we further utilise Sequential Forward Search (SFS) to select important features. It incrementally adds new features to the selection, and at every stage, a greedy approach is used to choose the feature that leads to be best performance among all choices, based on its performance on the training set. The selected features are then used when applying the model to the unseen test set. Although SFS can often be too expensive to be used in the deep learning setting, our deep network is small and can be trained efficiently, so this strategy is still reasonably efficient, Table 4.1 shows the selected features for Dataset [70] on dwarf shape, we found that the model only needs 7 features to reach optimum results for our model and Figure 4.6 shows the mapping between selected features and accuracy.

4.2.6 Experiments

4.2.6.1 Training Details

The model was trained for 250 epochs and batch size of three. The learning rate was set to 0.0001. The stochastic gradient decent was used as the optimiser. The model was trained on an NVIDIA GTX 1080ti GPU. The rendering of a shape in a specific style and view takes 22 milliseconds.

4.2.6.2 Datasets

Three datasets are used in the experiments which include both distorted shapes and subjective scores: [72], [70] and [48].

Algorithm 4.1 Sequential Forward Search (SFS)

Input A list of features: $F = \{f_1, f_2, \cdots, f_d\}$

Output A list of selected features: Sel_F

$Sel_F = \{\}$

repeat:

$f_{\text{next}} = \emptyset$

$c_{\text{next}} = corr(Sel_F)$; *$corr(\cdot)$ computes the correlation on the training set when trained with the given set of features*

for $f = f_1 \ldots f_d$**:**

if $f \notin Sel_F$ **then**

$c_f = corr(Sel_F \cup \{f\})$

if $c_f > c_{\text{next}}$ **then**

$f_{\text{next}} = f$

$c_{\text{next}} = c_f$

if $f_{\text{next}} \neq \emptyset$ **then** $Sel_F = Sel_F \cup \{f_{\text{next}}\}$

until $f_{\text{next}} = \emptyset$

For [72], 12 students from the Swiss Federal Institute of Technology and the University Claude Bernard of Lyon participated in a study where they evaluated 3D objects by interacting with them (rotation, scaling, translation) from a comfortable distance. Initially, participants were shown the original models alongside distorted versions, including the extreme cases of noise and smoothing, to set a reference for each object's distortion level. Specifically, they were instructed to note the most distorted version they observed. Then, they reviewed 66 objects, each displayed for 20 seconds, and rated the distortion on a scale from 0 (no distortion, identical to the original) to 10 (extremely distorted). The order of these objects was randomised for each participant to eliminate any bias from the order of presentation.

For [70], for each of the four models (Armadillo, Dyno, Lion Head, and Bimba) in the study, observers were presented with six altered versions of the original object.

They were tasked with rating each version based on its visual similarity to the original, on a scale from four (identical) to zero (most degraded). These objects were displayed for approximately three minutes, during which participants could interact with them through rotation, scaling, and translation. Notably, all six degraded versions were shown simultaneously on the same screen, eliminating the need for establishing a baseline for comparison. Consequently, participants naturally assigned a "0" to the most degraded version and a "4" to the version that most closely resembled the original.

For [48], the authors created and reviewed a broad range of distortions, selecting a subset that represented various levels of visual quality (Excellent, Good, Fair, and Poor) to include in the database.

The four shapes in dataset [72] are {Armadillo, Rockerarm, Venus, Dyno}. The five shapes in dataset [48] are {Dwarf, Hulk, Squirrel, Statue, Sports Car}. The shapes in dataset [70] are {Armadillo, Dyno, Lion, Bimba}. Several types of distortions are used in Dataset [72]: 1. smoothing with a different number of iterations; 2. simplification (removal of vertices) with different percentages; 3. uniform quantisation using different bit sizes; 4. JPEG texture compression; and 5. sub-sampling to reduce the texture size. As we only examine geometric distortion here, we eliminate textural (2D) distortion. The sports car shape could not be included in our experiments as we identified some issues in loading the geometry (files missing). As a result, each dataset contains 4 usable shapes. Although this may sound quite small, considering the range of distortions, and time consumption for collecting user subjective ratings, collecting such data is onerous, and we are not aware of larger datasets of this kind being available.

A total of 12 distortion types are used in [72]. The dataset [70] only applies noise to the shape surface with six different levels of noise for each shape. The work [48] applies two types of distortion with different levels. The first is noise addition, which was done by altering the location of vertices on different levels. The second is Taubin smoothing [130] distortion. A total of 21 distorted shapes were generated for each shape.

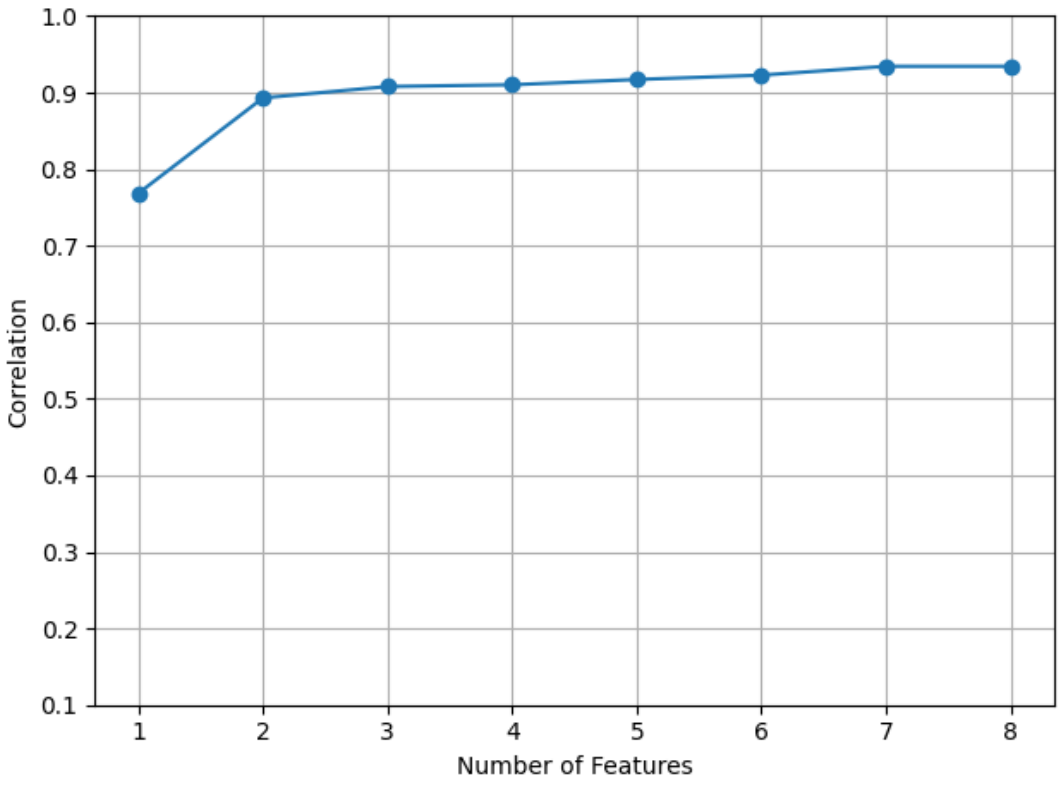


Figure 4.6: SFS result on dwarf in Dataset [48], where it shows using the 7 selected features has the highest performance.

4.3 Evaluation

To evaluate our model's results, we use Pearson correlation.We calculate the correlation between the Mean Opinion Score (MOS) and each method because the methods we used for comparison all have either low sensitivity or varying value ranges , as a result to unify it we used Pearson correlation.

To evaluate our model's performance, we compare it to five methods: (1) the Point SSIM Model [14]; this method leverages geometry, vector value and curvature to calculate the similarity; and (2) the Density-Aware Chamfer Distance [147] (DCD) Model; this approach is derived from the Chamfer distance and focuses on distribution quality. Two other metrics are traditional and widely used methods (3) MSE (Mean Squared

Table 4.1: Selected features using SFS when Dwarf is used as the test shape, Dataset [48].

selected features
Mask SSIM window ceramic lightbulb flat
Mask SSIM metal anisotropic smooth
Mask SSIM matt blue smooth
Mask SSIM brown flat
No mask metal lead smooth
Mask SSIM clay studio smooth
Mask SSIM metal anisotropic flat

Table 4.2: Cross-validation correlation results on Dataset [72]. Our and Our* refer to our results with all features and selected features. Note the setup for MS-SSIM is a flat shader and ceramic lightbulb style, similar to typical rendering styles in previous work. Note, this number represents Pearson correlation.

	chamfer	Hausdorff	PointSSIM	DCD	MSE	PSNR	MS-SSIM	Our	Our*
armadillo	0.13	0.13	0.20	0.12	0.19	0.17	0.20	0.41	**0.54**
dyno	0.02	0.47	**0.67**	0.16	0.17	0.19	0.34	0.52	0.54
rockerarm	0.08	0.27	0.15	0.23	0.19	0.18	0.17	0.36	**0.62**
venus	0.25	0.13	0.62	0.40	0.21	0.20	0.24	0.36	**0.65**
average	0.12	0.25	0.41	0.23	0.19	0.19	0.23	0.41	**0.58**

Error), and (4) PSNR (Peak Signal-to-Noise Ratio). Finally, we also compare with an image-based baseline which applies (5) MS-SSIM (multi Scale-SSIM) to 12 rendered views.

Table 4.3: Cross-validation correlation results on Dataset [48]. Our and Our* refer to our results with all features and selected features. Note the setup for MS-SSIM is a flat shader and ceramic lightbulb style. Note, this number represents Pearson correlation.

	chamfer	Hausdorff	PointSSIM	DCD	MSE	PSNR	MS-SSIM	Our	Our*
dwarf	0	0.16	0.62	0.19	0.32	0.31	0.34	0.75	**0.93**
hulk	0.35	0.45	0.65	0.56	0.35	0.35	0.46	0.80	**0.87**
squirrel	0.31	0.22	0.17	0.34	0.26	0.25	0.36	0.43	**0.63**
statue	0.05	0.52	0.92	0.61	0.31	0.29	0.38	0.92	**0.93**
average	0.17	0.33	0.59	0.43	0.31	0.30	0.38	0.72	**0.84**

Table 4.4: Cross-validation correlation results on Dataset [70]. Our and Our* refer to our results with all features and selected features. Note the setup for MS-SSIM is a flat shader and ceramic lightbulb style. Note, this number represents Pearson correlation.

	chamfer	Hausdorff	PointSSIM	DCD	MSE	PSNR	MS-SSIM	Our	Our*
Armadillo	0.13	0.13	0.26	0.09	0.12	0.11	0.31	0.40	**0.51**
Dyno	0.02	0.47	0.41	0.19	0.26	0.10	0.42	0.55	**0.63**
Lion	0.19	0.32	0.32	0.21	0.18	0.21	0.47	0.58	**0.64**
Bimba	0.12	0.24	0.22	0.11	0.15	0.17	0.28	0.59	**0.63**
average	0.11	0.29	0.30	0.15	0.17	0.14	0.37	0.53	**0.60**

4.3.1 Results

Our model was trained on three different datasets, as stated above. The results of the training for Lavoué. et al.'s dataset [72] are shown in Table 4.2, we achieve the best result in 3 out of 4 shapes, furthermore in average we outperformed other models. The results for Dataset [48] are shown in Table 4.3 we outperformed other models in all 5 shapes, and the results for Dataset [70] are reported in Table 4.4, our model outperformed other models in all 4 shapes. For all datasets, a leave-one-shape-out

cross-validation method was used to show model generalisation capabilities, as the datasets' sizes are relatively small, and we treat each shape in turn as the test shape with the remaining shapes as the training set. As the tables show, our approach outperforms compared state-of-the-art methods with a large margin. Our method with SFS features are consistently better than our method without feature selection. As the table shows Pearson correlation between each method scores and subject scores, the result proves that, styles can enhance model estimation.

4.3.2 Ablation Studies

We conduct three ablation studies, first on the batch-norm to show its necessity, followed by different versions of MaskSSIM, and finally cross-dataset feature selection.

Batch-norm layer. The batch-norm is introduced first in the model to normalise input data, we choose *Dataset [72]* for the experiments. The network shows worse results without the batch-norm layer, as shown in the comparison in Table 4.6.

We also tested different versions of Mask-SSIM. Four versions of Mask-SSIM: [*Mask-SSIM - Mask-SSIM Window - Mask-SSIM Negative -Mask-SSIM Merge*] are implemented. As described above, Mask-SSIM calculates the score for the foreground pixels and omits the background pixels. After calculating the pixel score for one image, we repeat the operation with the other image. Finally, we average the scores. In this ablation, we examine a different approach that utilises pixels at boundaries and spatial relations.

Mask-SSIM window. As discussed, we only consider pixels as in the foreground where all pixels in the neighbouring window (3×3) are foreground pixels.

Mask-SSIM negative is a pixel-wise operation that is similar to the original (Mask-SSIM). However if this operation encounters a pixel that is considered part of the foreground in the first image but not in the second image, it penalises the score by adding -1. The result was worse than Mask-SSIM.

Mask-SSIM merge. The operation tries to find a smooth middle line between Mask-SSIM and Mask-SSIM negative as the result so we take the average of both values.

Overall, MaskSSIM achieves best performance, and so is used in our model, for comparison see table 4.7.

4.3.3 Cross-Dataset Evaluation with Feature Selection

Our method relies on neural networks and feature selection to achieve the best performance. Although leave-one-shape-out testing ensures training/test separation, feature selection has to be performed for each training/test split. To further evaluate the generalisability of our method, we perfrom cross-dataset evaluation, where the whole dataset [48] is used for training (including SFS feature selection), and the trained model is then applied to the two other datasets [70] and [72]. The selected features are shown in Table 4.9 and the performance is reported in Table 4.8. We compare the cross-dataset performance with within-dataset performance (cross validation including SFS) and PointSSIM which is the best performing previous method. As can be seen, the model in the cross-dataset setting achieves slightly worse correlation: for the dataset [70], the average correlation drops from 0.60 to 0.58, but still much higher than existing method PointSSIM (0.30). Similar observations can also be made for the dataset [72]. This demonstrates that our learned model can be generalised to independent datasets with different types of distortions while still achieving good performance. Such models are also more efficient to deploy as only the selected rendering styles need to be generated during testing.

4.4 Conclusion

In this paper, we presented an image-based method to evaluate 3D shape distortion. Shapes are rendered from 12 views, along with a range of rendering styles. A deep

learning based approach is then used to learn to predict distortion measures more closely related to subjective evaluation. Experiments on three datasets demonstrate that our method outperforms existing methods by a large margin. Our cross-dataset evaluation further demonstrates the generalisability of our learning based model.

