

Learning Organizational Principles in Human Environments

Martin J. Schuster, Dominik Jain, Moritz Tenorth and Michael Beetz
Technische Universität München
martin.schuster@mytum.de, {jain,tenorth,beetz}@cs.tum.edu

Abstract—In the context of robotic assistants in human everyday environments, pick and place tasks are beginning to be competently solved at the technical level. The question of where to place objects or where to pick them up from, among other higher-level reasoning tasks, is therefore gaining practical relevance. In this work, we consider the problem of identifying the *organizational structure* within an environment, i.e. the problem of determining organizational principles that would allow a robot to infer where to best place a particular, previously unseen object or where to reasonably search for a particular type of object given past observations about the allocation of objects to locations in the environment. This problem can be reasonably formulated as a classification task. We claim that organizational principles are governed by the notion of similarity and provide an empirical analysis of the importance of various features in datasets describing the organizational structure of kitchens. For the aforementioned classification tasks, we compare standard classification methods, reaching average accuracies of at least 79% in all scenarios. We thereby show that, in particular, ontology-based similarity measures are well-suited as highly discriminative features. We demonstrate the use of learned models of organizational principles in a kitchen environment on a real robot system, where the robot identifies a newly acquired item, determines a suitable location and then stores the item accordingly.

I. INTRODUCTION

As many of the more fundamental problems in robotics (e.g. with regard to perception, navigation and motion planning) are being solved to a degree where we can consider the respective components to be sufficiently reliable to form the basis for complex tasks, the high-level reasoning capabilities of robots will become increasingly important in the years to come. In view of aging societies in many countries, the field of service robotics in general and the field of robotic household assistants in particular demands our attention. For competent behaviour in human everyday environments to become a reality, a deep understanding of human environments and the interactions between entities within them is a necessary precondition. Robots will have to reason about everyday environments, the containers and objects these are likely to contain, the properties and functions of objects and their relationships to tasks and other objects in order to achieve competent problem-solving behaviour. A multitude of research questions arise, out of which only a tiny subset has yet been addressed. With pick and place tasks being addressed by many researchers at the technical level, the question of where to place objects or where to pick them up from, among other higher-level reasoning tasks, gains practical relevance in robotic applications.

In this work, we consider the problem of identifying

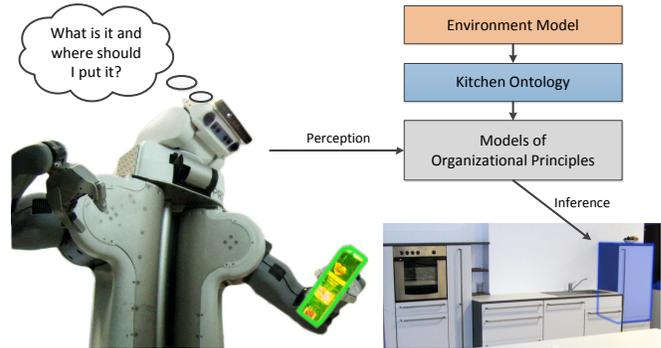


Fig. 1. Application of models representing organizational principles on a robot system: The robot acquires models given observations of the objects found within its environment. In order to determine a suitable storage location for a given object, it identifies the class to which the object belongs and performs inference over the model using, in particular, features pertaining to the similarity between the object and the other objects already stored in the environment.

the *organizational structure* within an environment, i.e. the problem of determining organizational principles that would allow a robot to infer where to best place a particular, previously unseen object or where to reasonably search for a particular type of object given knowledge about the object type and past observations about the allocation of objects to locations in the environment. In particular, we consider kitchen environments in which various utensils, food preparation devices, foodstuffs and food ingredients are typically assigned to storage locations such as cupboards, drawers, refrigerators and working surfaces. Given a set of previously observed objects with their associated storage locations, the robot is to acquire models that would allow it to, for instance, reasonably allocate each of the objects it might find within a shopping bag to appropriate storage locations (see Figure 1). As far as the granularity at which we solve the problem is concerned, we distinguish only between the various storage locations but not the placement within these storage locations. Furthermore, we are concerned only with capturing the actual storage locations of objects, not the task-dependent locations of objects that might change over time as actions are carried out.

We believe that for an organizational principle to be discernable (to robots or humans) at any given location, the objects placed at the location must share certain characteristics, i.e. the entropy of the random variable describing the distribution of object properties – in some attribute space – should ideally be low. In order to identify an organizational principle, we must either make direct use

of a suitable attribute space or consider an aggregate in the form of a similarity measure. At the global level, an organizational structure will be identifiable if the attribute subspaces pertaining to any two locations are sufficiently different from each other.

