The Neurally-Guided Shape Parser: Grammar-based Labeling of 3D Shape Regions with Approximate Inference

R. Kenny Jones
Brown University

Aalia Habib
Brown University

Rana Hanocka
University of Chicago

Daniel Ritchie
Brown University

Abstract

We propose the Neurally-Guided Shape Parser (NGSP), a method that learns how to assign fine-grained semantic labels to regions of a 3D shape. NGSP solves this problem via MAP inference, modeling the posterior probability of a label assignment conditioned on an input shape with a learned likelihood function. To make this search tractable, NGSP employs a neural guide network that learns to approximate the posterior. NGSP finds high-probability label assignments by first sampling proposals with the guide network and then evaluating each proposal under the full likelihood. We evaluate NGSP on the task of fine-grained semantic segmentation of manufactured 3D shapes from PartNet, where shapes have been decomposed into regions that correspond to part instance over-segmentations. We find that NGSP delivers significant performance improvements over comparison methods that (i) use regions to group per-point predictions, (ii) use regions as a self-supervisory signal or (iii) assign labels to regions under alternative formulations. Further, we show that NGSP maintains strong performance even with limited labeled data or noisy input shape regions. Finally, we demonstrate that NGSP can be directly applied to CAD shapes found in online repositories and validate its effectiveness with a perceptual study.

1. Introduction

The ability to semantically segment 3D shapes is important for numerous applications in vision, graphics, and robotics: reverse-engineering the part structure of an object to support editing and manipulation; producing training data for structure-aware generative shape models [10, 14, 24]; helping autonomous agents understand how to interact with objects in their environment [1]; and more. These applications often demand that the parts detected be fine-scale (e.g. wheels of an office chair) and hierarchically-organized (e.g. a cabinet door decomposes into a handle, door, and frame). Producing such segmentations has proved to be a challenging task, as it is expensive to gather large amounts of data at this granularity; PartNet [25] is the only existing large-scale dataset of this type.

Recent work on 3D shape semantic segmentation has mainly focused on end-to-end approaches that operate on shape *atoms* (e.g. mesh faces, point cloud points, occupancy grid voxels), i.e. the lowest-level geometric entity in the input representation [12, 28, 29, 38]. While these methods achieve impressive performance on many tasks, they do not often transfer well to domains with fine-grained labels or when access to labeled data is limited. We postulate that one reason for this phenomenon is that attempting to label shape *atoms* directly results in a massive search space, allowing learning-based methods to overfit unless the ratio of labeled shape instances to the label set complexity is high.

One way to address this issue is to design systems that make use of shape *regions*. When the number of shape *regions* becomes significantly smaller than the number of shape *atoms*, the label assignment problem becomes easier. Such a framing may allow methods to learn fine-grained semantic segmentation when access to labeled data is limited. When shape *regions* are provided, they can be used in various ways: (i) as a post-process aggregation on top of shape *atom* predictions, (ii) to formulate auxiliary self-supervised objectives, or (iii) as the object to be labeled. Methods that operate within this last paradigm can more directly reason about relationships *between* regions, which can help improve fine-grained segmentation performance by better considering the context of a region within the entire shape.

The problem of decomposing a shape into regions useful for semantic segmentation is application-dependent. For CAD shapes and scenes found in online repositories, this type of region decomposition is often produced as a by-product of the modeling process, e.g. each part instance will be made out of one or more connected mesh components [22, 34, 43]. Discovering region decompositions for shapes that do not already provide them is a well-studied problem within computer vision and graphics. There has been considerable recent effort on unsupervised techniques that approximate 3D shapes with primitives [6, 17, 27, 32, 33],

and there is a long history of research on shape segmentation through purely geometric analysis [3, 15, 36]. There is even reason to believe that region decomposition solutions can generalize across shape categories, i.e. the way that shapes (especially manufactured objects) decompose into parts is largely category-independent [11, 44].

In this paper, we propose the Neurally-Guided Shape Parser (NGSP), a method that learns to assign fine-grained labels from a semantic grammar to regions of a 3D shape. Our approach is based on maximum *a posteriori* (MAP) inference in a model of the probability that a label assignment to the shape's regions is correct. Our likelihood consists of a mixture of modules that each operate on some regions of the shape. One set of modules evaluates the validity of the implied geometry and spatial layout for each label in the semantic grammar. Another module evaluates groups of regions formed by the label assignment. As this combinatorial search problem is too complex to solve with exhaustive enumeration, we employ a neural guide network to approximate the posterior. The guide network reasons locally, predicting the label probability for each region independently. Using the per-region probabilities produced by the guide network, NGSP importance samples a set of proposed label assignments. To choose the best proposal out of this set, each label assignment is evaluated under the full likelihood, and the sample with highest posterior probability is chosen.

We compare NGSP against methods that use shape regions as a post-process, a self-supervisory signal, or assign labels to regions with different search strategies and likelihood formulations. We evaluate each method on the task of fine-grained semantic segmentation of manufactured 3D shapes from PartNet, where each method has access to regions from the annotated part instance over-segmentations (e.g. each semantic part instance may consist of multiple regions). NGSP achieves the best semantic segmentation performance, even in paradigms where access to labeled data is limited or when the input shape regions are noisy. To validate our design decisions, we run an ablation study measuring the effect of each likelihood term and the neural guide network. Finally, we show that NGSP can find good semantic segmentations on 'in the wild' CAD shapes found from online repositories, and evaluate its performance with a forced choice perceptual study against comparison methods. Code for our method and experiments can be found at found at https://github.com/rkjones4/NGSP .

In summary, our contributions are:

(i) We present the Neurally-Guided Shape Parser (NGSP), a method that learns how to assign labels from a semantic grammar to regions of a 3D shape. NGSP performs approximate MAP inference, using a guide network to find high-probability label assignments under a learned posterior probability of a label assignment conditioned on an input shape.

(ii) We demonstrate that NGSP finds better fine-grained semantic segmentations for manufactured shapes compared with methods that use shape regions in alternative learning paradigms.

2. Related Work

Semantic Segmentation with 3D Shape Atoms Most learning-based methods for 3D shape semantic segmentation have used shape atoms (points, faces, edges, voxels) as their fundamental unit to label. This practice dates back to pre-deep-learning work using conditional random fields on mesh faces [16] and extends to present-day, neural network methods including PointNet [28], PointNet++ [29], MeshCNN [12], and DGCNN [35]. Some methods have been designed for settings where labeled data is limited, either in terms of the number of labels provided for each shape [23, 40] or the number of shapes that contain any labels at all [5, 8, 30]. While approaches within this paradigm achieve state-of-the-art performance for coarse, non-hierarchical segmentation, we show experimentally that they do not work as well in hierarchical, fine-grained settings where more inter-part relational reasoning is helpful.