4.5 Limitations

The dataset used in this study primarily considers convex shapes, which may not accurately represent real-life scenarios. Moreover, the established distortion metric datasets include only a limited range of shapes. Additionally, the method is still time-consuming compared to other techniques mentioned.

4.6 Summary

In this study, we introduced a deep learning approach for 3D shape distortion assessment using images rendered from multiple views and styles. Our method, tested on three datasets, consistently outperformed traditional techniques and exhibited notable generalisability across different datasets.

Table 4.5: Comparisons of original SSIM and Mask-SSIM for resolutions of 500×500 and 1000×1000 canvas sizes. The experiment is based on the dwarf shape with various distortions, rendered using metal anisotropic material and smooth shading.

Distorted shapes	500 resolution Mask-SSIM	500 resolution original SSIM
dwarf quantization 8 bit	0.29	0.86
dwarf quantization 9 bit	0.52	0.90
dwarf quantization 10 bit	0.79	0.96
dwarf quantization 11 bit	0.92	0.98
dwarf Simplification 0.80	0.53	0.90
dwarf Simplification 0.92	0.46	0.89
dwarf Simplification 0.975	0.27	0.85
dwarf Simplification 0.987	0.27	0.85
dwarf Smoothing 15 iteration	0.72	0.94
dwarf Smoothing 25 iteration	0.64	0.92
dwarf Smoothing 40 iteration	0.57	0.91
dwarf Smoothing 50 iteration	0.53	0.90
Distorted shapes	1000 resolution Mask-SSIM	1000 resolution original SSIM
dwarf quantization 8 bit	0.29	0.97
dwarf quantization 9 bit	0.52	0.99
dwarf quantization 10 bit	0.79	0.99
dwarf quantization 11 bit	0.92	0.99
dwarf Simplification 0.80	0.53	0.98
dwarf Simplification 0.92	0.46	0.98
dwarf Simplification 0.975	0.27	0.97
dwarf Simplification 0.987	0.27	0.97
dwarf Smoothing 15 iteration	0.72	0.99
dwarf Smoothing 25 iteration	0.64	0.99
dwarf Smoothing 40 iteration	0.57	0.99
dwarf Smoothing 50 iteration	0.53	0.99

Table 4.6: An Ablation study on batch-norm layer using the selected features only. Note, this number represents Pearson correlation.

	without batch-norm	with batch-norm
armadillo	0.42	**0.54**
dyno	0.52	**0.54**
rockerarm	0.56	**0.62**
venus	0.62	**0.65**
average	0.53	**0.58**

Table 4.7: Comparison of different variants of MaskSSIM on the dataset [72] with cross-validation correlation results. Note, this number represents Pearson correlation.

	Mask-SSIM	Mask-SSIM window	Mask-SSIM negative	Mask-SSIM merge
armadillo	**0.28**	**0.28**	0.18	0.23
dyno	**0.44**	0.43	0.35	0.39
rockerarm	**0.24**	**0.24**	0.20	0.22
venus	**0.37**	**0.37**	0.29	0.34
average	**0.33**	**0.33**	0.25	0.29

Table 4.8: Cross-dataset correlation results. The model trained on Dataset [48] with SFS feature selection, and then tested on Datasets [70] and [72]. We compare the performance with same dataset leave-one-shape-out testing results (including feature selection), and the previous best performing model PointSSIM. Note, this number represents Pearson correlation.

Dataset [70]	within-dataset	cross-dataset	PointSSIM
Armadillo	0.51	0.43	0.26
Dyno	0.63	0.61	0.41
Lion	0.64	0.59	0.32
Bimba	0.63	0.62	0.22
average	0.60	0.58	0.30

Dataset [72]	within-dataset	cross-dataset	PointSSIM
armadillo	0.54	0.53	0.20
dyno	0.54	0.51	0.67
rockerarm	0.62	0.62	0.15
venus	0.65	0.55	0.62
average	0.58	0.55	0.41

Table 4.9: Selected features using SFS for cross dataset evaluation where features are selected based on Dataset [48].

selected features
Mask SSIM car paint flat
Mask SSIM matt blue smooth
Mask SSIM brown flat
No mask resin smooth
Mask SSIM clay studio smooth
Mask SSIM metal anisotropic flat
No mask SSIM check rim light smooth

Chapter 5

Learning to Generate Canonical Forms for Single Depth Images

5.1 Introduction

3D reconstruction aims to turn 2D input such as images into 3D shapes. Most 3D reconstruction methods are designed for rigid shapes (including the method introduced in Chapter 3). But for non-rigid objects that can bend or twist, like living creatures or flexible materials, it gets tricky. These objects can have a large range of deformation, making them hard to handle especially for reconstruction tasks, as significant training examples are required to cover the deformation space. To make the problem more manageable, an effective approach is to bring non-rigid shapes back to a default or standardised pose. This default pose is called the canonical form. Using this form can help simplify and improve various geometric processing tasks, from shape retrieval to shape reconstruction.

Canonical form refers to a normalised representation of a deformable shape such that various instances of similar objects are represented in a unified pose which removes the non-rigid deformation. This uniform representation aids in reducing variability [21], ensuring consistency, and simplifying subsequent computational processes [51]. The canonical form is commonly used in retrieval tasks, enabling us to search for and identify similar 3D models regardless of their deformations. However, current

canonical form methods often prioritise discriminating between shapes but fail to retain good quality of shape appearance. These approaches typically rely on either Euclidean distance [111] [80] or geodesic distance [37] which can distort the deformed shapes. Alternatively, some works suggest other approaches like mapping the deformed shape to a template to preserve shape appearance [80]. However, these methods all assume that the input is a complete deformed shape, so cannot be applied to cases with depth image input.

In this study, we consider the problem of turning a deformable shape as a single-view depth image to its canonical form. This is a more challenging task as the input is no longer a complete shape. It is also a useful processing step for deformable shape reconstruction, as once turned into its canonical form, shapes are aligned and existing single view reconstruction methods for rigid objects can be applied.

To address this challenging task, we introduce a learning-based model that turns a single depth image to a default pose. Given a 2D depth image and its corresponding mask, our model aims to produce a depth image that corresponds to the input shape in a canonical pose. Figure 5.1 displays an overview of the model, which begins with an encoder-decoder that produces high-dimensional local features. Additionally, we introduce parallel encoders utilising sparse convolution to detect varying neighbours, thereby fusing multi-scale features that contribute to preserving shape appearance. These fused features serve as a basis to generate high-dimensional attributes. Ultimately, we utilise an encoder-decoder model to reconstruct the canonical pose depth image.

Our contributions are:

- We propose Canonical pose model, an end-to-end 2D network designed for the canonical pose reconstruction of single-view depth images. It comprises three components, Local Features Extractor (LFE), Multi-Scale Features Extractor (MSFE) and reconstruction component.

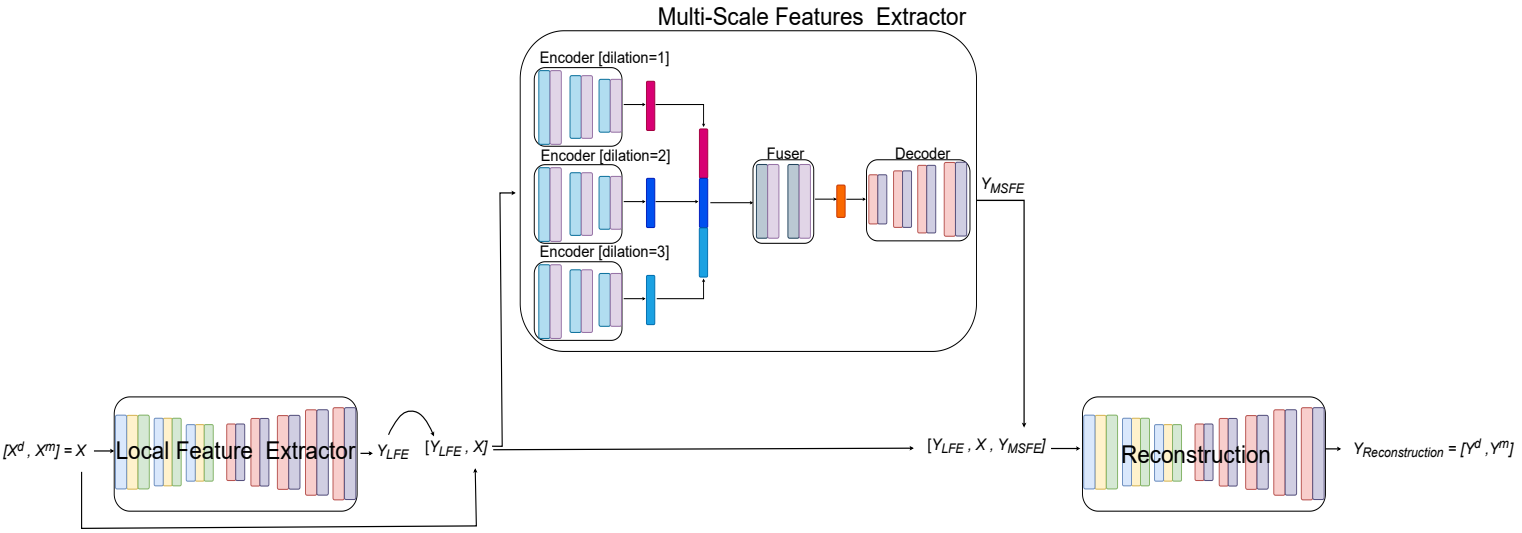


Figure 5.1: Overview of the model, which comprises three components. Initially, the model processes the input to extract local features. Subsequently, it uses both the original input and the extracted features for multi-scale feature extraction. Finally, a reconstruction component reconstructs the depth image of the canonical pose shape based on the outputs from both previous components.

- We propose parallel encoders and a single decoder block that extract features at different scales and use a fusing decoder to decode multi-scale, high-dimensional features.
- The extensive experimental results on TOSCA [19] and human [113] datasets demonstrate that our model outperforms the existing state-of-the-art methods and has competitive inference time. Moreover, our model is also capable of preserving high quality shape details while deforming shapes across different types of forms, such as humans and animals.

5.2 Methodology

The canonical form involves addressing deformation by eliminating it. The input, a depth image, encompasses values ranging from 0 to 1. We anticipate that the model

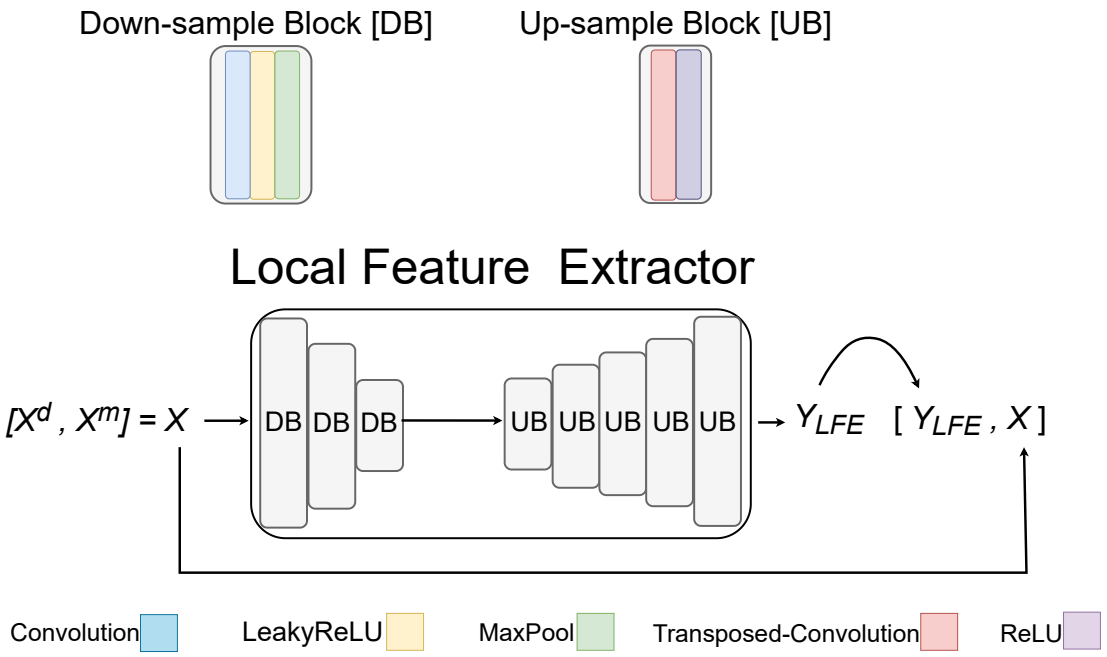


Figure 5.2: The Local Feature Extractor (LFE) takes the single-view depth image X^d and the corresponding mask X^m as input. It is composed of an encoder and a decoder. The encoder has three down-sample blocks, with each block featuring a convolution layer, ReLU, and max pooling. In contrast, the decoder encompasses five up-sample blocks, each having a transposed convolution and ReLU. The model takes a depth and its mask as input and produces a local feature output, of the same input size, denoted as Y_{LFE}.

will transform this depth image to align with a canonical form. This model consists of three distinct components. Initially, the first component interprets the depth image to derive high-dimensional local features, subsequently integrating this original depth with the extracted local features. Thereafter, the model leverages parallel-encoders in conjunction with a fusing decoder to generate multi-scale dimensional features. At the end of this process, these multi-scale features are concatenated with local features, serving as skip links for the last component as well as grouping local features with multi-scale features, which aid in reconstructing the depth into its canonical form.

5.2.1 Local Feature Extractor

Given an input depth image X^d and mask X^m, where $X^{d,m} = \{x_i^{d,m} \in \mathbb{R}^{500\times500}\}$, the LFE component processes both to generate local features. The component consists of N down-sample blocks and K up-sample blocks, where $N = 3$ and $K = 5$. For the down-sample blocks, each block consists of a convolution with a kernel size of 5×5 and strides of 1×1. We use `LeakyReLU` as the activation function, and a `Maxpool` layer is employed for spatial reduction. For the up-sample blocks, the transpose-convolutions utilise three different kernel sizes: $[5, 3, 2]$, which are applied in the order $[5, 3, 5, 2, 2]$. `LeakyReLU` is also utilised for each of these up-sample blocks. The output features, denoted as Y_{LFE} in Eq. 5.1, are concatenated with the original input $X^{d,m}$ as an extra channel. The network is shown in Figure 5.2.

$$Y_{LFE} = LFE(X^d, X^m) \tag{5.1}$$

LFE attaches local features to the original depth image and its mask, so each pixel is associated with both a local feature and a mask value. Consequently, in Section 5.2.3, the reconstruction component has access to both the local features and the original input depth image.