As key contributions, we present

- the selection of suitable attributes and similarity measures derived from an ontological knowledge base, which we partly build up based on web resources;
- an analysis of the performance of various modelling and classification schemes with regard to the aforementioned allocation problem;
- an analysis of the degree to which organizational principles can be identified at the locations within the kitchens we considered, and the degree to which there is a global structure that sets apart the locations;
- an open source implementation of our algorithm to solve the object allocation problem, integrated into the KNOWROB [1] knowledge processing system.

II. RELATED WORK

There is a large body of related work that considers organizational principles at a fairly coarse level, seeking to exploit knowledge about object-room associations for the purpose of room classification in the context of mapping and navigation [2] or, inversely, visual search for objects given the classes of rooms within a map [3], or both [4, 5]. Object-room associations are typically described using logical knowledge, e.g. represented in a description logics knowledge base [3, 6], and/or using conditional probability distributions [4, 5, 7]. The problem we consider is qualitatively different, because we do not assume that there is a given, globally applicable model which captures the structure of an arbitrary environment, regardless of, for example, the personalities and preferences of the environment’s inhabitants. Instead, we are interested in recovering principles based on observations of a single environment in order to generate a model that reflects its idiosyncrasies, capturing the unique characteristics of each place instead of assigning concepts to places. Moreover, we disregard the types of containers/rooms and therefore do not establish object-container relationships but rather object-object relationships that induce local clusterings. While [8] also makes use of object-object relations, the approach considers strictly spatial relations and therefore cannot infer locations for objects that are not yet found in the environment. Notably, relations are not considered for the problem of identifying the most probable location concept for a single object of interest. Moreover, the naive Bayesian classifier used in [8] can only reasonably operate on object types that have been observed in the training set. In contrast, we apply semantic object-object relations in order to determine the location of objects of types that have not previously been observed. These semantic similarities can be seen as an abstraction from the object types and thereby achieve a generalization that allows us to apply our model to object types that were not present in the training data.



Fig. 2. Examples for groups of objects found in several kitchens

III. ORGANIZATIONAL PRINCIPLES

To acquire a notion of what might constitute an organizational principle in real-world kitchen environments, we analyzed photographs of kitchens (see e.g. Figure 2) as well as blogs¹ and videos from the Internet. Focusing on the questions of where objects are located, which objects are grouped together and why, our analysis of this data led to the following prevalent organizational principles:

- *Class*: Most places contain objects that belong to similar classes as they might appear in a taxonomy. For example, there is often a distinction between food and non-food items. More specifically, most people store, for example, prepared food, ingredients, spices, dishes, cutlery or kitchen utensils at separate locations (see Figure 2).
- *Physical Constraints*: Objects are often placed with respect to constraints imposed by their physical properties. For example, large items can obviously be placed only at locations that provide sufficient space, perishable items are stored in a fridge or freezer, objects that can easily be stacked (e.g. different kinds of plates, see Figure 2) are placed on top of each other.
- *Purpose*: Objects are often grouped according to the purpose they serve. For example, sugar and coffee beans are used to make coffee and therefore may be placed close together. Similarly, all ingredients used for baking are often found in the same place. In Figure 2, objects needed to prepare hot drinks are grouped together (e.g. teabags, coffee and coffee filters).

We also discovered the following additional organizational principles but found them to be less relevant:

- *Packaging*: Oftentimes, large packs of products not intended for daily use are kept in stock, located at different places than single products intended for immediate use. For instance, a single bottle of beer may be kept in the fridge while a crate may be stored elsewhere.

¹e.g. <http://cakescraps.wordpress.com/2010/01/02/organizing-your-fridge/> and <http://www.beruly.com/?p=279>

