Tags2Parts: Discovering Semantic Regions from Shape Tags

Sanjeev Muralikrishnan¹   Vladimir G. Kim²   Siddhartha Chaudhuri¹

¹IIT Bombay   ²Adobe Research

Abstract

We propose a novel method for discovering shape regions that strongly correlate with user-prescribed tags. For example, given a collection of chairs tagged as either “has armrest” or “lacks armrest”, our system correctly highlights the armrest regions as the main distinctive parts between the two chair types. To obtain point-wise predictions from shape-wise tags we develop a novel neural network architecture that is trained with tag classification loss, but is designed to rely on segmentation to predict the tag. Our network is inspired by U-Net, but we replicate shallow U structures several times with new skip connections and pooling layers, and call the resulting architecture WU-Net. We test our method on segmentation benchmarks and show that even with weak supervision of whole shape tags, our method is able to infer meaningful semantic regions, without ever observing shape segmentations. Further, once trained, the model can process shapes for which the tag is entirely unknown. As a bonus, our network architecture is directly operational in a strongly-supervised scenario and outperforms state-of-the-art strongly-supervised methods on standard benchmarks.

1. Introduction

Online repositories contain millions of 3D shapes, providing rich data for a wide range of data-driven 3D modeling interfaces. While these repositories often provide tags, textual descriptions, and soft categorization to facilitate text-based search, these labels are typically provided for the entire shape, and not at the region level. Many applications require finer shape understanding, for example, parts and their labels are essential for assembly-based interactive modeling interfaces. While one can obtain these labels by training a fully supervised segmentation model [5], this level of supervision requires substantially more involved annotation interfaces and human effort, making it infeasible for massive and growing online repositories. Existing methods for discovering semantic regions without explicit supervision are typically guided by geometric features and similarities (e.g. [4]), but these methods are prone to failure and tend to be tailored to a specific notion of parts, implicitly encoded by algorithm design.

Weakly- or semi-supervised methods have been proposed as a compromise between supervised and unsupervised techniques. For example, Yi et al. [23] propose to leverage metadata available in existing repositories in the form of scene graphs, which provided some segments and labels for a small subset of shapes. This metadata is very sparse and specific to computer graphics models. In contrast, tags for entire shapes are abundant, and also accompany scanned or automatically reconstructed shapes. In the absence of any annotations, collecting detailed region-wise labeling is a very time-consuming and tedious process [24], that can be significantly simplified if only the presence/absence of a particular region of interest needs to be indicated. In this work we propose a novel method for discovering regions from shape tags without explicit region-wise labeling. For example, in a collection of shapes tagged as “has armrest” and “does not have armrest”, we are able to identify the armrest components of the chairs in the former category (Figure 1). Further, once trained, our method can process shapes for which the tag is entirely unknown.

Our main challenge is that the weak supervisory signal (whole object tag) is different from the target network output (point-wise labels). To address this challenge, we use a neural network that addresses both problems simultaneously, and train it for whole object tagging while relying on point-wise labels to infer the tags. In particular, we propose a novel neural network architecture with skip connections, which we call WU-Net (Figure 2), that is inspired by the U-Net [15] architecture previously used for image segmentation problems. We make two important modifications. First, to regularize the network and improve localization of segments we replicate the ‘U’ structure
Figure 1: Armrests of chairs identified from only shape-level tags: “has armrest” or “does not have armrest”. The weakly-supervised problem is solved by a novel deep neural network architecture which we called WU-Net. The highlighted red regions are the automatically generated outputs of the network, with no postprocessing other than symmetrization.

thrice (‘WU’) and add skip connections both within and across them. Second, since the network is originally designed for segmentation, we add two layers for tag classification: average pooling followed by max pooling. Average pooling encourages coherent regions, forcing the network to train for segments that can help as much as possible with the tag classification task. This network architecture is our main technical contribution.

