Analysis-by-Synthesis Transformer for Single-View 3D Reconstruction

Dian Jia[1], Xiaoqian Ruan[1], Kun Xia[1,2], Zhiming Zou[1], Le Wang[2], and Wei Tang[1]

[1] University of Illinois Chicago, Chicago, IL, USA
{djia7,xruan9,zzou6,tangw}@uic.edu
[2] Xi'an Jiaotong University, Xi'an, Shaanxi, P.R. China
xiakun@stu.xjtu.edu.cn
lewang@xjtu.edu.cn

Abstract. Deep learning approaches have made significant success in single-view 3D reconstruction, but they often rely on expensive 3D annotations for training. Recent efforts tackle this challenge by adopting an analysis-by-synthesis paradigm to learn 3D reconstruction with only 2D annotations. However, existing methods face limitations in both shape reconstruction and texture generation. This paper introduces an innovative Analysis-by-Synthesis Transformer that addresses these limitations in a unified framework by effectively modeling pixel-to-shape and pixel-to-texture relationships. It consists of a *Shape Transformer* and a *Texture Transformer*. The Shape Transformer employs learnable shape queries to fetch pixel-level features from the image, thereby achieving high-quality mesh reconstruction and recovering occluded vertices. The Texture Transformer employs texture queries for non-local gathering of texture information and thus eliminates the incorrect inductive bias. Experimental results on CUB-200-2011 and ShapeNet datasets demonstrate superior performance in shape reconstruction and texture generation compared to previous methods. The code is available at https://github.com/DianJJ/AST.

Keywords: Single-view 3D reconstruction · Shape Transformer · Texture Transformer

1 Introduction

Reconstructing the 3D shape of objects from images or videos is a long-standing task in computer vision. It holds significant promise for a variety of applications, such as virtual and augmented reality, robotics, and autonomous driving. In the past few years, data-driven approaches, in particular, deep neural networks, have shown the capability to achieve high-quality 3D reconstruction from a single-view image [8, 19, 24, 42, 46]. However, the practical utility of these approaches in real-world applications is significantly restricted because they rely on large-scale 3D annotations (shapes and/or poses) for training [9, 12, 16, 19, 46, 50, 66], which are expensive and often unobtainable.

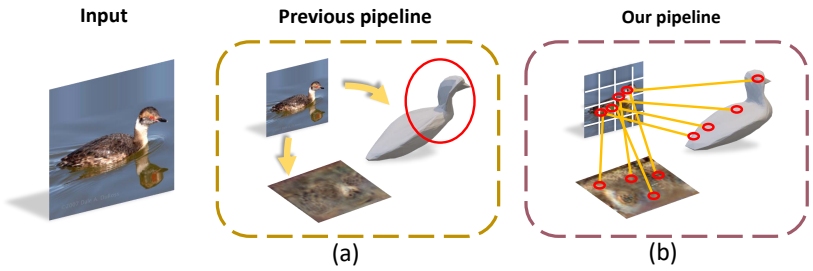

Fig. 1: Current 3D reconstruction methods from 2D annotations have limitations: shape models often lack pixel-level details, and texture models typically produce blurry UV map textures. For example, the reconstruction in (a) from [47] exhibits an unnatural neck shape and a blurry texture. This paper introduces the Analysis-by-Synthesis Transformer, which addresses these issues within a unified framework, enhancing both shape and texture modeling, as shown in (b).

Some recent works have attempted to learn 3D reconstruction from a collection of images with only 2D annotations [7, 18, 22, 25, 29, 39, 47, 48], such as keypoints and silhouette masks. They follow the *analysis-by-synthesis* paradigm. In this framework, a convolutional neural network (ConvNet) takes as input an image and predicts the object's mesh, texture, and pose. These predictions are then used to synthesize the input image and its 2D annotations through differentiable rendering [31, 43]. The network is trained by minimizing the synthesis error, along with some regularizations.

Despite the promising results demonstrated by the analysis-by-synthesis paradigm, the exploration is still in its early stages. Existing methods suffer from limitations in both shape reconstruction and texture generation, as illustrated in Fig. 1 and described below.

Limitation in Shape Reconstruction. Current shape reconstruction models rely on a single global shape code extracted from the entire image to predict mesh vertices, lacking access to pixel-level features and struggling with fine details. Prior research in supervised mesh reconstruction, exemplified by Pixel2Mesh [66], highlights the importance of modeling pixel-level features that are related to the vertices for achieving high-quality mesh reconstruction. Specifically, this approach involves projecting each coarse vertex into the image space and then pooling local features from the projected pixel location to refine the mesh progressively. Regrettably, this operation, known as *perceptual feature pooling*, is not applicable to the analysis-by-synthesis paradigm as it necessitates an accurate camera pose to ensure correct 3D-to-2D projection, as well as ground truth shape to guide coarse mesh generation. Moreover, there is no mechanism to handle vertices that are invisible in the image, which disrupts the perceptual feature pooling process.

Limitation in Texture Generation. To render an image from 3D reconstruction, current models employ a ConvNet to predict the mesh texture in the

form of a UV map. ConvNets are well-suited for dense prediction tasks such as semantic segmentation [44, 57], image enhancement [6, 14, 36], and depth estimation [11, 17] due to their *translation equivariance* property, where the output shifts by the same amount as the input. As a result, there is a one-to-one correspondence between input and output pixels at the same location. However, unlike segmentation maps, enhanced images, or depth maps, the UV map is *not* translation equivariant w.r.t. the input image. Each pixel in the UV map corresponds to a vertex on the mesh rather than to a pixel at the same location in the input image. Therefore, employing a ConvNet for UV map prediction is inferior and often results in blurry texture. This issue is especially crucial within the analysis-by-synthesis paradigm, as texture quality directly impacts the rendered image and thus the image synthesis loss.

This paper introduces the *Analysis-by-Synthesis Transformer (AST)*, a novel unified Transformer architecture specially designed to tackle the two aforementioned limitations of the analysis-by-synthesis paradigm through effective pixel-to-shape and pixel-to-texture modeling. It consists of two core components: the *Shape Transformer* and the *Texture Transformer*. The Shape Transformer models a set of learnable *shape queries*, each corresponding to a vertex of the object mesh. On one hand, each shape query automatically fetches pixel-level features related to its corresponding vertex for high-quality shape reconstruction. On the other hand, these shape queries interact with each other to recover vertices occluded in the image. The Texture Transformer models a set of learnable *texture queries*, each corresponding to a pixel in the UV map. Each texture query gathers texture information relevant to the corresponding UV map pixel from the image in a non-local manner, thereby eliminating the incorrect inductive bias of translation equivariance from texture generation. By unifying the Shape and Texture Transformers, our proposed approach offers clear advantages over previous methods in terms of both shape reconstruction and texture generation, making it highly suitable for learning single-view 3D reconstruction within the analysis-by-synthesis paradigm.

It is worth noting that our approach is obviously different from the Mesh Transformer (METRO) [41] for human pose and shape reconstruction. First, each query of METRO is a concatenation of a *constant* vertex coordinate from a fixed human mesh template and a *global* shape code extracted from the entire image. METRO only models self-attention between these queries, which fails to capture pixel-level features relevant to the vertices. Therefore, it cannot address the limitation in shape reconstruction. Second, METRO does not predict mesh texture, and is trained in a fully supervised manner. Therefore, it cannot address the limitation in texture generation.

