A Computational Model for the Alignment of Hierarchical Scene Representations in Human-Robot Interaction

Agnes Swadzba, Sven Wachsmuth
Applied Informatics
Bielefeld University
{aswadzba, swachsmu}@techfak.uni-bielefeld.de

Constanze Vorwerg, Gert Rickheit
Faculty of Linguistics
Bielefeld University
{constanze.vorwerg, gert.rickheit}@uni-bielefeld.de

Abstract

The ultimate goal of human-robot interaction is to enable the robot to seamlessly communicate with a human in a natural human-like fashion. Most work in this field concentrates on the speech interpretation and gesture recognition side assuming that a propositional scene representation is available. Less work was dedicated to the extraction of relevant scene structures that underlies these propositions. As a consequence, most approaches are restricted to place recognition or simple table top settings and do not generalize to more complex room setups. In this paper, we propose a hierarchical spatial model that is empirically motivated from psycholinguistic studies. Using this model the robot is able to extract scene structures from a time-of-flight depth sensor and adjust its spatial scene representation by taking verbal statements about partial scene aspects into account. Without assuming any pre-known model of the specific room, we show that the system aligns its sensor-based room representation to a semantically meaningful representation typically used by the human descriptor.

1 Introduction

Although robotic systems designed for communicating and interacting with humans have already achieved an impressive performance [Böhme *et al.*, 2003; Kim *et al.*, 2004; Li *et al.*, 2005; Montemerlo *et al.*, 2002; Simmons *et al.*, 2003; Tomatis *et al.*, 2002], they suffer from an insufficient understanding of scenes. In this paper, we will focus on indoor environments, i.e. living rooms, offices, etc. Given the current state of technology, the human interaction partner is either able to specify global room types, e.g. *"This is the living room"*, or individual objects, e.g. *"Take the cup"* [Mozos *et al.*, 2007; Torralba *et al.*, 2003]. In the case of more complex spatial descriptions, the scenario is typically restricted to a single table top allowing the user to specify simple binary spatial relations between objects [Brenner *et al.*, 2007; Mavridis and Roy, 2006; Wachsmuth and Sagerer, 2002]. Both scene representations abstract completely from sensoric data and relate verbal descriptions to a set of propositions that are judged by object detectors or localization procedures. They do not consider the top-down influence of a verbal description on establishing a model of the scene structure.

Such approaches are not generalizable to more complex scenes because they miss an intermediate level of scene representation. An indoor environment – such as a living room – typically consists of a configuration of several pieces of furniture with many smaller items placed on tables, shelves, or side-boards. In order to talk to the robot about *a pen lying beside a book on a table that should be placed back into a drawer under the desk*, the scene needs to be represented at different levels of granularity. However, the automatic extraction of geometric scene structures from sensoric data is a great challenge that suffers from occlusion and segmentation issues. Without a large amount of pre-knowledge it is nearly impossible to completely extract chairs, tables, shelves, or side-boards in a purely data-driven manner. Such a kind of process will always generate much over- or under-segmentation. Therefore, the robotic system will come up with a different scene structure than the human communication partner expects.

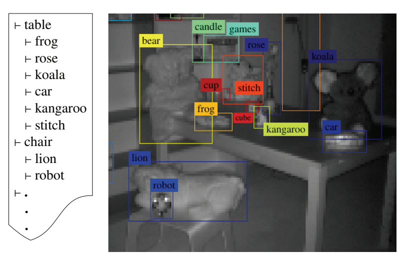


Figure 1: Object examples and their relations.

Rather than specifying room layouts beforehand, much scene information can be implicitly learnt from verbal user descriptions (see Fig. 1). The more the robot learns about a scene, the more consistent the scene representation will be. This leads – step by step – to spatial structures that are *aligned* between the robotic system and the user.

The paper is structured as follows. In Section 2 we discuss the state of the art in computational spatial models. Our main contribution is explained in Section 3 and 4, where a hierarchical model of static scenes is proposed motivating its assumptions from an empirical psycholinguistic perspective and the data-driven adaptation of these scene structures is described processing Time-of-Flight (ToF) depth data. Section 5 gives a final conclusion.

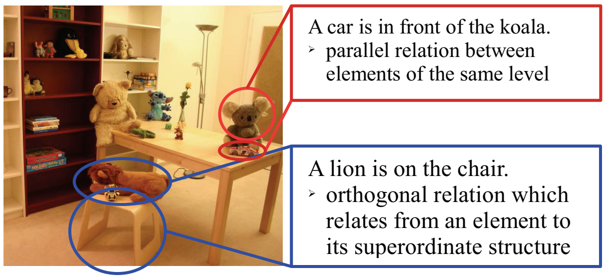


Figure 2: Photograph of the scene presented to the subjects for description. Two typical relations between scene elements (a parallel and an orthogonal one) are visualized.

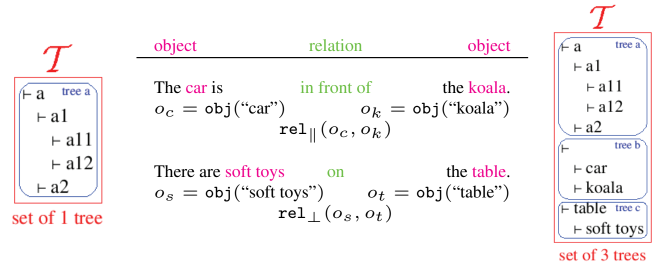


Figure 3: Introduction of the notation for $\mathcal{T}$, set of trees. $\mathcal{T}$ starts with one tree (tree a – a1 and a2 are children of a, a11 and a12 are child nodes of a1). Two expressions are given which are transferred to a parallel and an orthogonal relation. The resulting $\mathcal{T}$ contains three trees (tree a, tree b, tree c) when the objects were inserted with regard to the definition of the relations. $\vdash$ is used for items of tree structures.

