

Unsupervised Learning of Shape Concepts – From Real-World Objects to Mental Simulation

Christian A. Mueller and Andreas Birk *

Abstract

An unsupervised shape analysis is proposed to learn concepts reflecting shape commonalities. Our approach is two-fold: i) a spatial topology analysis of point cloud segment constellations within objects is used in which constellations are decomposed and described in a hierarchical and symbolic manner. ii) A topology analysis of the description space is used in which segment decompositions are exposed in. Inspired by Persistent Homology, groups of shape commonality are revealed. Experiments show that extracted persistent commonality groups can feature semantically meaningful shape concepts; the generalization of the proposed approach is evaluated by different real-world datasets. We extend this by not only learning shape concepts using real-world data, but by also using mental simulation of artificial abstract objects for training purposes. This extended approach is unsupervised in two respects: *label-agnostic* (no label information is used) and *instance-agnostic* (no instances preselected by human supervision are used for training). Experiments show that concepts generated with mental simulation, generalize and discriminate real object observations. Consequently, a robot may train and learn its own internal representation of concepts regarding shape appearance in a self-driven and machine-centric manner while omitting the tedious process of supervised dataset generation including the ambiguity in instance labeling and selection.

1 Introduction and Motivation

Studies of early object perception in infants [1] suggested that objects can be characterized by a set of properties such as continuity, i.e., objects successively move along a path, or solidity, i.e., objects can only move through free-space. Furthermore, shape is a key visual cue as it fundamentally contributes to reasoning and understanding of objects [2, 3]. Inferred shape commonalities among objects allow to infer similar object (including semantic) properties. Shape is used in many robotic application areas ranging from household to industry, e.g., in object

*The authors are with the Robotics Group of the Computer Science & Electrical Engineering Department, Jacobs University Bremen, Germany
e-mail: {chr.mueller, a.birk}@jacobs-university.de.

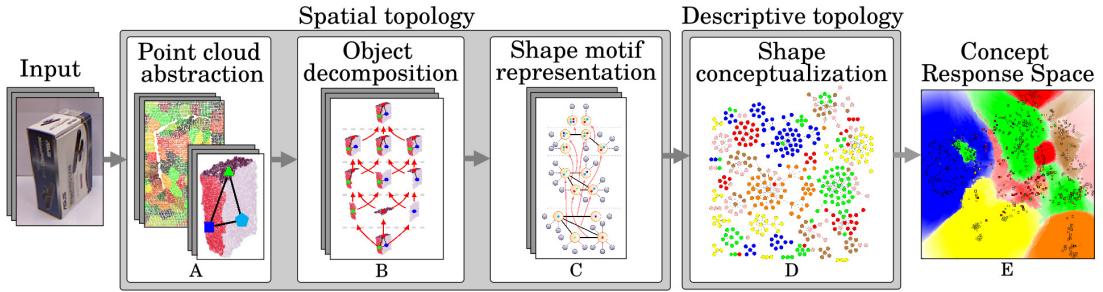


Figure 1: Illustration of the proposed object shape conceptualization approach.

shape categorization tasks [4], in generation of grasping primitives for similar object appearances in manipulation [5], or in finding substitutes for currently absent objects [6, 7], to name just a few examples.

In traditional object perception in form of object instance or category recognition, an association is formed between a label and a specific instance (e.g., *John’s mug*) or a generic group of instances (e.g., *mug*), which share commonalities in appearance [8]. A group of instances can be denoted as a *category* and the description and abstraction of group commonalities as a *concept*. Learning concepts from objects by associating meaning to a system’s percepts is often conducted through interaction [9] and supervision [10, 11]. Eventually, associations are generally human-made, individually and continuously evolved over lifetime experience [12] based on a set of modalities like tactial, auditory or visual sensations [13, 14]. The combination of those sensations allows us to reliably interpret perceived object information [15]. Humans are capable of incorporating further modalities including functional object knowledge to differentiate even though visual percepts can be similar, e.g., *mug*, *cup*, *vase* or *bowl*. Consequently, such natural concepts are often not inferable from a machine-perspective due to the lack of dimensionality representing the perceived observations (e.g., only images or point clouds). From a machine-vision perspective, human-supervised learning methods are particularly highly vulnerable to incorporate such knowledge, e.g., the function or affordances of objects, which is not inferable from pure sensor data. This is often inevitable when a supervised labeling process is conducted by humans, which will ultimately lead to biases in the learning phase.

Our work in contrast focuses on object understanding from a machine-perspective avoiding supervision. The work presented here builds upon our method from [16] that learns shape concepts in an unsupervised (label agnostic) and data-driven manner from point clouds irrespective of human-annotations, which may contain biases. Extracted segment constellations within object point clouds are used to learn patterns and eventually concepts of shape commonalities in a hierarchical manner. It is shown that concepts can be learned from real-world RGBD-snapshots of objects, or more precisely single view point clouds omitting the color information, using well-known datasets like the *Washington RGB-D Object Dataset* [17] (WD) or the *Object Segmentation Database* [18] (SD). From the machine perspective, the concepts learned are purely derived by the given data, i.e.,

they are not affected by variable biases that may be caused by individual human interpretations with respect to the instance label annotation. As can be seen, the concepts learned on one real-world dataset generalize well across other real-world datasets not seen before.

Nevertheless, biases can be in the selection of the dataset instances used for training and validation. Moreover, the dataset generation process is cumbersome and generally requires effort in preparation including object instances selection, defining the experimental setup or labeling of object ground truth with regard to the background. Therefore, we further investigate in this article the capability of learning the essence of object appearance from artificially generated object observations in simulation and whether the learned concepts are applicable to discriminate real-world object concepts. This capability allows an artificial system like a robot to train its own internal representation of concepts regarding shape appearance in a self-driven and machine-centric manner without human-bias. Thus, we present here an approach, which is unsupervised in two respects: it is *label-agnostic* (no label information is used) as well as *instance-agnostic* (no instances preselected by human supervision are used).

2 Approach and Related Work

Shape analysis relies on a robust description and representation [19, 20] of objects, particularly in real world scenarios where snapshots of objects are affected by sensor noise and occlusions [21]. Theories of object perception from Cognitive Science and Psychology suggest a hierarchical and component-based representation of object information [22]. Inspired by this, an analysis of topological patterns is applied here to sensor data in form of point clouds observed from single viewpoints with a Kinect-like camera. The analysis is two-fold (see Fig. 1): **i**) an analysis of the spatial topology in point cloud decompositions, **ii**) a topology analysis of these decompositions in description space.

Regarding **i**), point clouds are initially over-segmented [23] (Fig. 1 **A**) and further post-processed to segments that can reflect meaningful components of objects. Subsequently, a hierarchical decomposition of point clouds is generated in a bottom-up manner (Fig. 1 **B**). These segment compositions of objects allow to reason about shape characteristics and commonalities; commonalities observed within objects can be generalized to a shape concept.

Constellation models, which learn concepts from perceived feature (e.g., key-points or segments) constellations have been successfully used in recent years [4, 24, 25, 26, 27]. The inference is typically based on *local* analysis of feature coherences with a priori learned constellation models, i.e., local evidences in a constrained spatial range with respect to the features using, e.g., Markov Networks [28, 29]. This inference is robust to the absence of features due to noise and partial object occlusion. Shape facets, which become apparent on a *global* scale – especially in case of complex structures, are in contrast insufficiently reflected considering only local inferences. In the work presented here, a hierarchical

constellation model (Fig. 1 **C**) is proposed in which segment constellations are decomposed over multiple topological levels that gradually (from local to global) reflect shape facets: from individual segment occurrences over segment groups to a single group of segments, which represent an entire object. On each topological level, shape characteristics are observed and learned.

A related research field focuses on compositional hierarchies [30, 31, 32] in which general geometric building-blocks like edges or contours are hierarchically composed to unions of these building-blocks. Similarly, skeletonization methods [19, 33, 34] try to extract structure within objects from which regions and object components can be decomposed for reasoning purposes. Our work differs in several aspects; especially as here a) the building-blocks are represented as symbols which characterize underlying 3D point cloud segments, and b) their constellations are subsequently learned in a multi-hierarchical manner.

Regarding **ii**), i.e., the topology analysis of the decompositions in description space: observed decompositions over the topological levels are here analyzed to gather distinctive insights and patterns that can be interpreted and related to concepts of specific shape appearances. Persistent Homology (PH) is a concept related to Topological Data Analysis that has been applied in various areas related to high dimensional data visualization or to finding relations and coherencies in Big Data scenarios in general [35]. PH allows to extrapolate features from data by means of finding persistent (or stable) feature appearances through an iterative filtration of the data compared to standard clustering approaches. Standard cluster algorithm (e.g., k-Means, Expectation-Maximization, tree-based algorithms, etc.) associate data points to groups of data, which share similar properties, which is measured by a metric or similarity function. Inherent parameters of clustering algorithms are related to the number, size, variance of clusters, neighborhood distance between data points or in case of tree-like clustering, a splitting criterion. The parameterization is often computationally costly and it depends on the concrete data on which the clustering process is applied to. Furthermore, partitioning the topology of a continuous description space with a static parameterization is often not a good solution due to over- and under-fitting effects. Soft-clustering approaches like probability-based Expectation-Maximization (EM) provide a feedback of the actual fit of a query to the set of previously extracted clusters; but such approaches require additional post-processing in order to make a final decision about cluster membership.

PH in contrast allows to investigate the topological evolution of the data in a step-wise manner. The concept of PH has already shown its applicability in geometric shape analysis to detect persistent shape patterns when being directly applied on point cloud data [35, 36, 37]. But instead of directly applying PH on point cloud data, we use here the responses that are retrieved from our topological analysis of point cloud decompositions proposed in **i**). The PH-based analysis allows to detect persistent appearances of the responses during the filtration process, which reveal shape commonalities of instances that can form concepts (Fig. 1 **D**).

Consequently, we focus on shape reasoning with a symbolic representation of

geometric information, which is further exploited to learn visual patterns from observed object point cloud compositions on multiple granularity levels which allow to learn concepts from. Our final goal is to investigate whether the visual patterns can be learned in a data-driven manner by encoding real object observations or on the basis of abstract artificial data from simulation. The question arises whether artificial data from simulation allows to encode visual patterns that lead to concepts which can then be applied to discriminate real object observations (Fig. 1 E).

3 Spatial Topology Analysis

3.1 Object Segment Extraction

Building on our work from [38] as basis, an object point cloud is initially over-segmented into atomic patches and further processed to segments, also known as super patches, which can represent semantically meaningful shape components like planar surfaces of a *box* or cylindrical and planar surfaces of a *can* (see Fig. 2(a)). Subsequently, objects are represented as a set of *point cloud segments*. These segments can be interpreted as *building blocks* that constitute objects.