The contributions of this paper are as follows. (1) Shape reconstruction and texture generation are two core aspects of the analysis-by-synthesis paradigm for learning textured mesh reconstruction. To our knowledge, this work is the first of its kind that identifies critical limitations in both aspects of existing methods. It introduces a novel Analysis-by-Synthesis Transformer to address these limitations in a unified framework through effective pixel-to-shape and pixel-to-texture

modeling. (2) We propose the Shape Transformer. Different from existing Transformers, it utilizes a set of learnable shape queries to fetch pixel-level features related to each vertex from the image and models the interactions among these queries, which is essential to achieving high-quality mesh reconstruction and effectively recovering vertices occluded in the image. (3) We propose the Texture Transformer. Unlike existing methods, which employ a ConvNet for texture prediction, it utilizes a set of learnable texture queries to gather texture information relevant to the corresponding UV map pixels from the image in a non-local manner, which eliminates the incorrect inductive bias of translation equivariance. (4) Experimental results on the CUB-200-2011 and ShapeNet datasets demonstrate that our proposed approach significantly outperforms previous methods in terms of both shape reconstruction and texture generation.

2 Related Work

2.1 Supervised Single-view 3D Reconstruction

In the past few years, deep learning approaches have drastically advanced the area of single-view 3D reconstruction. Early efforts [8, 15, 20, 62, 70, 74, 78, 80] employ voxel-based shape representations, but suffer from the cubic growth in complexity [15]. To overcome this limitation, point cloud [27, 40, 45, 54, 55] and mesh-based representations [16, 19, 43, 49, 66, 69] emerge as alternatives, offering a better balance between efficiency and accuracy. Recent implicit methods, such as the occupancy network [46] and DeepSDF [50], represent the shape as a neural network, mapping a continuous 3D coordinate to an occupancy value or a signed distance to the surface. Therefore, they can model shapes with arbitrary topology at any resolution. Despite these significant advancements, current methods rely on large-scale 3D shape annotations for training, which are expensive and often unobtainable.

2.2 Single-view 3D Reconstruction without 3D Supervision

Thanks to the development of differentiable rendering [31, 43], recent research has attempted to learn single-view 3D reconstruction without 3D supervision [18,21,26,29,33,34,38,39,71,72]. State-of-the-art methods follow the analysis-by-synthesis paradigm. CMR [29] trains a ConvNet to predict the object mesh, pose, and texture from a single input image by exploiting the silhouette and keypoint annotations available in 2D image datasets. UCMR [18] infers the 3D shape of objects from a collection of images without using keypoint annotations. UMR [39] enforces semantic consistency across different views in a self-supervised learning framework. SMR [25] models interpolated consistency and landmark consistency to better learn the 3D mesh. UNICORN [47] gets rid of the silhouette mask annotations and common shape assumptions through a neighbor reconstruction loss and background modeling. MagicPony [71] and ShapeClipper [26] fuse features from pre-trained external models, such as CLIP [56] and DINO [4], to improve the

consistency of model predictions. Other related works [10,13,23,30,51,52,75,77] leverage in addition generative adversarial techniques for improved performance, but they rely on a powerful 2D generative model and a more complex training procedure. This paper follows the analysis-by-synthesis paradigm for learning 3D reconstruction with only silhouette mask annotations. Previous methods in this paradigm suffer from critical limitations in both shape reconstruction and texture generation, as described in Sec. 1. This paper introduces the Analysis-by-Synthesis Transformer, a novel Transformer architecture specially designed to address these limitations in a unified framework through effective pixel-to-shape and pixel-to-texture modeling.

2.3 Transformers for 3D Reconstruction

Transformers [63] have shown promise in various computer vision tasks including object detection [3], image enhancement [76], semantic segmentation [65,79], and vision-language modeling [37,60]. Recent works have adapted Transformer architectures for 3D reconstruction. Wang *et al*. [64] treat multi-view 3D reconstruction as a sequence prediction problem using a Transformer for view relationship modeling. Shi *et al*. [58] employ a Transformer encoder for feature extraction and a decoder for voxel prediction. Peng *et al*. [53] combine a 3D ConvNet with a Transformer decoder for voxel-based reconstruction. Bozic *et al*. [2] use a Transformer to fuse multi-view data into a volumetric grid, which is decoded into a 3D scene. Unlike these approaches that use Transformers as an alternative to ConvNets, our method specifically overcomes limitations in pixel-to-shape and pixel-to-texture modeling in the analysis-by-synthesis paradigm for self-supervised 3D reconstruction.

3 Method

This paper introduces the *Analysis-by-Synthesis Transformer* for single-view 3D reconstruction. It is a novel Transformer architecture specially designed to address the limitations in shape reconstruction and texture generation of the analysis-by-synthesis paradigm (Sec. 1) through effective pixel-to-shape and pixel-to-texture modeling. We focus on a problem setting commonly used in this paradigm. During training, we have an image dataset that covers the object category of interest, *e.g*., cars, chairs, and tables. The only annotations are the 2D silhouette masks. During inference, the input is a new image of an object instance; the output includes a textured mesh and the camera pose.

Our proposed network architecture is illustrated in Fig. 2. The input is a single-view image $\boldsymbol{I}$. Following the camera multiplex [18], the network generates multiple hypotheses of the object pose (including scale, translation, and rotation) and their probability distribution. The Shape Transformer and the Texture Transformer reconstruct the object mesh and texture, respectively. The network also predicts an object saliency map. On one hand, it will be used to guide the

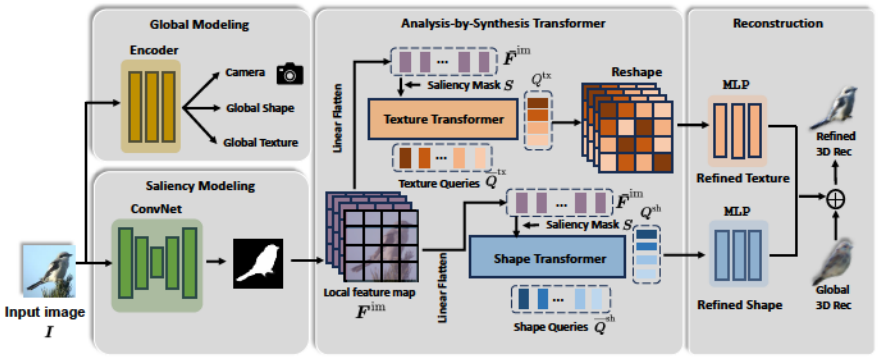

Fig. 2: An overview of our proposed Analysis-by-Synthesis Transformer. The Shape Transformer employs a set of learnable shape queries to fetch pixel-level features related to each vertex from the image for high-quality mesh reconstruction. The Texture Transformer utilizes a set of learnable texture queries to gather texture information relevant to the corresponding UV map pixels from the image in a non-local manner, thereby eliminating the incorrect inductive bias of translation equivariance from texture generation. Both the Shape and Texture Transformers are guided by a saliency map, focusing on object pixels, and they predict the shape and texture from coarse to fine.

attention mechanism in the Transformers. On the other hand, learning it under silhouette mask supervision helps the backbone acquire more discriminative features, thereby benefiting 3D reconstruction.