To evaluate our approach we use shapes from the standard segmentation dataset [24], but withhold region-wise labels and only tag shapes based on presence and absence of parts. Our method is able to detect regions with remarkable accuracy without observing a single segmented shape. As a bonus, we further observe that our approach is also suitable for standard fully-supervised segmentation, and demonstrate that our network architecture outperforms existing state-of-the-art techniques in fully supervised setting.

2. Related work

We overview related work on shape and image segmentation with various degrees of supervision.

Unsupervised shape segmentation One can leverage shape similarities across objects and local geometric cues to discover parts (e.g. [3] [17] [4]). These methods typically encode certain priors for what constitutes a part, and thus can only be used to discover parts that conform to these priors (e.g. partition the shape into similarly-sized elements, have boundaries align with sharp geometric features, and exhibit geometric similarities across shapes). Not all semantic regions conform to these priors: as some might span a few parts, or a subset of a part, or have feature-less boundary. To bias unsupervised methods towards more semantic regions, Yi et al. [23] proposed to leverage messy and inconsistent scene graphs provided by modelers, which is an implicit, ambient data available for many graphics models. Unfortunately, these data are sparse, only informative for some categories of shapes. None of these methods provide any means to control the output of the algorithm, which renders them unable to discover custom user-prescribed regions.

Supervised shape segmentation The easiest way to address this problem is to use shapes with manually-defined region-wise labels to train a model that can discover similar semantic regions in new shapes [6]. Most recent techniques use various kinds of neural network architectures to achieve this task, where the network design changes depending on the shape representation. Existing networks analyze shapes based on 2D renderings [5], local descriptors after spectral alignment to some canonical shape [26], unordered point sets [13], and mesh surfaces mapped to a canonical 2D domain [9]. In this work we choose a voxel grid as our shape representation, which has been used for classification in the past [13]. (Alternative representations such as point sets and collections of 2D views could
also potentially be used.) We focus our efforts on designing a network that leverages skip connections and stacked down- and up-sampling steps to facilitate segmentation. Although it was not designed specifically for this purpose, we found that our WU-Net outperforms state-of-the-art tools even in a fully supervised setting.

The need to collect labeled data is the main bottleneck for supervised methods. Several methods have been developed to minimize the cost of training a segmentation algorithm. For example, Wang et al. [22] actively choose the next shape to label such that it most improves a supervised method. Yi et al. [25] combine manual annotation with manual (quick) verification of the output of a supervised segmentation algorithm, which significantly reduces the time to label a dataset, since providing region-wise labels is the most time-consuming step. These techniques still require tedious manual segmentation of shapes. The goal of our work is to avoid this altogether by using a less taxing form of supervision, which is known as weakly supervised analysis in computer vision.

**Distinctive regions in shapes** In prior work most relevant to ours, Shilane and Funkhouser [16] propose a metric to highlight regions that are common to a category and different across categories, yielding distinctive regions. Their similarities and dissimilarities are based on hand-crafted per-point shape descriptors. In this work we demonstrate that it is possible to learn this representation via a neural network directly from a voxelized shape, and apply it to the problem of fine-grained shape segmentation within a single category. Further, unlike Shilane-Funkhouser, our method can be directly applied to test shapes where the tag is unknown, since it implicitly involves a classification step. In comparative evaluations, our method significantly outperforms that of Shilane and Funkhouser.

**Weakly-supervised image segmentation** Several computer vision methods can infer more localized object data from whole-scene tags (e.g. [19, 21, 2]). With the rise of deep neural networks, researchers observed that neuron activations in a trained classification network often pick up salient object parts [18]. Oquab et al. [11] append a global max-pooling layer to a fully convolutional segmentation network [8] to obtain a classification network suitable for object localization. They use the localization score in the last layer before pooling to estimate the object position. In our work we focus on segmentation, and found that global max-pooling does not favor detecting coherent regions. Thus we prefix it with average pooling to obtain a smoother segmentation. We also found that skip connections in WU-Net improve segmentation results over sequentially stacked convolutions. Pathak et al. [12] demonstrate that additional constraints can be used to favor segmentations, and some version of these constraints, suitable for 3D geometry, can potentially be incorporated into our framework.