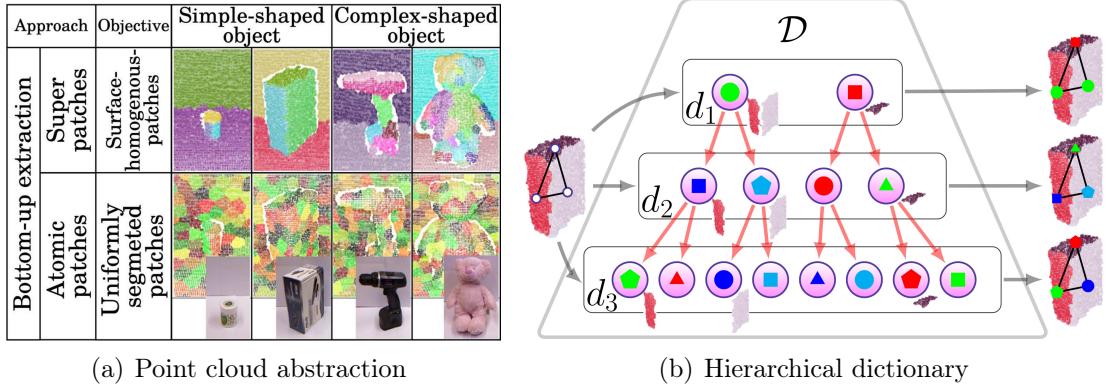


Figure 2: (a) A two-step segmentation [38] from atomic patch segments to super patches is used as basis – here illustrated by sample snapshots of 4 example objects (*can*, *box*, *cordless drill*, *teddy*). (b) An example hierarchical dictionary [4] $\mathcal{D}=\{d_1, d_2, d_3\}$ is shown that consists of 3 description levels using divisive clustering. For illustration, each visual word is depicted as a circle with a colored polygon. A segmented object is shown as a graph on the left of the dictionary; on its right, the visual words assigned for each segment according to the respective description level are shown.

Tackling with real world data, object observations are imperfect, e.g., noisy and partially occluded, which leads to a degradation of the detection of these building blocks. This leads to failures in associating observed data to known building blocks, which is also in general known as the correspondence problem. Therefore the stability of the detection of such building blocks in real world data

is a major challenge. To mitigate the correspondence problem among imperfect segments, a symbolic representation of segments is chosen as an abstraction step to facilitate further shape reasoning. Segment appearances are quantized to a set of discrete visual words following the well-known bag-of-words methodology [4], i.e., the visual words constitute a dictionary. The idea is that similar appearing segments are abstracted to the same symbol, respectively, visual word. The level of quantization plays a crucial role, since too few words may lead to under-fitting, whereas too many words may lead to over-fitting symptoms. For an unbiased and purely data-driven word generation, a hierarchical divisive clustering procedure is applied as introduced in our previous work [4]. Therein segments are initially described with a description vector that is generated by a point cloud descriptor like FPFH [39]. As a result of the clustering procedure, a hierarchical dictionary \mathcal{D} is created that consists of multiple description levels $\{d_1, d_2, \dots\}$, where level f consists of 2^f words (Fig. 2(b)). Each word represents a description vector whose position is inferred by the clustering procedure during the training phase using a set of segments captured from random scenes. Given an object segment, the extracted description vector of the segment is passed through the hierarchical dictionary \mathcal{D} . For each description level, the propagated description vector is accordingly labeled with the visual word whose description vector is closest using the l^2 -norm (Fig. 2(b)).

3.2 Hierarchical Object Decomposition and Representation

A segment composition of a captured object o is initially represented as graph g^o in which each segment corresponds to a vertex and neighboring vertices are connected with an edge. Each vertex is augmented with the corresponding point cloud segment and the visual word that is inferred from the set of visual words on the respective description level in the dictionary \mathcal{D} (see Sec. 3.1); the visual word inferences can hence differ according to the description level as illustrated on the right side in Fig. 2(b).

The spatial topology of segments is analyzed in an unsupervised manner and encoded in a hierarchical representation, which we denote as *Shape Motif Hierarchy*; an illustration of a hierarchy \mathcal{H} is shown in Fig. 3(a). \mathcal{H} is based on a graphical representation of visual word constellations which are denoted as *motifs*; note that these constellations can only contain visual words of a specific description level. Therefore for a dictionary $\mathcal{D}=\{d_1, d_2, \dots, d_n\}$ which contains n description levels, n hierarchies are created that constitute an ensemble $\mathcal{HE}=\{\mathcal{H}_1, \mathcal{H}_2, \dots, \mathcal{H}_n\}$, see Fig. 3(c).

In the training phase for each hierarchy \mathcal{H} , object observations are encoded in a bottom-up manner, beginning with single object segments over groups of segments until a single constellation of segments represents the entire object. A sample propagation () of a box (consisting of three segments) through the hierarchy \mathcal{H} is shown in Fig 3(a). Object segments are propagated through the hierarchy using the corresponding visual words associated to the segments. Within the propagation process, newly observed visual word constellations (*word*

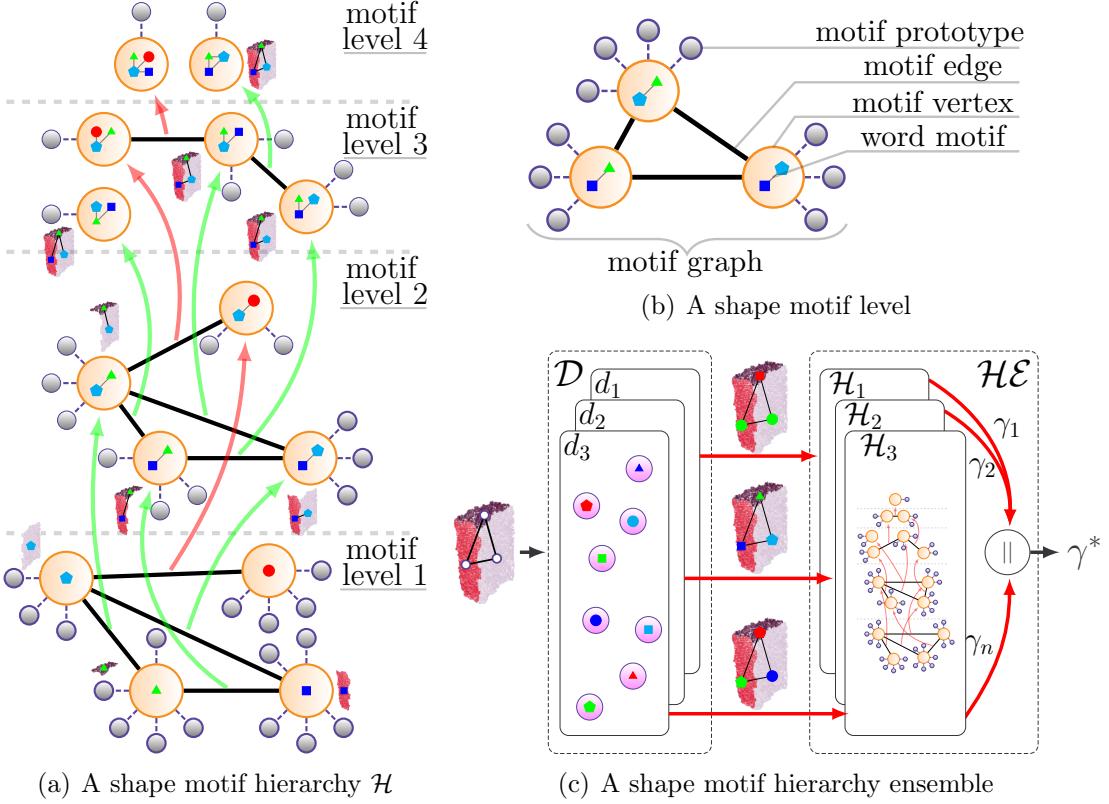


Figure 3: An example of a shape motif hierarchy \mathcal{H} is shown in (a); it consists of multiple *motif levels*. Each node \bigcirc represents a specific *motif vertex*, whereas each smaller linked node \circ represents a *motif prototype*. A sample propagation (\rightarrow) of a box \blacksquare (consisting of three segments) through \mathcal{H} is shown in (a). Feasible propagations, which have been previously encoded in the hierarchy during the training phase but which are not affected by the *box*, are depicted as \longrightarrow . Components of a *motif level* are illustrated in (b). In (c) the combined approach is illustrated: an example shape motif hierarchy ensemble $\mathcal{H}\mathcal{E}$ based on three shape motif hierarchies $\{\mathcal{H}_1, \mathcal{H}_2, \mathcal{H}_3\}$ using respective description levels $\{d_1, d_2, d_3\}$ of \mathcal{D} (see Fig. 2(b)).

motifs) are integrated into the hierarchy as *motif vertices* (see Fig. 3(b)). Each motif in the hierarchy is unique with respect to visual words, i.e., a newly observed word motif of an object leads to a creation of a *motif vertex* if the motif does not exist in the hierarchy. For further characterization of a motif vertex, a point cloud description is extracted of a propagated segment constellation and added as *motif prototype* \circ to the motif vertex \bigcirc that corresponds to the motif of the propagated constellation (Fig. 3(b)). As a result, each motif vertex represents a *shape motif* that can be exploited as *building block* and that can constitute – even unknown – objects. Further at motif level $l=1$, an edge (---) between two motif vertices is created if the corresponding object segments are neighbors. For $l>1$ an edge is created if two motif vertices contain a visual word that corresponds to the same segment of the propagated object. In each propagation step from level

l to $l+1$, the union of word motifs connected to an edge in level l forms a vertex in $l+1$ (→). Consequently, upper levels can consist of fewer edges or vertices, i.e., a single motif vertex can encompass a word constellation that represents an entire object; see, e.g., the *box* sample at motif level 4 in Fig. 3(a). In this manner, objects are decomposed in various motifs by the propagation through the hierarchy \mathcal{H} .

As a result, the Shape Motif Hierarchy Ensemble $\mathcal{HE}=\{\mathcal{H}_1, \mathcal{H}_2, \dots, \mathcal{H}_n\}$ does not only take the structural appearance with respect to the variety of the segment constellations into account but also the symbolic appearance of constellations by using a specific dictionary description level for the respective hierarchy. For illustration purposes, the propagation process of a segmented *teddy bear* is shown in Fig. 4 from a set of primitive motif vertices to more complex motifs vertices in which eventually a single motif represents the *teddy bear*.

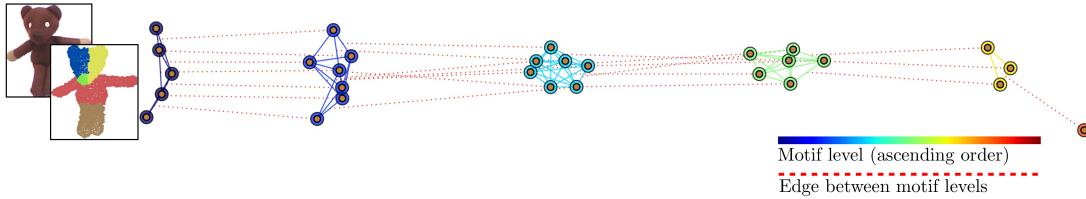


Figure 4: Illustration of a segmented teddy sample propagated through \mathcal{H}_4 (see Fig. 3(a)) showing *activated* motif vertices in each motif level (see Fig. 3(b)), i.e. level 1, 2, 3, 4, 5 and 6 (the teddy segments in the point cloud are randomly colored).