In the rest of this section, we describe the Shape Transformer and the Texture Transformer in Sec. 3.1 and Sec. 3.2, respectively, and describe the learning method in Sec. 3.3. The detailed network architecture can be found in the supplementary material.

3.1 Shape Transformer

Following the previous analysis-by-synthesis paradigm [25, 29, 39, 47], the Shape Transformer reconstructs the object mesh by deforming a predefined ellipsoid mesh. But different from previous methods, the Shape Transformer models the deformation as two complementary processes. First, it utilizes a global shape code inferred from the entire image to capture the overall object shape. Second, it utilizes a set of learnable shape queries to fetch pixel-level features related to each vertex for high-quality mesh reconstruction.

Global Shape Modeling. Let $\{v_i \in \mathbb{R}^3 : i = 1, \ldots, N\}$ be the vertices of a predefined ellipsoid mesh, where N is the number of vertices. An encoder network first extracts a global shape code $z^{\text{global-sh}} \in \mathbb{R}^D$ from the image, where D is the code dimension. Then, the overall object shape is modeled as a neural parametric surface as below:

$$\hat{\boldsymbol{v}}_i = \texttt{MLP}(\boldsymbol{v}_i, \boldsymbol{z}^{\text{global-sh}}; \Theta^{\text{global-sh}}),\ i = 1, \ldots, N \tag{1}$$

where $\hat{\boldsymbol{v}}_i \in \mathbb{R}^3$ is a new vertex, and $\Theta^{\text{global-sh}}$ is the parameters.

Local Shape Modeling. We model a set of learnable shape queries: $\bar{\boldsymbol{Q}}^{\text{sh}} = [\bar{\boldsymbol{q}}_i^{\text{sh}} \in \mathbb{R}^D : i = 1, \ldots, N]$, each corresponding to a mesh vertex. We extract a feature map $\boldsymbol{F}^{\text{im}} \in \mathbb{R}^{H \times W \times D}$ from the image through a ConvNet, where H, W, and D denote the height, width, and feature dimension, respectively. For convenience, we reshape the image feature map as $\bar{\boldsymbol{F}}^{\text{im}} \in \mathbb{R}^{HW \times D}$. The shape queries and the image feature map are fed into a Transformer decoder including two multi-head attention (MHA) units:

$$\hat{\boldsymbol{Q}}^{\text{sh}} = \texttt{MHA}(\boldsymbol{Q} = \bar{\boldsymbol{Q}}^{\text{sh}}, \boldsymbol{K} = \bar{\boldsymbol{F}}^{\text{im}}, \boldsymbol{V} = \bar{\boldsymbol{F}}^{\text{im}}, \boldsymbol{M} = \boldsymbol{S}; \Theta^{\text{sh-im}}) \tag{2}$$

$$\boldsymbol{Q}^{\text{sh}} = \texttt{MHA}(\boldsymbol{Q} = \hat{\boldsymbol{Q}}^{\text{sh}}, \boldsymbol{K} = \hat{\boldsymbol{Q}}^{\text{sh}}, \boldsymbol{V} = \hat{\boldsymbol{Q}}^{\text{sh}}, \boldsymbol{M} = \boldsymbol{1}; \Theta^{\text{sh-sh}}) \tag{3}$$

where $\boldsymbol{Q}$, $\boldsymbol{K}$, and $\boldsymbol{V}$ denote the query, key, and value matrices, respectively, $\boldsymbol{M}$ denotes the attention mask, $\boldsymbol{S}$ is the predicted object saliency map, $\boldsymbol{1}$ is an all-one matrix, and $\Theta^{\text{sh-im}}$ and $\Theta^{\text{sh-sh}}$ are parameters. Following the standard Transformer [63], we use the scaled dot product attention and augment the image feature map with pixel-wise positional embeddings. Using the saliency map as the attention mask in Eq. (2) guides the vertex queries to focus on the object pixels. The detailed formulation of MHA can be found in the supplementary material.

The updated shape queries $\boldsymbol{Q}^{\text{sh}} = [\boldsymbol{q}_i^{\text{sh}} \in \mathbb{R}^D : i = 1, \ldots, N]$ deform their corresponding mesh vertices through:

$$\boldsymbol{v}_i = \hat{\boldsymbol{v}}_i + \texttt{MLP}(\hat{\boldsymbol{v}}_i, \boldsymbol{q}_i^{\text{sh}}; \Theta^{\text{local-sh}}),\ i = 1, \ldots, N \tag{4}$$

where $\boldsymbol{v}_i \in \mathbb{R}^3$ is a vertex of the reconstructed object mesh by our model, and $\Theta^{\text{local-sh}}$ is parameters.

Discussion. Eq. (1) uses a single global shape code summarized from the entire image to deform all vertices jointly. Therefore, the global shape model is suitable for capturing the overall object shape. In contrast, Eq. (4) deforms each vertex separately using a specific local shape code, *i.e.*, $\boldsymbol{q}_i^{\text{sh}}$. Therefore, the local shape model will be able to characterize more subtle deformations. The global and local shape models are complementary.

Modeling these local shape codes necessitates extraction of pixel-level features related to each vertex from the image. This is, however, challenging as which image pixels correspond to each vertex is unknown and a large portion of vertices are invisible in the image. Pixel2Mesh [66] projects each vertex to the image space to pool vertex-relevant local features, but it necessitates an accurate camera pose to ensure correct 3D-to-2D projection, as well as shape supervision to guide coarse mesh generation. Additionally, this perceptual feature pooling process is easily disrupted by occluded vertices. In contrast, by taking the vertex queries as the query, and the image feature map as the key and value in an MHA,

Eq. (2) automatically fetches pixel-level features related to each vertex from the image feature map without explicit projection. Furthermore, Eq. (3) models the relational interactions among the vertex queries through self-attention, which helps infer occluded vertices.

Existing methods in the analysis-by-synthesis paradigm typically only set a few hundred vertices, which we also follow in the experiments. To be scalable to a significantly larger number of vertices, we can divide the template mesh into small patches and use a shape query to deform vertices within a patch instead of a single vertex.

3.2 Texture Transformer

Texture reconstruction is an essential aspect of the analysis-by-synthesis paradigm. Previous methods employ a ConvNet to predict a UV map from an input image. This is inferior as the ConvNet is equivariant to the translation transformation but the UV map is *not* translation equivariant w.r.t. the input image. As a result, these methods often produce blurry texture. To close this gap, the Texture Transformer utilizes a set of learnable texture queries to gather texture information relevant to the corresponding UV map pixels from the image in a non-local manner, thereby eliminating the incorrect inductive bias of translation equivariance. Inspired by the Shape Transformer, the Texture Transformer generates the UV map in a coarse-to-fine manner through global and refined texture modeling. Though the global model generates inferior texture, we find it beneficial to the training process.

Global Texture Modeling. The mesh texture is represented as a UV map of height H' and width W': $\boldsymbol{T} \in \mathbb{R}^{H' \times W' \times 3}$. An encoder network first extracts a global texture code $\boldsymbol{z}^{\text{global-tx}}$ from the image. Then, the coarse UV map $\hat{\boldsymbol{T}}$ is generated by:

$$\hat{\boldsymbol{T}} = \texttt{MLP}(\boldsymbol{z}^{\text{global-tx}}; \Theta^{\text{global-tx}}) \tag{5}$$

where $\Theta^{\text{global-tx}}$ is the parameters.