### 3. Method

#### 3.1. Data representation

In our system, a 3D shape is represented in voxelized form. Specifically, given a 64 × 64 × 64 cubical grid tightly fitting the shape, we set every voxel that intersects the shape surface to 1, and the remaining voxels to 0. In our experiments, we did not generate interior voxels. Apart from being the most natural do-
main for building 3D convolutional pipelines, the voxel representation ensures that we do not take any advantage of inherent part structure in the shape meshes. In fact, our input need not be meshes at all, as long as we have some sort of densely sampleable surface.

3.2. Network architecture

Our method for weakly-supervised 3D shape segmentation utilizes a novel feedforward neural network architecture, which we call WU-Net. It is inspired by the U-Net architecture of Ronneberger et al. [15], which was proposed as an effective way to segment biomedical images with limited training data in a strongly supervised setting. U-Net’s prominent feature, from which it derives its name, is a sequence of fully convolutional downsampling layers (the “contracting” arm of an ‘U’), followed by an inverse sequence of fully convolutional upsampling layers (the “expanding” arm of the ‘U’), with the two sequences bridged by skip connections.

The WU-Net architecture leverages this building block by linking three fully convolutional U structures in sequence, i.e. a ‘W’ followed by an U (Figure 2). Data flowing through the network therefore goes through three successive cycles of down- and upsampling, from $64 \times 64 \times 64$ to $32 \times 32 \times 32$ and back to $64 \times 64 \times 64$, encouraging spatial coherence and spread in the detected signal. Unlike U-Net, our U’s are very shallow, each one involving a single downsampling/upsampling sequence. We explain the rationale for this design choice below.

Our architecture also has skip connections like U-Net, which allow reasoning in later layers to be sensitive to structure in the original data which may have been lost during downsampling. Unlike the original U-Net, the WU-Net skip connections also provide bridging connections between different U structures (see the dashed arrows in the lower row of layers in Figure 2). This provides an elegant symmetry between the high and low resolution paths in the structure, with data winding back and forth between the resolutions while also having a secondary flow within the layers at each resolution. We also discuss this design choice below.

To map from a layer at one resolution to a layer at the same resolution (orange arrows), we employ several $5^3$ convolutional kernels. To downsample, we use a max pooling operator over a $2^3$ neighborhood. To upsample, we use bilinear interpolation of the feature map. All neurons in the ‘WU’ structure have ReLU activations.

Discussion of design choices. WU-Net has three shallow U’s instead of a single deep one, bridged by skip connections at both high and low resolutions. These design choices enable convolutional filters in later layers to have a high effective field of view (by composition with filters from preceding layers) even on the high resolution data. We can think of each shallow U as performing a mild summarization of the signal and then, by virtue of the “high-resolution” skip connection spanning its arms, analysing it jointly with the unsummarized signal. The “low resolution” skip connections, meanwhile, provide each summarization step access to previous summaries. We show via ablation studies (Section 4) that each successive stacked U improves performance.

Note the contrast to U-Net, where the latter half of the basic architecture reconstructs successively higher resolution signals from a single drastic summary in the bottleneck layer. While skip connections do provide access to undecimated signals, the results of the joint high- and low-resolution analysis at each level are not further summarized, but simply upsampled to the next level. The filters in the final layer cannot have a high field of view on the original signal unmodified by down-sampling. WU-Net enables stacked summarization + joint analysis steps, all at the input resolution, so that filters in later layers can (for appropriately learned weights) have high field of view not only on the summarized signal, but also on the original signal.

Investigating the interplay between depth of U’s and chaining multiple U’s is a fruitful direction for future work.