3.3 Stimuli Generation

In the training phase, object segment constellations represented by corresponding visual words are propagated through the hierarchy and are memorized as *motif prototypes* within motif vertices that match visual word constellations of the object. Inspired by the *Prototype Theory* [40], each motif vertex is formed by these prototypes, which are used to generate stimuli for unknown objects as described in the following: given a graph of segments g^o of object o , the segments are annotated with the corresponding words and subsequently propagated through the hierarchy as in the training phase, see the *box* example in Fig. 3(a) – note that the hierarchy is not modified during the stimuli generation. Through the propagation of segments, motif vertices are activated that correspond to the words of the propagated segments. An activation of a vertex v is represented by the Indicator function $\mathbb{1}_v(g^o)$, which returns 1 in case of a match, otherwise 0 if no match is found. For an activated v , a stimulus $\alpha(v, g^o)$ is computed based on point cloud descriptions of the memorized motif prototypes T^v of v and the respective description q of object segments in g^o , which activated v . By applying Probabilistic Neural Networks [41], the stimulus is computed with an adapted Gaussian kernel (bandwidth $\sigma=0.025$) in which Jenson-Shannon divergence (*JSD*) [42] is used as

distance measure, see Eq. 1.

$$\alpha(v, g^o) = \begin{cases} \frac{1}{|T^v|} \cdot \sum_{i=1}^{|T^v|} e^{\frac{\text{JSD}(t_i \in T^v, q)^2}{-2\sigma^2}}, & \text{if } \mathbb{1}_v(g^o)=1 \\ 0, & \text{otherwise} \end{cases} \quad (1)$$

As a result for each propagated object, stimuli of motif vertices in \mathcal{H}_i are accumulated and projected into vector form $\gamma_i^o = [\alpha(v_1, g^o), \alpha(v_2, g^o), \dots]$. Given n description levels and correspondingly trained n shape motif hierarchies that form the ensemble $\mathcal{HE} = \{\mathcal{H}_1, \mathcal{H}_2, \dots, \mathcal{H}_n\}$, the object graph g^o is propagated through each motif hierarchy. Subsequently, a final stimuli vector ${}^*\gamma^o = [\gamma_1^o, \gamma_2^o, \dots, \gamma_n^o]$ is composed (\parallel) of stimuli retrieved from n motif hierarchies, see Fig. 3(c).

4 Descriptive Topology Analysis

Commonalities among shape appearances can vary from *specific* to *generic* shape facets: a concept generation process is hence used, which in a gradual manner detects commonalities ranging from individual to common facets, i.e., very specific to often re-occurring facets. Persistent Homology (PH) provide the computational model that allows to gradually reveal topologically persistent patterns in generated stimuli ${}^*\gamma$, which are interpreted as commonalities and eventually as shape concepts.

4.1 Persistence Homology and Filtration

We briefly introduce terms from algebraic topology which are related to our shape concept learning approach. Comprehensive literature can be found in [35, 37, 43, 44, 45].

4.1.1 Simplices and Complexes

Given a continuous topological space $\mathcal{X} = \{x_0, x_1, \dots, x_m | x_i \in \mathcal{R}^n, 0 \leq i \leq m\}$ with m n -dimensional data points. A *simplex* π is a d -dimensional polytope, which is a graph consisting of a convex hull of $d+1$ affine independent vertices where each vertex is a point in \mathcal{X} . A composition of *simplices* is denoted as *simplicial complex* $K = \{\pi_0, \pi_1, \pi_2, \dots\}$. This composition is a union of vertices, edges, triangles or other higher dimensional polytopes.

4.1.2 Vietoris-Rips Complex

We focus on *vietoris-rips complexes* in which a complex K_i^{vr} is extracted from a subspace $\mathcal{X}_i \subseteq \mathcal{X}$ with a given scale parameter $\epsilon > 0$. K_i^{vr} consists of vertices that are only connected if the distances between the vertices is lower than the given parameter ϵ . The vietoris-rips complex K_i^{vr} can also be denoted as ϵ -complex, where ϵ is also denoted as radius or distance threshold.

4.1.3 Homology Groups

Homology is a concept in algebraic topology, which allows to reveal specific characteristics or features in \mathcal{X} . Characteristics are organized therein into homology groups $\mathcal{HG}=\{H_0(\mathcal{X}), H_1(\mathcal{X}), H_2(\mathcal{X}), \dots\}$. Often, the first three homology groups are analyzed: in the context of geometry $H_0(\mathcal{X})$ is related to *connected components* or *clusters* of vertices. $H_1(\mathcal{X})$ is related to the complexes in form of *loops* or *holes* and $H_2(\mathcal{X})$ is related to *voids* which represent fully connected complexes. Here, we focus on $H_0(\mathcal{X})$ since it complies with our goal to extract topological groups from stimuli vectors (see Sec. 3.3), which can represent concepts.

4.1.4 Topological Space Filtration and Persistent Homology

The filtration of the topological space \mathcal{X} is initiated by a subsequently nested application of a set of radii $\mathcal{E}=\{\epsilon_0, \epsilon_1, \dots, \epsilon_j\}$ where $\epsilon_{i-1} < \epsilon_i < \epsilon_{i+1}$. For $H_0(\mathcal{X})$, each point $x_i \in \mathcal{X}$ is represented at the beginning of the filtration process by a 0-simplex $\pi_i \in$ *vietoris-rips complexes* K_0^{vr} . These simplices are so to say *born* at radius 0. Note that the K_i^{vr} is extracted using radius ϵ_i . While the filtration progresses, the vietoris-rips complex grows since the radius increases, which can cause fusions of simplices that form a larger simplex: a *union* is performed between simplices while one simplex enlarges and sustains by annexing the other that *dies*. Eventually, a complex K^{vr} is filtered that contains a single high dimensional simplex – see Eq. 2.

$$\emptyset \subseteq K_0^{vr} \subseteq K_1^{vr} \subseteq \dots \subseteq K_j^{vr} = K^{vr} \quad (2)$$

Persistent Homology provides a way to analyze and track birth and death of simplices (also known as *homology classes*) along the filtration process: $H_0(K_i^{vr}) \rightarrow H_0(K_{i+1}^{vr})$. The according results can be represented in *persistence* or *barcode diagrams* (Fig. 9(a)). While considering the gradual evolution of vietoris-complex K_i^{vr} , the extraction of homology classes (birth and death) is inherently robust to deformation due to the topological organization of the data in a graphical manner.

4.2 Shape Concept Extraction

In the following, the shape concept extraction process is described – from topological space and concept generation to concept inference.

4.2.1 Topological Space Generation

Given a set of raw stimuli vector responses (Sec. 3.3), the responses are initially used to create a topological space in a graphical manner. Therein, a stimuli vector ${}^*\gamma$ can be interpreted as an independent point in the space, in which a distance metric can be used to measure the similarity to other stimuli vectors; these vectors serve as anchor points in a space of an unknown topology. The goal is to interrelate these vectors in order to discover topological relationships among the anchor points. We make use of a graphical representation, in which each anchor point represents a vertex. Initially a complete graph is created,

where each edge between vertices is augmented with the corresponding distance; distances are measured by the Jenson Shannon divergence (JSD).

To minimize the search space and to initiate the construction of the topological space \mathcal{X} , the Minimum Spanning Tree [46] is extracted using the respective JSD distances. Subsequently, a substantial amount of edges perishes and a minimum number of edges remain, which reveal the structural and topological organization of the stimuli vectors. Fig. 5 shows an example based on the object instances from the *Object Shape Category Dataset* (Sec. 6) that consists of seven shape categories (*sack*, *can*, *box*, *teddy*, *ball*, *amphora*, *plate*).

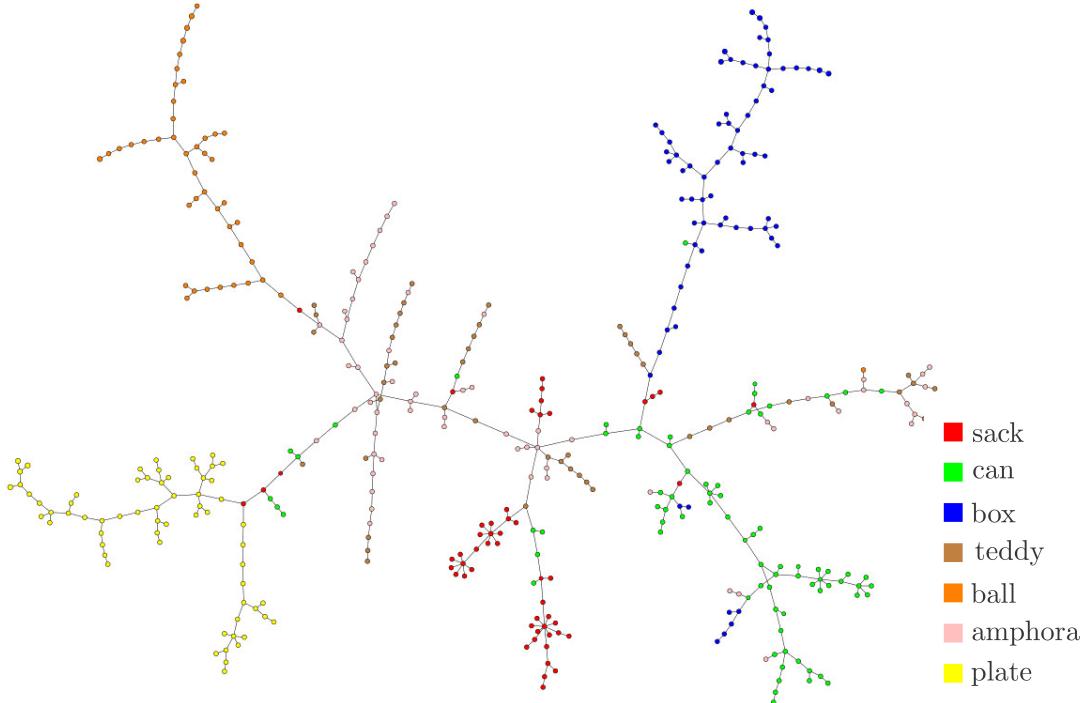


Figure 5: The minimum spanning tree, which spans the topological space \mathcal{X} of stimuli vectors extracted from instances of the *Object Shape Category Dataset* (OSCD), see Sec. 6. Note that each vertex represents a sample object of the dataset. Vertices are colored only for illustration purposes by their corresponding category label of the dataset, which is not used in our unsupervised learning phase.

From this point on, we focus on the *topological similarity* among stimuli in form of the *geodesic* distance within \mathcal{X} . Therefore each edge is uniformly weighted by assigning a distance of 1. Due to the inherent sparsity of edges in \mathcal{X} , Johnsons all-pair-shortest path algorithm allows to efficiently generate a distance map which is used to infer a heat for each vertex $x \in \mathcal{X}$. A vertex heat $h^\bullet(x)$ is inferred by the mean geodesic distances $d_{geo}(\cdot)$ to all other vertices in \mathcal{X} whereas the edge heat $h^{\bullet\bullet}(e_{j,k})$ is determined by the mean heat of the connected vertices x_j and x_k as shown in Eq. 3.

$$h^\bullet(x) = \frac{\sum_{i=0}^{|X|} d_{\text{geo}}(x, x_i \in X)}{|X|}, \quad h^{\bullet\bullet}(e_{j,k}) = \frac{h^\bullet(x_j) + h^\bullet(x_k)}{2} \quad (3)$$

Henceforth, we use edge heats as edge distances between respective vertices. By scaling the heat in X to the interval $[0, 1]$ and inverting the heat, vertices located at leaf regions of X come closer to each other whereas vertices in the inner region move farther away from each other. Furthermore, two observations can be made: a) the heat of exteriorly located edges is lower than the interiorly located ones; b) vertices which are interiorly located reflect more heterogeneity with respect to their neighbors, compared to vertices which are exteriorly located in X .

4.2.2 Topological Filtration

Given the topological space X , the filtration is applied over a range of radii $\mathcal{E} = \{\epsilon_0, \epsilon_1, \dots, \epsilon_j\}$. The step size $\epsilon_i \rightarrow \epsilon_{i+1}$ is determined by the minimum edge distance in X that also initializes the filtration at ϵ_0 . The filtration is completed when the maximum edge distance in X is reached at ϵ_j . In practice, the number of steps $|\mathcal{E}|$ can reach a computationally intractable number. An upper bound limit for $|\mathcal{E}|$ can be applied by increasing the step size until the upper bound is met. Consequently, the filtration is initialized with 0-simplices where each simplex represents a stimuli vector, i.e., a vertex of the topological space X . This filtration is performed on X as described in Sec. 4.1.4; note that the equidistant filtration steps from ϵ_0 to ϵ_j are often denoted as time.