Region-based Semantic Segmentation of Images and Scenes Our approach of decomposing a 3D shape into regions is conceptually similar to decomposing a 2D image into superpixels; there exist some prior work leveraging superpixels to improve image semantic segmentation. Some of these methods use superpixels or other larger image regions to increase the computational efficiency of semantic segmentation [26] or to produce segmentation masks with crisper edges [9, 41]. A few of these methods, like ours, focus on achieving high accuracy with less training data [2, 19, 42].

Similar ideas have also been proposed for segmenting 3D scenes. For large-scale scenes, points have been grouped into super-points to make learning approaches computationally tractable [13, 20]. Some 3D scene segmentation approaches explicitly compute labels per shape region. One approach over-segments an indoor scene point cloud then uses a recursive denoising autoencoder to infer a hierarchical organization of those segments [31]. Another converts over-segmented indoor scenes into consistent hierarchies via dynamic-programming-based, bottom-up grammar parsing [22]. The latter approach is similar to ours in that it also learns likelihoods from data; however, the scenes considered are more simplistic and easier to decompose into manageable sub-sections than the shapes we consider. In general, while scenes can be represented with point clouds, they have different characteristics from 3D shapes: scenes contain much fewer regular substructures and are more sparsely populated.

3D Shape Semantic Hierarchies There is a long tradition of organizing 3D shapes and scenes into hierarchies. Such hierarchies can be based on spatial locality or other metrics relating to convenience of editing and rendering, as

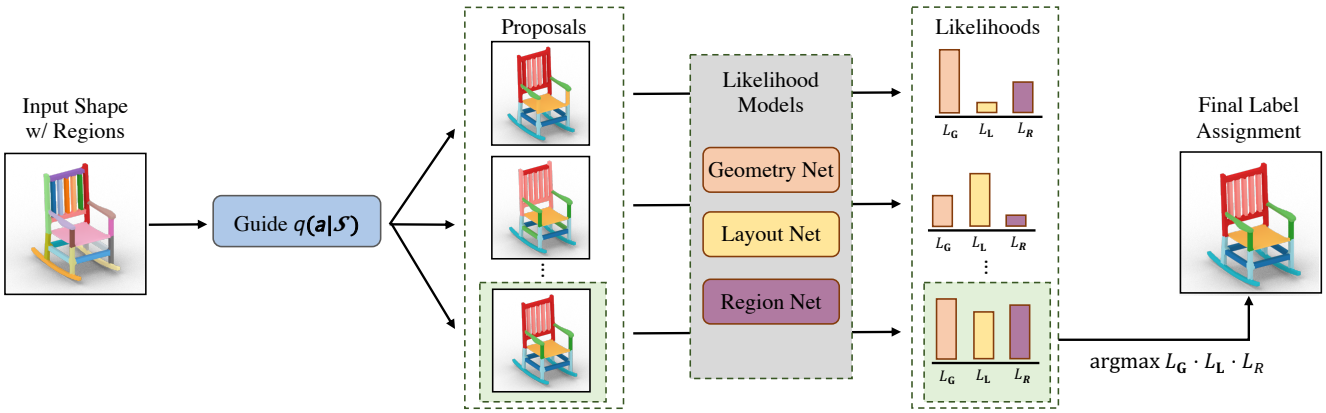

Figure 1. The Neurally-Guided Shape Parser (NGSP) learns to assign fine-grained semantic labels (rightmost) to shape regions (leftmost). A guide network generates a set of proposed label assignments. The label assignments are sent through likelihood modules that evaluate the global coherence of each proposal. These terms are combined into a posterior probability which determines the final label assignment.

in classical computer graphics. One can also arrange part-based shapes into binary hierarchies based on connectivity and symmetry relationships between their parts [37]; such a hierarchy can be a useful organization of shape data for training structure-aware generative models [21]. A generalization of this approach is to consider n-ary hierarchies; this is the data representation adopted by PartNet [25], which supports more sophisticated structure-aware generative shape models [14, 24]. Our method for semantic segmentation is designed with these kinds of hierarchies in mind and can help produce training data for such generative models.

Semantic Segmentation with 3D Shape Regions There has been some prior work that learns to assign semantic labels to 3D shape regions. One approach first learns how to group over-segmented shape regions from stock 3D models into part hypotheses, and then finds an optimal label assignment to each part hypothesis through a CRF formulation [34]. However, this method is not designed for hierarchical grammars, as it is unable to separate semantic parts that share similar bounding boxes, which is necessary for the fine-grained segmentations we desire (e.g. distinguishing a seat frame from a seat surface). Another approach proposes an MRF formulation where unary potentials capture per-region label probabilities and paired potentials encourage a smoothness term in relation to the grammar hierarchy [43]. We will show experimentally that NGSP outperforms this formulation on the task of fine-grained semantic segmentation.

Relatedly, some approaches have made use of a shape region decomposition to formulate self-supervised learning objectives. One such method trains a PointNet++ to perform semantic segmentation, but also enforces a contrastive loss on per-point embeddings, encouraging points from the same shape region to share similar embeddings [8]. This technique achieves impressive performance on few-shot coarse segmentation tasks when a large collection of unsupervised shapes augments the labeled data set. We compare NGSP against this approach, and find that NGSP makes better use of shape regions for fine-grained segmentations, even with limited labeled data.

3. Method

The input to our method is a shape $\mathcal{S}$ which has been decomposed into a set of regions R, i.e. $\mathcal{S} = \{R_i\}$. Our method also receives as input a label grammar $\mathcal{G} = (L, \omega, P)$, where L is a set of possible semantic labels, ω is the root label (the *axiom* of the grammar), and $P \subset L \times L^*$ is the set of production rules for the grammar (specifying which labels can be the children of other labels). The label set L can be divided into *terminal* labels $L_\mathbf{T}$ (those with no children) and *non-terminal* labels $L_\mathbf{V}$, such that $L = L_\mathbf{T} \cup L_\mathbf{V}$. We assume that there exists a unique path from the root to each terminal label $l_\mathbf{T} \in L_\mathbf{T}$, i.e. every label has at most one parent. This is a reasonable assumption for shape labeling; all PartNet [25] label grammars have this property.

Given these inputs, our goal is to find the maximum *a posteriori* (MAP) label assignment $\mathbf{A} = \{\mathbf{a}_i\}$, where $\mathbf{a}_i = \mathbf{A}(R_i)$ is the label assigned to region $R_i \in \mathcal{S}$. We assume a uniform prior distribution over labels and model the posterior $p(\mathcal{S}|\mathbf{A})$ with a data-driven likelihood function:

$$\mathcal{L}(\mathcal{S}, \mathbf{A}) = \mathcal{L}_\mathbf{G}(\mathcal{S}, \mathbf{A}) \cdot \mathcal{L}_\mathbf{L}(\mathcal{S}, \mathbf{A}) \cdot \mathcal{L}_\mathbf{R}(\mathcal{S}, \mathbf{A}) \quad (1)$$

$\mathcal{L}_\mathbf{G}$ and $\mathcal{L}_\mathbf{L}$ reason about properties of the semantic labels of $\mathcal{G}$, while $\mathcal{L}_\mathbf{R}$ reasons about properties of groups of regions implied by a given assignment.