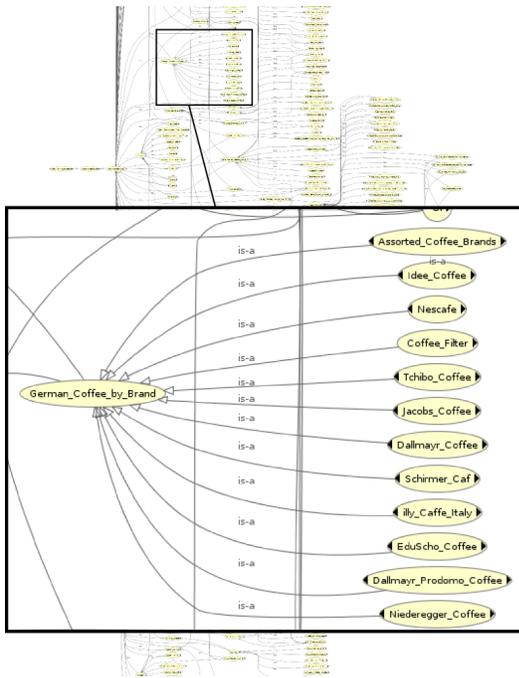


Fig. 3. Excerpt of the ontology generated from the category structure of the *germandeli.com* shopping website

- *Safety*: Some people place items in lower compartments because they might cause injury as a result of falling from a high location. Similarly, food items are not usually stored together with items that could spoil them.

The principles pertaining to class, physical constraints and packaging were also considered to be relevant for object placement in grocery stores [9].

All the above criteria can be reasonably translated into similarity measures between pairs of objects.

IV. ONTOLOGIES

A robot that is to learn about the structure of its environment and the objects located therein requires large amounts of background knowledge about all the objects it could potentially encounter. Such background knowledge can be conveniently represented within an ontology containing concepts (classes) for all the relevant types of objects as well as their attributes and relations.

Specifically, we extended the KNOWROB ontology [1], which uses the Web Ontology Language (OWL) as a representation formalism and incorporates parts of the OpenCyc upper ontology [10]. KNOWROB extends the OpenCyc ontology with more detailed knowledge about the household domain. For the task at hand, we require the ontology to contain information about a multitude of manufactured products found in kitchen environments – particularly foodstuffs. Manually encoding such knowledge for hundreds of objects is a tedious task, but it can be automated: Online shops provide very similar information, though not represented as an ontology. We thus implemented a system that automatically translates the category structure of a shopping website, in our example *germandeli.com*, into a class taxonomy in the knowledge base: For example, *Dallmayr Prodomo Coffee*

is represented as a sub-class of *Dallmayr coffee*, *Coffee (German Brands)*, *Beverages*, and finally *Groceries*. Using only the *germandeli.com* website, we generated an ontology that extends the KNOWROB ontology with knowledge about more than 7,000 manufactured products (see Figure 3 for a small excerpt). The translation engine and the generated ontology are publicly available as open-source software.²

In addition to the category structure, online shops also provide detailed descriptions of the properties of products, such as the perishability status, price, ingredients, etc. This information is usually presented in a semi-structured way in the form of tables or symbols, which can automatically be translated into attributes of the respective object classes. The product images found on the product page are used to construct recognition models that allow the robot to both detect these objects and to reason about their properties [11].

V. DATASETS

We gathered data about the organization within twelve different kitchen environments. Ten of these were acquired by simulating the process of placing objects within a fictitious kitchen, two were obtained by carefully annotating the object locations in two real kitchens. We divided each kitchen environment into *locations*, a location representing a container or tabletop, e.g. a cupboard, a drawer, the fridge, etc.

A. Kitchen Mockups

From our extended KNOWROB ontology, we selected 66 different concepts, to define an exemplary kitchen inventory with a total of 152 objects (instances of concepts). We printed each concept including a product image and the number of its instances on a small piece of paper. We also printed two different sketches of kitchen layouts on large sheets of paper, marking the available containers (cupboards, drawers, fridge) with numbers indicating locations where objects may be placed. The first kitchen mockup had twelve different locations, the second six. We then asked ten persons, five for each of the two layouts, to place the 66 pieces of paper representing the 152 objects at the different locations, grouping them together as if they were to establish an order in their own new kitchen. One example from this dataset can be seen in Figure 4.

We then annotated the mapping of products to locations in a database that we used for our evaluation (see Section VII).

²http://code.in.tum.de/pubsvn/knownrob/tags/latest/comp_germandeli

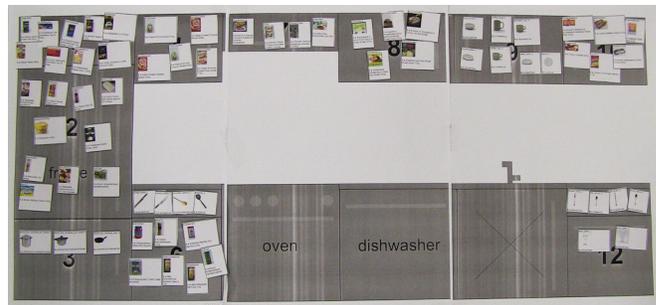


Fig. 4. Kitchen mockup with objects placed at twelve different locations

The layout itself served only as a visual aid for our test subjects, potentially improving their impression of acting in a real kitchen and therefore the quality of the data. We did not use information about the proximity of different places or the proximity of places to devices like the oven or sink, although this information might give further clues for reasonable object placement and could constitute an interesting subject for future research.