Persistent Homology allows to track the birth and death of simplices in K^{vr} of X during the filtration. Due to the nature of evolving simplices complexes (see Eq. 2) in each time step, the complex changes its appearance after annexations of simplices complexes of previous time steps. These changes during the filtration are encoded in graph \mathcal{F} , which is shown in Fig. 6(a).

An edge represents an annexation during the filtration process of a simplices complex to another complex – beginning with 0-simplices representing leaves in \mathcal{F} . Each edge is augmented with the annexation time. So, outer simplices lived shorter since they have been annexed earlier in time compared to inner ones. As a result, \mathcal{F} represents the filtration progression of X .

4.2.3 Extraction of Persistent Shape Concepts

The lifetime of simplices can be interpreted as a feature indicator in X , i.e., *persistent* or long living simplices tend to represent a significant feature, i.e., a shape property that is prominent for an object or even object category. At the same time, short living simplices can be interpreted as being insignificant. The goal is hence to detect persistent simplices. In order to ease the persistence analysis, the filtration time range is scaled within the interval $[0, 1]$, i.e., from 0 (start of filtration = ϵ_0) to 1 (end of filtration = ϵ_j). In the filtration process, trivial homology classes are obtained at time 0 where 0-simplices exist and at time 1 where a single simplex consists of all simplices in X . We are interested of finding persistent groups between these extrema. A *group* is a connected component

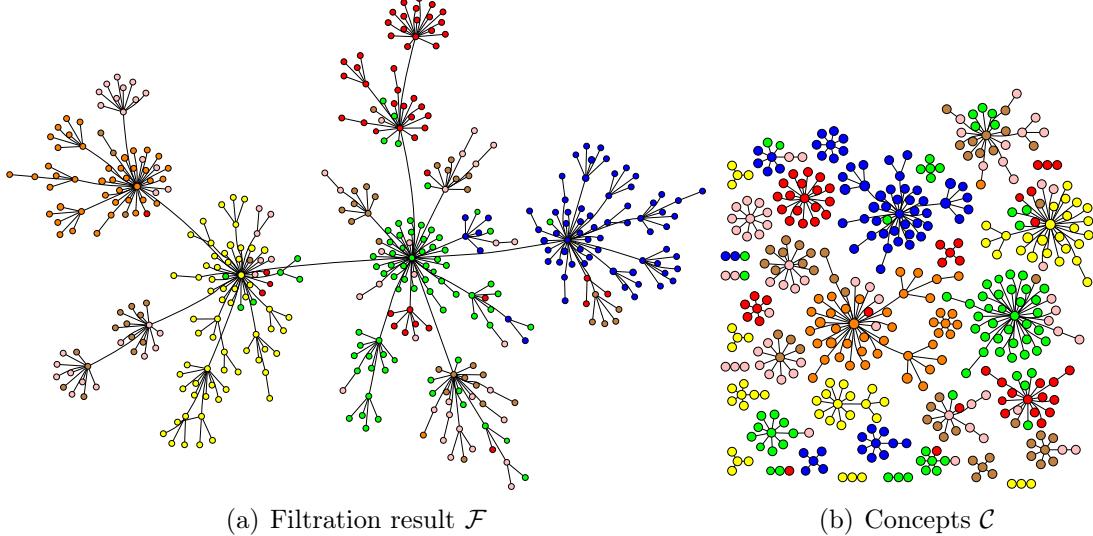


Figure 6: (a) A filtration graph \mathcal{F} showing annexations over time according to the given graph \mathcal{X} (see Fig. 5). For illustrations purposes, each vertex is colored with the corresponding label as shown in Fig. 5. (b) Connected components extracted from \mathcal{F} (a) that represent concepts \mathcal{C} ($|\mathcal{C}|=36$). For illustration purposes, each vertex (concept prototype) is colored with the corresponding label as in Fig. 5.

of vertices, i.e., a d -simplex ($d>0$). Due to the gradual filtration, each group consists of topologically similar vertices. Therefore, the groups can constitute shape concepts, where each vertex within a group is a representative *concept prototype*.

Given the entire time spectrum $[0, 1]$, Persistent Homology allows to access any state of detected concepts \mathcal{C} in \mathcal{X} at an arbitrary time in the spectrum; note that the filtration starts with $|\mathcal{C}|=|\mathcal{X}|$ and ends with $|\mathcal{C}|=1$. Consequently, a distinctive time can be determined. An optimal time varies according to the topology that is reflected by the given stimuli vector. Consider an optimal time when the global maximum of annexations (see Sec. 6.1) is reached, and subsequently edges in \mathcal{F} that are augmented with an older time than the optimal time are removed. This optimal time leads to a set of connected components in \mathcal{F} that can reflect useful shape concepts as illustrated in Fig. 6(b). Note that edges which are created at later time connect more heterogeneous groups and subsequently represent more, and possibly too generic concepts, in contrast to more specific concepts which emerge when edges are created at earlier time.

4.3 Shape Concept Inference

Given a stimuli vector ${}^*\gamma^o$ that is extracted from an unknown object o , a response is retrieved based on similarity to previously learned shape concepts (see Fig. 6(b)). Each concept $c \in \mathcal{C}$ consists of a set of *concept prototypes* $P^c=\{p_1, p_2, \dots\}$, which are used to derive the correspondence of unknown objects to concepts. In the spirit of *Prototype Theory* [40], unknown instances are classified

based on the similarity to known instances, which are associated to the previously learned shape concepts. To demonstrate the discrimination capability of our shape representation, the similarity $\phi^c(\cdot)$ to a concept c is determined by a (basic) mean similarity among ${}^*\gamma^o$ and prototypes P^c of concept c (see Eq. 4); as distance measure the *Mahalanobis distance* $d_{\text{mah}}(\cdot)$ is used.

$$\phi^c({}^*\gamma^o) = \frac{\sum_{i=1}^{|P^c|} d_{\text{mah}}({}^*\gamma^o, p_i \in P^c)}{|P^c|} \quad (4)$$

5 Machine-Centric Concept Generation Through Mental Simulation

An interesting question is how the training data is generated to learn the shape concepts. One option is to use datasets of real-world objects. In contrary, we propose the use of mental simulation to generate *abstract artificial* objects for shape concept learning purposes. This approach generates concepts in a machine-centric manner, i.e., concepts are learned in an unsupervised fashion in two respects: a) *label-agnostic* (no label information given by supervision is used) and additionally b) *instance-agnostic* (no real-world instances preselected by human supervision are used for training). As will be shown in the experiments in Sec. 7, the shape concepts learned in this way generalize well when applied to objects from real-world datasets.

The core idea for the mental simulation is described in the following. We start with primitive-shaped building-blocks or prototypes, namely *box*, *sphere*, and *cylinder*. Multiple prototypes can be randomly combined to a prototype composition which forms an abstract object. We denote the number of introduced prototypes of an abstract object as the *prototype order*. The Gazebo simulation environment [47] is then used to generate these artificial abstract objects in simulation and to capture samples of the generated objects with a virtual sensor in simulation. Fig. 7 shows samples of artificial abstract objects of different prototype orders, captured in simulation.

Algorithm 1 Artificial Sample Generation

Input: prototype order n , empty sample s

- 1: $i \leftarrow 0$
- 2: **while** $i < n$ **do**
- 3: $p \leftarrow \text{get_random_prototype}(\{\text{box}, \text{cylinder}, \text{sphere}\})$
- 4: $p \leftarrow \text{set_random_dimensions}(\{\text{length}, \text{width}, \text{height}, \text{radius}\})$
- 5: **if** $i > 0$ **then**
- 6: $p \leftarrow \text{set_random_pose}(\{\text{position}, \text{orientation}\})$
 so that p intersects with s
- 7: **end if**
- 8: $s \leftarrow p$ (introduce prototype p to sample s)
- 9: $i \leftarrow i+1$
- 10: **end while**

Output: sample s representing a composition of prototypes.

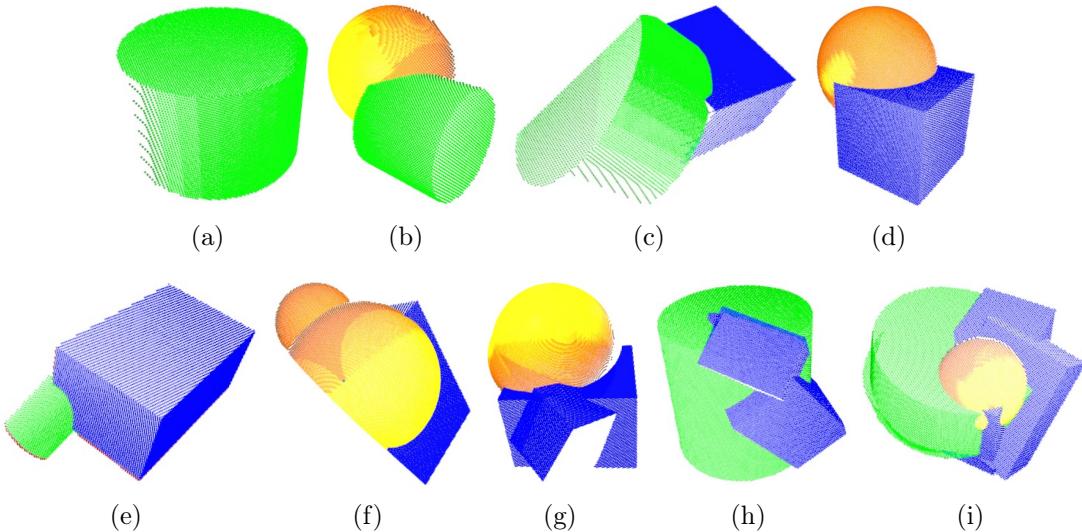


Figure 7: Examples of randomly generated abstract objects. Objects can encompass up to five primitive-shaped prototypes. Only for illustration purposes, the primitive-shaped prototypes of each object are distinctively colored: **box**, **can** and **sphere**.

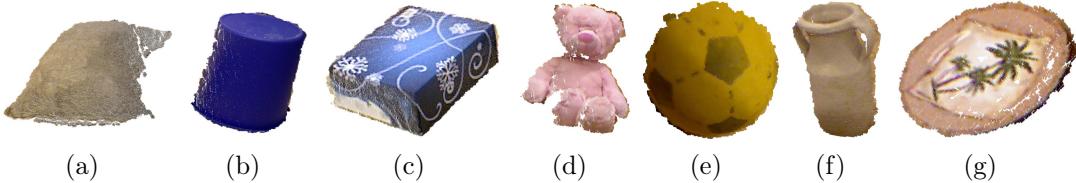


Figure 8: Examples of 2.5D scans from the OSCD dataset: *sack* (a), *can* (b), *box* (c), *teddy* (d), *ball* (e), *amphora* (f) and *plate* (g).

Each prototype of an artificial object sample is not only randomly generated with respect to its type (*box*, *sphere*, *cylinder*) but also with respect to its spatial dimensions (e.g., *length*, *width*, *height*, *radius*). Each prototype has to overlap with at least an other prototype in order to form a connected structure, which is considered as a valid object (see Alg. 1). Using this random approach, these object samples are obviously generated without any human bias.

6 Experiments: Label-Agnostic Learning

The experimental evaluation is two-fold. This section deals with the performance of the label-agnostic concept generation. This means that the shape concepts are learned in an unsupervised manner; semantic object labels generated by humans are only used to evaluate how reasonable the generated concepts are. Among others, it will be shown that concepts learned on one real-world dataset also generalize well to other real-world datasets consisting of different objects. In the

following Sec. 7, the focus is on the evaluation of machine-centric learning of the shape concepts, i.e., not real-world data but abstract artificial objects from mental simulation are used for training, which also leads to concepts that also perform well on the real-world datasets.