As the search space of label assignments to shape regions is large, especially with fine-grained label sets, we guide our search with a network that learns to locally approximated the posterior: $q(\mathbf{a}|\mathcal{S})$. Figure 1 outlines our approach. Using this guide network, we importance sample a set of complete label assignments, which we call proposals. These proposals

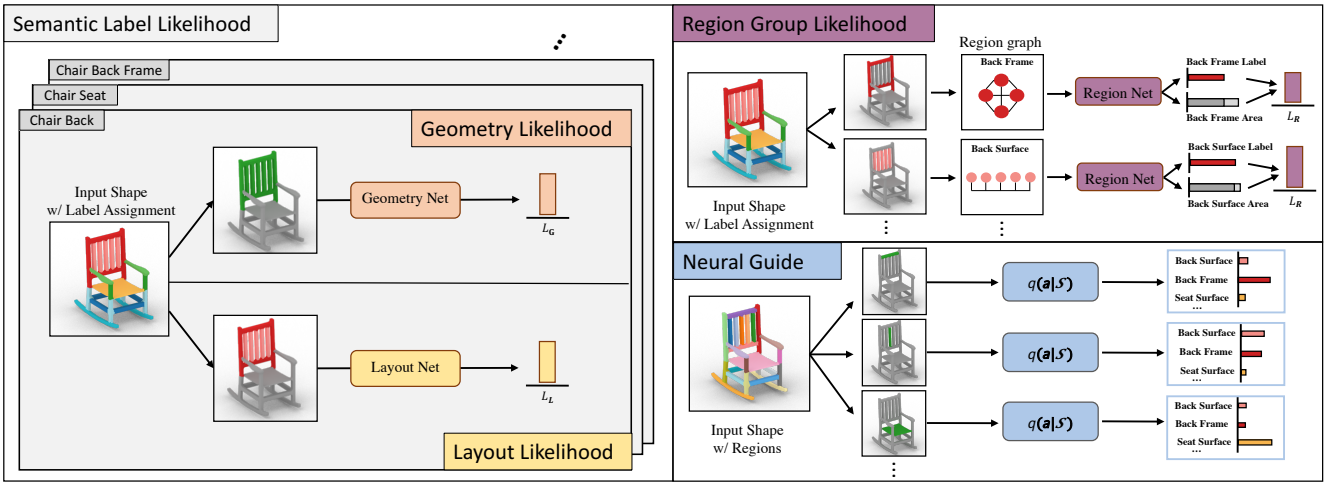

Figure 2. Design of NGSP's modules. The geometry and layout likelihoods consume a (shape, label assignment) pair, and are computed for each semantic label in the grammar *(left)*. Each geometry network sees which regions of the input shape have been assigned to its label (e.g. chair back). Each layout network sees which regions of the input shape have been assigned to its child labels (e.g. chair back surface and chair back frame). The region group likelihood term also takes a (shape, label assignment) pair as input *(top-right)*. For each group of regions implied by the label assignment, it creates a fully-connected graph, where nodes correspond to shape regions in the group. The neural guide network operates over individual shape regions, predicting the label for each region independently *(bottom-right)*

are then evaluated under Equation 1, and the proposal that returns the highest likelihood is chosen as the final output label assignment.

In the remainder of this section, we describe the different components of this pipeline in more detail: the semantic label likelihood terms (Section 3.1), the region group likelihood term (Section 3.2), and details of our neurally-guided search procedure (Section 3.3).

3.1. Semantic Label Likelihood Terms

For each label in the grammar, the semantic label likelihood terms reason about different properties of shape regions that were assigned to that label. Specifically, for each $l \in \mathcal{G}$, we learn to identify geometric properties of l with a geometry likelihood $\mathcal{L}_{\mathbf{G}}$ and semantic layout properties of l with a layout likelihood $\mathcal{L}_{\mathbf{L}}$. $\mathcal{L}_{\mathbf{G}}$ aims to capture information about the typical geometric properties of regions assigned to a given label (e.g. chair seats usually have a flat top surface); $\mathcal{L}_{\mathbf{L}}$ aims to capture typical spatial relationships between a label's children (e.g. within a chair base, a rocker is usually positioned underneath the legs). Both of these likelihoods are modeled with the same structure:

$$\mathcal{L}_{\mathbf{G}}(\mathcal{S}, \mathbf{A}) = \Big(\prod_{l \in L} p_{\mathbf{G}}(\mathcal{S}_{l:\mathbf{A}} | l) \Big)^{1/(\sum_{l \in L} \mathbb{1}[l \in \mathbf{A}])}$$

$$\mathcal{L}_{\mathbf{L}}(\mathcal{S}, \mathbf{A}) = \Big(\prod_{l \in L} p_{\mathbf{L}}(\mathcal{S}_{l:\mathbf{A}} | l) \Big)^{1/(\sum_{l \in L} \mathbb{1}[l \in \mathbf{A}])}$$

$$\mathcal{S}_{l:\mathbf{A}} = \{ R \in \mathcal{S} \text{ s.t. } \mathbf{A}(R) = l \}$$

where $\mathcal{S}_{l:\mathbf{A}}$ is the subset of regions in the shape $\mathcal{S}$ which are assigned to label l in assignment $\mathbf{A}$. The exponents normalize these probabilities by the number of labels which occur in the label assignment $\mathbf{A}$, e.g. the number of non-unity product terms.

We model the geometry network $p_{\mathbf{G}}(\mathcal{S}_{l:\mathbf{A}} | l)$ and layout network $p_{\mathbf{L}}(\mathcal{S}_{l:\mathbf{A}} | l)$ with PointNet++ architectures, where each input point cloud contains surface samples from $\mathcal{S}_{l:\mathbf{A}}$ (Figure 2, left). Conditioning on l is implemented by training separate $p_{\mathbf{G}}$ and $p_{\mathbf{L}}$ networks for each label $l \in L$.