Refined Texture Modeling. We model a set of learnable texture queries: $\bar{\boldsymbol{Q}}^{\text{tx}} = [\bar{\boldsymbol{q}}_i^{\text{tx}} \in \mathbb{R}^D : i = 1, \ldots, H'W']$, each corresponding to a pixel in the UV map. These texture queries are fed into a Transformer decoder including two multi-head attention units:

$$\hat{\boldsymbol{Q}}^{\text{tx}} = \texttt{MHA}(\boldsymbol{Q} = \bar{\boldsymbol{Q}}^{\text{tx}}, \boldsymbol{K} = \bar{\boldsymbol{F}}^{\text{im}}, \boldsymbol{V} = \bar{\boldsymbol{F}}^{\text{im}}, \boldsymbol{M} = \boldsymbol{S}; \Theta^{\text{tx-im}}) \tag{6}$$

$$\boldsymbol{Q}^{\text{tx}} = \texttt{MHA}(\boldsymbol{Q} = \hat{\boldsymbol{Q}}^{\text{tx}}, \boldsymbol{K} = \hat{\boldsymbol{Q}}^{\text{tx}}, \boldsymbol{V} = \hat{\boldsymbol{Q}}^{\text{tx}}, \boldsymbol{M} = \boldsymbol{1}; \Theta^{\text{tx-tx}}) \tag{7}$$

where $\boldsymbol{Q}^{\text{tx}} = [\boldsymbol{q}_i^{\text{tx}} \in \mathbb{R}^D : i = 1, \ldots, H'W']$ is the updated texture queries and also the texture codes of UV map pixels, and $\Theta^{\text{tx-im}}$ and $\Theta^{\text{tx-tx}}$ are parameters.

Then, each texture code is used to predict the RGB values of the corresponding pixel in the UV map:

$$\boldsymbol{t}_i = \hat{\boldsymbol{t}}_i + \texttt{MLP}(\boldsymbol{q}_i^{\text{tx}}; \Theta^{\text{local-tx}}),\ i = 1, \ldots, H'W' \tag{8}$$

Table 1: Quantitative results on CUB-200-2011. MeshInv [77] and HybridInv [52] are *test-time optimization* methods that optimize the shape for N steps at test time by inverting a generator; they are significantly slower than direct prediction methods. CMR and DIB-R use camera pose annotations in addition to silhouette masks; Unicorn eliminates mask supervision through background modeling.

Methods	**Mask IoU** (%, ↑)	**SSIM** (%, ↑)	**PCK** (%, ↑)
MeshInv ($N = 200$) [77]	75.2	-	-
HybridInv ($N = 0$) [52]	73.9	-	-
HybridInv ($N = 30$) [52]	**84.4**	-	-
CMR [29]	73.8	44.6	28.5
CSM [35]	-	-	48.0
DIB-R [7]	75.7	-	-
UMR [39]	73.4	71.3	58.2
IMR [61]	-	-	53.5
UCMR [18]	63.7	-	-
SMR [25]	80.6	83.2	62.2
Unicorn [47]	71.4	63.5	49.0
MagicPony [71]	-	-	55.5
AST (Ours)	**81.6**	**86.0**	**64.7**

where $\boldsymbol{t}_i \in \mathbb{R}^3$ and $\hat{\boldsymbol{t}}_i \in \mathbb{R}^3$ are the ith pixel in the refined UV map and coarse UV map, respectively, and $\Theta^{\text{local-tx}}$ is parameters. Eq. (8) can be implemented efficiently using convolutional layers.

The Shape and Texture Transformers share similar Transformer decoders but differ in the shape/texture query modeling and mesh/UV map reconstruction. Eq. (6) enables texture queries to gather texture features from the image in a non-local manner. Eq. (7) allows those texture queries to exchange information with each other to effectively recover occluded texture. Altogether, the Texture Transformer will be more suitable for UV map prediction than a ConvNet.

3.3 Learning

Our learning objective largely follows previous methods in the analysis-by-synthesis paradigm [18, 25, 29, 39, 43, 47]. It includes a rendering loss, a saliency loss, and a regularization loss.

We employ a differentiable renderer [43] to render an image $\tilde{\boldsymbol{I}}$ and a silhouette mask $\tilde{\boldsymbol{S}}$ from the predicted pose, mesh, and UV map. The pose corresponds to the camera-multiplex hypothesis with the highest probability. The rendering loss is formulated as:

$$\ell^{\text{REN}} = \|\boldsymbol{I} - \tilde{\boldsymbol{I}}\|_2^2 + \lambda^{\text{PER}} \|\phi(\boldsymbol{I}) - \phi(\tilde{\boldsymbol{I}})\|_2^2, \tag{9}$$

where ϕ is the `relu3_3` [28] layer of a pre-trained VGG16 [59], and λ^{PER} is a scalar hyperparameter set to 10.

We compare the predicted object saliency map $\boldsymbol{S}$ and the annotated silhouette mask $\boldsymbol{S}^{\text{gt}}$ through a binary cross entropy loss and a mask intersection-over-

Fig. 3: Qualitative results on CUB-200-2011.

union (IoU) loss. The saliency loss is formulated as:

$$\ell^{\mathrm{SAL}} = \ell^{\mathrm{BCE}}(\boldsymbol{S}^{\mathrm{gt}}, \boldsymbol{S}) + \ell^{\mathrm{IOU}}(\boldsymbol{S}^{\mathrm{gt}}, \boldsymbol{S}), \tag{10}$$

Following prior works [18, 43, 47], we adopt a regularization loss ℓ^{REG} which consists of a normal consistency loss [7], a Laplacian smoothing loss [7], a cross-instance consistency loss [47], and a uniformity regularization [21] on the multiplex pose hypotheses. The final objective is formulated as:

$$\ell = \ell^{\mathrm{REN}} + \lambda^{\mathrm{SAL}}\ell^{\mathrm{SAL}} + \lambda^{\mathrm{REG}}\ell^{\mathrm{REG}} \tag{11}$$

where λ^{SAL} and λ^{REG} are balancing weights. Our learning objective supervises both global and local shape/texture predictions. The detailed formulations of our loss function can be found in the supplementary material.

4 Experiments

4.1 Datasets

We evaluate our single-view 3D reconstruction method on CUB-200-2011 and ShapeNet datasets.

CUB-200-2011 [68] is one of the most widely used datasets for 3D reconstruction, featuring approximately 11,788 images across 200 bird species categories. We follow the community guideline to divide the dataset into 5,994 training images and 5,794 testing images.

ShapeNet [5] is a collaborative, large-scale dataset of richly-annotated synthetic 3D shape. We render each 3D object from ShapeNet into 64×64 images from 24 distinct angles and split the images into a training set, validation set and test set following the community guideline.