The *Object Shape Category Dataset (OSCD)*¹ [16] is used for the first part of the evaluation. It consists of 468 RGBD scans of real-world objects from 7 categories. A few examples are shown in Fig. 8).

6.1 Topological Filtration

In the training phase, each training sample scan (OSCD dataset) is propagated through $\mathcal{HE} = \{\mathcal{H}_1, \mathcal{H}_2, \dots, \mathcal{H}_n\}$, omitting any label-related information, i.e., each scan is applied in an unsupervised manner to the \mathcal{HE} ; in our evaluation $n=4$ has been heuristically selected – a smaller n may not allow \mathcal{HE} to sufficiently discriminate the observed range of object shape variety. Afterwards, extracted stimuli vectors are fed to the filtration process (see Sec. 4). Fig. 6(a) illustrates the filtration result of the stimuli vectors; the visualization does not reflect metric differences, it visualizes topological similarities among samples. Already at this stage, topological similarity can be observed with respect to the category labels of the objects. Note that the category labels are only associated to the prototypes for visualization purposes - as mentioned, they were not used in the training. In Fig. 9(a), the barcode is shown of the homology group 0.

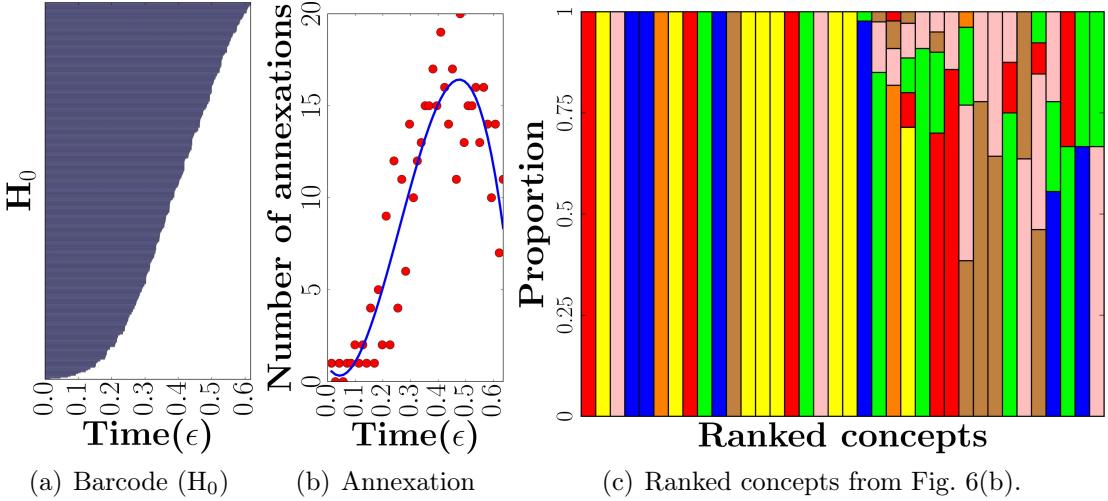


Figure 9: (a) Barcode of the homology group 0. (b) The number of annexations among Homology classes. (c) The proportional distribution of prototypes per concept. For visualization purpose, each proportion within a bar is colored with the corresponding label according to Fig. 5 and sorted in ascending order by $rs(\cdot)$, see Eq. 6.

¹<http://www.robotics.jacobs-university.de/datasets/2017-object-shape-category-dataset-v01/index.php>

At time ϵ_0 all *concept prototypes* – depicted as bars – are born. While the filtration progresses, more and more prototypes form larger homology classes that lead to the death (end of a bar) of prototypes, which have been annexed. As a result, only a single simplex at time ϵ_j survives the filtration (see Sec. 4.1.4). Moreover, Fig. 9(b) shows only the number of annexation of homology classes over time. It can be observed that the filtration reaches a global maximum of annexations at $\epsilon_{max}=0.48$, i.e., the annexation of classes decreases even though ϵ reaches its maximum value. It can be interpreted that the extracted homology classes after $\epsilon_{max}=0.48$ are already discriminative by their persistence.

6.2 Unsupervised Concept Selection

The gradual filtration process as described in Sec. 4.1 allows to analyze the topological space at any filtration step. Each filtration step offers insights about the topology and the relation among concept prototypes. Note that the choice of a specific number of concepts and concept size depends on the objective of the application scenario.

Using ϵ_{max} as indicator to stop the filtration process and to subsequently select the existing homology classes at time ϵ_{max} as concepts, we receive in total 36 concepts \mathcal{C} (see Fig. 6(b)) with a minimum concept size of 2. To assess the quality of the extracted concepts we can make use of the human-annotated category labels, which are associated to the prototypes (see Fig. 6(b)). Therein, the correlation between the concepts and the labels given a priori by a human can be interpreted as a quality measure for the concepts learned in an unsupervised manner. The amount of this correlation or *purity* $pu(\cdot)$ can be defined as the largest proportion in the distribution of prototypes of a category label, see Eq. 5, where concept $c \in \mathcal{C}$ consists of a set of concept prototypes $P^c = \{p_1, p_2, \dots\}$ which are accordingly attributed with labels $Y^c = \{y_1, y_2, \dots\}$, i.e., $y_i = \text{retrieve_label}(p_i)$, given the set of category labels \mathcal{Y} of the dataset where $y_i \in \mathcal{Y}$.

$$pu(c) = \arg \max_{y \in \mathcal{Y}} \frac{\sum_{i=1}^{|P^c|} \mathbb{1}_y(y_i \in Y^c)}{|P^c|} \quad (5)$$

Given the concepts inferred by ϵ_{max} as described in Sec. 4.2.3 and illustrated in Fig. 6(b), it can be observed that connected components of different sizes are extracted, which is caused by the shape heterogeneity of the prototypes in \mathcal{X} . A large portion of the concepts is pure (see Eq. 5), i.e., there is a perfect correlation and only prototypes of a specific category $y \in \mathcal{Y}$ are assigned to a concept $c \in \mathcal{C}$. In Fig. 9(c), the resulting distribution of prototypes within a concept is illustrated. Concepts are sorted in ascending order by the *rank score* $rs(c)$, which computes the concept purity $pu(c)$ with respect to the concept size $|P^c|$, see Eq. 6.

$$rs(c) = \frac{|P^c|}{1 - pu(c) + \varepsilon}, \text{ where } \varepsilon \text{ is a small constant } (0 < \varepsilon \ll 1) \quad (6)$$

While 57% of the concepts are pure, other concepts show a lower purity, i.e., samples of different categories are assigned to a particular concept. However, these

Table 1: Unsupervised concept selection: testing set (5 repetitions)

Label:	sack	can	box	teddy	ball	amphora	plate
Mean error (%):	4.2	6.5	2.5	8.8	0	10.4	0

categories show shape similarities like *sack* and *can* or *plate* and *box*. Furthermore, the mean concept purity is 86.2%.

Given the 36 concepts, responses are extracted for each sample of the dataset, i.e., each sample object o is represented by $\rho^o = \{\phi^1(*\gamma^o), \phi^2(*\gamma^o), \dots\}$ ($|\rho^o| = |\mathcal{C}| = 36$) and labeled with the corresponding dataset label. Accordingly, a Support Vector Machine (SVM) is trained and evaluated, see Table 1. Discriminative results have been obtained, which allow to assess how reasonable the extracted concepts are, e.g., shapes like *ball*, *plate* or *box* show low cross-validation error, whereas appearance variety of categories that include deformability or strong viewpoint dependability, e.g., *teddy* or *amphora* can appear more ambiguous.

6.3 Generalization to Other Real-World Datasets

The following experiment evaluates the generalization capability of the proposed approach. First, \mathcal{HE} is trained once with the training set of the OSCD dataset. This training process is unsupervised, i.e., \mathcal{HE} is solely trained with instances in a label-agnostic manner. Then, instances from the OSCD dataset are propagated through \mathcal{HE} (see Sec. 3). Based on the resulting stimuli vector of the propagation, concepts C are generated (see Sec. 4).

Given the previously trained \mathcal{HE} model and the generated concepts C , we evaluate in the following the discriminative power of the concepts with instances from different real-world datasets. In addition to the OSCD objects, additional datasets with completely different real-world objects are used, namely the *Washington RGB-D Object Dataset* [17] (WD) and the *Object Segmentation Database* [18] (SD) (see Table 2); note that all three datasets are sampled from *different distributions* as illustrated in Fig. 10(a)-(f).

In order to analyze the spectrum of responses for these dataset objects, each object o is initially represented with as graph of segments g^o (see Sec. 3.1) and applied to the two-step procedure: **1)** propagate g^o through \mathcal{HE} to generate a stimuli vector $*\gamma^o$ (see Sec. 3.3); **2)** compute for each concept $c \in \mathcal{C}$ the response with $\phi^c(*\gamma^o)$ (see Eq. 4 in Sec. 4.3). As a result, an object o generates a set of concept responses $\rho^o = \{\phi^1(*\gamma^o), \phi^2(*\gamma^o), \dots\}$ ($|\rho^o| = |\mathcal{C}| = 36$, see Fig. 6(b)).

Consequently, a $|\mathcal{C}|$ -dimensional space of concept responses $\mathcal{CR}^{|\mathcal{C}|}$ is created. The generalization capability can be assessed by $\mathcal{CR}^{|\mathcal{C}|}$, which allows to observe relations and similarities among sample objects. To visualize and reason about the $|\mathcal{C}|$ -dimensional $\mathcal{CR}^{|\mathcal{C}|}$ space, the *t-SNE* [48] embedding technique is applied to reduce the dimensionality to two; we denote this 2D space as \mathcal{CR}^2 . The embedding is performed in an unsupervised manner, i.e., it is label-agnostic. Fig. 11 shows instances from the WD, SD and OSCD datasets projected to the two-dimensional



Figure 10: Examples of appearance variations of sample point clouds related to the concept *can*, respectively cylinder, from different distributions (datasets): (a), (b) show *can 0* and *56* of OSCD-training set, (c), (d) show *food_can_1_1_1* and *food_can_14_1_1* of *WD* and (e), (f) show cylindrical instances from scenes *learn_34* and *test_42* of *SD*.

Table 2: Sample distribution of the \mathcal{CR}^2 space

Label	WD [17] scans	#	SD [18] scans	#	OSCD [16] scans	#	Σ
<i>sack</i>	<i>food bag 1-8</i>	40			<i>sack 0-56 (tr. set)</i>	57	115
					<i>sack 0-17 (te. set)</i>	18	
<i>can</i>	<i>food can 1-14</i>	70	<i>learn 33-44</i>	38	<i>can 0-59 (tr. set)</i>	60	259
	<i>soda can 1-6</i>	30	<i>test 31-42</i>	42	<i>can 0-18 (te. set)</i>	19	
<i>box</i>	<i>cereal box 1-5</i>	25	<i>learn 0-16</i>	38	<i>box 0-53 (tr. set)</i>	54	232
	<i>food box 1-12</i>	60	<i>test 0-15</i>	36	<i>box 0-18 (te. set)</i>	19	
<i>teddy</i>					<i>teddy 0-44 (tr. set)</i>	45	59
					<i>teddy 0-13 (te. set)</i>	14	
<i>ball</i>	<i>ball 1-7</i>	35			<i>ball 0-39 (tr. set)</i>	40	
	<i>lime 1-4</i>	20			<i>ball 0-9 (te. set)</i>	10	125
	<i>orange 1-4</i>	20					
<i>amphora</i>					<i>amphora 0-47 (tr. set)</i>	48	62
					<i>amphora 0-13 (te. set)</i>	14	
<i>plate</i>	<i>plate 1-7</i>	35			<i>plate 0-49 (tr. set)</i>	50	105
					<i>plate 0-19 (te. set)</i>	20	
Σ		-	335	-	154	-	468 957

Note, for each instance of WD the 1st to 5th point cloud scans are selected of the first video sequence.
(tr.=training, te.=testing)

\mathcal{CR}^2 space.