5.2.2 Multi-Scale Feature Extractor

The Multi-scale Feature Extractor (MSFE) described in Eq. 5.2 comprises three parallel encoders $E_{dilation1}$, $E_{dilation2}$ and $E_{dilation3}$.

$$Y_{MSFE} = MSFE(X^d, X^m, Y_{LFE}) \tag{5.2}$$

$$z_1 = E_{dilation1}(X^d, X^m, Y_{LFE}) \tag{5.3}$$

$$z_2 = E_{dilation2}(X^d, X^m, Y_{LFE}) \tag{5.4}$$

$$z_3 = E_{dilation3}(X^d, X^m, Y_{LFE}) \tag{5.5}$$

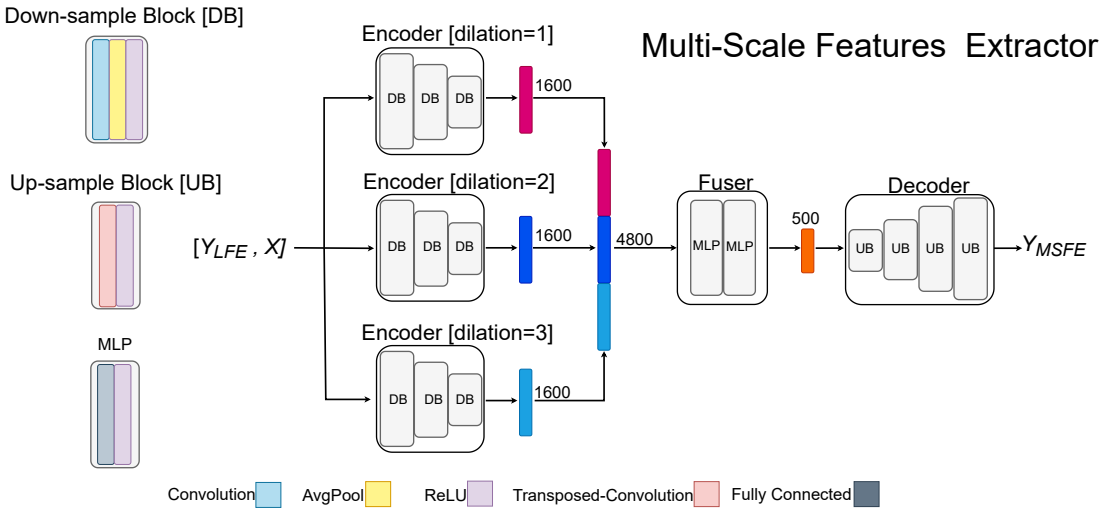


Figure 5.3: The model takes as input the original depth X^d, its mask X^m where $[X^d, X^m] = X$, and the local feature output Y_{LFE}. It features three encoders, each having a distinct dilation rate, with each encoder made up of down-sample blocks. Following the encoders, the latent codes are concatenated and passed through a fuser for inter-mapping. The subsequent decoder consists of up-sample blocks, culminating in the reconstructed multi-scale features, denoted as Y_{MSFE} .

Each encoder captures a different spatial neighbourhood size owing to the inherent nature of convolutions with distinct dilation values. Specifically, the three encoders possess dilation values of 1, 2, and 3 (Eq. 5.3, 5.4 and 5.5) in their convolution layers.We could not add more than 3 encoders as computation consumption exceeds GPU limits, also 1,2,3 variation is the natural way to expand as no previous work we found is suitable to rely on. Each encoder is equipped with three convolution layers followed by a `LeakyReLU` activation function and an `Avgpool`. Every encoder outputs a latent code of size 16000 ($z_1 = 1600$, $z_2 = 1600$ and $z_3 = 1600$). We found that trying less than 1600 for latent code actually reduce the reconstruction results. When concatenated, this results in a latent code with a length of 48000.

These parallel encoders handle pixels from different scales, thereby yielding multi-scale features. To fuse these multi-scale features, we utilise a single decoder, as de-

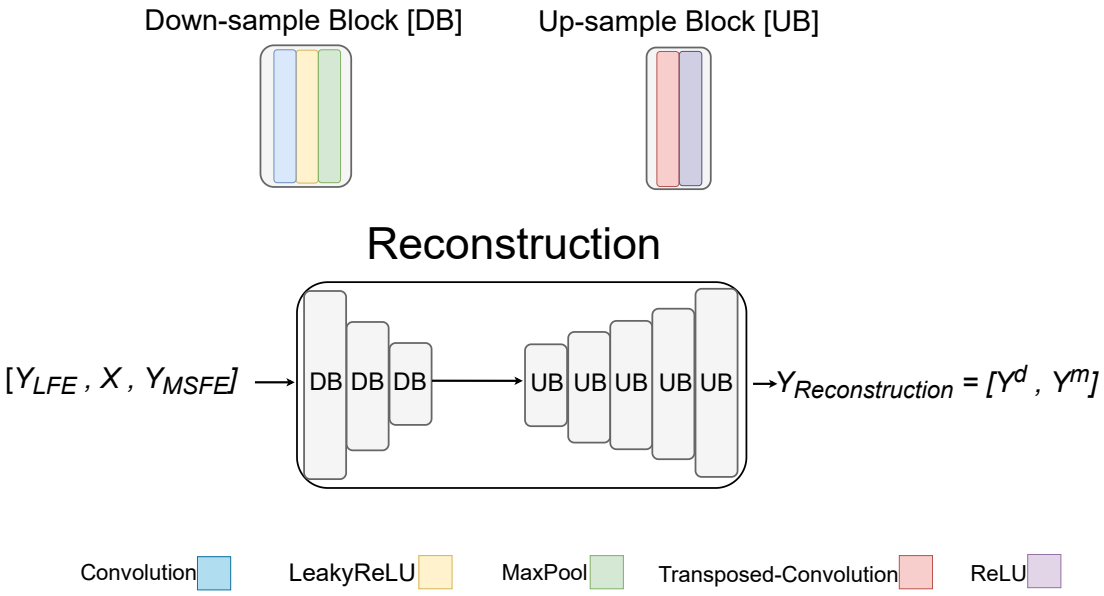


Figure 5.4: The reconstruction component leverages the original input X, the LFE output Y_{LFE}, and the MSFE output Y_{MSFE}. The model uses these inputs to determine the canonical form $Y_{Reconstruction}$ which consists of canonical form depth image Y^d and its mask Y^m. The network is similar to LFE network, containing down-sample and up-sample blocks.

scribed in Eq. 5.6.

$$Y_{MSFE} = D_{fuser}(z_1, z_2, z_3) \tag{5.6}$$

As an initial step, the 48000-d latent codes are processed through two MLP layers to identify inter-code relationships, ultimately generating 500 latent codes. Subsequently, six transpose-convolutions are applied. Following each convolution, a `ReLU` activation function is employed. The kernel sizes designated for these convolution layers are [3,3,6,2,2,2,3]. The output Y_{MSFE} has the spatial resolution aligned with the original input size. This design enables the association of multi-scale features with each input pixel. The overview of MSE is shown in figure 5.3.

5.2.3 Reconstruction component

Deformation involves transforming a shape from any pose to a default pose. In terms of an image, this means shifting the pixels to recreate a canonical pose. However, conventional convolution cannot adequately attend to long dependencies. As a solution, we generate both local and multi-scales features of the same size as the input image, allowing the reconstruction component Eq. 5.7 to access both feature types for each pixel.

Similar to LFE component, the reconstruction component incorporates four channels: the original input and its mask, local feature data generated by the LFE component, and multi-scale features produced by MSFE. Note combining features from different stages of the model help reduce vanishing gradient. The reconstruction component comprises N down-sample blocks and K up-sample blocks, where $N = 3$ and $K = 5$. Each down-sample block consists of a convolution layer, followed by a `LeakyReLU` and a `Maxpool` layer, with kernels of size 5 and stride 1. On the other hand, each up-sample block features a transpose-convolution and a `ReLU` layer, utilising kernels of sizes [5,3,5,2,2]. The final output from the reconstruction component $Y_{Reconstruction}$ is a reconstructed depth image alongside a reconstructed mask. The overview of reconstruction component is shown in figure 5.4.

$$Y_{Reconstruction} = reconstruction(X^d, X^m, Y_{LFE}, Y_{MSFE}) \tag{5.7}$$

5.2.4 Loss Function

The model employs two loss functions: depth loss and mask loss.

Depth Loss. We utilise the Mean Squared Error for the depth loss. However, we have modified this loss to focus on the foreground region.

$$L_{Depth} = \frac{1}{N}\sum_{i=1}^{N} \hat{y}_m y_m (\hat{y}_d - y_d)^2$$

Here, $\hat{y}_m$ and y_m denote the predicted mask and the ground truth mask, respectively. Likewise, $\hat{y}_d$ and y_d represent the predicted depth and the ground truth depth, respectively. By leveraging the intersection of the masks, we can exclude the background from the depth image, thereby reducing false positive predictions.

Mask Loss. For depth image reconstruction, we desire the model to concentrate on the target shape. Consequently, we aim for the model to learn the canonical form mask.

$$L_{Mask} = \frac{1}{N}\sum_{i=1}^{N} (\hat{y}_m - y_m)^2$$

Combined Loss. Since the model has two objectives, we introduce coefficients α and β to balance the training.

$$L_{weighted} = \alpha L_{Depth} + \beta L_{Mask}$$

5.3 Experiments

5.3.1 Training Details

The model was trained for 800 epochs. In the initial phase, specifically for the first 100 epochs, we prioritise mask learning. As mentioned in Eq. 5.8, depth images are sensitive to the intersection of masks; therefore, we set $\alpha = 10$ and $\beta = 1000$. In the subsequent 200 epochs, we leaned more towards the depth objective, setting both α and β to 1000. For the remaining epochs, we allow the model to focus primarily on the depth objective by setting $\alpha = 1000$ and $\beta = 100$. The learning rate is set to 0.001, and we employ the Adam optimiser [66].

5.3.2 Dataset

We conducted our experiments on three datasets, all of which contain non-rigid shapes. Specifically, the dataset from [113] features real human data. This dataset was constructed using the Civilian American and European Surface Anthropometry Resource (CAESAR) [119], wherein point clouds were fit to templates. In total, it comprises 40 subjects, equally split with 20 males and 20 females. Each subject is represented in 10 different poses.

The second dataset, also from [113], is a synthetic human dataset. It was created in a parameterised manner using 3D modelling software to control the shape and generate poses. This dataset contains 300 shapes, distributed among 15 subjects: 5 males, 5 females, and 5 children. Each subject has 20 poses.

While the aforementioned datasets focus on humans, real-life scenarios present a variety of non-human, non-rigid subjects. As such, we also chose the TOSCA dataset [19], which includes both humans and animals. In total, the dataset has 80 objects. Due to the varied nature of animals, the numbers of poses differ across objects: two males with 7 and 20 poses respectively; one female with 12 poses; one cat with 11 poses; one dog with 9 poses; one wolf with 3 poses; a horse with 8 poses; a centaur with 6 poses; and one gorilla with 4 poses.

For all datasets, the generation process is as follows: Each shape within the datasets is centred, after which we render an image of size 500×500. However, for the TOSCA dataset [19], the sizes of the shapes vary across classes, such as horses and cats. To address this, we scale the shapes to a fixed size (bounding box). We utilised blender for the dataset generation as we can bind python code to automate the process.

5.4 Evaluation

For canonical form results, the evaluation measure is typically based on retrieval results [110]. As previous works [110] [20], we utilised The Clock Matching and Bag-of-Features (CM-BOF) [78]. For retrieval results extraction. The framework starts by computing a descriptor for a given 3D shape. Initially, we centralise the mesh, normalise its scale, and employ a combination of principal component analysis (PCA) and rectilinearity for orientation normalisation. Following this, 66 depth images of the mesh are rendered from viewpoints situated at the vertices of a geometric figure resembling a soccer ball. Subsequently, SIFT features are extracted from these depth images. Using the bag-of-words method, we generate a histogram descriptor of length 1000 for each image of the shape. The degree of similarity between two shapes is determined by aggregating the similarities of their corresponding views.

The retrieval task involves ranking the shapes. For each shape in the dataset, we rank the remaining shapes in relation to it. Once ranked, we employ evaluation metrics to assess the retrieval outcomes. From the literature, we adopt four evaluation metrics: Nearest Neighbor (NN) where the 1-NN algorithm identifies the single nearest neighbour of a query point based on a distance metric (such as Euclidean distance) and assigns the category of this nearest neighbour to the query point. , First Tier (FT) refers to a metric that measures the precision at the first rank or the top-n results of the retrieval, assessing how many of the most relevant (or similar) items are correctly identified and ranked by the algorithm at the very top of its output list. Second Tier (ST) , While first tier focuses on the precision of the top-ranked results, second tier typically extends this evaluation to a broader set of top results, and Discounted Cumulative Gain (DCG) where DCG is a measure used to evaluate the effectiveness of ranking algorithms.

Comparison to prior work. To the best of our knowledge, there exists no learning-based canonical form model specifically tailored for non-rigid shapes. Consequently, all referenced works herein are non learning-based models.

The majority of the methods mentioned in the literature leverage the Multidimensional Scaling (MDS) technique [37]. Hence, MDS-based results are also included in our comparisons. MDS takes a distance as an input and calculates a Euclidean space embedding to retain that distance. For instance, Fast-MDS [39] projects geodesic distances into a Euclidean space, one dimension at a time. After determining geodesic distances across all vertex pairs, two vertices that are furthest apart in Euclidean space are selected for every dimension. Subsequently, the remaining vertices are projected onto the line formed by these two vertices.

Alternatively, *Non Metric MDS* emphasises preserving the ordering of distances rather than their exact values. It employs a stress function that incorporates both the geodesic and Euclidean distances of vertices, further optimised with a function emphasising dissimilarity.

Another method, *Least Squares MDS* [37], employs the SMACOF (Scaling by Majorising a Convex Function) algorithm. This iterative approach considers both geodesic and Euclidean distances. The Accelerated MDS method [125] was designed to offer a more efficient approximation of the pairwise geodesic distance maps, reducing computational burdens.

Furthermore, *Constrained MDS* [122] capitalises on the exact correspondence between an original shape and its Landmark MDS embedding. Through vertex adjustments and the utilisation of deformation regularisation energy, a detail-rich pose can be realised using MDS.