2 Related Work

There have been a few approaches that consider more complex scene structures in human-robot scenarios. Zender *et al.* [2008] propose a multi-layered spatial representation consisting of a metric map, a graph-based navigation map, a topological map dividing the set of graph nodes to areas, and a conceptual map linking the low-level maps and the communication system. Beeson *et al.* [2007] introduce a Hybrid Spatial Semantic Hierarchy (HSSH) as a rich interface for human-robot interaction. It combines large-scale space structures with knowledge about small-scale spaces and allows reasoning on four levels (local metrical, local symbolic, global symbolic, global metrical). In Hois *et al.* [2006], the scene description is based on a set of planes that are detected by a laser sensor. They focus on the combination of vision and language in order to classify objects placed in the scene into functional object categories.

All approaches described assume a correctly extracted scene structure that is compatible with verbal descriptions of human interaction partners. Based on this information, Hois et al. are able to map verbal descriptions to scene objects. In the following, we explore the opposite direction. If we are able to map verbal object descriptions to scene objects, what can we infer about the scene structure?

3 Computational Model and Empirical Foundation

This section deals with the structural elements of a human-given description about a static indoor scene. They are examined empirically in a study (Sec. 3.1) and the insights are used to propose a computational model which provides a hierarchical model of the scene layout referring to meaningful structures (Sec. 3.2).

3.1 Empirical Foundation

People's descriptions of spatial scenes reflect aspects of their mental representations of the perceived scenes relevant for communication. While there are many psycholinguistic studies using so-called 'ersatz scenes' (displays of arbitrarily arranged objects) few have addressed the way people talk about 'true scenes' (real or depicted views of natural environments [Henderson and Ferreira, 2004]). These are semantically coherent and comprised of both background elements and objects which are spatially arranged [Henderson and Hollingworth, 1999]. To investigate what people's descriptions reveal about their internal model of a visually perceived complex room setup in general and what specific information can be gained by analyzing their verbal statements with respect to spatial relations between objects and background planes, we conducted a psycholinguistic study in which participants gave verbal descriptions of a depicted room. Ten native speakers of German participated in this study. They were shown a photograph of a real room containing shelves, a table, a chair, and some small objects located on them (e.g., a toy car, a toy koala, a cup, etc. see Fig. 2). Their task was to describe what they saw in the picture. The verbal descriptions produced were analyzed with respect to the relative frequency of object references (for small objects, items of furniture, and room parts), the scanning paths expressed by linearization strategies (sequence of object references presents the attention of the subject), and the types of spatial relations named. A basic analysis of the experimental data confirmed the importance of spatial room structures (formed by pieces of furniture and room parts) as crystallization points of room descriptions and a hierarchical spatial representation of the perceived scene. The use of a hierarchical spatial model as a basis for the scene descriptions is evidenced by the fact that objects are verbally localized relative to their supporting room structure or to another object supported by the same room structure. In other words, the spatial relations verbalized in the room descriptions belong either to the *orthogonal* type (relation to a superordinate structure, e.g., "on the chair") or to the *parallel* type (relation to another element at the same level and located on the same superordinate structure, e.g., "in front of the koala"). These data support the conclusion that small objects and background elements of the visual scene are restructured in the mental model in a hierarchical way reflecting 3D spatial relations.

3.2 Computational Model

Given a scenario introduced in Sec. 3.1, these verbal descriptions are often organized in sequences of so-called *parallel* and *orthogonal* relations between pairs of objects. These two types of spatial relations are defined as following:

- $\texttt{rel}_{\parallel}(o_1, o_2)$:
  describes a *parallel* relation between two objects o_1 and o_2 in the sense of o_1 lies in front of/behind/next to/above/below o_2 (e.g "a car is in front of the koala"). It can be inferred that both objects can be assigned to the

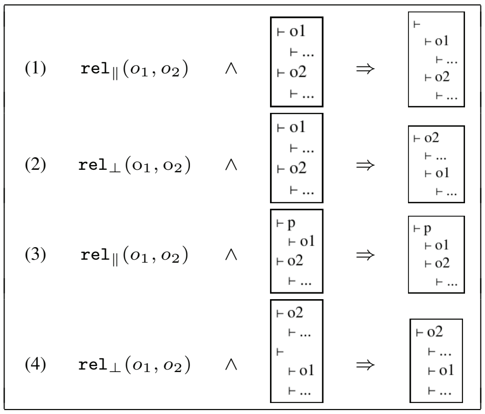


Figure 4: Graphical visualization of the rules introduced in Fig. 5

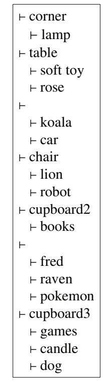
$$\texttt{rel}_{\parallel}(o_1, o_2) \Rightarrow \exists p = \texttt{obj}(\text{“ ”}) \rightarrow n_{\text{p}}, \texttt{child}(n_{\text{o1}}, n_{\text{p}}), \texttt{child}(n_{\text{o2}}, n_{\text{p}}) \quad (1)$$

$$\texttt{rel}_{\perp}(o_1, o_2) \Rightarrow \texttt{child}(n_{\text{o1}}, n_{\text{o2}}) \quad (2)$$

$$\texttt{rel}_{\parallel}(o_1, o_2) \wedge \exists n_{\text{p}} \in \mathcal{T} : \texttt{ischild}(n_{\text{o1}}, n_{\text{p}}) \Rightarrow \texttt{child}(n_{\text{o2}}, n_{\text{p}}) \quad (3)$$

$$\texttt{rel}_{\perp}(o_1, o_2) \wedge \exists n_{\text{p}} \in \mathcal{T} : \texttt{ischild}(n_{\text{o1}}, n_{\text{p}}) \Rightarrow \forall n : \texttt{ischild}(n, n_{\text{p}}) \rightarrow \texttt{child}(n, n_{\text{o2}}), \texttt{delete}(n_{\text{p}}) \quad (4)$$