B. Real Kitchens

We gathered two additional datasets from real kitchens, where we manually annotated all objects along with the location at which they were placed. We then added any missing product classes to our kitchen ontology. Our real kitchen datasets D_{r1} and D_{r2} contain 166 and 87 different classes with totals of 408 and 149 objects placed at 19 and 15 different locations respectively. In the real kitchens, not all objects belonging to a single class were placed at the same location, because the owners distinguished between different states of objects, e.g. partially used products and new products which do not require cooling. In our dataset, we did not consider this distinction, moving all objects of the same class to a single location for practical evaluation reasons (less than seven objects relocated in each dataset); it is not a general restriction of our approach.

VI. LEARNING ORGANIZATIONAL PRINCIPLES

Having identified the important principles in Section III, we want to learn a model that allows us to solve the classification task of choosing the best location for a previously unseen object in the kitchen. We additionally want to gain insights into the relevant organizational principles at each location in a kitchen by analyzing feature importance.

A. Features

As outlined above, we believe that organizational principles are governed by the notion of similarity. Similarity, however, can be defined in manifold ways: We could consider the similarity along any dimension, including the size, shape, weight, colour, value, fitness for a particular purpose, etc. of the object as features that would allow to identify the organizational principles that are characteristic for a specific location. Of course, we can also consider aggregates of the aforementioned similarities that consider an arbitrary number of dimensions at the same time.

In our experiments, we consider the following features, all of which correspond to principles identified in Section III:

- *WUP similarity*: a semantic degree of similarity between concepts in an ontology (where concepts in the ontology correspond to types of objects); it gives an indication of how similar the types of objects are.
- *Purpose*: what the object can be used for, as defined through super-concepts in our ontology to which the type of the object belongs (four binary features indicating whether the object is a *FoodVessel*, *PhysicalDevice*, *FoodOrDrink* or *FoodIngredient*)

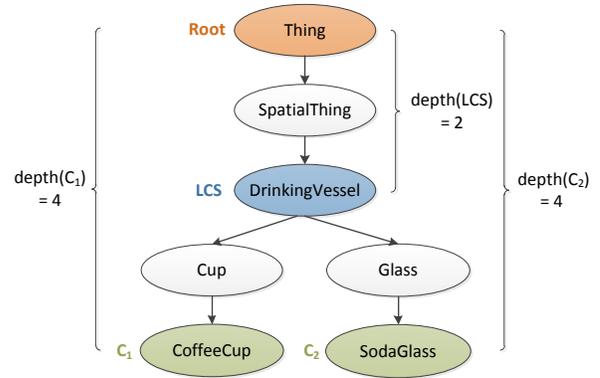


Fig. 5. Example for path lengths used to calculate the WUP-Similarity

- *MealRelevance*: five binary features indicating whether the type of object is typically used for *Breakfast*, *PrincipalMeal*, *Coffeebreak*, *Snack*, *Sandwich*
- *Size*: discretized size of the object according to its largest dimension, $size \in \{s, m, l\}$
- *Shape*: discrete values for the shape of the object, $shape \in \{\text{box, cylindrical, flat, bag, other}\}$

The WUP similarity is one particularly versatile similarity measure. It was originally defined by Wu and Palmer in [12] in the context of automatic translations. For two concepts in an ontology, it defines a similarity value in the interval $[0; 1]$, taking the depth of the concepts and the depth of their lowest common super-concept (LCS) into account:

$$wupSim(C_1, C_2) = \frac{\text{depth}(\text{LCS}(C_1, C_2))}{\frac{1}{2}(\text{depth}(C_1) + \text{depth}(C_2))} \quad (1)$$

The reflexive case is defined as $wupSim(C, C) = 1$. The computation is illustrated in Figure 5, showing a simplified version of our kitchen ontology with $wupSim(\text{CoffeeCup}, \text{SodaGlass}) = 0.5$.