Output segmentation map. The output of the final U is fed to two or more segmentation branches, one for each class to be detected. In a standard binary classification setup, e.g. “has back” vs “does not have back” for chairs, there are two branches. In our strongly supervised segmentation setup, there is one branch for each part label: ‘seat’, ‘back’, ‘leg’, ‘arm’ etc. Each branch consists of a single convolutional layer with a $3 \times 3 \times 3$ filter, with a sigmoid activation function. This layer acts as the segmentation map – it is in one-to-one correspondence with the input, and its output values are taken to represent the probability of each voxel having a particular class label.

Loss function. For strongly supervised segmentation, a per-voxel cross-entropy loss is directly applied to the output of the segmentation map layer. For weakly supervised segmentation, we apply a layer of $2 \times 2 \times 2$ average pooling to this output, and then taking the maximum over the entire pooled response. The average pooling layer encourages a wider response region. Greater pooling helps identify larger regions salient to training labels. We study the effect of varying the pooling radius in some detail in Section 4. To prevent ac-
tivating empty voxels near the shape boundaries, we first multiply each element of the segmentation map layer by the corresponding element of the input voxel grid, letting the network focus only on errors over the shape.

**Symmetrization.** Most shapes in our dataset have prominent symmetries, typically global reflectional symmetry. Since this implies the shape has redundant local information, a classification network can achieve high accuracy without needing to look at the complete shape. WU-Net is no exception, and our part detection results frequently demonstrate prominent and consistent asymmetry. This yields high precision but lower recall, for instance when only the right arms of chairs are detected (see figure on the right). To correct this, we simply mirror inferred salient regions on both sides of the symmetry plane.

### 3.3. Training

In the weakly supervised setting, the WU-Net architecture is trained in two phases. We found the two-phase training to give better results than a single phase alone. The phases are described below.

**Phase 1 (no output segmentation map).** In this phase, the final segmentation branches are removed and a simple classification layer is temporarily appended to the ‘WU’ structure. Specifically, this layer computes the maximum, over all voxels, of each of the 12 ‘WU’ output channels, followed by a fully-connected map from the 12 maxima to two outputs (the label of the complete shape, e.g. “armrest” vs “no armrest”). This network is trained with a cross-entropy classification loss until the classification accuracy on both the training set and a held-out validation set go above 95%. As soon as this happens, we adjudge the network to have achieved a high generalization accuracy and move to the next phase of training. Any further phase 1 training tends to cause overfitting and poorer results.

**Phase 2 (with output segmentation maps).** In the second phase, we remove the temporary classification layer, restore the segmentation branches, and train the whole network end-to-end. Here, too, we found a benefit in slowly increasing the size of the average-pooling kernel, starting from 0 (no average pooling) for 50 epochs, followed by 10 epochs for each expansion of the kernel. We obtained the best across-the-board performance with a $2 \times 2 \times 2$ average-pooling kernel, and report all comparative results with this setting. However, for specific datasets even higher performance may be obtained by gradually increasing the kernel size, as we show in our evaluation. In our experience, detection of larger salient parts is aided by larger average-pooling kernels (Figure 3).

For strongly supervised segmentation, we dispense with two-phase training and final pooling, and directly train the network end-to-end with a per-voxel cross-entropy loss over the segmentation maps for each output label.

### 4. Results

We evaluated our method on standard datasets that contain various semantic region labels. Here we present results both in the weakly-supervised segmentation setting, which is the principal focus of this paper, as well as in the strongly-supervised setting, where the same network achieves state-of-the-art performance on the standard ShapeNetCore benchmark.

#### 4.1. Weakly Supervised Region Labeling

In these validation experiments, we test whether our WU-Net architecture can successfully detect the salient parts that distinguish one category of shapes from another. We collated four different pairs of fine-grained shape classes, each pair distinguished by a prominent semantic component. These classes were: (a) chairs with and without armrests, (b) chairs with and without backs, (c) airplanes with and without engines/propellers mounted on the wings, and (d) cars with and without roofs. These classes were chosen because they are freely available in the ShapeNet repository [1], and have manually annotated ground-truth

![Figure 3: The effect of increasing the kernel size in average pooling. While in this example the largest kernel works best, for categories where finer parts are to be detected this is not the case.](image-url)
labeled segmentations into semantic parts, allowing us to automatically generate weak shape-level labels for training and directly validate the results. Chairs without backs (stools) were missing in ShapeNet, so we obtained these meshes from ModelNet. Each class was further randomly divided into train and test sets. While segmentation and labeling accuracy on the training set is as important as the accuracy on a test set in a weakly supervised setting, the test set allows us to do a direct comparison with a strongly supervised segmentation baseline. Our dataset statistics are shown in Table 1. The meshes were voxelized using Binvox [10].