For illustration, regions in \mathcal{CR}^2 are colored according to their correlation with a certain label (see Fig. 11) by exploiting the projected instances as anchor points in space. A uniform grid is created in the 2D \mathcal{CR}^2 space; for each cell in the grid the k -nearest instances are determined (e.g., $k=5\%$ of total number of instances); then the majority label of the k instances is determined and the cell is colored according to the majority label; each cell is weighted and visually depicted in form of cell opacity. The weight represents the observed proportion of the k instances associated to the majority label, which is depicted in an interval $[0, 1]$ from low to high proportion [low: transparent (white)=0, high: opaque (solid majority label color)=1].

The continuous space \mathcal{CR}^2 shown in Fig. 11 allows to observe regional characteristics and relations among locations in \mathcal{CR}^2 and instances of the three datasets. A main observation is that instances from different datasets are propagated through

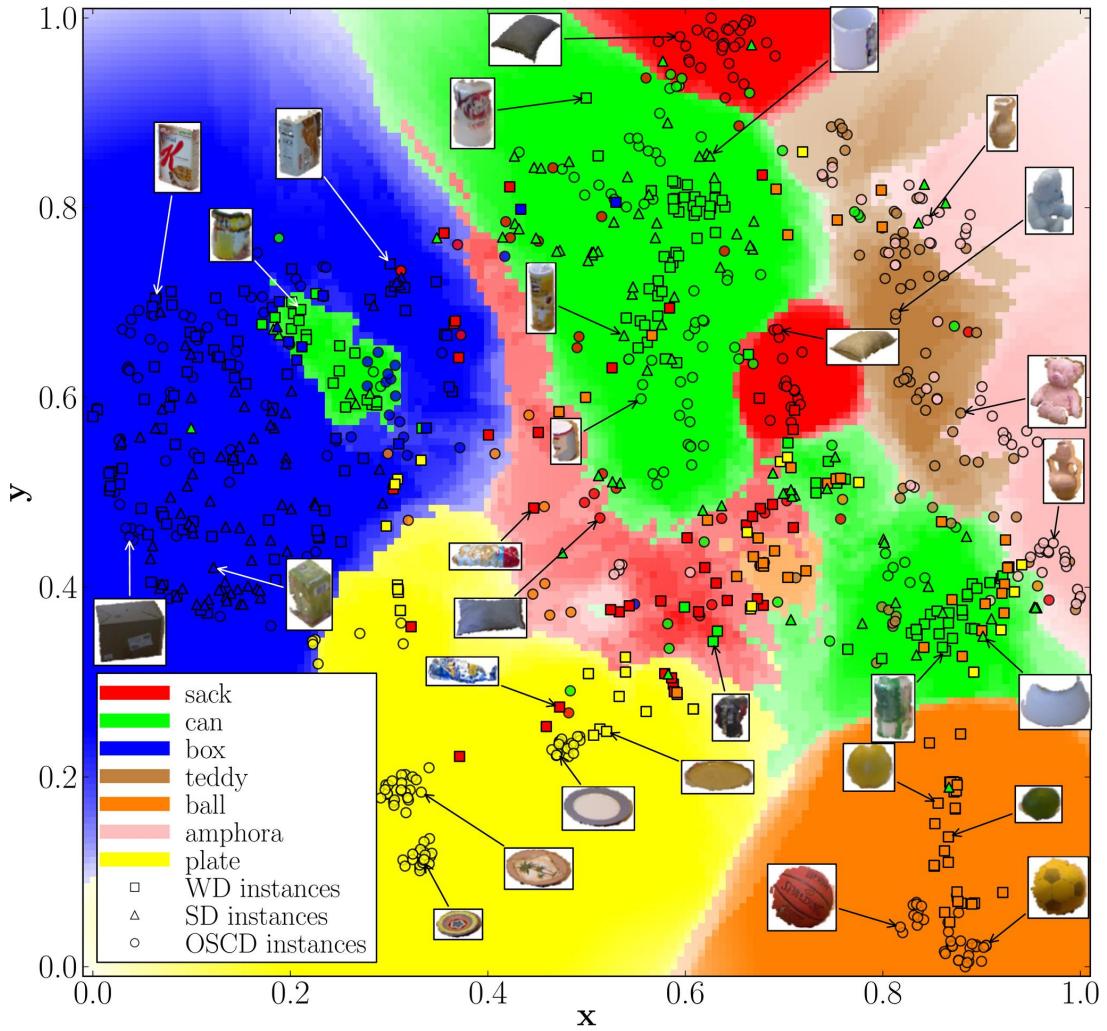


Figure 11: \mathcal{CR}^2 with instances from the WD, SD and OSCD datasets (see Table 2). The instance annotations are scaled for better visibility.

the \mathcal{HE} and the resulting concept responses show a strong coherency with respect to shape appearance: different instances from the different datasets that can be considered to be similar on a human semantic level, form interrelated and coherent groups, as shown by the uniformly colored regions in Fig. 11. This is also reflected in Fig. 12(a) and (b) that illustrate the distribution of instances in \mathcal{CR}^2 space. Instances labeled as *can*, *box*, *ball*, *amphora*, *plate* form distinct regions whereas deformable instances like *sack* and *teddy* lead to more scatter. However, *teddies* are still represented as a connected region and regions dedicated to *sack* are located at transitions to other labeled regions, e.g., *can* to *plate*, *can* to *box* or *can* to *teddy*. This observation can be explained that *sacks* can be interpreted as an intermediate shape, e.g., between a *box* and a *can* in \mathcal{CR}^2 space due to their roundish, bulgy or cylindric appearance depending on viewpoint and deformation.

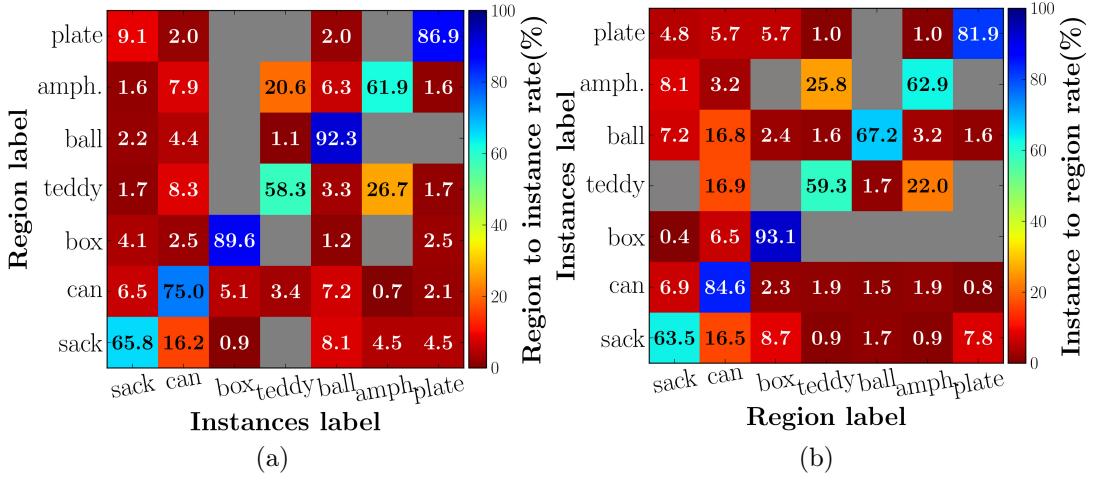


Figure 12: According to \mathcal{CR} in Fig. 11, the distribution is shown of instances within a region (a) and assignment of instances to particular regions (b).

7 Experiments: Mental Simulation

In this section, the performance is evaluated when using the mental simulation for training (Sec.5). To allow a comparison of our approach with other work, the concrete random samples that are used in this experiment are provided as open dataset, which is denoted in the following as *Artificial Object Dataset*² (AOD). Examples of samples of the artificial abstract objects from this dataset are shown in Fig. 7. The dataset contains 250 training samples, which were artificially generated with an equally distributed number of samples per prototype order (1 to 5). These artificially generated samples are used to generate shape concepts including the \mathcal{HE} generation.

7.1 Shape Knowledge Transfer from Mental Simulation to Real-World Data

We start the evaluation with an illustrative example. In Fig. 13 three (very simple) simulated objects are shown with their respective extracted segment graph (g^o). The simulated instances consist of noise-free point clouds; thus segments are optimally segmented. When using simulation-based training sample generation, an open question is whether the perception system is able to transfer the knowledge observed in simulation to real object observations. To test this in this simple illustrative example, the three artificial instances in Fig. 13 are consecutively fed (from Fig. 13(a) to (c)) to \mathcal{HE} and the learned motif prototypes are labeled with the respective label.

In Fig. 14(b) the classification results of this illustrative example on a real-world scene is shown. More precisely, the label with the highest accumulated

²<http://www.robotics.jacobs-university.de/datasets/2018-artificial-object-dataset-v01/index.php>

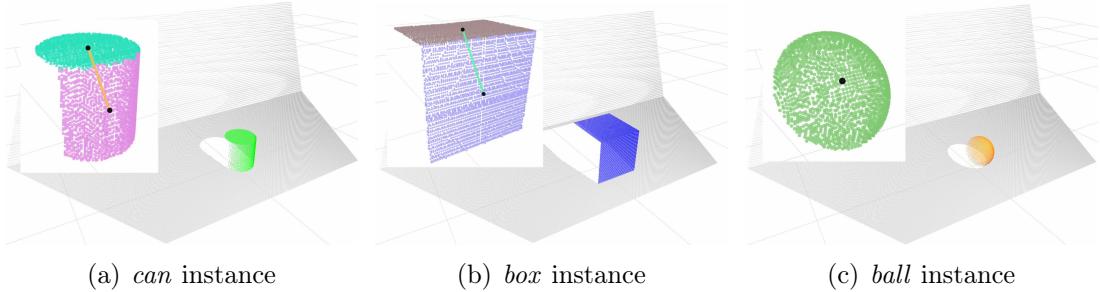


Figure 13: Examples of simulated instances of (simple) shape categories and the corresponding extracted super patch graphs (top left in (a), (b), (c)).