The WUP similarity’s versatility is due to the flexible ways in which we can define concepts in our ontology. Concepts can have multiple super-concepts, creating different subtrees in the ontology that correspond to distinct aspects of a particular type of object. For example, a refrigerator can be seen as an electrical household appliance, a cooling device, or just as a box-shaped container. Depending on the situation at hand, each of these views may be more or less relevant. The different sub-trees in the ontology lead to multiple connections between two concepts, each having possibly different lengths that correspond to the semantic distance in that particular respect. The WUP similarity typically computes the minimum of all these distances because it considers the lowest common super-concept (LCS).

The structure of the ontology thus influences the computed similarity values. Note that the ontologies we used were not specifically designed for the computation of similarities. Rather, we directly extended an existing ontology with classes that were automatically derived from an online shop’s website, as described above. Since the distances computed from this ontology appear to be meaningful to humans, it seems to be close to a “natural ontology” of household objects.

We visualize WUP similarities in Figure 6, where we show an excerpt of a graph containing a small subset of the concepts from one of our real kitchen datasets. We defined a distance measure $M_{ij} := 1 - wupSim(C_i, C_j)$, calculating the distances between all pairs of concept and using multi-dimensional scaling to visualize the distance matrix M in two dimensions. Concepts located close to each other in the graph have high pairwise similarity (low distance). Each shape/color indicates a different location where objects of the corresponding concepts are stored in our dataset. We observe that most concepts found at the same location are located in clusters separate from the others, which indicates high discriminative power of the WUP similarity for our classification task.

We refer to the group of products located at one place as a *location*. Based on the WUP similarities between pairs of objects, we define two similarity measures between a single object O and a location L as follows:

$$\max Wup(O, L) = \max_{O' \in L} wupSim(class(O), class(O')) \quad (2)$$

$$\text{avg} Wup(O, L) = \frac{\sum_{O' \in L} wupSim(class(O), class(O'))}{|L|} \quad (3)$$

B. Classifiers

We now describe the classifiers we tested for the task of allocating an object to its most appropriate storage location. For our classifiers, we use one *avgWup* and one *maxWup* feature for each location, unless noted otherwise.

1) *Maximum WUP Similarity*: The first set of primitive classifiers return the location with the maximum WUP similarity between the object and the location. We define two classifiers, one using the maximum *avgWup*, the other using the maximum *maxWup* similarity. The latter is equivalent to

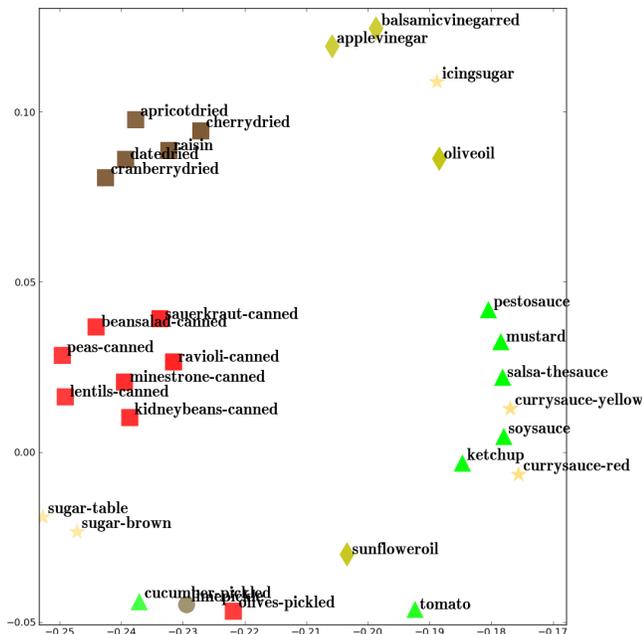


Fig. 6. Visualization of pairwise distances between concepts, based on the WUP similarity. Each shape/color indicates a location in the kitchen.

returning the location that contains the most similar object with respect to WUP similarity.

2) *Decision Trees*: We applied unpruned C4.5 decision trees with the discrete and continuous features described in section VI-A. We use the implementation from the Weka machine learning suite [13] called J48.

3) *Boosted Decision Trees*: We applied AdaBoost (Weka implementation called AdaBoostM1) with pruned C4.5 decision trees (J48), using the Weka default parameters.

4) *Support Vector Machine (SVM)*: We used the Weka SVM implementation called SMO with polynomial kernels and the default parameters ($C = 1$).