The *Global Point Signatures (GPS)* technique computes the embedding of a mesh. Initially, the mesh's discrete Laplace-Beltrami operator is computed using cotangent weights. The foremost smallest eigenvalues are then determined. Given the invariant nature of the Laplace-Beltrami operator's eigenspaces to metric-preserving deformations, the GPS embedding provides a pose-invariant representation of the mesh.

The *skeleton based* method [112] suggests that a skeleton is derived from a mesh to

Table 5.1: Retrieval results for Synthetic human dataset.

	NN	FT	ST	DCG
Classic MDS	0.10	0.22	0.39	0.54
Fast MDS	0.14	0.20	0.35	0.53
Non-metric MDS	0.09	0.24	**0.41**	0.55
Least Square MDS	0.01	0.13	0.31	0.45
Constrained MDS	0.04	0.14	0.25	0.46
GPS	0.40	0.20	0.32	0.56
Mesh Unfolding	0.04	0.18	0.34	0.49
Skeleton-based	0.01	0.14	0.32	0.46
Our	**0.51**	**0.32**	**0.41**	**0.63**

produce a canonical form. Following this, the Multidimensional SMACOF is utilised on the skeleton, positioning it into a standard pose.

Lastly, the *Detail-preserving Mesh Unfolding* method [123] is based on finite elements and omits the use of geodesics. This method, combining springs and finite elements, delivers superior outcomes concerning element inversions and retrieval performance.

5.4.1 Results

Our model trained on two datasets and tested on three as stated earlier in Section 5.3.2. For the synthetic human dataset [113], the results are shown in Table 5.1. The model is trained using a cross validation method where we do cross validation across the subjects and poses as poses are similar across the whole subjects. Specifically, the subjects and poses are split into groups. Every time, shapes belonging to a chosen group of subjects and a chosen group of poses are used as the test set, while we only use shapes not containing any of these subjects or any of these poses as the training set. This process ensures strict separation of training and test sets during cross validation.

Table 5.2: Retrieval results for real human dataset, trained on synthetic human dataset and tested on real human dataset.

	NN	FT	ST	DCG
Classic MDS	0.01	0.03	0.07	0.28
Fast MDS	0.00	0.02	0.04	0.27
Non-metric MDS	0.02	0.04	0.08	0.30
Least Square MDS	0.00	0.00	0.01	0.26
Constrained MDS	0.00	0.01	0.03	0.27
GPS	**0.07**	**0.06**	**0.12**	**0.33**
Mesh Unfolding	0.00	0.01	0.03	0.28
Skeleton-based	0.01	0.01	0.02	0.27
Our	0.04	0.023	0.051	0.23

Table 5.3: Retrieval results for TOSCA dataset.

	NN	FT	ST	DCG
Classic MDS	0.74	0.54	0.80	0.80
Fast MDS	0.73	0.52	0.77	0.77
Non-metric MDS	0.76	0.67	0.87	0.85
Least Square MDS	0.79	0.63	0.86	0.84
Constrained MDS	0.88	0.71	**0.89**	**0.89**
GPS	0.71	0.52	0.72	0.76
Mesh Unfolding	0.88	0.65	0.86	0.85
Skeleton-based	0.78	0.62	0.85	0.84
Our	**0.91**	**0.76**	0.80	**0.89**

The same protocol is applied to other experiments as well.

For the real human dataset [113], the results are shown in Table 5.2. As stated earlier, due to the nature of the dataset there are no T-poses (ground truth pose), so as a result

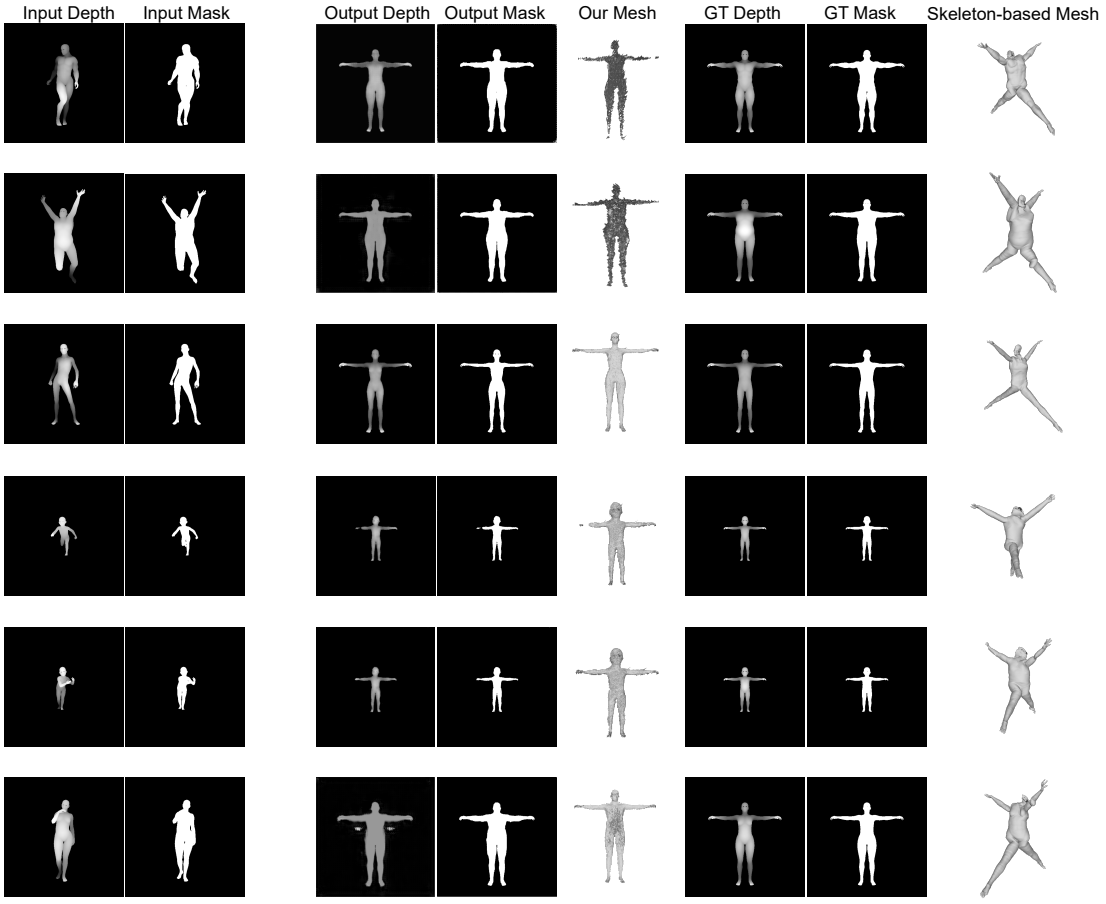


Figure 5.5: Some canonical form results on the synthetic dataset. The meshes are extracted from the output depth images.

we trained the model on the synthetic dataset and then tested on the real human dataset. Lastly, for the TOSCA dataset [19] the results are shown in Table 5.3. In the quantitative results, our model outperforms the state-of-the-art models, except for real human results, our result was the second on the NN metric, probably due to the domain gap. All the methods performed quite poorly on this dataset, indicating the difficulties for this task. For the qualitative results, for synthetic human dataset [113], the results are shown in Figure 5.5, and for real human dataset [113], the results are shown in Figure 5.6. For the TOSCA dataset [19] the results are shown in Figure 5.7.

Table 5.4: Ablation study on LFE and MSFE on TOSCA dataset.

	NN	FT	ST	DCG
Complete	**0.91**	**0.76**	**0.80**	**0.89**
without LFE	0.88	0.62	0.76	0.84
without MSFE	0.72	0.48	0.61	0.73

5.4.2 Ablation Studies

In this section, we conduct two ablation studies using the TOSCA dataset, chosen due to its varied content.

LFE. Training the model without the LFE component resulted in lower performance compared to the full model, as shown in Figure 5.9. Results are presented in Table 5.4.

MSFE. Without the MSFE component, the model's performance was worse compared to the complete model (Table 5.4). As observed in Figure 5.8, for classes like dog or cat (which do not have hand or T-pose features), the model could reconstruct the canonical pose. However, for shapes with outstretched hands and legs, such as centaur or human, the results often missed those body parts.

5.5 Conclusion

In conclusion, our research presents a novel learning-based approach that transforms a single depth image into a standard pose. Utilising both a depth image and its associated mask, our model is able to estimate the canonical form even for unseen poses. As illustrated in Figure 5.1, the model's foundation is an encoder-decoder structure designed to yield intricate features. We innovatively incorporate parallel encoders with sparse convolution, allowing the capture of diverse neighbours and subsequently mixing multi-scale features crucial for maintaining shape integrity. These amalgamated

features act as the foundational layer to produce detailed characteristics. Conclusively, the encoder-decoder architecture is harnessed to approximate the depth image in its canonical form.

5.6 Limitations

The model exhibits sub-optimal performance on real human datasets, and the variety of objects currently supported is limited. Additionally, the model primarily deals with 2D depth images, making it suitable for 3D reconstruction and single-view applications, but not directly applicable to complete 3D shapes. Transitioning from 3D shapes to these 2D images could result in a loss of certain shape details. Consequently, the output might not faithfully represent all aspects of the 3D shape in its canonical pose.

5.7 Summary

In this work, we presented our method and experiments on depth image reconstruction. We introduced three components: a Local Feature Extractor to capture local features, a Multi-Scale Feature Extractor to capture features across different scales, and a reconstruction component specifically for reconstructing depth images. During evaluation, we tested the model on three distinct datasets comprising various objects and demonstrated favourable results compared to other methods.

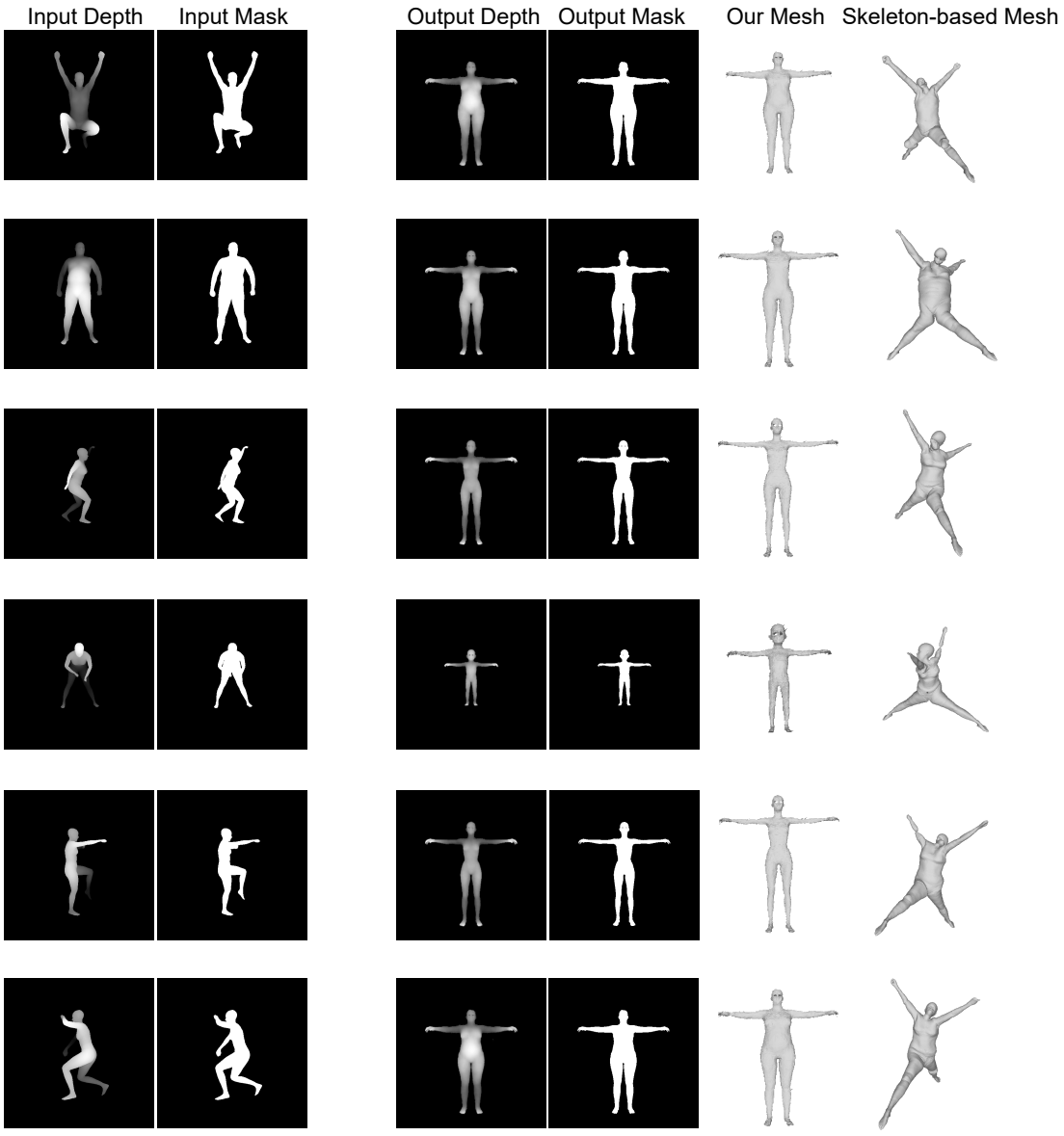


Figure 5.6: Some canonical form results on real dataset. The model is first trained on synthetic human dataset and then tested on real human dataset. There are no ground truth available (T-pose). our meshes are extracted from the output depth images.