Each network is trained in a binary classification paradigm, tasked with assessing whether the regions in $\mathcal{S}_{l:\mathbf{A}}$ are a valid instance of a semantic part with label l. Positive examples are sourced from the training dataset: the networks for label l receive one positive example of $\mathcal{S}_{l:\mathbf{A}}$ from each shape where l appears. Negative examples come from synthetically-generated corruptions of each positive example (i.e. changing region labels). To encourage the geometry and layout networks to focus on the properties after which they are named, we introduce the following inductive biases (details in the supplemental):

Geometry Network: The geometry network should learn to reason about whether the shape of the union of regions in $\mathcal{S}_{l:\mathbf{A}}$ is consistent with the label l. Thus, each negative example is derived by adding or removing regions from a positive example.

Layout Network: The layout network should focus on whether the relationships between the child labels of regions assigned to l are consistent with that label. To enable this reasoning, the network receives the child label as an additional one-hot attribute concatenated to every point. Each negative example is derived by modifying the child label assignment of at least one region from a positive example.

3.2. Region Group Likelihood Term

The region group likelihood term reasons about properties of region groups implicitly formed when a labeling is assigned to an input shape. Specifically, it models the probability that $(\mathcal{S}, \mathbf{A})$ pairs are valid with respect to region groups $\mathcal{R}$ of $\mathcal{S}$ formed under $\mathbf{A}$. For each $l_{\mathbf{T}} \in \mathbf{A}$, the region group $\mathcal{R}_l$ is defined to be $\{R_i \in \mathcal{S} \mid \mathbf{a}_i = l_{\mathbf{T}}\}$.

$\mathcal{L}_{\mathbf{R}}$ reasons over two properties of each $\mathcal{R}_l$: if l is the best label for $\mathcal{R}_l$ and what percentage of area within $\mathcal{R}_l$ belongs to l. We model these properties with a region network $p_{\mathbf{R}}$. It consumes a region group $\mathcal{R}_l$, and predicts the probability that $\mathcal{R}_l$ has l as its majority label, $p_{\mathbf{R}}^{\text{label}}$, and the percentage of the area within $\mathcal{R}$ that has l as its true label, $p_{\mathbf{R}}^{\text{area}}$. These predictions are then combined and normalized across all region groupings:

$$\mathcal{L}_{\mathbf{R}}(\mathcal{S}, \mathbf{A}) = \Big(\prod_{l \in L} p_{\mathbf{R}}^{\text{label}}(\mathcal{R}_l|l) \cdot p_{\mathbf{R}}^{\text{area}}(\mathcal{R}_l|l) \Big)^{1/|\mathcal{R}|}$$

We model $p_{\mathbf{R}}$ with a region-based graph convolutional network (Figure 2, top-right). We convert each $\mathcal{R}_l$ into a fully-connected graph where the nodes correspond to the regions of $\mathcal{R}_l$. We initialize node and edge features with embeddings predicted by a pretrained point cloud auto-encoder; details are provided in the supplemental. $p_{\mathbf{R}}$ performs 4 rounds of gated graph convolution, then creates a single latent representation for the entire graph with a max-pooling layer [4, 7]. $p_{\mathbf{R}}^{\text{label}}$ is modeled with a linear layer that predicts a probability distribution over the terminal label set. $p_{\mathbf{R}}^{\text{area}}$ is conditioned on l and modeled with a linear layer that predicts a scalar value in $[0, 1]$, where 0 implies none of the area within $\mathcal{R}_l$ belongs to l and 1 implies all of the area within $\mathcal{R}_l$ belongs to l.

3.3. Neurally-Guided Search

While the search space over regions is much smaller than the search space over atoms, it is still computationally infeasible to exhaustively evaluate $\mathcal{L}$ on all possible label assignments to regions. To guide our search procedure towards good areas of the search space, we learn a guide network $q(\mathbf{a}|\mathcal{S})$ to locally approximate the posterior.

We model $q(\mathbf{a}|\mathcal{S})$ with a neural network trained to predict the probability of each possible label assignment $\mathbf{a}_i$ for each region R_i of the shape $\mathcal{S}$. $q(\mathbf{a}|\mathcal{S})$ uses a PointNet++ architecture [29], where the input point cloud contains samples from the entire shape, but each point has an extra one-hot dimension indicating whether it belongs to the region of interest (Figure 2, bottom-right). We train $q(\mathbf{a}|\mathcal{S})$ in a classification paradigm, where each shape $\mathcal{S}$ in the dataset produces $|\mathcal{S}|$ training examples (one for each region), and the classification target for each example is the ground truth semantic label of that region. We can then calculate the approximate posterior guide probability, $\mathcal{L}_{\mathbf{Q}}$, of a $(\mathcal{S},\mathbf{A})$ pair with the following equation:

$$\mathcal{L}_{\mathbf{Q}}(\mathcal{S}, \mathbf{A}) = \prod_{i=1}^{|\mathcal{S}|} q(\mathbf{a}_i)$$

At inference time, our goal is to find high likelihood label assignments $\mathbf{A}$ for a given shape $\mathcal{S}$. To achieve this, our procedure creates a set of proposed label assignments by using $q(\mathbf{a}|\mathcal{S})$ to importance sample the top k label assignments to $\mathcal{S}$ under $\mathcal{L}_{\mathbf{Q}}$. We then evaluate each proposed assignment under $\mathcal{L}$ and select the label assignment within this set which maximizes Equation 1.

4. Experiments

In this section, we evaluate NGSP's ability to assign semantic labels to regions of 3D shapes. Our experiments use CAD manufactured objects from the PartNet dataset [25] (Section 4.1). We describe the details of our training procedure in Section 4.2. In Section 4.3, we compare NGSP against region-aware comparison methods on the task of semantic segmentation under varying amounts of labeled training data. We provide an ablation study on the components of NGSP in Section 4.4. We examine how NGSP is affected when input shape regions are artificially corrupted (Section 4.5) or are produced by an ACD method (Section 4.6). Finally, in Section 4.7 we run NGSP on 'in the wild' CAD shapes, and compare its predicted segmentations against alternative methods with a forced choice perceptual study.

4.1. Data

We consider six categories of manufactured shapes from PartNet [25]: chairs, lamps, tables, storage furniture, vases, and knives. We use PartNet's hierarchical labelings as our ground truth: on average, each label grammar contains 34 total labels and 21 leaf labels. The dataset for each category contains between 300 and 1200 shapes, split between train, validation and test sets. We over-segment each shape using the mesh components for each part instance in PartNet (a part instance may consist of multiple components). For training and inference, we convert each mesh into point clouds with a surface sampling. Full details are provided in the supplemental material.