Figure 5: These rules define how to rearrange the current tree set $\mathcal{T}$, namely add new nodes and insert new edges for a given relationship ($\texttt{rel}_{\{\parallel,\perp\}}$) between two objects o_1 and o_2. n_{o1} and n_{o2} refer to nodes in $\mathcal{T}$ representing these objects. The ischild−, child−, delete−methods operate on $\mathcal{T}$.



subject 4

Figure 6: Example set of trees $\mathcal{T}$ generates from the description of subject 4 (Fig. 7) using the rules of Fig. 5

In the corner is a lamp. Soft toys are on the table, a rose is on the table, and a car is in front of the koala. A lion is on the chair. A small robot lies in front of the lion. In the left cupboard (cupboard2) are books. Also there are games in the cupboard2. Next to fred is a raven. Below the raven are the pokemon. In the right cupboard (cupboard3) are games. Also a candle is in cupboard3. Above the candle is a dog.

Figure 7: This is the scene description of subject 4. The magenta framed words are the objects, the double green underlined the parallel relations, and the single green underlined the orthogonal relations.

same superordinate structure which would be the table in this example. This function is, in the mathematical sense, commutative as switching the objects' order does not change the superior structure.

- $\texttt{rel}_{\perp}(o_1, o_2)$:
  describes an *orthogonal* relation between two objects o_1 and o_2 where o_1 is assigned to o_2 as superordinate structure in the sense of o_1 lies on/in o_2 (e.g. "there are soft toys on the table"). This function cannot be considered to be commutative as switching the objects' order would result in a completely different superior structure.

As stated in Section 3.1, objects in/on different structures (e.g. the books in the cupboard and the bear on the table) are not related to each other. Therefore, we are going to build a set of dependency trees in which verbally related objects are organized in a hierarchical way, which means that the superordinate structure is a parent node of the subordered elements in the tree. The notation used to represent the trees can be seen in Fig. 3. For a certain object label in a verbal expression, there is a function obj which generates an object o. The verbal expression in Fig. 3 also provides a relation between two objects. The current set of trees is extended or transformed depending on the relation.

Our computational model is based on rules that define the way of how to add new nodes, edges, and trees in a given set of trees $\mathcal{T}$. When starting with the first expression, $\mathcal{T}$ will be an empty set. An expression "o1 is related to o2" is transformed to two objects $o_1 = \texttt{obj}(\text{“o1”})$ and $o_2 = \texttt{obj}(\text{“o2”})$ and a relation $\texttt{rel}_{\{\parallel,\perp\}}(o_1, o_2)$ between them. In general, new isolated nodes n_{o1} (⊢ o1) and n_{o2} (⊢ o2) are inserted in $\mathcal{T}$ representing the mentioned objects. Labels expressing distinct scene objects (e.g. "koala") are added only once into $\mathcal{T}$, while category labels (e.g. "soft toys") will be newly added every time they are mentioned as it cannot be assumed without additional knowledge that the same objects were meant. The rules treat nodes with and without children identically. First, there are three operations on object nodes and the current set of trees $\mathcal{T}$:

- $\texttt{child}(n_{\text{o}}, n_{\text{p}})$:
  inserts a directed edge from node n_{p} known as parent to the child node n_{o}.
- $\texttt{bool} = \texttt{ischild}(n_{\text{o}}, n_{\text{p}})$:
  returns true if $\exists\{n_{\text{o}}, n_{\text{p}}\} \in \mathcal{T}$ with a directed edge between n_{p} and n_{o}.
- $\texttt{delete}(n_{\text{p}})$:
  deletes the node n_{p} from the set of trees $\mathcal{T}$.

The rules for extending $\mathcal{T}$ from a given relation $\texttt{rel}_{\{\parallel,\perp\}}(o_1, o_2)$ are presented in Fig. 4 and 5. They are explained as follows:

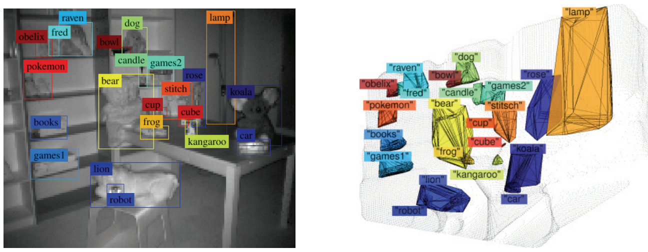
(1) The basic rule for a given parallel relation ($\texttt{rel}_{\parallel}(o_1, o_2)$) between two objects o_1 and o_2 state that there exists an object $p = \texttt{obj}(\text{“ ”})$ with an empty label which will be inserted as new node n_{p} into $\mathcal{T}$. The hierarchical relation between o_1, o_2 and p is expressed via setting the nodes n_{o1} and n_{o2} as child nodes of n_{p} using the child-operation.