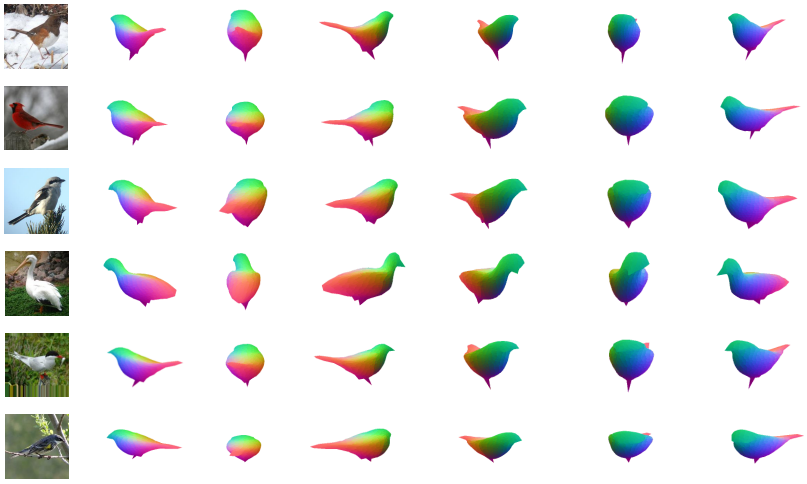

Fig. 4: Shape reconstruction obtained by our approach on CUB-200-2011 from 6 different viewpoints.

4.2 Evaluation Metrics

On CUB-200-2011, we use Mask IoU and PCK to measure the accuracy of 3D reconstruction, and SSIM for assessing texture synthesis quality. For ShapeNet, we measure the Chamfer-L1 distance (with ICP alignment) between reconstructed and ground truth meshes. **Mask IoU** calculates the intersection over union between generated and actual silhouette masks, assessing the accuracy of 2D projections. **Percent of Correct Keypoints (PCK)** [35] evaluates the precision of keypoint localization, with higher PCK reflecting better 3D reconstruction accuracy. **The Structural Similarity Index (SSIM)** [67] measures the similarity between two images by considering structural information, texture, luminance, and contrast. **Chamfer-L1 distance** [46] combines accuracy (average distance from points on the generated mesh to nearest points on the ground truth) and completeness (average distance from points on the ground truth to nearest points on the generated mesh). Following [47], we use Iterative Closest Point (ICP) [1] to align predicted shapes for fair comparisons.

4.3 Implementation Details

Following prior studies [25, 29, 39, 47], we start with a spherical mesh of 642 vertices and 1280 faces, scaling it into an ellipsoid with a fixed anisotropic factor of 0.6. Our setup uses 64×64 resolution for both image and texture maps across the ShapeNet and CUB-200-2011 datasets. We utilize the Soft Rasterizer [43] for differentiable rendering and a U-Net architecture [57] with six encoder and four decoder layers with channels of [16, 16, 32, 64, 128, 256] and [256 256,

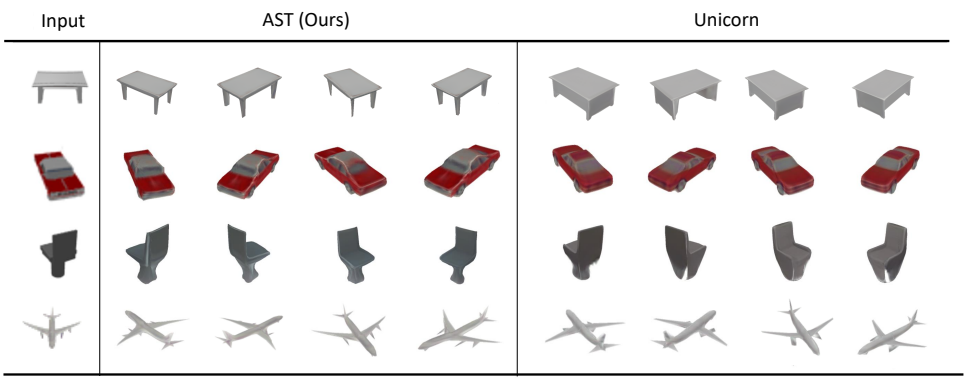

Fig. 5: Qualitative results on ShapeNet.

Table 2: Chamfer-L1 distance and Structural Similarity (SSIM) evaluation on ShapeNet. DVR uses camera pose annotations in addition to silhouette masks; Unicorn eliminates mask supervision through background modeling.

Chamfer-L1 ↓	**Table**	**Car**	**Chair**	**Airplane**	**Bench**	**Cabinet**	**Display**	**Lamp**	**Phone**	**Rifle**	**Sofa**	**Speaker**	**Vessel**
DVR [48]	0.303	0.203	0.371	0.114	0.255	0.254	0.257	0.363	0.191	0.130	0.321	0.312	0.180
Unicorn [47]	0.243	0.168	0.253	0.110	**0.159**	0.154	0.220	0.523	0.127	0.097	0.203	**0.235**	0.173
AST (Ours)	**0.218**	**0.159**	**0.245**	**0.109**	0.164	**0.132**	**0.217**	**0.356**	**0.096**	**0.089**	**0.193**	0.244	**0.160**
SSIM (%, ↑)	**Table**	**Car**	**Chair**	**Airplane**	**Bench**	**Cabinet**	**Display**	**Lamp**	**Phone**	**Rifle**	**Sofa**	**Speaker**	**Vessel**
Unicorn [47]	82.3	87.2	80.4	87.2	83.1	83.8	80.4	79.8	88.5	**90.7**	80.9	76.5	89.5
AST (Ours)	**87.0**	**89.2**	**86.4**	**94.1**	**88.1**	**87.9**	**87.6**	**90.6**	**92.4**	89.6	**85.5**	**83.7**	**92.0**

128, 128], respectively. The feature map output is $64 \times 64 \times 128$. A Resnet-18 extracts pose and global shape/texture codes, with six pose hypotheses. Each Transformer (Shape and Texture) operates with a single decoder layer. We apply various loss weights: rendering and cross-instance consistency at 1, and saliency, normal consistency, Laplacian smoothing, and uniformity regularization at 0.01, 0.01, 0.01, and 0.05, respectively. The Shape Transformer uses 642 queries, while the Texture Transformer employs 1024 to create a 32×32 texture map, upscaled to 64×64. Training occurs on a Nvidia V100 GPU using Adam [32] at a learning rate of 1×10^{-4}.

4.4 Evaluation on CUB

We present a quantitative comparison on the CUB-200-2011 dataset using metrics like Mask IoU, PCK for shape reconstruction, and SSIM for texture quality, as shown in Tab. 1. Our method surpasses previous state-of-the-art models in all metrics, demonstrating superior shape accuracy and texture realism. Qualitatively, as shown in Fig. 3, our approach outperforms methods like Unicorn [47], which compromises results due to its background modeling approach. UMR [39] and SMR [25] miss finer shape details and realistic textures, respectively, while HybridInv [52] offers competitive results but requires inefficient test-time opti-

mization. The input image size is 64×64. For better visual appeal, we display the original size as 256×256. Overall, our method effectively handles the inherent variability in natural images, producing precise and realistic 3D reconstructions, more reconstruction results are provided in in the supplementary material.

4.5 Evaluation on ShapeNet

We compare our approach with DVR [48], and Unicorn [47]. DVR relies on additional annotations like camera and keypoints along with silhouette masks for supervision, whereas Unicorn operates unsupervised, avoiding mask annotations through background modeling. In our quantitative evaluation (Tab. 2), we use Chamfer-L1 distance to assess shape dissimilarity and SSIM to evaluate texture quality. Results show our model outperforming Unicorn, particularly in complex shapes such as tables, and surpassing DVR which utilize extensive annotations. We excluded SMR from our comparison due to irreproducible results, also observed by Monnier *et al.* [47]; Similarly, we were unable to fully reproduce the Unicorn results in the cabinet, sofa, speaker, and vessel categories, so we have replaced them with our own reproduced results. Our model significantly excels in texture realism, evidenced by superior SSIM scores over Unicorn. Qualitative comparisons in Fig. 5 reveal our model's enhanced handling of subtle texture details, like the gray roof of the car and table legs, showcasing greater realism than Unicorn. Overall, our method provides more accurate shapes and realistic textures compared to previous models.