**Segmentation performance.** In Figure 4 we report the per-voxel labeling accuracy of WU-Net with exactly the same hyperparameters (including $2^3$ average pooling) and automatic training protocol in each of the 4 weakly-supervised segmentation and labeling ex-
Figure 7: The statistical effect of increasing the kernel size for average pooling at the end of the network.

experiments, on the training set. (Note: validation on the training set is a meaningful experiment in weakly-supervised settings, where the guiding assumption is that weak shape labels are easily gathered but fine per-point annotations are not.) For comparison we use the following alternative methods:

- The saliency map of the trained WU-Net network, computed as the gradient of the output w.r.t. the input. The magnitude of the component for each input voxel can be interpreted as the degree to which it influences the final shape classification. This is a theoretically feasible alternative to the final segmentation map layer.

- An ablated version of WU-Net without skip connections. This represents a conventional fully convolutional architecture.

- An ablated version of WU-Net, without the final U structure, dubbed W-Net. We do not present results with just a single shallow U because this contains too few layers to learn an interesting signal.

- A 3D analogue of the original U-Net architecture [15], with a single deep U structure that repeatedly halves the grid resolution until a 4^3 bottleneck layer, and then repeatedly doubles it back to 64^3 again, with layers at every resolution linked by skip connections.

In addition, we present results for WU-Net both with and without symmetrization.

It can be observed that WU-Net, with or without symmetrization, substantially improves upon the performance of these representative alternative methods. Training of the ablated networks did not converge in some of the experiments, leading to very poor results. In the cases where they did converge, W-Net performs reasonably well though not as well as WU-Net. The version without skip connections shows much worse performance. This reinforces the critical role played by the skip connections in the performance of these down-(and up-)sampling architectures. The deep 3D U-Net training converges, but it identifies incorrect parts (car bonnets instead of roofs, chair seats instead of backs, etc), leading to poor scores.

We also present visual examples of the symmetrized WU-Net output, for a threshold of 0.9, in the teaser and in Figure 12. In addition we also show some visual results on swivel chairs, for which ground truth segmentations were not available: the roller wheels were identified as salient in these shapes. Visualizations of all shapes in our datasets are provided in the supplementary material.

**Comparison to a strongly supervised baseline.** For further insight into the performance of weakly-supervised WU-Net, we train it with strong supervision, with a single segmentation branch which we can threshold to produce a precision-recall plot. (Note: we cannot directly use the more standard strongly supervised version of WU-Net, which we discuss in the next section, because it compares pairs of corresponding voxels in multiple branches directly, and hence has no tunable threshold.) The strongly supervised network is not, in this case, a classifier. When it makes a classification error, it ends up trying to identify the semantic part in a shape which does not have it. This can lead to very poor performance, substantially below the weakly supervised case on the test set in two categories (Figure 5). However, if we use the trained weak

| Parent category | Fine-grained category | Has part | Lacks part |
|-----------------|-----------------------|---------|-----------|
| Chair           | Armrest               | 481     | 1359      |
| Chair           | Back                  | 150     | 75        |
| Airplane        | Engine                | 1034    | 266       |
| Car             | Roof                  | 806     | 106       |

Table 1: Weakly supervised segmentation dataset.
network simply as a binary shape classification oracle (where it typically achieves 99% accuracy in our experiments) for the strongly supervised network, then the latter performs as expected and establishes a high baseline. This indicates the extremely valuable role the shape tags play in identifying semantic parts. In this case, they appear to be significantly more important than fine-grained training annotations!