(2) In the case of an orthogonal relation ($\texttt{rel}_{\perp}(o_1, o_2)$) between two objects a directed edge will be inserted so that the node n_{o1} will become a child node of n_{o2}.

For both basic cases, there exists an exception which has to be treated by an own rule.

(3) Given a parallel relation and the fact that n_{o1} (node of object o_1) has already a parent node n_{p} in $\mathcal{T}$, the node n_{o2} with its children – if existing – will become a child node of n_{p}.

(4) Assuming o_1 has a parent node n_{p} with an empty label in $\mathcal{T}$ and an orthogonal relation between o_1 and o_2, all child nodes of n_{p} including n_{o1} become child nodes of n_{o2}, n_{p} will be deleted from the set of trees.



(a)

(b)

Figure 8: Swissranger output: (a) Amplitude image recorded from the scene shown in Fig. 2. The boxes and labels around the movable objects represent the output $\mathcal{O}$ of a typical object detector. (b) For each pixel also a 3D point is provided. The convex 3D object hulls are computed on the 3D points determined by the 2D bounding boxes.

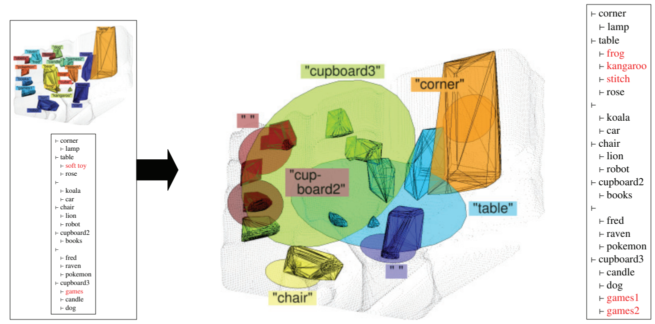


Figure 9: Using the set of trees $\mathcal{T}$ (see Fig. 6) generated from the description of subject 4 and the convex hulls of small objects (see Fig. 8(b)) this initial set of potential planes $\{\mathcal{P}^p_{\text{pot}}\}_{p=1\ldots7}$ can be computed. Also the ambiguous labels of $\mathcal{T}$ are resolved by distinct objects (red marked in the tree figures).

Applying these rules on descriptions of ten native speakers who participated in the study (example see Fig. 7, hand-annotated for objects and relations), dependency tree sets can be generated as presented in Fig. 6 and 11. We assume that a person will provide consistent relations and labels of structures. Otherwise, inconsistent relations are omitted as an issue to be resolved later in the process using sensor data or to be clarified in a human-robot interaction scenario via a query.

4 Extracting 3D Scene Structures

The experimental setup for obtaining human room descriptions was designed in such a way that our mobile robot [Haasch *et al.*, 2004] would be able to use the descriptions to build up a representation of its environment. We aim for a 3D representation, as it resolves depth disambiguities and provides more information for navigation and manipulation tasks. Our robot is equipped with a Swissranger SR3000 [Weingarten *et al.*, 2004], which is a 3D time-of-flight (ToF) near-infrared sensor delivering in real-time a depth map of 176×144 pixels resolution. The advantage of this sensor is that it provides a dense and reliable 3D point cloud of the scene shown in Fig. 8(b) and simultaneously a gray-scale image of amplitude values encoding for each 3D point the amount of infra-red light reflected (see Fig. 8(a)).

In a scene representation consisting of a set of trees as built by applying the computational model of Sec. 3.2 (Fig. 6), the movable objects like soft toys, cups, or books are located at the leaves of the trees while objects like furniture which are a structural part of the room can be found on the higher levels of the trees. This represents the physical constraint that no object is flying in the room but lies on or in a supporting structure. Here, the movable objects are hand-labeled by 2D bounding boxes and object names (see Fig. 8(a)) which is a typical representation provided by object detectors like Lowe's SIFT detector [Lowe, 2004] or the Viola-Jones detector [Viola and Jones, 2001]. Using the 3D ToF data it is even possible to extract automatically 3D convex hulls of these objects (Fig. 8(b)). These object hulls and the spatial relations between them given as a set of trees $\mathcal{T}$ can be used to determine the supporting structures of the leaf objects by assigning them to their parent nodes. The following sections will explain how potential supporting planes are specified and how they are adapted to real sensor data.

4.1 Computing Potential Planar Patches

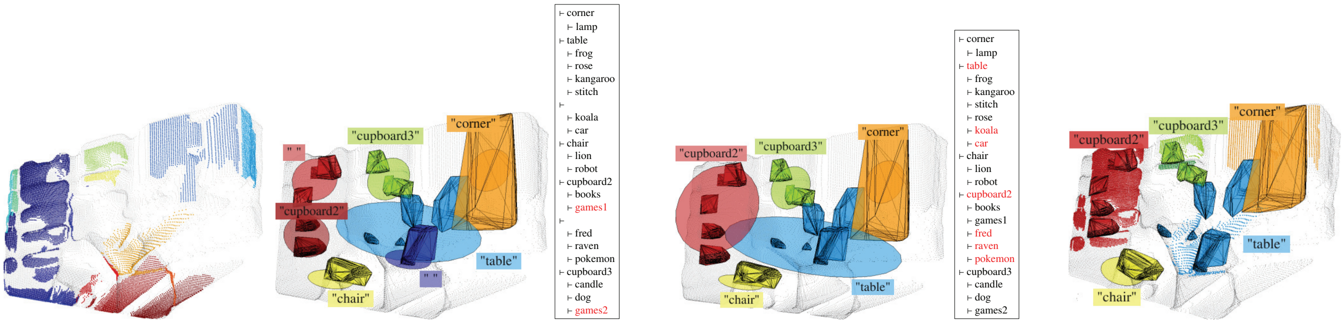
Several papers [Stamos and Allen, 2002; Lakaemper and Latecki, 2006; Swadzba and Wachsmuth, 2008] have shown the suitability of planar patches as meaningful structures for tasks like environment representation, landmarks for navigation, and room categorization. In our case, planar surfaces are the supporting structures for the movable objects. Thus, an intermediate level of scene representation is introduced in the sense that an object lies on or in such a planar patch. It is assumed that all child nodes of a parent node n_{p} in $\mathcal{T}$ belong to the same patch. Therefore, we are going to compute from the child objects *potential* planar patches and assign them to the corresponding parent node in such a way, that these patches represent a meaningful area in the room labeled with the name provided by the parent node.