4.2. Training Details

The layout, geometry, and region label networks are trained with binary cross entropy. The region area network is trained with L1 loss. The guide network is trained with focal cross entropy loss [39]. We use the Adam optimizer [18] with a learning rate of 10^{-3} for the guide network and 10^{-4} for all other networks. All networks perform early stopping using the validation set. Models were trained sequentially on a machine with a GeForce RTX 2080 Ti GPU with an Intel i9-9900K CPU, consuming up to 10GB of GPU memory and taking between 1-2 days to train for the categories with

# Train	Method	Mean	Chair	Lamp	Table	Vase	Knife	Storage
10	PartNet (R)	18.1	25.3	10.2	3.2	12.6	33.2	24.2
	BAE-NET (R)	20.7	23.3	10.7	11.0	35.7	22.2	21.8
	LEL (R)	20.1	31.1	14.3	8.6	12.6	27.4	26.8
	LHSS	24.3	24.7	16.7	13.0	33.3	**34.1**	23.9
	NGSP	**33.6**	**36.6**	**24.7**	**16.3**	**58.8**	29.3	**35.9**
40	PartNet (R)	31.6	39.4	24.5	19.1	44.9	25.5	36.0
	BAE-NET (R)	26.5	30.5	19.0	13.1	42.4	27.9	25.9
	LEL (R)	38.6	45.4	26.4	26.1	48.0	45.3	40.3
	LHSS	35.4	35.7	23.3	20.1	50.0	44.3	39.1
	NGSP	**50.9**	**53.6**	**42.8**	**30.4**	**76.2**	**49.7**	**52.9**
400	PartNet (R)	41.2	49.0	24.6	37.8	53.9	42.1	39.9
	BAE-NET (R)	30.4	34.7	29.6	16.6	44.3	28.7	28.3
	LEL (R)	41.9	48.0	38.0	38.2	46.4	41.2	39.4
	LHSS	36.3	43.7	29.0	31.2	45.0	33.1	36.0
	NGSP	**57.9**	**63.6**	**44.6**	**45.3**	**84.6**	**55.9**	**53.2**

Table 1. Fine-grained semantic segmentation results across different PartNet categories. The metric is mIoU (higher values are better). NGSP significantly outperforms other methods that make alternative use of shape regions. This trend remains consistent even in limited labeled data regimes (# Train column).

more semantic labels. See the supplemental material for full details about network architectures.

4.3. Fine-Grained Semantic Segmentation

We compare NGSP against alternative region labeling methods on the task of semantic segmentation. All evaluations are performed on a held-out test set. Unless otherwise stated, the number of sampled proposals from the guide network, k, is set to 10000. Following PartNet, we use mIoU as our evaluation metric: the intersection over union between predicted and ground-truth per-point labels, averaged over labels in the grammar.

We compare NGSP to the following methods. Methods appended by (R) make per-point predictions which are aggregated with an average operation into per-region predictions to form a full label assignment.

- **PartNet (R)**: De-facto approach for fine-grained semantic segmentation that uses a PointNet++ to predict into the terminal label set [25].
- **BAE-NET (R)**: Implicit field network that jointly learns to semantically segment and reconstruct shapes; designed for limited labeled data [5].
- **LEL (R)**: PointNet++ back-bone where shape region decompositions formulate a self-supervised training objective augmenting the classification loss; designed for limited labeled data [8].
- **LHSS**: Constructs an MRF where nodes correspond to shape regions. Finds low-cost label assignments over learned unary and grammar-based pairwise potentials with an alpha-expansion algorithm [43].

Each method is trained with access to the same labeled shape instances. BAE-NET and LEL are additionally provided with up to 1000 shape instances per class that lack semantic label annotations but contain region decompositions. Full details are provided in the supplemental.

Model	10 Train	40 Train	400 Train
No $\mathcal{L}_{\mathbf{G}}$	30.7	47.2	57.3
No $\mathcal{L}_{\mathbf{L}}$	29.0	46.1	56.3
No $\mathcal{L}_{\mathbf{R}}$	32.7	48.0	54.0
No $\mathcal{L}$	29.3	43.0	51.6
No $q(\mathbf{a}\|\mathcal{S})$	11.7	13.3	13.0
NGSP	**33.6**	**50.9**	**57.9**

Table 2. Semantic segmentation performance of NGSP under different ablation conditions (metric is mIoU, averaged across categories). Each component of NGSP helps it find good label assignments.

Results: Quantitative results of this experiment are shown in Table 1. When labeled data is plentiful (400 max training shapes, bottom rows), NGSP outperforms the comparison methods by a significant margin. Looking at the mean result across categories, NGSP offers a 38% improvement over the next best method (LEL). When access to labeled data is limited, NGSP also outperforms alternatives with a 31% improvement when 10% of the training data is used and a 38% improvement when 2.5% of the training data is used. In fact, NGSP's mean category performance with 10% of the labeled data outperforms any comparison method that has access to all of the labeled data by almost 10 absolute percentage points. This result suggests that NGSP could be useful for semantic segmentation of 3D shapes from uncommon categories for which datasets of semantically annotated instances are not readily available.

We present some qualitative comparisons from the same experiment in Figure 3, and provide additional examples in the supplemental. NGSP is able to find label assignments that are more coherent, and better reflect the ground-truth labels, compared with the alternative methods. Methods that rely on regions to group per-atom predictions often produce segmentations that lack global consistency. LHSS attempts to reason about global consistency with its pairwise potentials, but these encourage the output segmentation to become overly smooth, missing fine-grained part distinctions.

4.4. Ablation Study

To evaluate the design of NGSP, we conduct a series of ablations, where each formulation has one component of NGSP removed:

- **No** $\mathcal{L}_{\mathbf{G}}$: Geometry likelihood is removed from $\mathcal{L}$.
- **No** $\mathcal{L}_{\mathbf{L}}$: Layout likelihood is removed from $\mathcal{L}$.
- **No** $\mathcal{L}_{\mathbf{R}}$: Region group likelihood is removed from $\mathcal{L}$.
- **No** $\mathcal{L}$: The best proposal under $\mathcal{L}_{\mathbf{Q}}$ is chosen.
- **No** $q(\mathbf{a}|\mathcal{S})$: $\mathcal{L}$ evaluates proposals from a uniform prior.