4.6 Additional Results

We also conducted experiments on a new large-scale real-world dataset, OmniObject3D [73]. We selected the *banana* category and evaluated the Chamfer-L1 distance and SSIM, comparing our method with Unicorn [47]. As shown in Tab. 3, the experimental results show that our method maintains superior performance on this latest dataset.

Table 3: Evaluation on OmniObject3D using Chamfer-L1 and SSIM.

Methods	**Banana**	
	Chamfer-L1 ↓	SSIM (%, ↑)
Unicorn [47]	0.375	81.7
Ours	**0.272**	**85.8**

4.7 Ablation Study

We conducted an ablation study to assess the impact of each module by separately removing the Shape Transformer (ST) and Texture Transformer (TT) and measuring Mask IoU, SSIM, and PCK on the CUB-200-2011 dataset. The results, detailed in Tab. 4, confirm that both ST and TT are crucial. The study shows that improvements in shape accuracy and texture quality are interdependent, facilitated by enhanced texture maps aiding accurate shape learning through image

Table 4: Ablation study on CUB-200-2011.

Methods	**Mask IoU** (%, ↑)	**SSIM** (%, ↑)	**PCK** (%, ↑)
Ours w/o ST	74.6	77.8	59.2
Ours w/o TT	75.3	65.3	55.5
Ours	**81.6**	**86.0**	**64.7**

rendering loss backpropagation. Visual results in Fig. 3 demonstrate degraded performance when either ST or TT is removed. For ShapeNet, removing ST results in a notable decrease in Chamfer-L1 distance performance, particularly for complex objects like chairs and tables, highlighting ST's role in refining shape. While removing TT has a lesser impact on shape metrics, it significantly deteriorates texture quality, as shown in Tab. 5 with further visual ablation results available in the supplementary material.

Table 5: Ablation study on ShapeNet using Chamfer-L1 and SSIM.

Methods	**Table**	**Car**	**Chair**	**Airplane**
	Chamfer-L1 ↓			
Ours w/o ST	0.314	0.172	0.282	0.111
Ours w/o TT	0.240	0.161	0.262	0.109
Ours	**0.218**	**0.159**	**0.245**	**0.109**
Methods	**Table**	**Car**	**Chair**	**Airplane**
	SSIM (%, ↑)			
Ours w/o ST	86.6	88.9	85.9	92.4
Ours w/o TT	83.4	86.3	83.1	91.9
Ours	**87.0**	**89.2**	**86.4**	**94.1**

5 Conclusion

This paper introduces a novel Analysis-by-Synthesis Transformer for single-view 3D reconstruction. It addresses the limitations of existing analysis-by-synthesis methods in shape reconstruction and texture generation through effective pixel-to-shape and pixel-to-texture modeling. Extensive experiments on the CUB-200-2011 and ShapeNet datasets demonstrate that our approach enhances reconstruction accuracy and texture quality, surpassing previous state-of-the-art methods.

Limitations: A limitation of our method is that it can only reconstruct meshes of a fixed topology that is homeomorphic to a sphere and cannot handle concave shapes. This is a common limitation shared by all analysis-by-synthesis methods that perform shape reconstruction by deforming a sphere mesh, including CMR [29], UCMR [18], SMR [25], and Unicorn [47]. In future work, we will integrate the proposed Analysis-by-Synthesis Transformer and the mesh topology modification network [49] for more flexible 3D reconstruction.

Acknowledgements. This work was supported in part by the National Science Foundation (NSF) grants ECCS-2400900 and CNS-1828265.

References

1. Besl, P.J., McKay, N.D.: Method for registration of 3-d shapes. In: Sensor fusion IV: control paradigms and data structures. vol. 1611, pp. 586–606. Spie (1992)
2. Bozic, A., Palafox, P., Thies, J., Dai, A., Nießner, M.: Transformerfusion: Monocular rgb scene reconstruction using transformers. Advances in Neural Information Processing Systems **34**, 1403–1414 (2021)
3. Carion, N., Massa, F., Synnaeve, G., Usunier, N., Kirillov, A., Zagoruyko, S.: End-to-end object detection with transformers. In: European conference on computer vision. pp. 213–229. Springer (2020)
4. Caron, M., Touvron, H., Misra, I., Jégou, H., Mairal, J., Bojanowski, P., Joulin, A.: Emerging properties in self-supervised vision transformers. In: Proceedings of the IEEE/CVF international conference on computer vision. pp. 9650–9660 (2021)
5. Chang, A.X., Funkhouser, T., Guibas, L., Hanrahan, P., Huang, Q., Li, Z., Savarese, S., Savva, M., Song, S., Su, H., et al.: Shapenet: An information-rich 3d model repository. arXiv preprint arXiv:1512.03012 (2015)
6. Chen, C., Chen, Q., Xu, J., Koltun, V.: Learning to see in the dark. In: Proceedings of the IEEE conference on computer vision and pattern recognition. pp. 3291–3300 (2018)
7. Chen, W., Ling, H., Gao, J., Smith, E., Lehtinen, J., Jacobson, A., Fidler, S.: Learning to predict 3d objects with an interpolation-based differentiable renderer. Advances in neural information processing systems **32** (2019)
8. Choy, C.B., Xu, D., Gwak, J., Chen, K., Savarese, S.: 3d-r2n2: A unified approach for single and multi-view 3d object reconstruction. In: Computer Vision–ECCV 2016: 14th European Conference, Amsterdam, The Netherlands, October 11-14, 2016, Proceedings, Part VIII 14. pp. 628–644. Springer (2016)
9. Deitke, M., Schwenk, D., Salvador, J., Weihs, L., Michel, O., VanderBilt, E., Schmidt, L., Ehsani, K., Kembhavi, A., Farhadi, A.: Objaverse: A universe of annotated 3d objects. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 13142–13153 (2023)
10. Di, Y., Zhang, C., Wang, P., Zhai, G., Zhang, R., Manhardt, F., Busam, B., Ji, X., Tombari, F.: Ccd-3dr: Consistent conditioning in diffusion for single-image 3d reconstruction. arXiv preprint arXiv:2308.07837 (2023)
11. Eigen, D., Puhrsch, C., Fergus, R.: Depth map prediction from a single image using a multi-scale deep network. Advances in neural information processing systems **27** (2014)
12. Fan, H., Su, H., Guibas, L.J.: A point set generation network for 3d object reconstruction from a single image. In: Proceedings of the IEEE conference on computer vision and pattern recognition. pp. 605–613 (2017)
13. Gadelha, M., Maji, S., Wang, R.: 3d shape induction from 2d views of multiple objects. In: 2017 International Conference on 3D Vision (3DV). pp. 402–411. IEEE (2017)
14. Gharbi, M., Chen, J., Barron, J.T., Hasinoff, S.W., Durand, F.: Deep bilateral learning for real-time image enhancement. ACM Transactions on Graphics (TOG) **36**(4), 1–12 (2017)
15. Girdhar, R., Fouhey, D.F., Rodriguez, M., Gupta, A.: Learning a predictable and generative vector representation for objects. In: Computer Vision–ECCV 2016: 14th European Conference, Amsterdam, The Netherlands, October 11-14, 2016, Proceedings, Part VI 14. pp. 484–499. Springer (2016)