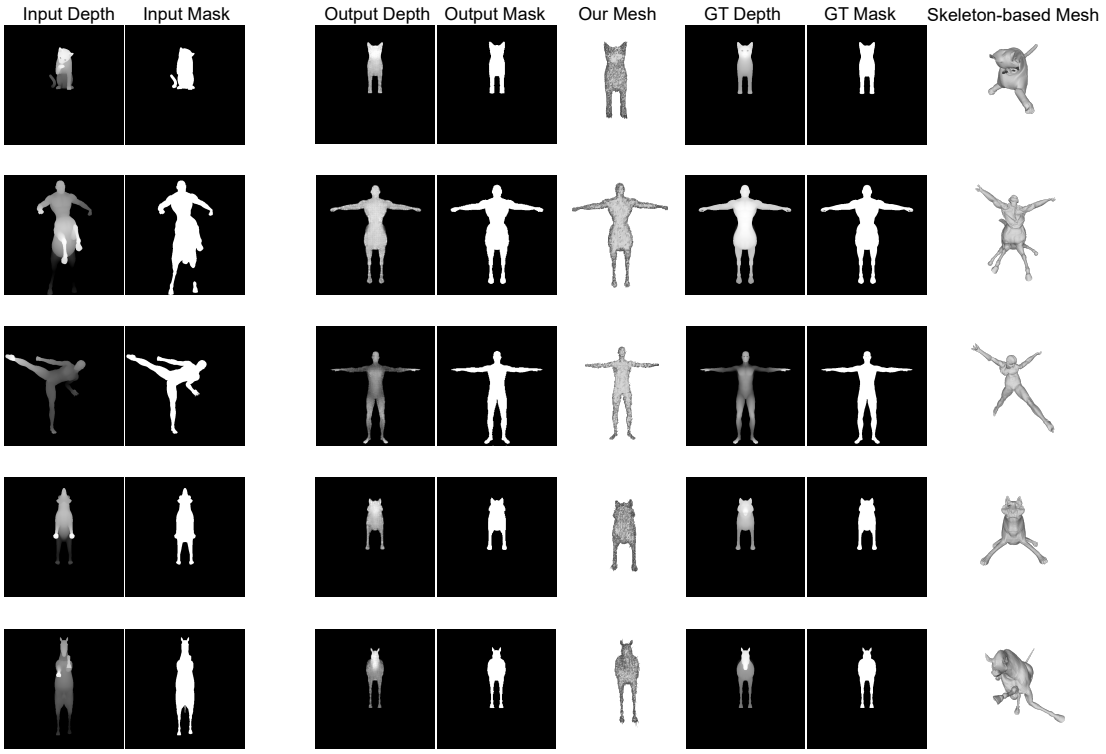


Figure 5.7: Some canonical form results on the TOSCA dataset. The meshes are extracted from the output depth images.

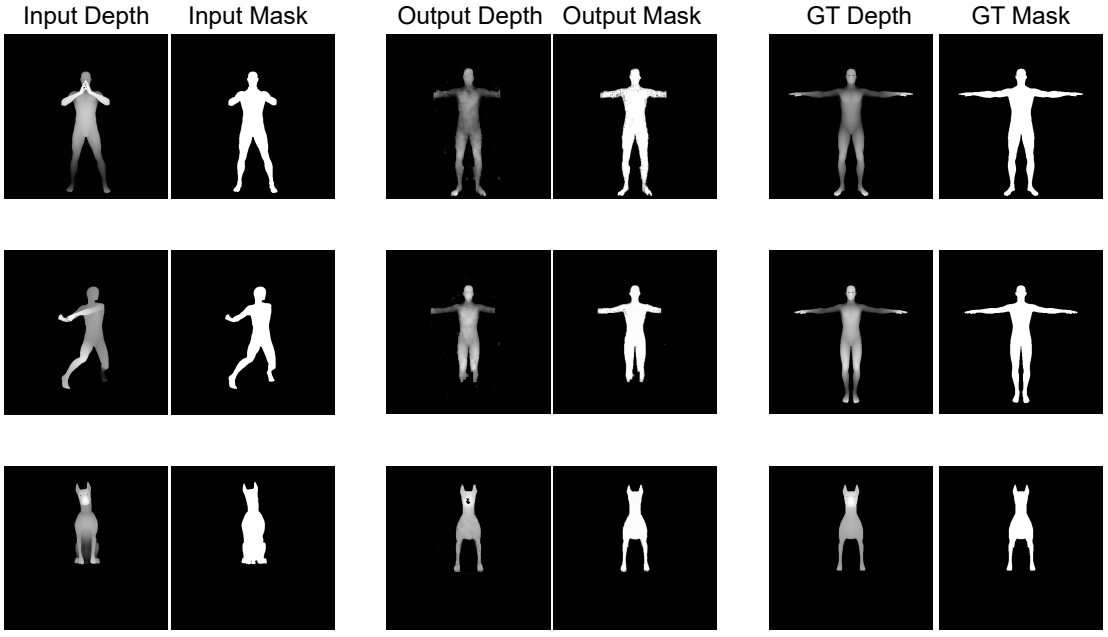


Figure 5.8: Ablation results for MSFE show that: without MSFE, the model is unable to estimate long dependencies.

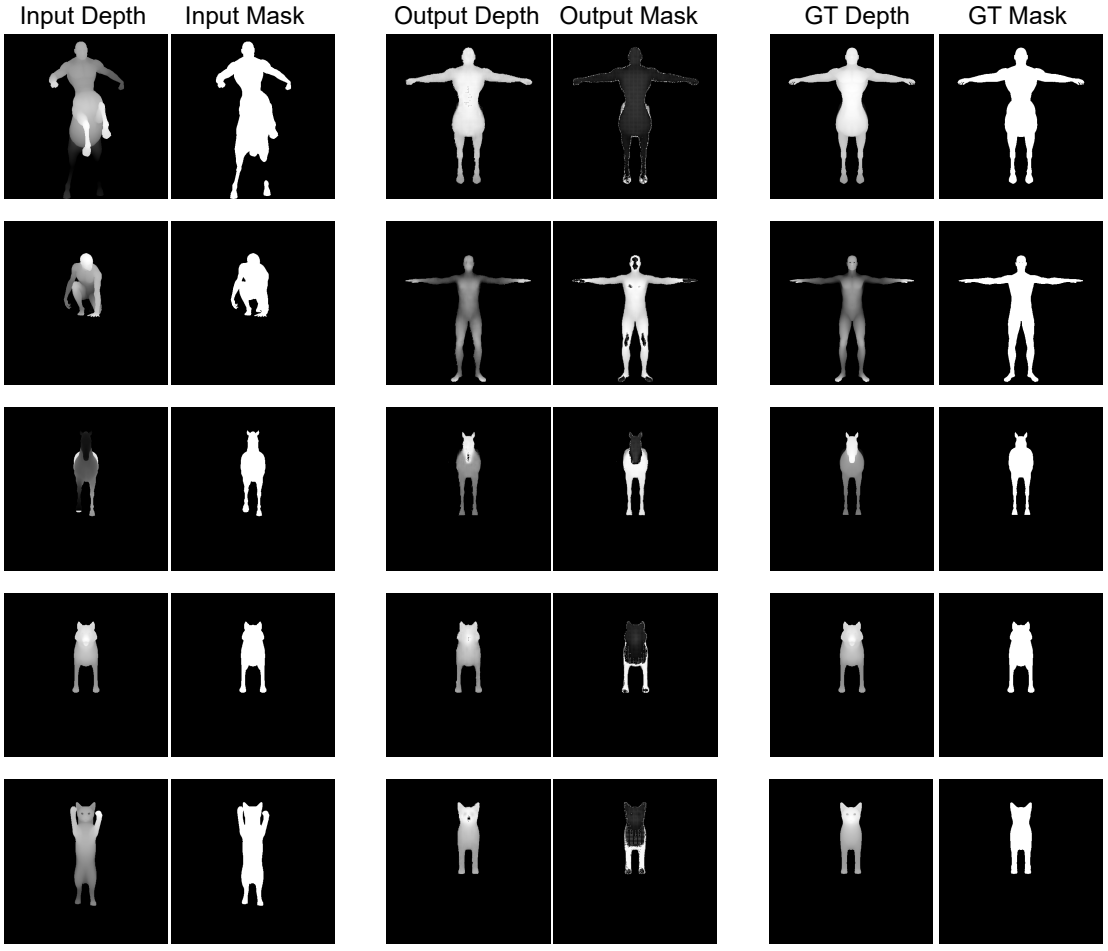


Figure 5.9: Ablation results for LFE.

Chapter 6

Conclusion

Overview

This thesis undertakes an in-depth exploration of tasks related to three-dimensional (3D) reconstruction. Our comprehensive investigation spans multiple elements of this domain, starting with the full reconstruction of rigid shapes from depth images. This is followed by an examination of appropriate evaluation methods for assessing the reconstructed shapes, ensuring a fair and valid appraisal of the outcomes. Finally, we delve into the aspect of disentangling deformations in the context of non-rigid shape reconstruction, thereby offering a holistic perspective on 3D shape reconstruction.

The comprehensive investigation undertaken within this research is encapsulated in Section 6.1, providing a summary of our methodology, results, and insights. Subsequent sections from 6.2 to 6.4 offer a detailed review of the key findings from each chapter, emphasising our contributions to the field of 3D reconstruction.

In conclusion, Section 6.5 envisages future directions for this research, indicating the prospective avenues to further enhance our understanding and capability in this area. It provides a road map for subsequent investigations, thereby ensuring continuity in our academic exploration of 3D reconstruction tasks.

6.1 Summary

This research presents innovative models and techniques aimed at addressing the challenges of 3D reconstruction from a single depth image. We navigate issues with occluded views and complex object representations by deploying a self-attention mechanism, a dynamic latent code selection, and a cascaded sequential encoder-decoder configuration. These enhancements ensure the efficient and effective reconstruction of shapes.

Furthermore, we introduce a unified representation that mitigates the issues of distortion that can arise when converting between different 3D representations like meshes, point clouds, voxels, or implicit forms like Signed Distance Fields or Unsigned Distance Fields. Our unique approach employs a Regular dodecahedron for rendering, where each face serves as a camera and light source point, ensuring a uniform coverage of the shape. We further refine the accuracy of our model with the mask-SSIM (Structural Similarity Index), a variety of metrics, and forward feature selection, aiming to align closely with human perception.

Finally, we then tackle the issue of non-rigid shapes, particularly their susceptibility to deformation. We propose a model that disentangles deformation by transforming depth images into a canonical form and fusing differentially distorted features for more detailed reconstructions. Our results in generating a canonical form surpass those of previous studies, reinforcing the effectiveness of our approach.

6.2 3D Reconstruction from Single Depth Images

Chapter 3 probes the complexities associated with reconstructing volumetric entities from a single depth image, an endeavour that presents substantial challenges, particularly when dealing with occluded views. This chapter outlines our novel model, which

successfully mitigates these hurdles by employing a self-attention mechanism to focus on global spatial features.

In the context of this complex task, we introduced a classifier to aid the model in finding boundaries between categories. This classification mechanism plays an important role in improving the overall reconstruction process, leading to more accurate and distinct object forms.

In a further enhancement, we developed a mechanism for dynamic latent code selection. This mechanism enhances efficiency by ensuring that the reconstruction process utilises only the crucial codes, thereby eliminating redundancy and enhancing the model's performance.

Moreover, the chapter presents our approach of adopting a sequential encoder-decoder configuration in a cascade format to decompose the reconstruction process into manageable stages. This sequential and layered approach to reconstruction helps make the completion more manageable and easier to learn, thus improving the task's overall comprehensibility and effectiveness.

The performance of the proposed model was evaluated under a variety of conditions to measure its robustness and versatility. Firstly, the model was subjected to tests within a single category scenarios, thereby examining its ability to handle specific instances of volumetric reconstruction.

Following this, we extended the testing environment to multi-category contexts, providing a more complex, real-world-like environment. This helped assess the model's capacity to handle greater variety and complexity, which is integral to its applicability in diverse practical situations.

In an additional and critical round of evaluation, we trained the model on certain predefined categories and then tested it on categories that had been previously unseen during the training phase. This experiment was aimed at understanding the model's adaptability and generalisability. These tests confirmed our model's efficacy in dealing

with novel scenarios, further establishing its superior performance over state-of-the-art models.

6.3 Rendering based 3D Shape Evaluation

The complexity of three-dimensional (3D) reconstruction tasks necessitates an approach that can deal with a variety of representations, including meshes, point clouds, voxels, or implicit representations like Signed Distance Fields (SDF) or Unsigned Distance Fields (UDF). However, each of these representations requires its own specific evaluation metric tailored to the inherent characteristics of the representation. Furthermore, attempts to convert these representations into a unified form may result in distortion, potentially compromising the integrity of the original data.

To address mentioned issue, In chapter 4 we propose a novel unified representation that obviates the need for multiple evaluation metrics and mitigates the risk of distortion. Our method based on rendering the shape using an Regular dodecahedron where each face is used as a position of a camera and source of light while the target is centred in the middle of the dodecahedron, which being a Platonic solid, offers uniform coverage of the shape, providing a more holistic and unbiased representation.

We render 12 images that encompass the entire shape, applying various styles and shaders to simulate different scenarios. Subsequently, we employ the mask-SSIM (Structural Similarity Index) to differentiate between each reconstructed shape and the ground truth. Mask-SSIM is a modified version of the SSIM that is masked to separate the background from the foreground, thus offering a more focused comparison.

We further refine our evaluation by generating a variety of metric results and integrating them to form the input for our neural network model. The objective is to derive a score that aligns closely with human perception, thereby enhancing the relevance and practical utility of the model.

Our approach also incorporates sequential forward feature selection to determine which features have the most significant impact on the results. This allows us to effectively utilise the model, focusing on the most influential attributes and discarding redundant ones.

The model underwent extensive testing and training using three diverse datasets, where it consistently demonstrated superior performance. In addition to internal cross-validation within each dataset, we conducted cross-dataset validation, selecting features based on one dataset and then training and testing another dataset using those features. This approach further validated the robustness and versatility of our model, reinforcing its potential for practical applications in 3D reconstruction tasks.

6.4 Learning to Generate Canonical Forms for Single Depth Images

Chapter 3 of this thesis addresses the prominent research topic of 3D reconstruction from a single depth image. While this field of research has seen considerable advancements, the main focus has typically been on rigid shapes. Non-rigid shapes, characterised by their susceptibility to deformation, present unique challenges that limit a model's ability to accurately reconstruct the complete shape.

To bridge this research gap, Chapter 5 introduces our novel model designed to disentangle the deformation inherent in non-rigid shapes. Our approach involves deforming a depth image into a canonical form, using various distillation to encompass a broader global spatial structure. A range of features are then fused, creating a more detailed feature vector, thereby enhancing the precision of the canonical form reconstruction.

In a further advancement, we propose a supplementary model aimed at completing the shape after the disentanglement process. Notably, our results in generating a canonical form surpassed those of previous studies, demonstrating the superiority of our

approach.

To ensure the robustness and generalizability of our model, we conducted extensive cross-validation. Additionally, to mimic real-world scenarios, we adopted a 'leave-one-out' strategy for deformations and subjects. This approach involved training the model on all but three deformation and five subjects, which was then used as a test case. Such an experimental design enabled us to more accurately evaluate our model's performance under unseen conditions, thereby reinforcing its potential for practical application in the field of 3D reconstruction tasks.