A parent node may have child nodes with distinct labels (e.g., "koala") and labels referring to a set of objects (e.g., "soft toys"). The main categories and the corresponding objects in the set of known objects $\mathcal{O}$ (see Fig. 8(a)) are:

- toy: car, robot, ...
- decoration: candle, rose, ...
- soft toy: koala, bear, ...
- games: games1, games2

Considering all tree nodes in $\mathcal{T}$ (of e.g., subject 4) the set $\mathcal{O}$ of available objects given by an object detector can be divided into a set of *confirmed* objects $\mathcal{O}_{\text{con}}$ (e.g., "car", "koala", "table") which are part of $\mathcal{T}$ and a set of *potential* objects $\mathcal{O}_{\text{pot}}$ (e.g., "bowl", "cup", "cube") which are not part of $\mathcal{T}$). The goal is to find in $\mathcal{O}_{\text{pot}}$ the correct items the subject had in mind when uttering, e.g., "soft toys". These items have to lie in/on the same spatial structure (here: planar patch) like the confirmed objects of a certain parent node.

Therefore, plane parameters ($\mathcal{P} : \vec{n} \cdot \vec{x} - d = 0$) are computed for each parent node n_p from the set of confirmed objects $\mathcal{O}^p_{\text{con}} \subset \mathcal{O}_{\text{con}}$. If the children are known to be *on* the parent structure, the normal vector $\vec{n}$ is $(0, 1, 0)^{\text{T}}$ as the data was calibrated beforehand such that table and ground plane are parallel to the $xz-$plane. The constant d is determined by that object of $\mathcal{O}^p_{\text{con}}$ having the smallest y-value. If the children are *in* the parent structure this structure is approximated by a vertical plane with its normal $\vec{n}$ is the cross product of $\vec{b}_1 = (0, 1, 0)^{\text{T}}$ and $\vec{b}_2$ obtained by finding the best line via RANSAC through the points of $\mathcal{O}^p_{\text{con}}$ projected onto the $xz-$plane. The centroid of all object points determines d. If nothing is known about the relation of the children to



(a) extracted planar patches (b) correct wrong assignments of objects (c) adapt potential structures to real structures (d) match model on $\{\mathcal{P}^i_{\text{real}}\}_{i=1\ldots m}$

Figure 10: The models in this figure are based on the description given by subject 4. (a) shows the automatically extracted planar patches $\{\mathcal{P}^i_{\text{real}}\}_{i=1\ldots m}$ using a region growing technique. (b) shows the scene model (visualized by planar patches and a set of trees) after correcting wrong assignments of objects in the initial model given in Fig. 9. (c) oversegmentation of structures (e.g., the table) are resolved using real planar patches which results into a rearranged set of trees and adapted planar patches. (d) the scene model of Fig. 10(c) is matched on the set of real planes $\{\mathcal{P}^i_{\text{real}}\}_{i=1\ldots m}$. This results into a subset of meaningful patches for which labels are provided by the model (e.g., "table", "cupboard2", . . .). The labels refer to that 3D point cloud pigmented with the color of the corresponding label.

their parent, the arrangement of the lowest point (regarding the y-value) per object in $\mathcal{O}^p_{\text{con}}$ is considered. A plane is computed through these points and tested whether it is parallel to the $xz-$plane or not. Depending on the result an *in* or *on* relation is assumed.

The computed *potential* plane is exploited to resolve ambiguous child labels. Those objects of $\mathcal{O}_{\text{pot}}$ which are located on/in this plane are assigned to the corresponding parent node. Finally, the region of interest (here ideally assumed as a circle) in each node plane is determined as the smallest circle holding all child objects. Fig. 9 shows a sets of potential planar patches $\{\mathcal{P}^p_{\text{pot}}\}_{p=1\ldots7}$ obtained from the set of dependency trees given in Fig. 6.

4.2 Adaption of Tree Represention and Potential Planar Patches to Real Data

The potential planar patches were derived without any knowledge about real planar patches in the 3D data. As can be seen in Figure 9, there are two main errors. First, objects are misleadingly assigned to a wrong parent while resolving ambiguous labels by distinct objects if different structures are aligned along an infinite plane (e.g., n_{games1} and n_{games2} are assigned to $n_{\text{cupboard3}}$). Secondly, real structures sometimes consist of two or more potential patches as the verbal description did not provide relations between certain objects (e.g., left and right part of "table"). These two problems can be addressed via considering real planar surfaces $\{\mathcal{P}^i_{\text{real}}\}_{i=1\ldots m}$ (see Fig. 10(a)) extracted using a region growing approach based on coplanarity and conormality measurements between 3D points [Stamos and Allen, 2002].