16. Gkioxari, G., Malik, J., Johnson, J.: Mesh r-cnn. In: Proceedings of the IEEE/CVF international conference on computer vision. pp. 9785–9795 (2019)
17. Godard, C., Mac Aodha, O., Brostow, G.J.: Unsupervised monocular depth estimation with left-right consistency. In: Proceedings of the IEEE conference on computer vision and pattern recognition. pp. 270–279 (2017)
18. Goel, S., Kanazawa, A., Malik, J.: Shape and viewpoint without keypoints. In: Computer Vision–ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part XV 16. pp. 88–104. Springer (2020)
19. Groueix, T., Fisher, M., Kim, V.G., Russell, B.C., Aubry, M.: A papier-mâché approach to learning 3d surface generation. In: Proceedings of the IEEE conference on computer vision and pattern recognition. pp. 216–224 (2018)
20. Häne, C., Tulsiani, S., Malik, J.: Hierarchical surface prediction for 3d object reconstruction. In: 2017 International Conference on 3D Vision (3DV). pp. 412–420. IEEE (2017)
21. Henderson, P., Ferrari, V.: Learning single-image 3d reconstruction by generative modelling of shape, pose and shading. International Journal of Computer Vision **128**(4), 835–854 (2020)
22. Henderson, P., Tsiminaki, V., Lampert, C.H.: Leveraging 2d data to learn textured 3d mesh generation. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition. pp. 7498–7507 (2020)
23. Henzler, P., Mitra, N.J., Ritschel, T.: Escaping plato's cave: 3d shape from adversarial rendering. In: Proceedings of the IEEE/CVF International Conference on Computer Vision. pp. 9984–9993 (2019)
24. Hong, Y., Zhang, K., Gu, J., Bi, S., Zhou, Y., Liu, D., Liu, F., Sunkavalli, K., Bui, T., Tan, H.: Lrm: Large reconstruction model for single image to 3d. arXiv preprint arXiv:2311.04400 (2023)
25. Hu, T., Wang, L., Xu, X., Liu, S., Jia, J.: Self-supervised 3d mesh reconstruction from single images. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition. pp. 6002–6011 (2021)
26. Huang, Z., Jampani, V., Thai, A., Li, Y., Stojanov, S., Rehg, J.M.: Shapeclipper: Scalable 3d shape learning from single-view images via geometric and clip-based consistency. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 12912–12922 (2023)
27. Insafutdinov, E., Dosovitskiy, A.: Unsupervised learning of shape and pose with differentiable point clouds. Advances in neural information processing systems **31** (2018)
28. Johnson, J., Alahi, A., Fei-Fei, L.: Perceptual losses for real-time style transfer and super-resolution. In: Computer Vision–ECCV 2016: 14th European Conference, Amsterdam, The Netherlands, October 11-14, 2016, Proceedings, Part II 14. pp. 694–711. Springer (2016)
29. Kanazawa, A., Tulsiani, S., Efros, A.A., Malik, J.: Learning category-specific mesh reconstruction from image collections. In: Proceedings of the European Conference on Computer Vision (ECCV). pp. 371–386 (2018)
30. Kato, H., Harada, T.: Learning view priors for single-view 3d reconstruction. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition. pp. 9778–9787 (2019)
31. Kato, H., Ushiku, Y., Harada, T.: Neural 3d mesh renderer. In: Proceedings of the IEEE conference on computer vision and pattern recognition. pp. 3907–3916 (2018)
32. Kingma, D.P., Ba, J.: Adam: A method for stochastic optimization. arXiv preprint arXiv:1412.6980 (2014)

33. Kokkinos, F., Kokkinos, I.: Learning monocular 3d reconstruction of articulated categories from motion. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 1737–1746 (2021)
34. Kokkinos, F., Kokkinos, I.: To the point: Correspondence-driven monocular 3d category reconstruction. Advances in Neural Information Processing Systems **34**, 7760–7772 (2021)
35. Kulkarni, N., Gupta, A., Tulsiani, S.: Canonical surface mapping via geometric cycle consistency. In: Proceedings of the IEEE/CVF International Conference on Computer Vision. pp. 2202–2211 (2019)
36. Li, J., Fang, P.: Hdrnet: Single-image-based hdr reconstruction using channel attention cnn. In: Proceedings of the 2019 4th International Conference on Multimedia Systems and Signal Processing. pp. 119–124 (2019)
37. Li, L.H., Yatskar, M., Yin, D., Hsieh, C.J., Chang, K.W.: Visualbert: A simple and performant baseline for vision and language. arXiv preprint arXiv:1908.03557 (2019)
38. Li, X., Liu, S., De Mello, S., Kim, K., Wang, X., Yang, M.H., Kautz, J.: Online adaptation for consistent mesh reconstruction in the wild. Advances in Neural Information Processing Systems **33**, 15009–15019 (2020)
39. Li, X., Liu, S., Kim, K., De Mello, S., Jampani, V., Yang, M.H., Kautz, J.: Self-supervised single-view 3d reconstruction via semantic consistency. In: Computer Vision–ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part XIV 16. pp. 677–693. Springer (2020)
40. Lin, C.H., Kong, C., Lucey, S.: Learning efficient point cloud generation for dense 3d object reconstruction. In: proceedings of the AAAI Conference on Artificial Intelligence. vol. 32 (2018)
41. Lin, K., Wang, L., Liu, Z.: End-to-end human pose and mesh reconstruction with transformers. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition. pp. 1954–1963 (2021)
42. Liu, M., Xu, C., Jin, H., Chen, L., Varma T, M., Xu, Z., Su, H.: One-2-3-45: Any single image to 3d mesh in 45 seconds without per-shape optimization. Advances in Neural Information Processing Systems **36** (2024)
43. Liu, S., Li, T., Chen, W., Li, H.: Soft rasterizer: A differentiable renderer for image-based 3d reasoning. In: Proceedings of the IEEE/CVF International Conference on Computer Vision. pp. 7708–7717 (2019)
44. Long, J., Shelhamer, E., Darrell, T.: Fully convolutional networks for semantic segmentation. In: Proceedings of the IEEE conference on computer vision and pattern recognition. pp. 3431–3440 (2015)
45. Mandikal, P., Radhakrishnan, V.B.: Dense 3d point cloud reconstruction using a deep pyramid network. In: 2019 IEEE Winter Conference on Applications of Computer Vision (WACV). pp. 1052–1060 (2019). https://doi.org/10.1109/WACV.2019.00117
46. Mescheder, L., Oechsle, M., Niemeyer, M., Nowozin, S., Geiger, A.: Occupancy networks: Learning 3d reconstruction in function space. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition. pp. 4460–4470 (2019)
47. Monnier, T., Fisher, M., Efros, A.A., Aubry, M.: Share with thy neighbors: Single-view reconstruction by cross-instance consistency. In: European Conference on Computer Vision. pp. 285–303. Springer (2022)
48. Niemeyer, M., Mescheder, L., Oechsle, M., Geiger, A.: Differentiable volumetric rendering: Learning implicit 3d representations without 3d supervision. In: Pro-

ceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 3504–3515 (2020)
49. Pan, J., Han, X., Chen, W., Tang, J., Jia, K.: Deep mesh reconstruction from single rgb images via topology modification networks. In: Proceedings of the IEEE/CVF International Conference on Computer Vision. pp. 9964–9973 (2019)
50. Park, J.J., Florence, P., Straub, J., Newcombe, R., Lovegrove, S.: Deepsdf: Learning continuous signed distance functions for shape representation. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition. pp. 165–174 (2019)
51. Pavllo, D., Spinks, G., Hofmann, T., Moens, M.F., Lucchi, A.: Convolutional generation of textured 3d meshes. Advances in Neural Information Processing Systems **33**, 870–882 (2020)
52. Pavllo, D., Tan, D.J., Rakotosaona, M.J., Tombari, F.: Shape, pose, and appearance from a single image via bootstrapped radiance field inversion. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 4391–4401 (2023)
53. Peng, K., Islam, R., Quarles, J., Desai, K.: Tmvnet: Using transformers for multi-view voxel-based 3d reconstruction. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 222–230 (2022)
54. Qi, C.R., Su, H., Mo, K., Guibas, L.J.: Pointnet: Deep learning on point sets for 3d classification and segmentation. In: Proceedings of the IEEE conference on computer vision and pattern recognition. pp. 652–660 (2017)
55. Qi, C.R., Yi, L., Su, H., Guibas, L.J.: Pointnet++: Deep hierarchical feature learning on point sets in a metric space. Advances in neural information processing systems **30** (2017)
56. Radford, A., Kim, J.W., Hallacy, C., Ramesh, A., Goh, G., Agarwal, S., Sastry, G., Askell, A., Mishkin, P., Clark, J., et al.: Learning transferable visual models from natural language supervision. In: International conference on machine learning. pp. 8748–8763. PMLR (2021)
57. Ronneberger, O., Fischer, P., Brox, T.: U-net: Convolutional networks for biomedical image segmentation. In: Medical Image Computing and Computer-Assisted Intervention–MICCAI 2015: 18th International Conference, Munich, Germany, October 5-9, 2015, Proceedings, Part III 18. pp. 234–241. Springer (2015)
58. Shi, Z., Meng, Z., Xing, Y., Ma, Y., Wattenhofer, R.: 3d-retr: end-to-end single and multi-view 3d reconstruction with transformers. arXiv preprint arXiv:2110.08861 (2021)
59. Simonyan, K., Zisserman, A.: Very deep convolutional networks for large-scale image recognition. arXiv preprint arXiv:1409.1556 (2014)
60. Su, W., Zhu, X., Cao, Y., Li, B., Lu, L., Wei, F., Dai, J.: Vl-bert: Pre-training of generic visual-linguistic representations. arXiv preprint arXiv:1908.08530 (2019)
61. Tulsiani, S., Kulkarni, N., Gupta, A.: Implicit mesh reconstruction from unannotated image collections. arXiv preprint arXiv:2007.08504 (2020)
62. Tulsiani, S., Zhou, T., Efros, A.A., Malik, J.: Multi-view supervision for single-view reconstruction via differentiable ray consistency. In: Proceedings of the IEEE conference on computer vision and pattern recognition. pp. 2626–2634 (2017)
63. Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A.N., Kaiser, Ł., Polosukhin, I.: Attention is all you need. Advances in neural information processing systems **30** (2017)
64. Wang, D., Cui, X., Chen, X., Zou, Z., Shi, T., Salcudean, S., Wang, Z.J., Ward, R.: Multi-view 3d reconstruction with transformers. In: Proceedings of the IEEE/CVF International Conference on Computer Vision. pp. 5722–5731 (2021)

65. Wang, H., Zhu, Y., Adam, H., Yuille, A., Chen, L.C.: Max-deeplab: End-to-end panoptic segmentation with mask transformers. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition. pp. 5463–5474 (2021)
66. Wang, N., Zhang, Y., Li, Z., Fu, Y., Liu, W., Jiang, Y.G.: Pixel2mesh: Generating 3d mesh models from single rgb images. In: Proceedings of the European conference on computer vision (ECCV). pp. 52–67 (2018)
67. Wang, Z., Bovik, A.C., Sheikh, H.R., Simoncelli, E.P.: Image quality assessment: from error visibility to structural similarity. IEEE transactions on image processing **13**(4), 600–612 (2004)
68. Welinder, P., Branson, S., Mita, T., Wah, C., Schroff, F., Belongie, S., Perona, P.: Caltech-ucsd birds 200 (2010)
69. Wen, C., Zhang, Y., Li, Z., Fu, Y.: Pixel2mesh++: Multi-view 3d mesh generation via deformation. In: Proceedings of the IEEE/CVF international conference on computer vision. pp. 1042–1051 (2019)
70. Wu, J., Wang, Y., Xue, T., Sun, X., Freeman, B., Tenenbaum, J.: Marrnet: 3d shape reconstruction via 2.5 d sketches. Advances in neural information processing systems **30** (2017)
71. Wu, S., Li, R., Jakab, T., Rupprecht, C., Vedaldi, A.: Magicpony: Learning articulated 3d animals in the wild. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 8792–8802 (2023)
72. Wu, S., Makadia, A., Wu, J., Snavely, N., Tucker, R., Kanazawa, A.: De-rendering the world's revolutionary artefacts. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition. pp. 6338–6347 (2021)
73. Wu, T., Zhang, J., Fu, X., Wang, Y., Ren, J., Pan, L., Wu, W., Yang, L., Wang, J., Qian, C., et al.: Omniobject3d: Large-vocabulary 3d object dataset for realistic perception, reconstruction and generation. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 803–814 (2023)
74. Wu, Z., Song, S., Khosla, A., Yu, F., Zhang, L., Tang, X., Xiao, J.: 3d shapenets: A deep representation for volumetric shapes. In: Proceedings of the IEEE conference on computer vision and pattern recognition. pp. 1912–1920 (2015)
75. Ye, Y., Tulsiani, S., Gupta, A.: Shelf-supervised mesh prediction in the wild. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 8843–8852 (2021)
76. Zamir, S.W., Arora, A., Khan, S., Hayat, M., Khan, F.S., Yang, M.H.: Restormer: Efficient transformer for high-resolution image restoration. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition. pp. 5728–5739 (2022)
77. Zhang, J., Ren, D., Cai, Z., Yeo, C.K., Dai, B., Loy, C.C.: Monocular 3d object reconstruction with gan inversion. In: European Conference on Computer Vision. pp. 673–689. Springer (2022)
78. Zhang, X., Zhang, Z., Zhang, C., Tenenbaum, J., Freeman, B., Wu, J.: Learning to reconstruct shapes from unseen classes. Advances in neural information processing systems **31** (2018)
79. Zheng, S., Lu, J., Zhao, H., Zhu, X., Luo, Z., Wang, Y., Fu, Y., Feng, J., Xiang, T., Torr, P.H., et al.: Rethinking semantic segmentation from a sequence-to-sequence perspective with transformers. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition. pp. 6881–6890 (2021)
80. Zhu, R., Kiani Galoogahi, H., Wang, C., Lucey, S.: Rethinking reprojection: Closing the loop for pose-aware shape reconstruction from a single image. In: Proceedings of the IEEE International Conference on Computer Vision. pp. 57–65 (2017)