6.5 Future Work

6.5.1 3D Reconstruction from Depth Images

The advancements made in Chapter 3 provide a solid foundation for future research in 3D reconstruction from a single depth image. Based on the findings and developments in this thesis, the following areas have been identified as promising avenues for future work:

1. **Enhancing Self-Attention Mechanisms:** The self-attention mechanism used in this study proved to be an effective strategy for addressing global spatial features. Future work could consider enhancing this mechanism with adaptive attention strategies, allowing the model to better account for the variety and complexity of features in different object categories.

2. **Optimising Latent Code Selection:** The dynamic latent code selection mechanism used in this study improved computational efficiency and results, but future studies could focus on further optimisations. This could involve developing methods to dynamically adapt the number of the latent codes based on the complexity of the object being reconstructed.

3. **Expanding the Sequential Encoder-Decoder Framework:** The sequential encoder-decoder configuration was pivotal in our model's success. Future research could explore variations of this architecture, such as incorporating recurrent or transformer structures, to further enhance performance.

4. **Dealing with Occlusions:** Despite considerable progress made in handling occluded views, this remains a challenging aspect of 3D reconstruction.Future studies might explore generation capabilities to offer multiple solutions.

Each of these potential areas of research could yield significant contributions to the field of 3D reconstruction and could enhance our understanding and capabilities in tackling complex 3D tasks.

6.5.2 Rendering based 3D Shape Evaluation

Building on the innovations of our research, several areas for future work emerge. These include:

1. **Extend Unified Representation:** The unified representation proposed in our study proved effectiveness; however we could extend experiments to support implicit representation (SDF and UDF), as the current implementation does not support Implicit representation.

2. **Improved Rendering Techniques:** We utilised a dodecahedron-based method to determine the viewpoint for rendering process to derive the uniform representation. Exploration of alternative rendering styles and shaders or another platonic shape could potentially yield more accurate or efficient results.

These potential directions promise to advance the state of the art in 3D reconstruction, bringing us closer to practical, efficient, and accurate methods for complex 3D tasks.

6.5.3 Learning to Generate Canonical Forms for Single Depth Images

The findings and advancements of our research in the area of 3D reconstruction from a single depth image, particularly in the context of non-rigid shapes, pave the way for various future endeavours. Some of the potential directions for future work include:

1. **Extending the non-rigid classes:** While our approach effectively disentangles the deformation inherent in non-rigid shapes, the dataset encompasses only a few classes. Future work could explore extending the dataset to include a wider variety of shapes.

2. **Improving the Canonical Form Generation:** Our model excels at estimating the canonical pose, which is a pose chosen during training. In the future, we could enhance it to automatically estimate a suitable pose as the canonical form, reducing the burden of applying the technique.

3. **Enhancing the Shape Completion Model:** We have proposed a learning-based model for canonical pose estimation using depth images. A potential future work would be to incorporate shape completion to reconstruct non-rigid 3D shapes from depth images.

These future directions offer exciting prospects for the further advancement of 3D reconstruction, particularly in the context of non-rigid shapes, and have the potential to make a significant impact in both academia and industry.

Bibliography

[1] Ilyass Abouelaziz, Aladine Chetouani, Mohammed El Hassouni, and Hocine Cherifi. No-reference mesh visual quality assessment using graph-based deep learning. In *2021 IEEE 23rd International Workshop on Multimedia Signal Processing (MMSP)*, pages 1–6. IEEE, 2021.

[2] Ilyass Abouelaziz, Aladine Chetouani, Mohammed El Hassouni, Hocine Cherifi, and Longin Jan Latecki. Learning graph convolutional network for blind mesh visual quality assessment. *IEEE Access*, 9:108200–108211, 2021.

[3] Ilyass Abouelaziz, Aladine Chetouani, Mohammed El Hassouni, Longin Jan Latecki, and Hocine Cherifi. Convolutional neural network for blind mesh visual quality assessment using 3d visual saliency. In *2018 25th IEEE International Conference on Image Processing (ICIP)*, pages 3533–3537. IEEE, 2018.

[4] Ilyass Abouelaziz, Aladine Chetouani, Mohammed El Hassouni, Longin Jan Latecki, and Hocine Cherifi. Mesh visual quality based on the combination of convolutional neural networks. In *2019 Ninth International Conference on Image Processing Theory, Tools and Applications (IPTA)*, pages 1–5. IEEE, 2019.

[5] Ilyass Abouelaziz, Aladine Chetouani, Mohammed El Hassouni, Longin Jan Latecki, and Hocine Cherifi. 3D visual saliency and convolutional neural network for blind mesh quality assessment. *Neural Computing and Applications*, 32(21):16589–16603, 2020.

[6] Ilyass Abouelaziz, Aladine Chetouani, Mohammed El Hassouni, Longin Jan Latecki, and Hocine Cherifi. Combination of handcrafted and deep learning-based features for 3D mesh quality assessment. In *2020 IEEE International Conference on Image Processing (ICIP)*, pages 171–175. IEEE, 2020.

[7] Ilyass Abouelaziz, Aladine Chetouani, Mohammed El Hassouni, Longin Jan Latecki, and Hocine Cherifi. No-reference mesh visual quality assessment via ensemble of convolutional neural networks and compact multi-linear pooling. *Pattern Recognition*, 100:107174, 2020.

[8] Ilyass Abouelaziz, Mohammed El Hassouni, and Hocine Cherifi. A curvature based method for blind mesh visual quality assessment using a general regression neural network. In *2016 12th International Conference on Signal-Image Technology & Internet-Based Systems (SITIS)*, pages 793–797. IEEE, 2016.

[9] Ilyass Abouelaziz, Mohammed El Hassouni, and Hocine Cherifi. No-reference 3D mesh quality assessment based on dihedral angles model and support vector regression. In *Image and Signal Processing: 7th International Conference, ICISP 2016, Trois-Rivières, QC, Canada, May 30-June 1, 2016, Proceedings 7*, pages 369–377. Springer, 2016.

[10] Ilyass Abouelaziz, Mohammed El Hassouni, and Hocine Cherifi. A convolutional neural network framework for blind mesh visual quality assessment. In *2017 IEEE International Conference on Image Processing (ICIP)*, pages 755–759. IEEE, 2017.

[11] Ilyass Abouelaziz, Mohammed El Hassouni, and Hocine Cherifi. Blind 3D mesh visual quality assessment using support vector regression. *Multimedia Tools and Applications*, 77:24365–24386, 2018.

[12] Ilyass Abouelaziz, Mounir Omari, Mohammed El Hassouni, and Hocine Cherifi. Reduced reference 3D mesh quality assessment based on statistical models. In *2015 11th International Conference on Signal-Image Technology & Internet-Based Systems (SITIS)*, pages 170–177. IEEE, 2015.

[13] Rémy Alcouffe, Simone Gasparini, Geraldine Morin, and Sylvie Chambon. Blind quality of a 3D reconstructed mesh. In *2022 IEEE International Conference on Image Processing (ICIP)*, pages 3406–3410. IEEE, 2022.

[14] Evangelos Alexiou and Touradj Ebrahimi. Towards a point cloud structural similarity metric. In *2020 IEEE International Conference on Multimedia & Expo Workshops (ICMEW)*, pages 1–6. IEEE, 2020.

[15] Antonio Alliegro, Diego Valsesia, Giulia Fracastoro, Enrico Magli, and Tatiana Tommasi. Denoise and contrast for category agnostic shape completion. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 4629–4638, 2021.

[16] Dominik Bauer, Timothy Patten, and Markus Vincze. Reagent: Point cloud registration using imitation and reinforcement learning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 14586–14594, 2021.

[17] Zhe Bian, Shi-Min Hu, and Ralph R Martin. Evaluation for small visual difference between conforming meshes on strain field. *Journal of Computer Science and Technology*, 24(1):65–75, 2009.

[18] Andrew Brock, Theodore Lim, James M Ritchie, and Nick Weston. Generative and discriminative voxel modeling with convolutional neural networks. *arXiv:1608.04236*, 2016.

[19] Alexander M Bronstein, Michael M Bronstein, and Ron Kimmel. *Numerical geometry of non-rigid shapes*. Springer Science & Business Media, 2008.

[20] AM Bronstein, MM Bronstein, U Castellani, B Falcidieno, A Fusiello, Afzal Godil, LJ Guibas, I Kokkinos, Zhouhui Lian, M Ovsjanikov, et al. SHREC 2010: robust large-scale shape retrieval benchmark. *Proc. 3DOR*, 5(4):1–8, 2010.

[21] B Bustos, H Tabia, JP Vandeborre, and R Veltkamp. Coulomb shapes: Using electrostatic forces for deformation-invariant shape representation. In *Proceedings of the 7th eurographics workshop on 3D Object Retrieval*, pages 9–15, 2014.

[22] Yingjie Cai, Kwan-Yee Lin, Chao Zhang, Qiang Wang, Xiaogang Wang, and Hongsheng Li. Learning a structured latent space for unsupervised point cloud completion. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 5543–5553, 2022.

[23] John Canny. A computational approach to edge detection. *IEEE Transactions on pattern analysis and machine intelligence*, PAMI-8(6):679–698, 1986.

[24] Angel X Chang, Thomas Funkhouser, Leonidas Guibas, Pat Hanrahan, Qixing Huang, Zimo Li, Silvio Savarese, Manolis Savva, Shuran Song, Hao Su, et al. ShapeNet: An information-rich 3D model repository. *arXiv preprint arXiv:1512.03012*, 2015.

[25] Weikai Chen, Cheng Lin, Weiyang Li, and Bo Yang. 3PSDF: Three-pole signed distance function for learning surfaces with arbitrary topologies. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 18522–18531, 2022.

[26] Zhang Chen, Yinda Zhang, Kyle Genova, Sean Fanello, Sofien Bouaziz, Christian Häne, Ruofei Du, Cem Keskin, Thomas Funkhouser, and Danhang Tang. Multiresolution deep implicit functions for 3D shape representation. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 13087–13096, 2021.

[27] An-Chieh Cheng, Xueting Li, Sifei Liu, Min Sun, and Ming-Hsuan Yang. Autoregressive 3D shape generation via canonical mapping. In *Computer Vision–ECCV 2022: 17th European Conference, Tel Aviv, Israel, October 23–27, 2022, Proceedings, Part III*, pages 89–104. Springer, 2022.

[28] Aladine Chetouani. Three-dimensional mesh quality metric with reference based on a support vector regression model. *Journal of Electronic Imaging*, 27(4):043048–043048, 2018.

[29] Aladine Chetouani, Maurice Quach, Giuseppe Valenzise, and Frédéric Dufaux. Convolutional neural network for 3D point cloud quality assessment with reference. In *2021 IEEE 23rd International Workshop on Multimedia Signal Processing (MMSP)*, pages 1–6. IEEE, 2021.

[30] Aladine Chetouani, Maurice Quach, Giuseppe Valenzise, and Frédéric Dufaux. Deep learning-based quality assessment of 3D point clouds without reference. In *2021 IEEE International Conference on Multimedia & Expo Workshops (ICMEW)*, pages 1–6. IEEE, 2021.

[31] G.K. Cheung, T. Kanade, J.Y. Bouguet, and M. Holler. A real time system for robust 3D voxel reconstruction of human motions. In *IEEE CVPR*, volume 2, pages 714–720, 2000.

[32] Julian Chibane, Thiemo Alldieck, and Gerard Pons-Moll. Implicit functions in feature space for 3D shape reconstruction and completion. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 6970–6981, 2020.

[33] Dan Claudiu Cireşan, Ueli Meier, Luca Maria Gambardella, and Jürgen Schmidhuber. Deep, big, simple neural nets for handwritten digit recognition. *Neural computation*, 22(12):3207–3220, 2010.

[34] A. Dai, C. Ruizhongtai Qi, and M. Nießner. Shape completion using 3D-encoder-predictor CNNs and shape synthesis. In *CVPR*, pages 5868–5877, 2017.

[35] João Pedro Carvalho de Souza, Carlos M Costa, Luís F Rocha, Rafael Arrais, A Paulo Moreira, EJ Solteiro Pires, and José Boaventura-Cunha. Reconfigurable grasp planning pipeline with grasp synthesis and selection applied to picking operations in aerospace factories. *Robotics and Computer-Integrated Manufacturing*, 67:102032, 2021.

[36] Shivam Duggal and Deepak Pathak. Topologically-aware deformation fields for single-view 3Dreconstruction. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 1536–1546, 2022.

[37] Asi Elad and Ron Kimmel. On bending invariant signatures for surfaces. *IEEE Transactions on pattern analysis and machine intelligence*, 25(10):1285–1295, 2003.

[38] Clemens Eppner, Arsalan Mousavian, and Dieter Fox. ACRONYM: A large-scale grasp dataset based on simulation. In *2021 IEEE International Conference on Robotics and Automation (ICRA)*, pages 6222–6227. IEEE, 2021.

[39] Christos Faloutsos and King-Ip Lin. FastMap: A fast algorithm for indexing, data-mining and visualization of traditional and multimedia datasets. In *Proceedings of the 1995 ACM SIGMOD international conference on Management of data*, pages 163–174, 1995.

[40] 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.

[41] Zhaoxin Fan, Hongyan Liu, Jun He, Qi Sun, and Xiaoyong Du. A graph-based one-shot learning method for point cloud recognition. In *Computer Graphics Forum*, volume 39, pages 313–323. Wiley Online Library, 2020.

[42] Jonathan Freer, Kwang Moo Yi, Wei Jiang, Jongwon Choi, and Hyung Jin Chang. Novel-view synthesis of human tourist photos. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*, pages 3069–3076, 2022.

[43] Jonas Gehring, Michael Auli, David Grangier, and Yann N Dauphin. A convolutional encoder model for neural machine translation. *arXiv preprint arXiv:1611.02344*, 2016.

[44] Kyle Genova, Forrester Cole, Avneesh Sud, Aaron Sarna, and Thomas Funkhouser. Local deep implicit functions for 3D shape. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 4857–4866, 2020.

[45] Ian Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, and Yoshua Bengio. Generative adversarial nets. *Advances in neural information processing systems*, 27, 2014.

[46] Benoit Guillard, Edoardo Remelli, Artem Lukoianov, Stephan R Richter, Timur Bagautdinov, Pierre Baque, and Pascal Fua. DeepMesh: Differentiable isosurface extraction. *arXiv preprint arXiv:2106.11795*, 2021.

[47] I. Gulrajani, F. Ahmed, M. Arjovsky, V. Dumoulin, and A. Courville. Improved training of Wasserstein GANs. In *NIPS*, pages 5767–5777, 2017.