We present results of this experiment in Table 2. As we show across multiple training set sizes, removing any

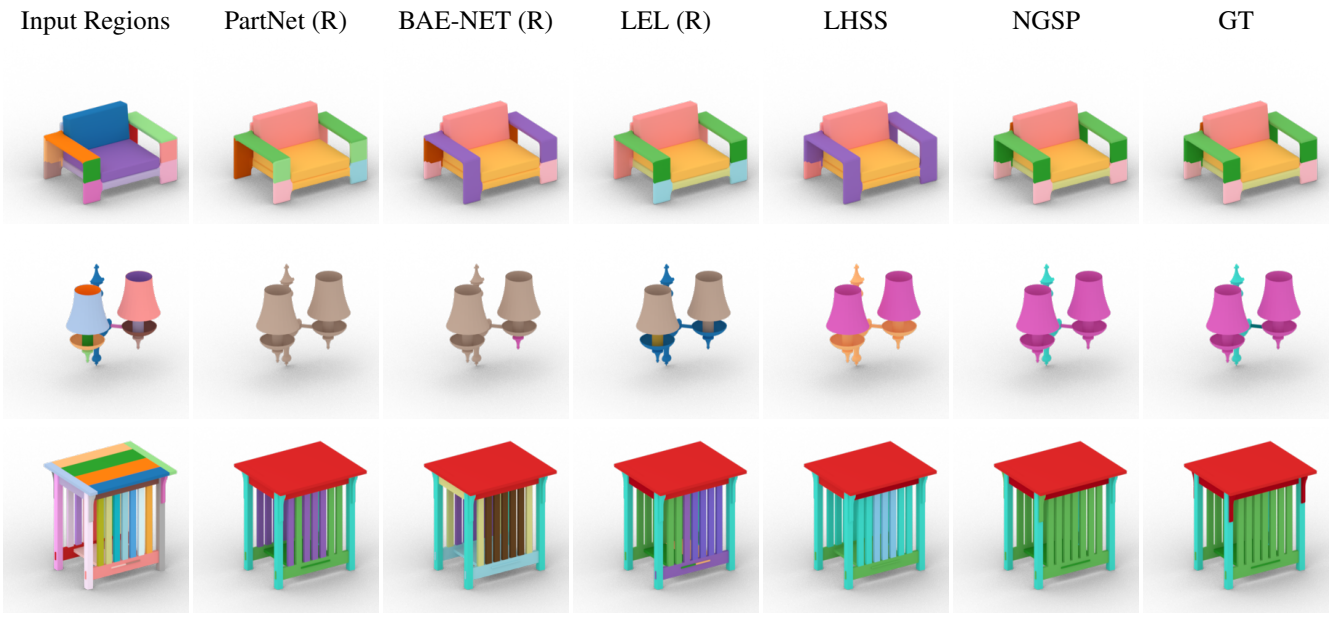

Figure 3. Qualitative comparison of fine-grained semantic segmentations. We show the input shape regions *(left)*, the ground-truth label assignment *(right)*, and the label assignments produced by different methods *(middle)*. Each semantic label is represented by a unique color. NGSP predicts label assignments that best agree with the ground-truth. We present additional qualitative results in the supplemental.

Method	1X Reg	2X Reg	4X Reg
PartNet (R)	41.2	40.7	40.7
BAE-NET(R)	30.4	30.3	29.9
LEL(R)	41.9	41.7	41.3
LHSS	36.3	35.9	35.4
NGSP	**57.9**	**49.0**	**45.3**

Table 3. We evaluate the semantic segmentation performance of different methods in regimes where shape regions have undergone artificial corruption (metric is mIoU, averaged across categories). NGSP's performance declines gradually as the corruption increases, but in all cases remains better than alternative methods.

component of $\mathcal{L}$ (top 3 rows) leads to a worse mIoU. The "No $q(\mathbf{a}|\mathcal{S})$" row demonstrates the importance of the neural guide network: the search space is too large to effectively explore in a naive manner. However, as seen in the "No $\mathcal{L}$" row, the predictions of $q(\mathbf{a}|\mathcal{S})$ can be furthered improved by evaluating its proposals under a better estimate of the posterior. As $q(\mathbf{a}|\mathcal{S})$ only evaluates regions locally, it is unable to benefit by reasoning about part-to-part relationships implied by the global label assignment in the same way as $\mathcal{L}$.

4.5. Sensitivity to Region Corruption

We analyze how sensitive NGSP is to corruptions of the part instance over-segmented regions of the input shapes. For this analysis, we construct datasets of shapes whose regions have been artificially split into smaller sub-regions. In the 2X (4X) paradigm, each region is split into 2 (4) regions; details of how these splits are produced are provided in the supplemental. For each corruption paradigm, the neural guide network is retrained on training shapes whose regions have undergone similar corruption. Results of this experiment are shown in Table 3, where we track semantic segmentation performance against baselines which receive the same corrupted regions. As the amount of region corruption increases, the performance of NGSP declines, but in every condition it continues to offer performance improvements over all comparison methods.

4.6. Applications to Unstructured Data

As NGSP requires a region decomposition as input, it can't be directly applied to some types of unstructured data without the help of auxiliary methods. While there are many methods that aim to convert unstructured shape data into a reasonable region decomposition, all existing methods have limitations, and this remains a hard, unsolved problem. However, even though these region decompositions may contain errors, NGSP can still use them to improve semantic segmentation performance when access to labeled data is limited. We run an experiment comparing NGSP against alternative region labeling methods over unstructured input data, with regions created by the ACD method from [8]. We report the mean category mIoU that each method achieves with ACD produced regions while training over 10 training shapes (Table 4). In this paradigm, NGSP makes the best use of the ACD regions, but all methods perform worse compared with using the PartNet provided regions (Table 1).

Method	Mean mIoU
PartNet + NR	0.155
PartNet + ACD	0.161
BAE-NET + ACD	0.180
LEL + ACD	0.206
LHSS + ACD	0.202
NGSP + ACD	**0.244**

Table 4. Semantic segmentation performance over unstructured input data with ACD generated regions and 10 labeled training shapes (NR is no regions).

NGSP vs.	Mean	95% CI
PartNet (R)	79.1	[66.1, 92.1]
LHSS	79.6	[68.1, 91.1]

Table 5. Quantitative results of our perceptual study comparing semantic segmentations produced by different methods on 'in the wild' CAD shapes. NGSP's label assignments were significantly preferred over those predicted by Partnet (R) or LHSS.

4.7. Applications to 'in the wild' CAD Shapes

As a byproduct of CAD modeling procedures, many 'in the wild' 3D shapes come with part instance over-segmentations. NGSP can segment such objects by treating each mesh connected component as a shape region. To demonstrate this application, we compile a small dataset of 26 meshes from the chair category of ShapeNet, where each shape's connected components form a reasonable approximation to a part instance over-segmentation. We run NGSP and two comparison methods (PartNet (R) and LHSS) on each shape and record each method's predicted label assignment. As we lack ground-truth label annotations for these shapes, we evaluate NGSP with a two-alternative forced choice perceptual study. Each participant was shown a sequence of examples, where each example visualized two ways that parts of a chair could be labeled, and was asked to select the part labeling that better matched the given shape. Further details provided in the supplemental.

Results We present the results of this perceptual study in Table 5. Participants had a strong preference for the part labelings generated by NGSP. In comparisons against PartNet, NGSP was preferred 79.1% on average, with a 95% confidence interval lower-bound of 66.1%. In comparisons against LHSS, NGSP was preferred 79.6% on average, with a 95% confidence interval lower-bound of 68.1%.