Considering $\{\mathcal{P}^i_{\text{real}}\}_{i=1\ldots m}$ the wrong object assignments can be corrected. For each potential patch $\mathcal{P}^p_{\text{pot}}$, all possible real patches have to be identified. The coplanarity measurement and the Euclidean distance to the center of $\mathcal{P}^p_{\text{pot}}$ are computed for all points of $\mathcal{P}^i_{\text{real}}$. If there is any point of $\mathcal{P}^i_{\text{real}}$ for which both values are below a certain threshold then this patch is related to the current potential patch. Then, all objects of $\mathcal{P}^p_{\text{pot}}$ are tested whether they lie in/on one of the assigned real planes. Those not assigned to a real plane are removed. Afterwards, if a real plane $\mathcal{P}^i_{\text{real}}$ is assigned to different potential patches with different labels (not considering empty labels " ") it can be concluded that some of the objects are mismatched. $\mathcal{P}^i_{\text{real}}$ will be put to that potential patch holding the biggest percentage of objects lying in $\mathcal{P}^i_{\text{real}}$ ($\mathcal{P}^p_{\text{pot}}$) and all objects lying in $\mathcal{P}^i_{\text{real}}$ are assigned to $\mathcal{P}^p_{\text{pot}}$. Fig. 10(b) shows a corrected object assignment of Fig. 9. The node n_{games1} is now assigned to "cupboard2".

After correcting mismatched objects and recomputing potential patches, the real planes can be used to establish new relations. In Figure 10(b), it can be seen that the objects on the table are grouped into two sets one labeled as "table" and one as " ". Originally, the subject did not provide a relation which indicated to fuse these two sets. Obviously, a human would conclude that both sets have the same supporting structure which would be the table. In our framework such inferences can be done based on real planes in our scene. All potential patches pointing to the same real plane will be merged to one patch and their objects will be assigned to the new parent node, if at most one label is not empty. Unless there exists a non-empty label, it will be assigned to the new patch (see Figure 10(c)) and the corresponding real plane (Figure 10(d)).

4.3 Results

The planar surfaces in Fig. 10 and their labels obtained from human descriptions convincingly meet the expected groundtruth, as meaningful structural elements were chosen and the correct labels were provided. In contrast to "cupboard2", it can be seen that two planes (colored in different greens) are annotated with "cupboard3" as this furniture consists of several patches in the data. No patch is found for the label "chair" (only the potential patch is displayed) because in the current data the chair is hidden completely by its objects on top. However, our algorithm would be able to find it in subsequent data, when the objects are removed. In this case, a reliable chair patch would be extracted and labeled correctly, since a model representation of the static scene layout exists (see Figure 10(c) for potential patches and their annotations).

The developed computational model (Sec. 3.2) is applied to the verbal expression of all ten subjects participating in our experiment. Fig. 11 shows the sets of trees generated from the given descriptions. Six of the ten participants described the scene quite detailed pointing to each object separately. The

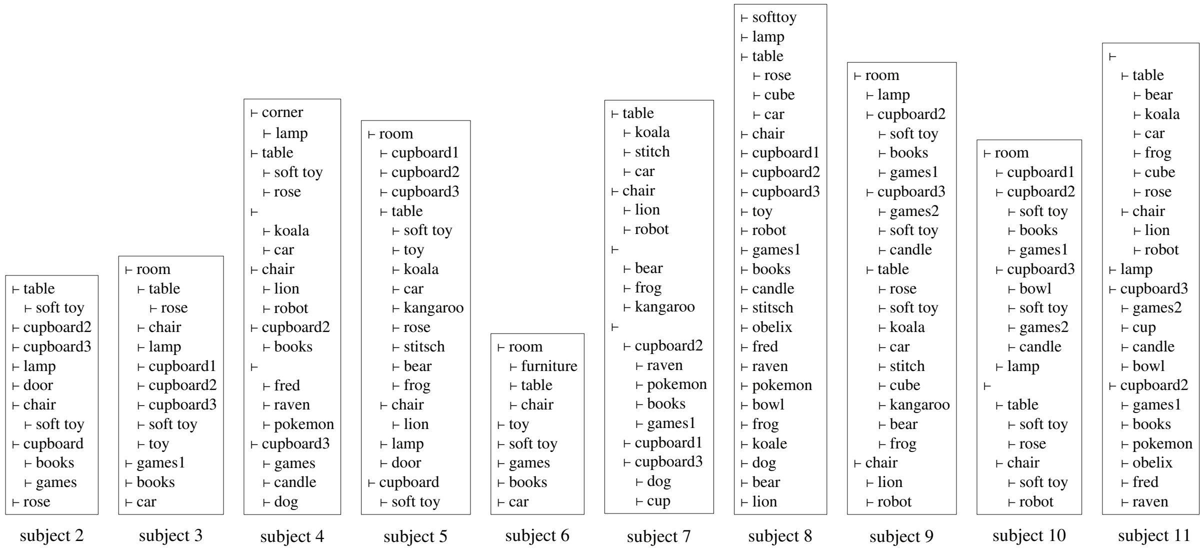


Figure 11: For all descriptions collected in our study a set of dependency trees is generated using the computational model proposed in Sec. 3.2.

remaining subjects grouped the movable objects by their categories or picked some representative examples. Two persons provided almost no structural relations, they simply itemized the things they saw.