[48] Jinjiang Guo, Vincent Vidal, Irene Cheng, Anup Basu, Atilla Baskurt, and Guillaume Lavoue. Subjective and objective visual quality assessment of textured 3D meshes. *ACM Transactions on Applied Perception (TAP)*, 14(2):1–20, 2016.

[49] Sara Hahner and Jochen Garcke. Mesh convolutional autoencoder for semi-regular meshes of different sizes. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*, pages 885–894, 2022.

[50] Hela Haj Mohamed, Samir Belaid, and Wady Naanaa. Local feature-based 3D canonical form. In *Representations, Analysis and Recognition of Shape and*

Motion from Imaging Data: 6th International Workshop, RFMI 2016, Sidi Bou Said Village, Tunisia, October 27-29, 2016, Revised Selected Papers 6, pages 3–14. Springer, 2017.

[51] Hela Haj Mohamed, Samir Belaid, Wady Naanaa, and Lotfi Ben Romdhane. Local commute-time guided MDS for 3D non-rigid object retrieval. *Applied Intelligence*, 48:2873–2883, 2018.

[52] C. Hane, S. Tulsiani, and J. Malik. Hierarchical surface prediction for 3D object reconstruction. In *Intl. Conf. 3D Vision (3DV)*, 2017.

[53] 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.

[54] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 770–778, 2016.

[55] Paul Henderson and Vittorio Ferrari. Learning single-image 3D reconstruction by generative modelling of shape, pose and shading. *International Journal of Computer Vision*, 128(4):835–854, 2020.

[56] Tao Hu, Zhizhong Han, and Matthias Zwicker. 3D shape completion with multi-view consistent inference. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 34, pages 10997–11004, 2020.

[57] Tao Hu, Liwei Wang, Xiaogang Xu, Shu Liu, and Jiaya Jia. Self-supervised 3D mesh reconstruction from single images. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 6002–6011, 2021.

[58] Qidong Huang, Xiaoyi Dong, Dongdong Chen, Hang Zhou, Weiming Zhang, and Nenghai Yu. Shape-invariant 3D adversarial point clouds. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 15335–15344, 2022.

[59] Zitian Huang, Yikuan Yu, Jiawen Xu, Feng Ni, and Xinyi Le. PF-Net: Point fractal network for 3D point cloud completion. In *Proceedings of the IEEE/CVF*

Conference on Computer Vision and Pattern Recognition, pages 7662–7670, 2020.

[60] Zixuan Huang, Stefan Stojanov, Anh Thai, Varun Jampani, and James M Rehg. Planes vs. chairs: Category-guided 3D shape learning without any 3D cues. In *Computer Vision–ECCV 2022: 17th European Conference, Tel Aviv, Israel, October 23–27, 2022, Proceedings, Part I*, pages 727–744. Springer, 2022.

[61] Daniel P Huttenlocher, Gregory A. Klanderman, and William J Rucklidge. Comparing images using the Hausdorff distance. *IEEE Transactions on pattern analysis and machine intelligence*, 15(9):850–863, 1993.

[62] Abouelaziz Ilyass, Chetouani Aladine, Mohammed El Hassouni, and Cherifi Hocine. Full reference mesh visual quality assessment using pre-trained deep network and quality indices. In *2019 15th International Conference on Signal-Image Technology & Internet-Based Systems (SITIS)*, pages 693–697. IEEE, 2019.

[63] A. Janoch, S. Karayev, Y. Jia, J. T. Barron, M. Fritz, K. Saenko, and T. Darrell. A category-level 3-D object dataset: Putting the Kinect to work. In *ICCV Workshop*, 2011.

[64] Jiongchao Jin, Huanqiang Xu, Pengliang Ji, and Biao Leng. IMC-NET: Learning implicit field with corner attention network for 3D shape reconstruction. In *2022 IEEE International Conference on Image Processing (ICIP)*, pages 1591–1595. IEEE, 2022.

[65] Majdi Jribi and Faouzi Ghorbel. A novel canonical form for the registration of non rigid 3D shapes. In *Computer Analysis of Images and Patterns: 16th International Conference, CAIP 2015, Valletta, Malta, September 2-4, 2015, Proceedings, Part II 16*, pages 230–241. Springer, 2015.

[66] D.P. Kingma and J. Ba. Adam: A method for stochastic optimization. *arXiv:1412.6980*, 2014.

[67] Alex Krizhevsky. One weird trick for parallelizing convolutional neural networks. *arXiv preprint arXiv:1404.5997*, 2014.

[68] Andrey Kurenkov, Jingwei Ji, Animesh Garg, Viraj Mehta, JunYoung Gwak, Christopher Choy, and Silvio Savarese. DeformNet: Free-form deformation

network for 3D shape reconstruction from a single image. In *2018 IEEE Winter Conference on Applications of Computer Vision (WACV)*, pages 858–866. IEEE, 2018.

[69] Charis Lanaras, José Bioucas-Dias, Silvano Galliani, Emmanuel Baltsavias, and Konrad Schindler. Super-resolution of Sentinel-2 images: Learning a globally applicable deep neural network. *ISPRS Journal of Photogrammetry and Remote Sensing*, 146:305–319, 2018.

[70] Guillaume Lavoué. A local roughness measure for 3D meshes and its application to visual masking. *ACM Transactions on Applied perception (TAP)*, 5(4):1–23, 2009.

[71] Guillaume Lavoué. A multiscale metric for 3D mesh visual quality assessment. In *Computer graphics forum*, volume 30, pages 1427–1437. Wiley Online Library, 2011.

[72] Guillaume Lavoué, Elisa Drelie Gelasca, Florent Dupont, Atilla Baskurt, and Touradj Ebrahimi. Perceptually driven 3D distance metrics with application to watermarking. In *Applications of Digital Image Processing XXIX*, volume 6312, page 63120L. International Society for Optics and Photonics, 2006.

[73] Yann LeCun, Léon Bottou, Yoshua Bengio, and Patrick Haffner. Gradient-based learning applied to document recognition. *Proceedings of the IEEE*, 86(11):2278–2324, 1998.

[74] Manyi Li and Hao Zhang. D2im-net: Learning detail disentangled implicit fields from single images. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 10246–10255, 2021.

[75] Shikun Li, Yang Ye, Jianya Liu, and Liang Guo. Vprnet: Virtual points registration network for partial-to-partial point cloud registration. *Remote Sensing*, 14(11):2559, 2022.

[76] X. Li, W. Yi, H.L. Chi, X. Wang, and A.P. Chan. A critical review of virtual and augmented reality (VR/AR) applications in construction safety. *Automation in Construction*, 86:150–162, 2018.

[77] Xi Li and Ping Kuang. 3D-VRVT: 3D voxel reconstruction from a single image with vision transformer. In *2021 International Conference on Culture-oriented Science & Technology (ICCST)*, pages 343–348. IEEE, 2021.

[78] Zhouhui Lian, Afzal Godil, Xianfang Sun, and Jianguo Xiao. CM-BOF: visual similarity-based 3D shape retrieval using clock matching and bag-of-features. *Machine Vision and Applications*, 24:1685–1704, 2013.

[79] Zhouhui Lian, Afzal Godil, Xianfang Sun, and Hai Zhang. Non-rigid 3D shape retrieval using multidimensional scaling and bag-of-features. In *2010 IEEE International Conference on Image Processing*, pages 3181–3184. IEEE, 2010.

[80] Zhouhui Lian, Afzal Godil, and Jianguo Xiao. Feature-preserved 3D canonical form. *International journal of computer vision*, 102:221–238, 2013.

[81] Chen-Hsuan Lin, Chaoyang Wang, and Simon Lucey. SDF-SRN: Learning signed distance 3D object reconstruction from static images. *Advances in Neural Information Processing Systems*, 33:11453–11464, 2020.

[82] Jianjie Lin, Markus Rickert, Alexander Perzylo, and Alois Knoll. PCTMA-Net: Point cloud transformer with morphing atlas-based point generation network for dense point cloud completion. In *2021 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 5657–5663. IEEE, 2021.

[83] Yaoyao Lin, Mei Yu, Ken Chen, Gangyi Jiang, Fen Chen, and Zongju Peng. Blind mesh assessment based on graph spectral entropy and spatial features. *Entropy*, 22(2):190, 2020.

[84] Yaoyao Lin, Mei Yu, Ken Chen, Gangyi Jiang, Zongju Peng, and Fen Chen. Blind mesh quality assessment method based on concave, convex and structural features analyses. In *2019 IEEE International Conference on Multimedia & Expo Workshops (ICMEW)*, pages 282–287. IEEE, 2019.

[85] Yaoyao Lin, Mei Yu, Gangyi Jiang, Yang Song, and Hua Shao. Novel 3D mesh quality assessment method based on curvature analysis. In *Optoelectronic Imaging and Multimedia Technology V*, volume 10817, pages 96–106. SPIE, 2018.

[86] Yujun Lin, Zhekai Zhang, Haotian Tang, Hanrui Wang, and Song Han. PointAcc: Efficient point cloud accelerator. In *MICRO-54: 54th Annual*

IEEE/ACM International Symposium on Microarchitecture, pages 449–461, 2021.

[87] Ruoshi Liu, Rundi Wu, Basile Van Hoorick, Pavel Tokmakov, Sergey Zakharov, and Carl Vondrick. Zero-1-to-3: Zero-shot one image to 3D object. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 9298–9309, 2023.

[88] Shi-Lin Liu, Hao-Xiang Guo, Hao Pan, Peng-Shuai Wang, Xin Tong, and Yang Liu. Deep implicit moving least-squares functions for 3D reconstruction. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 1788–1797, 2021.

[89] Xinpu Liu, Guoquan Xu, Ke Xu, Jianwei Wan, and Yanxin Ma. Point cloud completion by dynamic transformer with adaptive neighbourhood feature fusion. *IET Computer Vision*, 16(7):619–631, 2022.

[90] Jameel Malik, Ibrahim Abdelaziz, Ahmed Elhayek, Soshi Shimada, Sk Aziz Ali, Vladislav Golyanik, Christian Theobalt, and Didier Stricker. HandVoxNet: Deep voxel-based network for 3D hand shape and pose estimation from a single depth map. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 7113–7122, 2020.

[91] Bogdan Maxim and Sergiu Nedevschi. OccTransformers: Learning occupancy using attention. In *2021 IEEE 17th International Conference on Intelligent Computer Communication and Processing (ICCP)*, pages 219–226. IEEE, 2021.

[92] Kirill Mazur and Victor Lempitsky. Cloud transformers: A universal approach to point cloud processing tasks. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 10715–10724, 2021.

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

[94] Fausto Milletari, Nassir Navab, and Seyed-Ahmad Ahmadi. V-net: Fully convolutional neural networks for volumetric medical image segmentation. In *2016*

fourth international conference on 3D vision (3DV), pages 565–571. IEEE, 2016.

[95] Paritosh Mittal, Yen-Chi Cheng, Maneesh Singh, and Shubham Tulsiani. AutoSDF: Shape priors for 3D completion, reconstruction and generation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 306–315, 2022.

[96] Diksha Moolchandani, Nivedita Shrivastava, Anshul Kumar, and Smruti R Sarangi. PredStereo: An accurate real-time stereo vision system. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*, pages 731–740, 2022.

[97] KL Navaneet, Ansu Mathew, Shashank Kashyap, Wei-Chih Hung, Varun Jampani, and R Venkatesh Babu. From image collections to point clouds with self-supervised shape and pose networks. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 1132–1140, 2020.

[98] Thu H Nguyen-Phuoc, Chuan Li, Stephen Balaban, and Yongliang Yang. RenderNet: A deep convolutional network for differentiable rendering from 3D shapes. *Advances in Neural Information Processing Systems*, 31, 2018.

[99] Michael Niemeyer, Jonathan T Barron, Ben Mildenhall, Mehdi SM Sajjadi, Andreas Geiger, and Noha Radwan. RegNeRF: Regularizing neural radiance fields for view synthesis from sparse inputs. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 5480–5490, 2022.

[100] M. Nießner, M. Zollhöfer, S. Izadi, and M. Stamminger. Real-time 3D reconstruction at scale using voxel hashing. *ACM Trans. Graph.*, 32(6):169, 2013.

[101] Anass Nouri, Christophe Charrier, and Olivier Lézoray. Full-reference saliency-based 3D mesh quality assessment index. In *2016 IEEE International Conference on Image Processing (ICIP)*, pages 1007–1011. IEEE, 2016.

[102] Anass Nouri, Christophe Charrier, and Olivier Lézoray. 3D blind mesh quality assessment index. In *IS&T International Symposium on Electronic Imaging*, 2017.

[103] Anass Nouri, Christophe Charrier, and Olivier Lézoray. A genetically based combination of visual saliency and roughness for FR 3D mesh quality assessment: A statistical study. *The Computer Journal*, 65(3):606–620, 2022.

[104] Liang Pan, Xinyi Chen, Zhongang Cai, Junzhe Zhang, Haiyu Zhao, Shuai Yi, and Ziwei Liu. Variational relational point completion network. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 8524–8533, 2021.

[105] 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/CVF Conference on Computer Vision and Pattern Recognition*, pages 165–174, 2019.

[106] Diego Patino, Carlos Esteves, and Kostas Daniilidis. Level set mesher: Single-image to 3D reconstruction by following the level sets of the signed distance function. In *2022 26th International Conference on Pattern Recognition (ICPR)*, pages 3994–4000. IEEE, 2022.

[107] Kebin Peng, Rifatul Islam, John Quarles, and Kevin Desai. Tmvnet: Using transformers for multi-view voxel-based 3D reconstruction. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 222–230, 2022.

[108] Xidong Peng, Xinge Zhu, Tai Wang, and Yuexin Ma. SIDE: Center-based stereo 3D detector with structure-aware instance depth estimation. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*, pages 119–128, 2022.

[109] Felix Petersen, Amit H Bermano, Oliver Deussen, and Daniel Cohen-Or. Pix2vex: Image-to-geometry reconstruction using a smooth differentiable renderer. *arXiv preprint arXiv:1903.11149*, 2019.

[110] David Pickup, Juncheng Liu, Xianfang Sun, Paul L Rosin, Ralph R Martin, Zhiquan Cheng, Zhouhui Lian, Sipin Nie, Longcun Jin, Gil Shamai, et al. An evaluation of canonical forms for non-rigid 3D shape retrieval. *Graphical Models*, 97:17–29, 2018.