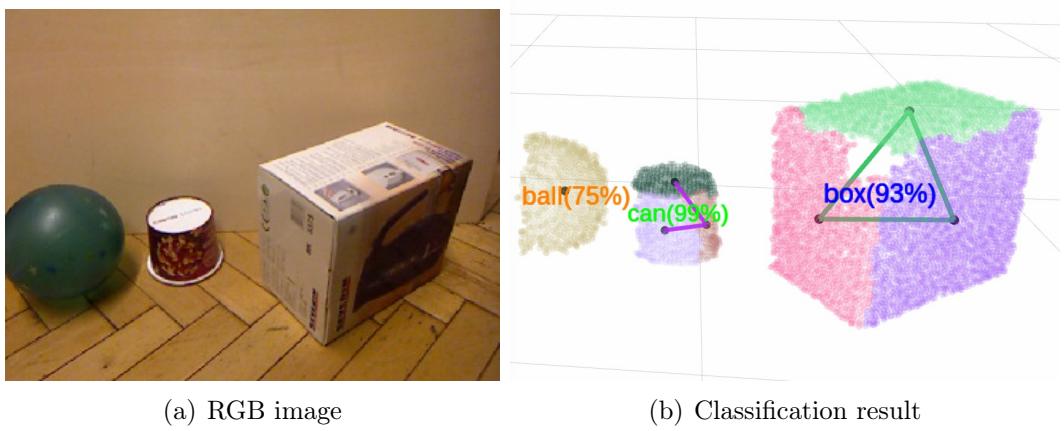


Figure 14: A sample classification result on real-world scene data using a model trained with simulated objects only (Fig. 13). Note that, objects are segmented from the scene with our previous work [38] and then classified with the trained model.

stimulus considering the observed (labeled) motif prototypes (Sec. 3.3) is shown for each object. Note that, \mathcal{HE} has been only trained with a single artificial instance per label (*can*, *box* and *ball*). Several observations can be made from the classification results. Considering the correct classification, one may interpret it as a knowledge transfer from simulated data to real noisy observed data. Regarding sensor noise, segmented surfaces are distorted and may even contain holes (Fig. 14(b)). These distortions lead to segment constellations, which have not been observed in the training phase. The simulated cylinder (*can*) in Fig. 13(a) naturally consists of an upper planar segment and a cylindric body, whereas the real *can* shown in Fig. 14(b) is over-segmented and subsequently consists of three segments caused by sensor noise. Nevertheless, this segment constellation has not been observed in training phase but still led to a correct classification as it is closest to the ideal cylinder concept. Further on, different viewpoints on objects can lead to different segment constellations due to self-occlusion effects. The viewpoint on the box in Fig. 13(b) results to two planar segments whereas the

viewpoint on the box shown in Fig. 14(b) leads to three segments and a hole (red colored segment) caused by sensor noise. Also in this case, this segment constellation has not been observed in the training phase but it still leads to a correct and confident classification. Furthermore, note that the simulated instances used for training have in addition completely different spatial dimensions compared to the real objects shown in Fig. 14(b).

7.2 Generalization to Real-World Datasets

This experiment evaluates the generalization ability using extensive artificial training data from mental simulation, i.e., the Artificial Object Dataset (AOD) with 250 simulated samples of abstract objects. Initially \mathcal{HE} is trained and concepts are generated once in an unsupervised and label-agnostic manner with the artificial samples of the AOD dataset. Given the \mathcal{HE} model and the concepts \mathcal{C} generated with the AOD, the generalization capability of \mathcal{C} is evaluated with real-world instances from the *Object Shape Category Dataset* [16] (OSCD), the *Washington RGB-D Object Dataset* [17] (WD) and the *Object Segmentation Database* [18] (SD), see Table 2. Note that all three real-world datasets are sampled from different distributions (see Fig. 10), i.e., the datasets consist of various, very different objects and they differ with respect to the experimental setups for the sensor data generation. In order to analyze the spectrum of responses for these dataset objects, each object o is applied to a two-step procedure: **1)** propagate o through \mathcal{HE} to generate a stimuli vector ${}^*\gamma^o$ (see Sec. 3.2); **2)** compute for each concept $c \in \mathcal{C}$ the response with $\phi^c({}^*\gamma^o)$. As a result, an object o is represented by the set of concept responses $\rho^o = \{\phi^1({}^*\gamma^o), \phi^2({}^*\gamma^o), \dots\}$ ($|\rho^o| = |\mathcal{C}| = 28$).

Consequently, in order to investigate the generalization capability, the approach as described in Sec. 6.3 is followed, i.e., a $|\mathcal{C}|$ -dimensional space of concept responses $\mathcal{CR}^{|\mathcal{C}|}$ is created and the embedding is performed to reduce the dimensionality to two. As a result, instances from the WD, the SD and the OSCD datasets are projected to this two-dimensional \mathcal{CR}^2 space (Fig. 15).

When looking at \mathcal{CR}^2 , an important observation is that after propagating the instances of the three datasets through the \mathcal{HE} , the resulting concept responses show also here coherency regarding shape appearance as in shown in Sec. 6.3. Instances of all evaluated datasets together form interrelated and coherent groups, see in Fig. 15 uniformly colored regions according to the labels of the real datasets. This is also reflected in Fig. 16 illustrating the instance distribution in \mathcal{CR}^2 space. By averaging the diagonal (bottom-left to top-right) one can observe 68.7% (Fig. 16(a)) / 67.6% (Fig. 16(b)) vs. 75.7% (Fig. 12(a)) / 73.2% (Fig. 12(b)), i.e., a similar discrimination has been achieved with the artificially generated training set (Fig. 16) compared to a real object training set (Fig. 12). Note that in an unsupervised manner \mathcal{CR} forms regions of various shapes and degree of label-association in a continuous space (Fig. 15) compared to these hard-assigned *discrete* results w.r.t. labels in Fig. 12 and 16, which may also contain noise in point clouds and in the labeling process. Thus, the discrete results may only partially reflect the underlying label-association strength

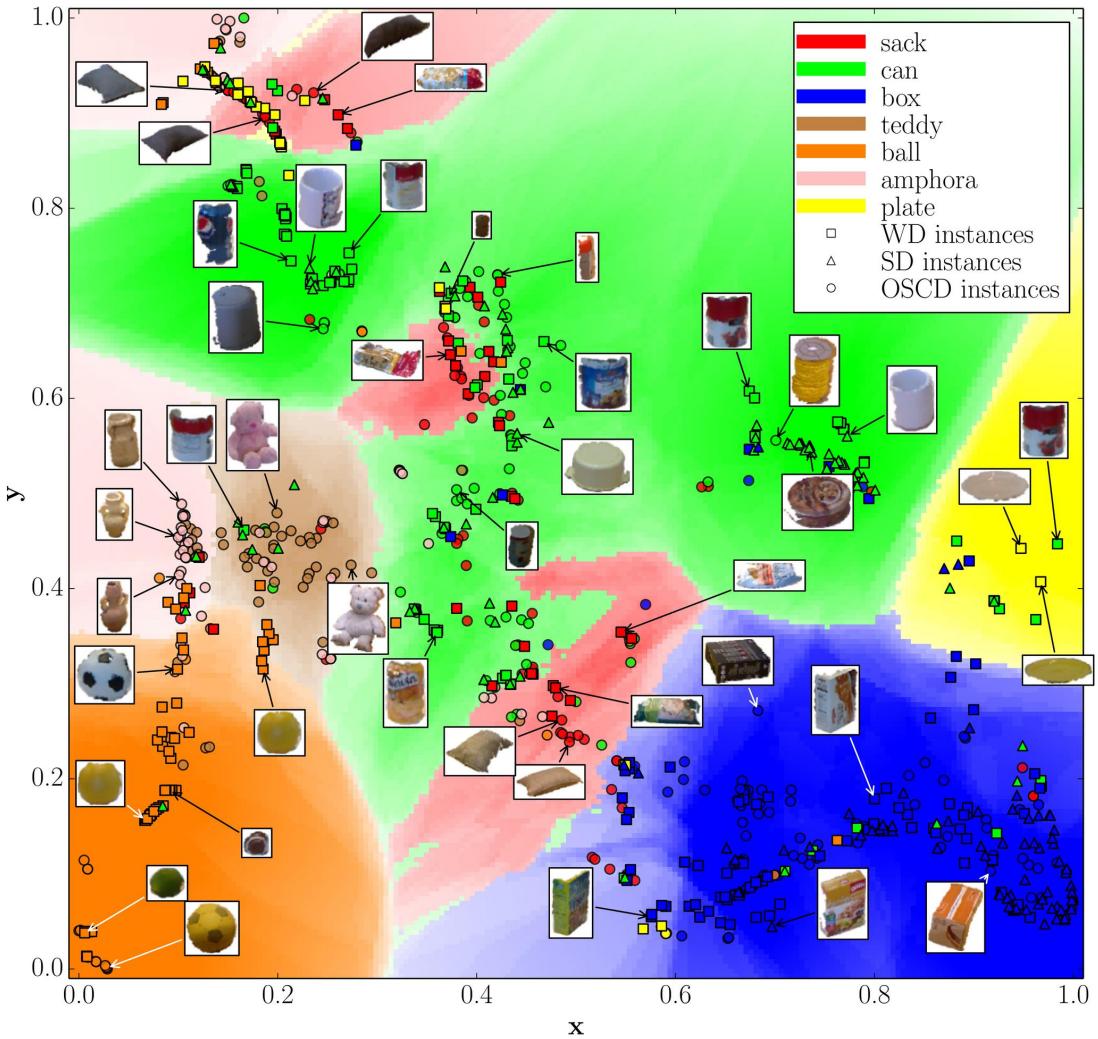


Figure 15: The projection of real-world samples from the OSCD, the WD and the SD dataset to the \mathcal{CR} space that is generated with mental simulation (Fig. 7). A summary of the instances used is shown in Table 2.

of objects compared to the continuous \mathcal{CR} space.

Consequently, this indicates that randomly generated, abstract instances based on composition of primitive shape prototypes from mental simulation carry information about facets of shape appearance that allow to create shape concepts which facilitate the generation of an abstract space suited to discriminate and categorize real object observations in a reasonable way. From the perspective of Cognitive Science, specifically in the field of representation architectures, \mathcal{CR} can be seen as a *Conceptual Space* [49, 50, 51] where points (prototypes) in the abstract space represent multidimensional vectors of *stimuli* and regions in space *concepts*. These stimuli are often denoted as *Quality Dimensions* and can be interpreted as concept responses ρ^o with respect to \mathcal{C} given an object o . Another property can be observed that supports that concept responses of similar instances appear close in \mathcal{CR} in comparison to dissimilar ones: the majority of

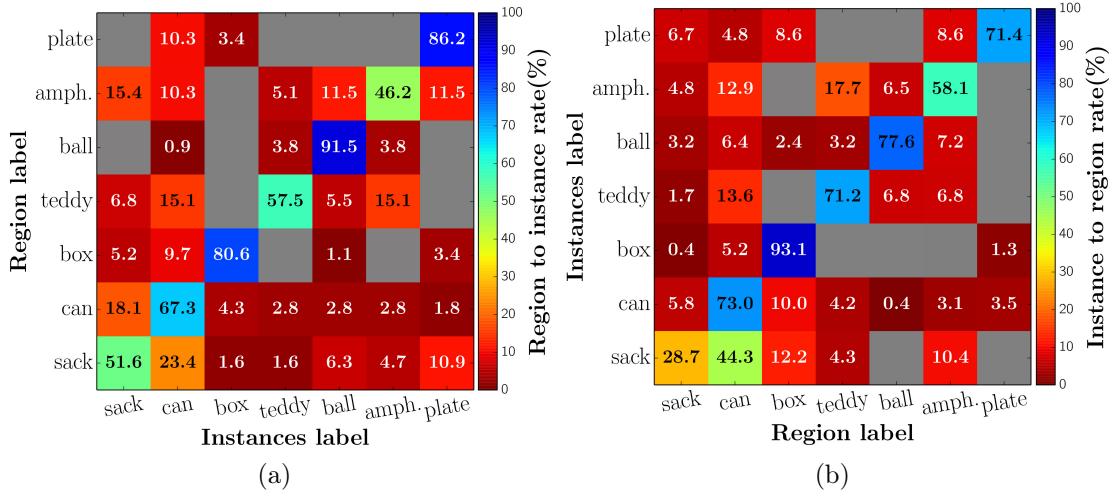


Figure 16: According to \mathcal{CR} in Fig. 15, the distribution is shown of instances within *a region* (a) and assignment of instances to *particular regions* (b).

instances of the respective label given by humans are closest or within the same region and form groups (Fig. 15 and Fig. 11).