Comparison to Shilane and Funkhouser [16]. There is very little prior work on weakly supervised 3D shape segmentation. The most relevant research is by Shilane and Funkhouser, who studied the problem of identifying distinctive regions of shapes in different categories. While the problem domain is slightly different from ours (fine-grained intra-category differences), their method can be evaluated directly in our training setup. (Note that Shilane-Funkhouser cannot be directly applied in our test setup, where the shape tag is unknown.) We show the results of the comparison in Figure 6. The Shilane-Funkhouser results were not symmetrized, since symmetrization actually slightly worsened the results because of false positives. In three out of four cases, our method significantly outperforms theirs. In one case (car roofs), our method is somewhat worse, though the parameter setting for which Shilane-Funkhouser shows the best results here turns out to be suboptimal, often dramatically so, in the other cases.

The role of the average-pooling layer. The kernel size of the average-pooling layer following the segmentation map serves as a tunable hyperparameter network that directly affects the identified regions in a visually interpretable way. For large semantic parts, a larger final kernel size often yields better results. The effect is one of degree, as can be seen in Figure 7 and depends on the data. However, we found that a fixed $2 \times 2 \times 2$ kernel achieves good performance in all cases, and this is the setting we present for our fully automatic method and for all evaluations.

4.2. Strongly Supervised Region Labeling

The WU-Net architecture has the great advantage of being directly deployable in a strongly supervised setting, where per-point labels are available. We therefore test it on a standard benchmark: ShapeNetCore [1]. This dataset has manually annotated ground-truth segmentations for thousands of shapes in 16 categories (Figure 8). We compare our method to the recent state-of-the-art work of Kalogerakis et al. [5], using exactly the same train/test splits. Our performance is reported in Table 2. Our method improves upon the prior state-of-the-art in 10 out of 16 categories, often by significant amounts, and is competitive in all categories. While this is an extremely fast-moving area, we believe WU-Net establishes the current benchmark accuracy for this dataset at the time of writing.

The high performance in the fully-supervised setting, in situations where prior voxel-based methods had achieved significantly poorer results (Kalogerakis et al. rely on projective transforms that map the problem to the 2D domain) indicates that the WU-Net architecture has core advantages that make it suitable for segmentation tasks.

5. Applications

In this section, we demonstrate the wide utility of our method by presenting a few potential applications. We believe that our approach can benefit anyone who needs to quickly identify and discover semantic parts that characterize large classes of shapes, without having the resources to gather fine-grained part annotations. Hence, this list is hardly exhaustive.

Part-sensitive shape search. 3D shape search has been extensively studied [20]. Our approach enables part-sensitive search, where shapes with a particular part similar to a given shape are sought, to be implemented cheaply and quickly, since crowdsourced shape tags can be easily obtained (cf. the abundance of image tags in Flickr, but the relative paucity of pixel-level image segmentations). Further since WU-Net has very
| Category   | #train/#test | #labels | Shape-Boost | Shape-PFCN | WU-Net |
|------------|--------------|---------|-------------|------------|--------|
| Airplane   | 250/250      | 4       | 85.8        | 90.3       | 90.13  |
| Bag        | 38/38        | 2       | 93.1        | 94.6       | 96.02  |
| Bike       | 101/101      | 6       | 77.2        | 87.0       | 84.77  |
| Cap        | 27/28        | 2       | 85.9        | 94.5       | 89.82  |
| Car        | 250/250      | 4       | 79.5        | 86.7       | 89.44  |
| Chair      | 250/250      | 4       | 70.1        | 82.9       | 91.82  |
| Earphone   | 34/35        | 3       | 81.4        | 84.9       | 78.53  |
| Guitar     | 250/250      | 3       | 89.0        | 91.8       | 95.98  |
| Knife      | 196/196      | 2       | 81.2        | 82.8       | 90.96  |
| Lamp       | 250/250      | 4       | 71.7        | 78.0       | 77.37  |
| Laptop     | 222/223      | 2       | 86.1        | 95.3       | 96.61  |
| Mug        | 92/92        | 3       | 98.1        | 96.0       | 99.05  |
| Pistol     | 137/138      | 3       | 88.2        | 91.5       | 95.75  |
| Rocket     | 33/33        | 3       | 79.2        | 81.6       | 79.94  |
| Skateboard | 76/76        | 3       | 91.0        | 91.9       | 94.66  |
| Table      | 250/250      | 3       | 74.5        | 84.8       | 92.91  |
| Category average |         |         | 83.0        | 88.4       | 90.24  |
| Dataset average |        |         | 81.2        | 87.5       | 90.81  |