[111] David Pickup, Xianfang Sun, Paul L Rosin, and Ralph R Martin. Euclidean-distance-based canonical forms for non-rigid 3D shape retrieval. *Pattern Recognition*, 48(8):2500–2512, 2015.

[112] David Pickup, Xianfang Sun, Paul L Rosin, and Ralph R Martin. Skeleton-based canonical forms for non-rigid 3D shape retrieval. *Computational visual media*, 2:231–243, 2016.

[113] David Pickup, Xianfang Sun, Paul L Rosin, Ralph R Martin, Z Cheng, Zhouhui Lian, Masaki Aono, A Ben Hamza, A Bronstein, M Bronstein, et al. Shape retrieval of non-rigid 3D human models. *International Journal of Computer Vision*, 120:169–193, 2016.

[114] Peter Pinggera, Sebastian Ramos, Stefan Gehrig, Uwe Franke, Carsten Rother, and Rudolf Mester. Lost and found: detecting small road hazards for self-driving vehicles. In *2016 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 1099–1106. IEEE, 2016.

[115] Matus Pleva, Yuan-Fu Liao, Wuhua Hsu, Daniel Hladek, Jan Stas, Peter Viszlay, Martin Lojka, and Jozef Juhar. Towards slovak-english-mandarin speech recognition using deep learning. In *2018 International Symposium ELMAR*, pages 151–154. IEEE, 2018.

[116] Sebastian Ramos, Stefan Gehrig, Peter Pinggera, Uwe Franke, and Carsten Rother. Detecting unexpected obstacles for self-driving cars: Fusing deep learning and geometric modeling. In *2017 IEEE Intelligent Vehicles Symposium (IV)*, pages 1025–1032. IEEE, 2017.

[117] Konstantinos Rematas, Ricardo Martin-Brualla, and Vittorio Ferrari. ShaRF: Shape-conditioned radiance fields from a single view. *arXiv preprint arXiv:2102.08860*, 2021.

[118] Edoardo Remelli, Artem Lukoianov, Stephan Richter, Benoit Guillard, Timur Bagautdinov, Pierre Baque, and Pascal Fua. MeshSDF: Differentiable iso-surface extraction. *Advances in Neural Information Processing Systems*, 33:22468–22478, 2020.

[119] K Robinette, S Blackwell, and D Hoeferlin. CAESAR: Civilian american and european surface anthropometry resource, 2002.

[120] Olaf Ronneberger, Philipp Fischer, and Thomas Brox. U-net: Convolutional networks for biomedical image segmentation. In *International Conference on Medical image computing and computer-assisted intervention*, pages 234–241. Springer, 2015.

[121] Abhinav Sagar. Monocular depth estimation using multi scale neural network and feature fusion. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*, pages 656–662, 2022.

[122] Yusuf Sahillioğlu. A shape deformation algorithm for constrained multidimensional scaling. *Computers & Graphics*, 53:156–165, 2015.

[123] Yusuf Sahillioğlu and Ladislav Kavan. Detail-preserving mesh unfolding for nonrigid shape retrieval. *ACM Transactions on Graphics (TOG)*, 35(3):1–11, 2016.

[124] Thomas W Sederberg and Scott R Parry. Free-form deformation of solid geometric models. In *Proceedings of the 13th annual conference on Computer graphics and interactive techniques*, pages 151–160, 1986.

[125] Gil Shamai, Michael Zibulevsky, and Ron Kimmel. Accelerating the computation of canonical forms for 3D nonrigid objects using multidimensional scaling. In *Proceedings of the 2015 Eurographics Workshop on 3D Object Retrieval*, pages 71–78, 2015.

[126] Mo Shan, Qiaojun Feng, You-Yi Jau, and Nikolay Atanasov. ELLIPSDF: joint object pose and shape optimization with a bi-level ellipsoid and signed distance function description. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 5946–5955, 2021.

[127] Zai Shi, Zhao Meng, Yiran Xing, Yunpu Ma, and Roger Wattenhofer. 3D-RETR: End-to-end single and multi-view 3D reconstruction with transformers. *arXiv preprint arXiv:2110.08861*, 2021.

[128] Karen Simonyan and Andrew Zisserman. Very deep convolutional networks for large-scale image recognition. *arXiv preprint arXiv:1409.1556*, 2014.

[129] David Stutz and Andreas Geiger. Learning 3d shape completion under weak supervision. *International Journal of Computer Vision*, 128:1162–1181, 2020.

[130] Gabriel Taubin. A signal processing approach to fair surface design. In *Proceedings of the 22nd annual conference on Computer graphics and interactive techniques*, pages 351–358, 1995.

[131] Konstantinos Tertikas, Despoina Paschalidou, Boxiao Pan, Jeong Joon Park, Mikaela Angelina Uy, Ioannis Emiris, Yannis Avrithis, and Leonidas Guibas. Generating part-aware editable 3D shapes without 3D supervision. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 4466–4478, 2023.

[132] J. Varley, C. DeChant, A. Richardson, J. Ruales, and P. Allen. Shape completion enabled robotic grasping. In *IEEE/RSJ IROS*, pages 2442–2447, 2017.

[133] L. Wan, J. Jiang, and H. Zhang. Incomplete 3D shape retrieval via sparse dictionary learning. In *Pacific Graphics Short Papers*, 2015.

[134] Kai Wang, Fakhri Torkhani, and Annick Montanvert. A fast roughness-based approach to the assessment of 3D mesh visual quality. *Computers & Graphics*, 36(7):808–818, 2012.

[135] Xiaogang Wang, Marcelo H Ang Jr, and Gim Hee Lee. Cascaded refinement network for point cloud completion. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 790–799, 2020.

[136] Xu-Lei Wang and Hongbin Zha. Contour canonical form: An efficient intrinsic embedding approach to matching non-rigid 3D objects. In *Proceedings of the 2nd ACM International Conference on Multimedia Retrieval*, pages 1–8, 2012.

[137] Yida Wang, David Joseph Tan, Nassir Navab, and Federico Tombari. ForkNet: Multi-branch volumetric semantic completion from a single depth image. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 8608–8617, 2019.

[138] Zhou Wang, Alan C Bovik, Hamid R Sheikh, and Eero P Simoncelli. Image quality assessment: from error visibility to structural similarity. *IEEE transactions on image processing*, 13(4):600–612, 2004.

[139] Xin Wen, Zhizhong Han, Yan-Pei Cao, Pengfei Wan, Wen Zheng, and Yu-Shen Liu. Cycle4completion: Unpaired point cloud completion using cycle trans-

formation with missing region coding. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 13080–13089, 2021.

[140] Xin Wen, Tianyang Li, Zhizhong Han, and Yu-Shen Liu. Point cloud completion by skip-attention network with hierarchical folding. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 1939–1948, 2020.

[141] Xin Wen, Peng Xiang, Zhizhong Han, Yan-Pei Cao, Pengfei Wan, Wen Zheng, and Yu-Shen Liu. PMP-Net: Point cloud completion by learning multi-step point moving paths. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2021.

[142] Xin Wen, Peng Xiang, Zhizhong Han, Yan-Pei Cao, Pengfei Wan, Wen Zheng, and Yu-Shen Liu. PMP-Net++: Point cloud completion by transformer-enhanced multi-step point moving paths. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 45(1):852–867, 2022.

[143] J. Wu, Y. Wang, T. Xue, X. Sun, B. Freeman, and J. Tenenbaum. MarrNet: 3D shape reconstruction via 2.5D sketches. In *Proceedings of the neural information processing systems (NIPS)*, pages 540–550, 2017.

[144] J. Wu, C. Zhang, X. Zhang, Z. Zhang, W. T. Freeman, and J. B. Tenenbaum. Learning shape priors for single-view 3D completion and reconstruction. In *ECCV*, pages 673–691, 2018.

[145] Jiajun Wu, Yifan Wang, Tianfan Xue, Xingyuan Sun, Bill Freeman, and Josh Tenenbaum. MarrNet: 3D shape reconstruction via 2.5D sketches. *Advances in neural information processing systems*, 30, 2017.

[146] Jiajun Wu, Chengkai Zhang, Xiuming Zhang, Zhoutong Zhang, William T Freeman, and Joshua B Tenenbaum. Learning shape priors for single-view 3D completion and reconstruction. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pages 646–662, 2018.

[147] Tong Wu, Liang Pan, Junzhe Zhang, Tai Wang, Ziwei Liu, and Dahua Lin. Density-aware chamfer distance as a comprehensive metric for point cloud completion. In *In Advances in Neural Information Processing Systems (NeurIPS), 2021*, 2021.

[148] Peng Xiang, Xin Wen, Yu-Shen Liu, Yan-Pei Cao, Pengfei Wan, Wen Zheng, and Zhizhong Han. Snowflakenet: Point cloud completion by snowflake point deconvolution with skip-transformer. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 5499–5509, 2021.

[149] Peng Xiang, Xin Wen, Yu-Shen Liu, Yan-Pei Cao, Pengfei Wan, Wen Zheng, and Zhizhong Han. Snowflake point deconvolution for point cloud completion and generation with skip-transformer. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2022.

[150] 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/CVF International Conference on Computer Vision*, pages 2690–2698, 2019.

[151] Qiangeng Xu, Weiyue Wang, Duygu Ceylan, Radomir Mech, and Ulrich Neumann. DISN: Deep implicit surface network for high-quality single-view 3D reconstruction. *Advances in neural information processing systems*, 32, 2019.

[152] Yan Xu, Tao Mo, Qiwei Feng, Peilin Zhong, Maode Lai, I Eric, and Chao Chang. Deep learning of feature representation with multiple instance learning for medical image analysis. In *2014 IEEE international conference on acoustics, speech and signal processing (ICASSP)*, pages 1626–1630. IEEE, 2014.

[153] Xinchen Yan, 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.

[154] B. Yang, S. Rosa, A. Markham, N. Trigoni, and H. Wen. Dense 3D object reconstruction from a single depth view. *IEEE Trans. Patt. Anal. Mach. Intell.*, 41(12):2820–2834, 2019.

[155] Shuo Yang, Min Xu, Haozhe Xie, Stuart Perry, and Jiahao Xia. Single-view 3D object reconstruction from shape priors in memory. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 3152–3161, 2021.

[156] Wei Yang, Chris Paxton, Arsalan Mousavian, Yu-Wei Chao, Maya Cakmak, and Dieter Fox. Reactive human-to-robot handovers of arbitrary objects. In *2021*

IEEE International Conference on Robotics and Automation (ICRA), pages 3118–3124. IEEE, 2021.

[157] Yuan Yao, Nico Schertler, Enrique Rosales, Helge Rhodin, Leonid Sigal, and Alla Sheffer. Front2back: Single view 3D shape reconstruction via front to back prediction. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 531–540, 2020.

[158] Zeynep Cipiloglu Yildiz, A Cengiz Oztireli, and Tolga Capin. A machine learning framework for full-reference 3D shape quality assessment. *The Visual Computer*, 36(1):127–139, 2020.

[159] Kangxue Yin, Zhiqin Chen, Hui Huang, Daniel Cohen-Or, and Hao Zhang. LOGAN: Unpaired shape transform in latent overcomplete space. *ACM Transactions on Graphics (TOG)*, 38(6):1–13, 2019.

[160] Yu-Jie Yuan, Yang-Tian Sun, Yu-Kun Lai, Yuewen Ma, Rongfei Jia, Leif Kobbelt, and Lin Gao. Interactive NeRF geometry editing with shape priors. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2023.

[161] Eloi Zalczer, François-Xavier Thomas, Laurent Chanas, Gabriele Facciolo, and Frédéric Guichard. Depth map quality evaluation for photographic applications. *Electronic Imaging*, 2020(9):370–1, 2020.

[162] Hui Zeng, Qi Wang, and Jiwei Liu. Multi-feature fusion based on multi-view feature and 3D shape feature for non-rigid 3D model retrieval. *IEEE access*, 7:41584–41595, 2019.

[163] H. Zhang, Goodfellow I., D. Metaxas, and A. Odena. Self-attention generative adversarial networks. *arXiv:1805.08318*, 2018.

[164] Junzhe Zhang, Xinyi Chen, Zhongang Cai, Liang Pan, Haiyu Zhao, Shuai Yi, Chai Kiat Yeo, Bo Dai, and Chen Change Loy. Unsupervised 3D shape completion through GAN inversion. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 1768–1777, 2021.

[165] Lin Zhang, Lei Zhang, Xuanqin Mou, and David Zhang. FSIM: A feature similarity index for image quality assessment. *IEEE transactions on Image Processing*, 20(8):2378–2386, 2011.

[166] Wenxiao Zhang, Zhen Dong, Jun Liu, Qingan Yan, Chunxia Xiao, et al. Point cloud completion via skeleton-detail transformer. *IEEE Transactions on Visualization and Computer Graphics*, 2022.

[167] X. Zhang, Z. Zhang, C. Zhang, J. Tenenbaum, B. Freeman, and J. Wu. Learning to reconstruct shapes from unseen classes. In *NIPS*, pages 2257–2268, 2018.

[168] Xiuming Zhang, Zhoutong Zhang, Chengkai Zhang, Josh Tenenbaum, Bill Freeman, and Jiajun Wu. Learning to reconstruct shapes from unseen classes. *Advances in neural information processing systems*, 31, 2018.

[169] Xuancheng Zhang, Yutong Feng, Siqi Li, Changqing Zou, Hai Wan, Xibin Zhao, Yandong Guo, and Yue Gao. View-guided point cloud completion. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 15890–15899, 2021.

[170] Yanqiang Zhang, Lijin Fang, and Chengpeng Li. Generalized deep implicit surface network for image-based three-dimensional object reconstruction. In *2021 China Automation Congress (CAC)*, pages 276–281. IEEE, 2021.

[171] Zicheng Zhang, Wei Sun, Xiongkuo Min, Tao Wang, Wei Lu, and Guangtao Zhai. No-reference quality assessment for 3D colored point cloud and mesh models. *IEEE Transactions on Circuits and Systems for Video Technology*, 32(11):7618–7631, 2022.

[172] X Zheng, Yang Liu, P Wang, and Xin Tong. Sdf-stylegan: Implicit sdf-based stylegan for 3d shape generation. In *Computer Graphics Forum*, volume 41, pages 52–63. Wiley Online Library, 2022.

[173] Haoran Zhou, Yun Cao, Wenqing Chu, Junwei Zhu, Tong Lu, Ying Tai, and Chengjie Wang. SeedFormer: Patch seeds based point cloud completion with upsample transformer. *arXiv preprint arXiv:2207.10315*, 2022.