5. Conclusion

We presented the Neurally-Guided Shape Parser (NGSP), a method that performs semantic segmentation on region-decomposed 3D shapes. NGSP assigns labels to shape regions via MAP inference in a learned model of the probability that a label assignment is correct conditioned on the shape's regions. Search is made tractable through an approximate inference scheme, where the exploration of label assignments is constrained by a neural guide network. We experimentally demonstrated that NGSP outperforms methods that (i) use regions to aggregate point predictions (ii) incorporate regions into self-supervised training objectives or (iii) assign labels to regions in alternative search-based formulations. We observed that these trends remain consistent with limited labeled data and with noisy shape regions. Finally, we applied NGSP to a set of 'in the wild' CAD shapes and validated that it produced better semantic decompositions than alternative approaches with a perceptual study.

When presented with an unstructured shape that lacks a region decomposition, NGSP must rely on other methods to produce suitable regions. Many methods that decompose shapes represented as raw sensor input (e.g. point clouds) into primitive parts do so at too coarse a granularity for fine-grained segmentation [6, 17, 27, 32, 33]. However, the input region requirements for NGSP may actually be weaker than what most of these approaches aim to produce: as shown in Sections 4.5 and 4.6, NGSP offers advantages even when the input regions poorly approximate the target part instances. Developing unsupervised methods for producing such 'instance over-segmentations' is a good direction for future work.

Looking forward, we believe that NGSP's framing of 3D shape semantic segmentation as approximate inference in a probabilistic model suggests a vision for how this task could be scaled beyond carefully-curated research datasets to 'in-the-wild' scenarios. In the future, we plan to design likelihood terms that cannot be easily accommodated by end-to-end approaches; these could include hard-to-differentiate terms that consider functional part relationships such as adjacency, symmetry, or physical support (e.g. the chair base should physically support the chair seat). These terms could potentially be provided by a person via explicit rules, either in advance or with a human-in-the-loop system. Paradigms that allow integration of such symbolic rules with data-driven models could be a key step towards producing high-quality semantic segmentations in few-shot or zero-shot scenarios.

Acknowledgments

We would like to thank the participants in our user study for their contribution to our research. We would also like to thank the anonymous reviewers for their helpful suggestions. Renderings of part cuboids and point clouds were produced using the Blender Cycles renderer. This work was funded in parts by NSF award #1941808 and a Brown University Presidential Fellowship. Daniel Ritchie is an advisor to Geopipe and owns equity in the company. Geopipe is a start-up that is developing 3D technology to build immersive virtual copies of the real world with applications in various fields, including games and architecture

References

[1] Ben Abbatematteo, Stefanie Tellex, and George Konidaris. Learning to generalize kinematic models to novel objects. In *Proceedings of the 3rd Conference on Robot Learning*, 2019. 1

[2] Iñigo Alonso and Ana C. Murillo. Semantic segmentation from sparse labeling using multi-level superpixels. In *2018 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 5785–5792, 2018. 2

[3] Shmuel Asafi, Avi Goren, and Daniel Cohen-Or. Weak convex decomposition by lines-of-sight. In *Computer graphics forum*, volume 32, pages 23–31. Wiley Online Library, 2013. 2

[4] Xavier Bresson and Thomas Laurent. Residual gated graph convnets. *arXiv preprint arXiv:1711.07553*, 2017. 5

[5] Zhiqin Chen, Kangxue Yin, Matthew Fisher, Siddhartha Chaudhuri, and Hao Zhang. Bae-net: Branched autoencoder for shape co-segmentation. *Proceedings of International Conference on Computer Vision (ICCV)*, 2019. 2, 6

[6] Boyang Deng, Kyle Genova, Soroosh Yazdani, Sofien Bouaziz, Geoffrey Hinton, and Andrea Tagliasacchi. Cvxnet: Learnable convex decomposition. June 2020. 1, 8

[7] Vijay Prakash Dwivedi, Chaitanya K Joshi, Thomas Laurent, Yoshua Bengio, and Xavier Bresson. Benchmarking graph neural networks. *arXiv preprint arXiv:2003.00982*, 2020. 5

[8] Matheus Gadelha, Aruni RoyChowdhury, Gopal Sharma, Evangelos Kalogerakis, Liangliang Cao, Erik Learned-Miller, Rui Wang, and Subhransu Maji. Label-efficient learning on point clouds using approximate convex decompositions. In *European Conference on Computer Vision (ECCV)*, 2020. 2, 3, 6, 7

[9] Jun Gao, Zian Wang, Jinchen Xuan, and Sanja Fidler. Beyond fixed grid: Learning geometric image representation with a deformable grid. In *ECCV*, 2020. 2

[10] Lin Gao, Jie Yang, Tong Wu, Yu-Jie Yuan, Hongbo Fu, Yu-Kun Lai, and Hao (Richard) Zhang. Sdm-net: Deep generative network for structured deformable mesh. In *SIGGRAPH Asia*, 2019. 1

[11] Songfang Han, Jiayuan Gu, Kaichun Mo, Li Yi, Siyu Hu, Xuejin Chen, and Hao Su. Compositionally generalizable 3d structure prediction. 2020. 2

[12] Rana Hanocka, Amir Hertz, Noa Fish, Raja Giryes, Shachar Fleishman, and Daniel Cohen-Or. Meshcnn: A network with an edge. *ACM Transactions on Graphics (TOG)*, 38(4):90, 2019. 1, 2

[13] Shi-Min Hu, Jun-Xiong Cai, and Yu-Kun Lai. Semantic labeling and instance segmentation of 3d point clouds using patch context analysis and multiscale processing. *IEEE Transactions on Visualization and Computer Graphics*, 26(7):2485–2498, 2020. 2

[14] R. Kenny Jones, Theresa Barton, Xianghao Xu, Kai Wang, Ellen Jiang, Paul Guerrero, Niloy J. Mitra, and Daniel Ritchie. Shapeassembly: Learning to generate programs for 3d shape structure synthesis. *ACM Transactions on Graphics (TOG), Siggraph Asia 2020*, 39(6):Article 234, 2020. 1, 3

[15] Oliver Van Kaick, Noa Fish, Yanir Kleiman, Shmuel Asafi, and Daniel Cohen-Or. Shape segmentation by approximate convexity analysis. *ACM Transactions on Graphics (TOG)*, 34(1):1–11, 2014. 2

[16] Evangelos Kalogerakis, Aaron Hertzmann, and Karan Singh. Learning 3D Mesh Segmentation and Labeling. *ACM Transactions on Graphics*, 29(3), 2010. 2

[17] Yuki Kawana, Yusuke Mukuta, and Tatsuya Harada. Neural star domain as primitive representation. In *NeurIPS 2020*, 2020. 1, 8

[18] Diederik P. Kingma and Jimmy Ba. Adam: A method for stochastic optimization. *CoRR*, abs/1412.6980, 2014. 5