Table 2: Dataset statistics and strongly-supervised segmentation and labeling accuracy per category for test shapes in ShapeNetCore, versus ShapePFCN [5] and ShapeBoost [6].

high pure classification performance, it can directly infer tags on unseen shapes as well, allowing the dataset to be rapidly expanded and segmented.

We show a prototype interface in Figure 9. A query shape and a tag are provided. Shapes in the dataset preprocessed by WU-Net for this tag are retrieved based on similarity in the salient region (our simple implementation uses weighted average distance between the salient voxels, after the centroids of the salient regions are aligned).

**Fine-grained exploration of a shape dataset.**

The part-sensitive similarity metric computed for the search application can also be used to organize the complete dataset. In Figure 10 we show a t-SNE embedding of our chair dataset based on similarity of the detected “armrest” region. Note that the embedding places shapes with broadly similar arms together even if the rest of shape is quite different. A person can quickly get a visual overview of the variation of armrests in the dataset from this embedding, and perhaps discover new types of armrests.

In the same embedding, we highlight representative shapes which have very different armrests. This type of information, which is very valuable in gauging the variance and range of the dataset, is a direct byproduct of our inferred part annotations.

**Thumbnail creation.** When large numbers of shapes need to be presented for rapid browsing, it is helpful to have access to thumbnail representations which nevertheless capture the salient aspects of the shapes [16]. The WU-Net output directly enables such an application where the generated icons capture semantic parts. In Figure 11 we show examples of different thumbnails generated for the same shape that highlight different semantic aspects of the shape.

6. **Discussion, limitations, and future work**

We presented a method to obtain fine-grained semantic part annotations of 3D shapes from only weak shape-level tags. Given two sets of shapes, one known to have a part or parts present and one without, our method discovers where these parts are located in each shape of the former set. It achieves this through a deep neural network that is trained simply to classify the shape as possessing or lacking the part. The novel
Figure 10: A t-SNE embedding of chairs organized by similarity of the “armrest” regions detected by WU-Net. Below, we show several zoomed-in regions of the image. The larger icons on top represent diverse representatives of the collection that can be obtained from this similarity metric.

Figure 11: Different thumbnails of the same shapes (first column) created to highlight detected “armrest” (second column) and “back” (third column) regions.

structure of this network, which forms our core technical contribution, encourages finding large consistent regions across shape that characterize the differentiating part. The approach does not use inherent mesh topology or other representational or annotational information about the shape in any way. We also presented state-of-the-art results on strongly supervised shape segmentation using the same network.

Weakly-supervised segmentation is an insufficiently studied problem in 3D shape analysis. Our approach has several limitations. We do not currently support multiple tags per shape, or shapes where the characteristic parts are extremely heterogenous. Parts in our experiments occur in largely similar configurations: it would be interesting to explore whether, for example, parts that support human affordances can be extracted robustly using such a framework. For instance, finding handles in a dataset containing bikes, hammers, bags and sports equipment, given only weak tags (“can be gripped” vs “cannot be gripped”) would be an extremely challenging problem because of the diversity of geometry involved.

It would also be interesting to explore the range of applications that can be enabled or accelerated by our approach. WU-Net performs twin roles of part annotation and part discovery. In the former role, the user is mainly interested in rapidly and cheaply labeling salient semantic regions of shapes in a large dataset. In the latter role, the user is interested in discovering which parts distinguish two semantic classes of shapes. These are both promising directions for further work and new applications.