8 Conclusion

We presented an unsupervised abstraction process for machine learning of shape concepts: from 3D point clouds over hierarchically organized motifs to (semantically meaningful) concepts of shape commonalities. The proposed Shape Motif Hierarchy Ensemble encodes object segment compositions in a hierarchical symbolic manner. Inspired by the concept of Persistent Homology, stimuli generated by the ensemble are filtered in a gradual manner to reveal topological structures. The filtration leads to stimuli groups which can be interpreted as shape concepts that reflect commonalities of shape appearances.

An important question is how this unsupervised learning is trained. Even when not using human labels, biases can be in the selection of the dataset instances used for training. Moreover, the generation of real-world datasets is cumbersome and generally requires substantial effort. Therefore, the use of mental simulation is investigated in this article, i.e., the generation of virtual sensor data from artificial abstract objects. This approach is unsupervised in two respects: it is label-agnostic (no label information is used) and instance-agnostic (no instances preselected by human supervision are used).

In a first set of experiments, the shape concepts are learned in an unsupervised, label-agnostic fashion from a single real-world dataset and it is shown that a) semantically meaningful categories emerge, i.e., associations to shape categories linked to human-annotated labels appear, and that b) the concepts generalize to other real-world datasets, i.e., the concepts learned on one dataset lead to meaningful label associations when being applied to completely different real-

world datasets. In a second set of experiments, these results are extended to mental simulation, i.e., the training is both label-agnostic and instance-agnostic. It is shown that training with virtual sensor data from artificial abstract objects leads to a semantically meaningful shape concept space, which generalizes to real-world object datasets. I.e., it leads to a shape concept space, in which unknown objects of real-world sensor data are grouped (based on their commonalities) into regions in concept space that can be, for instance, linked to human-annotated labels.

References

- [1] E. S. Spelke, “Principles of object perception,” *Cognitive Science*, vol. 14, no. 1, pp. 29 – 56, 1990.
- [2] L. B. Smith, “Learning to recognize objects,” *Psychological Science*, vol. 14, no. 3, pp. 244–250, 2003.
- [3] M. Graf, *Categorization and Object Shape*. Springer Berlin Heidelberg, 2010, pp. 73–101.
- [4] C. A. Mueller, K. Pathak, and A. Birk, “Object shape categorization in rgbd images using hierarchical graph constellation models based on unsupervisedly learned shape parts described by a set of shape specificity levels,” in *International Conference on Intelligent Robots and Systems*, 2014.
- [5] C. Eppner and O. Brock, “Grasping unknown objects by exploiting shape adaptability and environmental constraints,” in *International Conference on Intelligent Robots and Systems*, 2013.
- [6] P. Abelha, F. Guerin, and M. Schoeler, “A model-based approach to finding substitute tools in 3d vision data,” in *International Conference on Robotics and Automation*, 2016.
- [7] M. Thosar, C. A. Mueller, and S. Zug, “What stands-in for a missing tool? a prototypical grounded knowledge-based approach to tool substitution,” in *International Cognitive Robotics Workshop on Principles of Knowledge Representation and Reasoning (KR)*, 2018, arXiv:1808.06423 [cs.RO].
- [8] Sloutsky Vladimir M., “From Perceptual Categories to Concepts: What Develops?” *Cognitive Science*, vol. 34, no. 7, pp. 1244–1286, 2010.
- [9] V. Höglman, M. Björkman, A. Maki, and D. Kragic, “A sensorimotor learning framework for object categorization,” *IEEE Transactions on Cognitive and Developmental Systems*, vol. 8, no. 1, pp. 15–25, 2016.
- [10] T. Nakamura and T. Nagai, “Ensemble-of-concept models for unsupervised formation of multiple categories,” *IEEE Transactions on Cognitive and Developmental Systems*, 2018.

- [11] J. Nishihara, T. Nakamura, and T. Nagai, “Online algorithm for robots to learn object concepts and language model,” *IEEE Transactions on Cognitive and Developmental Systems*, vol. 9, no. 3, pp. 255–268, 2017.
- [12] F. G. Ashby and E. M. Waldron, “On the nature of implicit categorization,” *Psychonomic Bulletin & Review*, vol. 6, no. 3, pp. 363–378, 1999.
- [13] A. M. S. Barry, *Visual Intelligence : Perception, Image, and Manipulation in Visual Communication*. State University of New York Press, 1997.
- [14] Palmeri Thomas J. and Gauthier Isabel, “Visual object understanding,” *Nature Reviews Neuroscience*, vol. 5, no. 4, pp. 291–303, 2004.
- [15] S. Zmigrod and B. Hommel, “Feature integration across multimodal perception and action: A review,” *Multisensory Research*, vol. 26, no. 1-2, pp. 143–157, 2013.
- [16] C. A. Mueller and A. Birk, “Conceptualization of Object Compositions Using Persistent Homology,” in *International Conference on Intelligent Robots and Systems*, 2018.
- [17] K. Lai, L. Bo, X. Ren, and D. Fox, “A large-scale hierarchical multi-view rgbd object dataset,” in *International Conference on Robotics and Automation*, 2011.
- [18] A. Richtsfeld, T. Morwald, J. Prankl, M. Zillich, and M. Vincze, “Segmentation of unknown objects in indoor environments,” in *International Conference on Intelligent Robots and Systems*, 2012.
- [19] S. Biasotti, L. De Floriani, B. Falcidieno, P. Frosini, D. Giorgi, C. Landi, L. Papaleo, and M. Spagnuolo, “Describing shapes by geometrical-topological properties of real functions,” *ACM Computing Surveys*, vol. 40, no. 4, pp. 12:1–12:87, 2008.
- [20] J. J. DiCarlo and D. D. Cox, “Untangling invariant object recognition,” *Trends in Cognitive Sciences*, vol. 11, pp. 333–341, 2007.
- [21] R. Jonschkowski, C. Eppner, S. Höfer, R. M. Martin, and O. Brock, “Probabilistic multi-class segmentation for the amazon picking challenge,” in *International Conference on Intelligent Robots and Systems*, 2016.
- [22] J. a. Fodor and Z. W. Pylyshyn, “Connectionism and cognitive architecture: a critical analysis.” *Cognition*, vol. 28, pp. 3–71, 1988.
- [23] J. Papon, A. Abramov, M. Schoeler, and F. Wörgötter, “Voxel cloud connectivity segmentation - supervoxels for point clouds,” in *Computer Vision and Pattern Recognition*, 2013.

- [24] A. Anand, H. S. Koppula, T. Joachims, and A. Saxena, “Contextually guided semantic labeling and search for three-dimensional point clouds,” *The International Journal of Robotics Research*, vol. 32, no. 1, pp. 19–34, 2013.
- [25] B. Leibe, A. Leonardis, and B. Schiele, “Combined Object Categorization and Segmentation With An Implicit Shape Model,” in *European Conference on Computer Vision Workshop on Statistical Learning in Computer Vision*, 2004.
- [26] M. Prasad, J. Knopp, and L. Van Gool, “Class-specific 3D Localization using Constellations of Object Parts,” in *British Machine Vision Conference*, 2011.
- [27] U. Asif, M. Bennamoun, and F. Sohel, “Efficient rgbd object categorization using cascaded ensembles of randomized decision trees,” in *International Conference on Robotics and Automation*, 2015.
- [28] R. Kindermann and J. L. Snell, *Markov Random Fields and Their Applications*, 1980.
- [29] S. Geman and D. Geman, “Stochastic Relaxation, Gibbs Distributions, and the Bayesian Restoration of Images,” *Pattern Analysis and Machine Intelligence*, 1984.
- [30] J. Utans, “Learning in compositional hierarchies: Inducing the structure of objects from data,” in *Advances in Neural Information Processing Systems 6*, 1993, pp. 285–292.
- [31] S. Fidler, M. Boben, and A. Leonardis, “Learning hierarchical compositional representations of object structure,” in *Object Categorization: Computer and Human Vision Perspectives*, S. Dickinson, A. Leonardis, B. Schiele, and M. J. Tarr, Eds. Cambridge University Press, 2009.
- [32] M. Ozay, U. R. Aktas, J. L. Wyatt, and A. Leonardis, “Compositional hierarchical representation of shape manifolds for classification of non-manifold shapes,” in *International Conference on Computer Vision*, 2015, pp. 1662–1670.
- [33] K. R. Jerripothula, J. Cai, J. Lu, and J. Yuan, “Object co-skeletonization with co-segmentation,” in *Conference on Computer Vision and Pattern Recognition*, 2017.
- [34] W. Shen, K. Zhao, Y. Jiang, Y. Wang, X. Bai, and A. Yuille, “Deepskeleton: Learning multi-task scale-associated deep side outputs for object skeleton extraction in natural images,” *IEEE Transactions on Image Processing*, vol. 26, no. 11, pp. 5298–5311, 2017.
- [35] G. Carlsson, “Topological pattern recognition for point cloud data,” *Acta Numerica*, vol. 23, pp. 289–368, 005 2014.

- [36] C. Li, M. Ovsjanikov, and F. Chazal, “Persistence-based structural recognition,” in *Conference on Computer Vision and Pattern Recognition*, 2014.
- [37] W. J. Beksi and N. Papanikolopoulos, “3d point cloud segmentation using topological persistence,” in *International Conference on Robotics and Automation*, 2016.
- [38] C. A. Mueller and A. Birk, “Hierarchical Graph-Based Discovery of Non-Primitive-Shaped Objects in Unstructured Environments,” in *International Conference on Robotics and Automation*, May 2016.
- [39] R. Rusu, N. Blodow, and M. Beetz, “Fast Point Feature Histograms (FPFH) for 3D registration,” in *International Conference on Robotics and Automation*, 2009.
- [40] E. H. Rosch, “Natural categories,” *Cognitive Psychology*, vol. 4, no. 3, pp. 328–350, 1973.
- [41] C.-J. Huang and W.-C. Liao, “Application of probabilistic neural networks to the class prediction of leukemia and embryonal tumor of central nervous system,” *Neural Process. Lett.*, vol. 19, no. 3, pp. 211–226, 2004.
- [42] J. Lin, “Divergence measures based on the shannon entropy,” *IEEE Transactions on Information Theory*, vol. 37, no. 1, pp. 145–151, 1991.
- [43] H. Edelsbrunner, D. Letscher, and A. Zomorodian, “Topological persistence and simplification,” *Discrete & Computational Geometry*, vol. 28, no. 4, 2002.
- [44] A. Zomorodian and G. Carlsson, “Computing persistent homology,” *Discrete Computational Geometry*, vol. 33, no. 2, 2005.
- [45] X. Zhu, “Persistent homology: An introduction and a new text representation for natural language processing,” in *Proceedings of the Twenty-Third International Joint Conference on Artificial Intelligence*, 2013.
- [46] J. B. Kruskal, “On the shortest spanning subtree of a graph and the traveling salesman problem,” *Proceedings of the American Mathematical Society*, vol. 7, no. 1, pp. 48–50, 1956.
- [47] N. Koenig and A. Howard, “Design and Use Paradigms for Gazebo, An Open-Source Multi-Robot Simulator,” in *International Conference on Intelligent Robots and Systems*, 2004.
- [48] L. van der Maaten and G. E. Hinton, “Visualizing high-dimensional data using t-sne,” *Journal of Machine Learning Research*, vol. 9, pp. 2579–2605, 2008.
- [49] P. Gärdenfors, *Conceptual Spaces: The Geometry of Thought*. MIT Press, 2000.

- [50] F. Zenker and P. Gärdenfors, *Applications of Conceptual Spaces: The Case for Geometric Knowledge Representation*. Springer International Publishing, 2015.
- [51] S. Rama Fiorini, P. Gärdenfors, and M. Abel, “Representing part–whole relations in conceptual spaces,” *Cognitive Processing*, vol. 15, no. 2, pp. 127–142, 2014.