[19] Suha Kwak, Seunghoon Hong, and Bohyung Han. Weakly supervised semantic segmentation using superpixel pooling network. In *AAAI Conference on Artificial Intelligence*, 2017. 2

[20] Loic Landrieu and Martin Simonovsky. Large-scale point cloud semantic segmentation with superpoint graphs. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 4558–4567, 2018. 2

[21] Jun Li, Kai Xu, Siddhartha Chaudhuri, Ersin Yumer, Hao Zhang, and Leonidas Guibas. GRASS: Generative recursive autoencoders for shape structures. *ACM Transactions on Graphics (TOG)*, 36(4):52, 2017. 3

[22] Tianqiang Liu, Siddhartha Chaudhuri, Vladimir G. Kim, Qixing Huang, Niloy J. Mitra, and Thomas Funkhouser. Creating consistent scene graphs using a probabilistic grammar. *ACM Trans. Graph.*, 33(6), Nov. 2014. 1, 2

[23] Zhengzhe Liu, Xiaojuan Qi, and Chi-Wing Fu. One thing one click: A self-training approach for weakly supervised 3d semantic segmentation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 1726–1736, 2021. 2

[24] Kaichun Mo, Paul Guerrero, Li Yi, Hao Su, Peter Wonka, Niloy Mitra, and Leonidas Guibas. StructureNet: Hierarchical graph networks for 3D shape generation. In *SIGGRAPH Asia*, 2019. 1, 3

[25] Kaichun Mo, Shilin Zhu, Angel X. Chang, Li Yi, Subarna Tripathi, Leonidas J. Guibas, and Hao Su. PartNet: A large-scale benchmark for fine-grained and hierarchical part-level 3D object understanding. In *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2019. 1, 3, 5, 6

[26] Hyojin Park, Jisoo Jeong, Youngjoon Yoo, and Nojun Kwak. Superpixel-based semantic segmentation trained by statistical process control. *arXiv preprint arXiv:1706.10071*, 2017. 2

[27] Despoina Paschalidou, Angelos Katharopoulos, Andreas Geiger, and Sanja Fidler. Neural parts: Learning expressive 3d shape abstractions with invertible neural networks. In *Proceedings IEEE Conf. on Computer Vision and Pattern Recognition (CVPR)*, 2021. 1, 8

[28] Charles R Qi, Hao Su, Kaichun Mo, and Leonidas J Guibas. Pointnet: Deep learning on point sets for 3D classification and segmentation. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 652–660, 2017. 1, 2

[29] Charles Ruizhongtai Qi, Li Yi, Hao Su, and Leonidas J Guibas. Pointnet++: Deep hierarchical feature learning on point sets in a metric space. In *Advances in neural information processing systems*, pages 5099–5108, 2017. 1, 2, 5

[30] Gopal Sharma, Evangelos Kalogerakis, and Subhransu Maji. Learning point embeddings from shape repositories for few-shot segmentation. In *2019 International Conference on 3D Vision, 3DV 2019, Québec City, QC, Canada, September 16-19, 2019*, pages 67–75. IEEE, 2019. 2

[31] Yifei Shi, Angel X. Chang, Zhelun Wu, Manolis Savva, and Kai Xu. Hierarchy denoising recursive autoencoders for 3d scene layout prediction. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2019. 2

[32] Chun-Yu Sun, Qian-Fang Zou, Xin Tong, and Yang Liu. Learning adaptive hierarchical cuboid abstractions of 3d shape collections. *ACM Trans. Graph.*, 38(6), Nov. 2019. 1, 8

[33] Shubham Tulsiani, Hao Su, Leonidas J. Guibas, Alexei A. Efros, and Jitendra Malik. Learning Shape Abstractions by Assembling Volumetric Primitives. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2017. 1, 8

[34] Xiaogang Wang, Bin Zhou, Haiyue Fang, Xiaowu Chen, Qinping Zhao, and Kai Xu. Learning to group and label fine-grained shape components. *ACM Trans. Graph.*, 37(6), Dec. 2018. 1, 3

[35] Yue Wang, Yongbin Sun, Ziwei Liu, Sanjay E. Sarma, Michael M. Bronstein, and Justin M. Solomon. Dynamic graph cnn for learning on point clouds. *ACM Transactions on Graphics (TOG)*, 2019. 2

[36] Yanzhen Wang, Kai Xu, Jun Li, Hao Zhang, Ariel Shamir, Ligang Liu, Zhiquan Cheng, and Yueshan Xiong. Symmetry hierarchy of man-made objects. In *Computer graphics forum*, volume 30, pages 287–296. Wiley Online Library, 2011. 2

[37] Yanzhen Wang, Kai Xu, Jun Li, Hao Zhang, Ariel Shamir, Ligang Liu, Zhiquan Cheng, and Yueshan Xiong. Symmetry hierarchy of man-made objects. *Computer Graphics Forum (Eurographics 2011)*, 30(2):287–296, 2011. 3

[38] Z. Wu, S. Song, A. Khosla, F. Yu, L. Zhang, X. Tang, and J. Xiao. 3d shapenets: A deep representation for volumetric shapes. In *Computer Vision and Pattern Recognition*, 2015. 1

[39] Kai Xu, Rui Ma, Hao Zhang, Chenyang Zhu, Ariel Shamir, Daniel Cohen-Or, and Hui Huang. Organizing heterogeneous scene collection through contextual focal points. *ACM Transactions on Graphics (TOG)*, 33(4):Article 35, 2014. 5

[40] Xun Xu and Gim Hee Lee. Weakly supervised semantic point cloud segmentation: Towards 10x fewer labels. In *CVPR*, 2020. 2

[41] Zhiwei Xu, Thalaiyasingam Ajanthan, and Richard Hartley. Refining semantic segmentation with superpixel by transparent initialization and sparse encoder, 2020. 2

[42] Xiong Yan and Xiaohua Liu. Weakly supervised image semantic segmentation based on clustering superpixels. In Hui Yu and Junyu Dong, editors, *Ninth International Conference on Graphic and Image Processing (ICGIP 2017)*, volume 10615 of *Society of Photo-Optical Instrumentation Engineers (SPIE) Conference Series*, page 106151Y, Apr. 2018. 2

[43] Li Yi, Leonidas Guibas, Aaron Hertzmann, Vladimir G. Kim, Hao Su, and Ersin Yumer. Learning hierarchical shape segmentation and labeling from online repositories. *SIGGRAPH*, 2017. 1, 3, 6

[44] Chenyang Zhu, Kai Xu, Siddhartha Chaudhuri, Li Yi, Leonidas J. Guibas, and Hao Zhang. Adacoseg: Adaptive shape co-segmentation with group consistency loss. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2020. 2