References

[1] A. X. Chang, T. A. Funkhouser, L. J. Guibas, P. Hanrahan, Q.-X. Huang, Z. Li, S. Savarese, M. Savva, S. Song, H. Su, J. Xiao, L. Yi, and F. Yu. ShapeNet: An information-rich 3D model repository. arXiv preprint arXiv:1512.03012, 2015.

[2] R. G. Cinbis, J. J. Verbeek, and C. Schmid. Weakly supervised object localization with multi-fold multiple instance learning. CoRR, abs/1503.00949, 2015.

[3] A. Golovinskiy and T. Funkhouser. Consistent segmentation of 3D models. Computers and Graphics (Proc. SMI), 33(3), 2009.

[4] Q. Huang, V. Koltun, and L. Guibas. Joint shape segmentation with linear programming. Trans. Graphics, 30(6), 2011.

[5] E. Kalogerakis, M. Averkiou, S. Maji, and S. Chaudhuri. 3D shape segmentation with projective convolutional networks. In Proc. CVPR, 2017.
Figure 12: Examples of weakly supervised segmentation by WU-Net. Top: detecting roofs of cars. Middle: detecting wing-mounted engines and propellers of airplanes. Bottom: detecting backs (left) and swivel wheels of chairs (right).
[6] E. Kalogerakis, A. Hertzmann, and K. Singh. Learning 3D mesh segmentation and labeling. Trans. Graphics, 29(4), 2010.
[7] V. G. Kim, S. Chaudhuri, L. Guibas, and T. Funkhouser. Shape2Pose: Human-centric shape analysis. Trans. Graphics, 33(4), 2014.
[8] J. Long, E. Shelhamer, and T. Darrell. Fully convolutional networks for semantic segmentation. In Proc. CVPR, 2015.
[9] H. Maron, M. Galun, N. Aigerman, M. Trope, N. Dym, E. Yumer, V. G. Kim, and Y. Lipman. Convolutional neural networks on surfaces via seamless toric covers. Trans. Graphics, 36(4), 2017.
[10] P. Min. Binvox. http://www.patrickmin.com/binvox/ 2017.
[11] M. Oquab, L. Bottou, I. Laptev, and J. Sivic. Is object localization for free? – weakly-supervised learning with convolutional neural networks. In Proc. CVPR, 2015.
[12] D. Pathak, P. Krähenbühl, and T. Darrell. Constrained convolutional neural networks for weakly supervised segmentation. In Proc. ICCV, 2015.
[13] C. R. Qi, H. Su, K. Mo, and L. J. Guibas. PointNet: Deep learning on point sets for 3D classification and segmentation. arXiv preprint arXiv:1612.00593, 2016.
[14] C. R. Qi, H. Su, M. Nießner, A. Dai, M. Yan, and L. Guibas. Volumetric and multi-view CNNs for object classification on 3D data. In Proc. CVPR, 2016.
[15] O. Ronneberger, P. Fischer, and T. Brox. U-Net: Convolutional networks for biomedical image segmentation. In Proc. MICCAI, 2015.
[16] P. Shilane and T. Funkhouser. Distinctive regions of 3D surfaces. Trans. Graphics, 26(2), 2007.
[17] O. Sidi, O. van Kaick, Y. Kleiman, H. Zhang, and D. Cohen-Or. Unsupervised co-segmentation of a set of shapes via descriptor-space spectral clustering. Trans. Graphics, 30(6), 2011.
[18] K. Simonyan, A. Vedaldi, and A. Zisserman. Deep inside convolutional networks: Visualising image classification models and saliency maps. CoRR, abs/1312.6034, 2013.
[19] H. O. Song, R. B. Girshick, S. Jegelka, J. Mairal, Z. Harchaoui, and T. Darrell. One-bit object detection: On learning to localize objects with minimal supervision. CoRR, abs/1403.1024, 2014.