Apart from the default structures given by the fact that each (movable) object defines a patch where it lies on, Fig. 12 presents for each subject the additional structures learnt from the verbal descriptions. Fig. 12(k) gives an overview of how often each structure (here: "table", "chair", "cupboard2", "cupboard3", "cupboard", and "corner") was generated. In eight of the ten cases the "table"-structure and in six of ten cases the "chair"-structure was inferred from the provided relations. This fact supports the impression that these two structures had a prominent role in the given scenario. In one case (subject 6) no structures could be learnt as only a rough scene description was delivered with almost no relations between objects. In three cases potential patches could not be computed as the subjects provided ambiguous information which could not be resolved. In most cases they said "There are soft toys on/in ..." without specifying the objects more detailed like "namely a koala, bear ...". Our system needs at least one specific object from which it can gather a position and orientation of the potential patch. Then it can solve such ambiguities.

5 Conclusion

In this paper, we propose a computational model for arranging objects into a set of dependency trees via spatial relations given by human descriptions. It is assumed that objects are arranged in a hierarchical manner. The method predicts intermediate structures which *support* other object structures as expressed in "soft toys lie on the table". The objects at the leaves of the trees are assumed to be known and used to compute potential planar patches for their parent nodes leading to a model of the scene. Finally, these patches are adapted to real planar surfaces correcting wrong object assignments and introducing new object relations which were not given in the verbal descriptions, explicitly. Results show that our approach provides reliable scene models which match meaningful labels to planar surfaces in the real 3D world and supports the empirical hypotheses about a hierarchical spatial model.

So far, we have used the planar model for supporting structures. The planar model holds in the case of "something lies *on* a structure", but generalizes only partly for "something lies *in* a structure". In future work our approach will be extended by using different models and degrees of shape abstraction to handle the in-relations comprehensively. Further, it would be interesting to learn the degrees of freedom of the spatial arrangements in the obtained model.

6 Acknowledgement

This work was partially funded by the German Research Foundation within the Collaborative Research Center CRC673 "Alignment in Communication".

References

P. Beeson, M. Macmahon, J. Modayil, A. Murarka, B. Kuipers, and B. Stankiewicz. Integrating multiple representations of spatial knowledge for mapping, navigation, and communication. In *AAAI Spring Symposium on Control Mechanisms for Spatial Knowledge Processing in Cognitive / Intelligent Systems*, 2007.

H.-J. Böhme, T. Wilhelm, J. Key, C. Schauer, C. Schröter, H.-M. Groß, and T. Hempel. An approach to multi-modal human-machine interaction for intelligent service robots. In *Robotics and Autonomous Systems*, 2003.

M. Brenner, N. Hawes, J. Kelleher, and J. Wyatt. Mediating between qualitative and quantitative representations for task-orientated human-robot interaction. In *International Joint Conference on Artificial Intelligence*, 2007.

A. Haasch, S. Hohenner, S. Hüwel, M. Kleinehagenbrock, S. Lang, I. Toptsis, G. A. Fink, J. Fritsch, B. Wrede, and G. Sagerer. BIRON – The Bielefeld Robot Companion. In *International Workshop on Advances in Service Robotics*, 2004.

J.M. Henderson and F. Ferreira. Scene perception for psycholinguists. *The Interface of Language, Vision, and Action: Eye movements and the visual world*, pages 1–58, 2004.

J. M. Henderson and A. Hollingworth. High-level scene perception. *Annual Review of Psychology*, 50:243–271, 1999.

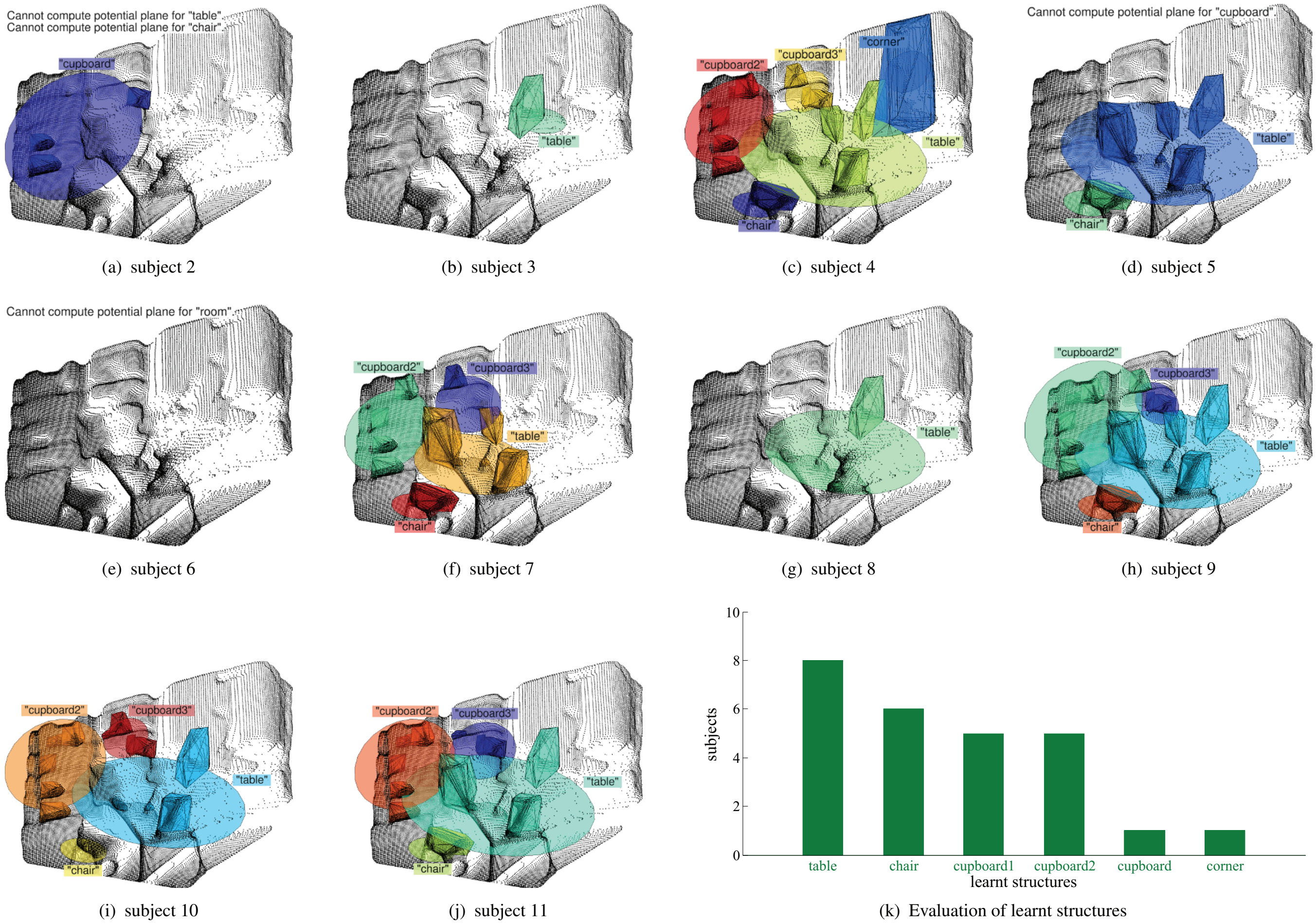


Figure 12: This figure shows for each subject the learnt potential structures ((a) – (j)) using the initial set of trees of Fig. 11 and adapting them to real sensor data. (k) presents a numerical evaluation of the generated structures. For each structure it is checked how often this structure has been deduced.

J. Hois, M. Wünstel, J. A. Bateman, and T. Röfer. Dialog-based 3D-image recognition using a domain ontology. In *Spatial Cognition V: Reasoning, Action, Interaction*, number 4387 in Lecture Notes in Artificial Intelligence, pages 107–126. 2006.

G. Kim, W. Chung, S. Han, K. Kim, M. Kim, and R. H. Shinn. The autonomous tour-guide robot jinny. In *International Conference on Intelligent Robots and Systems*, 2004.

R. Lakaemper and L. J. Latecki. Using extended em to segment planar structures in 3D. In *International Conference on Pattern Recognition*, pages 1077–1082, 2006.

S. Li, A. Haasch, B.Wrede, J. Fritsch, and G. Sagerer. Human-style interaction with a robot for cooperative learning of scene objects. In *International Conference on Multimodal Interfaces*, 2005.

D. G. Lowe. Distinctive image features from scale-invariant keypoints. *International Journal on Computer Vision*, 60:91–110, 2004.

N. Mavridis and D. Roy. Grounded situation models for robots: Where words and percepts meet. In *International Conference on Intelligent Robots and Systems*, 2006.

M. Montemerlo, J. Pineau, N. Roy, S. Thrun, and V. Verma. Experiences with a mobile robotic guide for the elderly. In *National Conference on Artificial Intelligence*, 2002.

O. M. Mozos, R. Triebel, P. Jensfelt, A. Rottmann, and W. Burgard. Supervised semantic labeling of places using information extracted from laser and vision sensor data. *Robotics and Autonomous Systems Journal*, 55(5):391–402, 2007.

R. Simmons, A. Bruce, D. Goldberg, A. Goode, M. Montemerlo, N. Roy, B. Sellner, C. Urmson, A. Schultz, W. Adams, M. Bugajska, M. MacMahon, J. Mink, D. Perzanowski, S. Rosenthal, S. Thomas, I. Horswill, R. Zubek, D. Kortenkamp, B. Wolfe, T. Milam, and B. Maxwell. Grace and george: Autonomous robots for the AAAI robot challenge. In *AAAI Mobile Robot Competition 2003: Papers from the AAAI Workshop*, 2003.

I. Stamos and P. K. Allen. Geometry and texture recovery of scenes of large scale. *Computer Vision and Image Understanding*, 88(2):94–118, 2002.

A. Swadzba and S. Wachsmuth. Categorizing perceptions of indoor rooms using 3D features. In *International Workshop on Structural and Syntactic Pattern Recognition and Statistical Pattern Recognition*, pages 744–754, 2008.

N. Tomatis, R. Philippsen, B. Jensen, K.O. Arras, G. Terrien, R. Piguet, and R. Siegwart. Building a fully autonomous tour guide robot: Where academic research meets industry. In *International Symposium on Robotics*, 2002.

A. Torralba, K. Murphy, W. Freeman, and M. Rubin. Context-based vision system for place and object recognition. In *International Conference on Computer Vision*, 2003.

P. Viola and M. Jones. Robust real-time object detection. In *International Journal on Computer Vision*, 2001.

S. Wachsmuth and G. Sagerer. Bayesian networks for speech and image integration. In *National Conference on Artificial Intelligence*, pages 300–306, 2002.

J. Weingarten, G. Gruener, and R. Siegwart. A state-of-the-art 3D sensor for robot navigation. In *International Conference on Intelligent Robots and Systems*, 2004.

H. Zender, Ó. M. Mozos, P. Jensfelt, G.-J. M. Kruijff, and W. Burgard. Conceptual spatial representations for indoor mobile robots. *Robotics and Autonomous Systems, Special Issue "From Sensors to Human Spatial Concepts"*, 56(6):493–502, 2008.