5) *Naive Bayes*: In a naive Bayesian classifier, the WUP similarities, as continuous features, can be treated in various ways. The authors of [14] compare different methods of handling continuous variables, including discretization and the approximation by a normal distribution. They conclude that no method systematically outperforms the others and propose a strategy to automatically select the best one. [15] presents an empirical study on the influence of dependencies on the error of naive Bayesian classifiers and conclude that they work best not only on completely independent features but also on functionally dependent features. We therefore compare three variants in our tests:

- *NB discrete*: We used k -means clustering to discretize each of the continuous features with $k = 5$.
- *NB continuous*: We approximate the distribution of each continuous feature with a Gaussian, increasing the variance by 0.05 to avoid overfitting.
- *NB soft*: We treat the degree of similarity as a degree of belief, using a boolean variable for each continuous feature and applying soft evidential updates (analogous to [16]) to compute posterior probabilities.

An interesting topic for future work would be to test kernel methods instead of a single Gaussian to approximate the distribution of the continuous features, for example the nonparametric kernel density estimation presented in [17].

When interpreting similarity values as degrees of belief, we intuitively want to have the minimum similarity correspond to “not similar”, the maximum similarity to “similar” and values in between to “similar to some degree”. Most WUP similarities in our dataset are approximately between 0.4 and 0.8, therefore normalization is desirable. We normalize the WUP similarity features for the soft and continuous naive Bayesian classifiers to the $[0; 1]$ interval by applying a linear scaling on the values occurring in the training data. Values in the test data are scaled in the same way and clipped to the $[0; 1]$ interval in case they exceed its limits.

C. Organizational Principles: Feature Importance Measure

In addition to the classification task, we analyze the degree to which features are capable of defining organizational principles at each particular location, fostering an intuitive understanding of the principles (implicitly) represented in a classifier. To this end, the conditional distributions of the attributes given a location, as represented in a naive Bayesian classifier trained on the dataset D , can be used. Within a

TABLE I
RESULTS: ACCURACY FOR ALL CLASSIFIERS ON BOTH DATASETS

	mockup kitchens				real kitchens					
	avgWup and maxWup		all features		avgWup and maxWup			all features		
	mean	std	mean	std	D_{r1}	D_{r2}	mean	D_{r1}	D_{r2}	mean
max. avgWup	77.45%	20.85%	—	—	48.19%	70.24%	59.22%	—	—	—
max. maxWup	87.52%	17.91%	—	—	72.29%	71.43%	71.86%	—	—	—
DecisionTrees	86.61%	12.46%	88.12%	14.16%	84.94%	73.81%	79.37%	79.52%	69.05%	74.28%
Boosted DecisionTrees	87.68%	13.95%	89.50%	9.92%	84.94%	71.43%	78.18%	80.72%	69.05%	74.89%
SVM	77.46%	24.11%	89.49%	17.68%	57.23%	69.05%	63.14%	73.49%	76.19%	74.84%
NB Discrete	78.37%	15.64%	85.69%	15.56%	50.00%	50.00%	50.00%	57.83%	64.29%	61.06%
NB Continuous	69.97%	28.63%	82.61%	17.75%	41.57%	58.33%	49.95%	60.24%	63.10%	61.67%
NB Soft	42.16%	39.38%	82.32%	18.73%	13.86%	50.00%	31.93%	65.66%	59.52%	62.59%

given location L , there is certainly structure with respect to a particular feature F if the conditional distribution of the feature given L exhibits little entropy. Thus, the degree to which F defines an organizational principle at L can be computed as the inverse normalized Shannon entropy,

$$I_F^D(L) := 1 - \frac{\sum_{f \in \text{dom}(F)} P_D(F=f|L) \log(P_D(F=f|L))}{\log(|\text{dom}(F)|)} \quad (4)$$

where $\text{dom}(F)$ is the domain of F , the numerator is the entropy of the distribution over $\text{dom}(F)$ and the denominator is the maximum possible entropy (uniform distribution over F 's domain). We thus obtain an importance value in the interval $[0, 1]$, with low values (high relative entropy) representing low importance and high values (low relative entropy) high importance.

In order to analyze the discriminative power of the various aggregated features, we compute the Hellinger distance [18] $H_F^D \in [0, 1]$ between the distributions of a feature F given the locations L_1 and L_2 as follows

$$H_F^D(L_1, L_2) = \sqrt{1 - \sum_{f \in \text{dom}(F)} \sqrt{P_D(F=f|L_1)P_D(F=f|L_2)}} \quad (5)$$

As a measure of distribution dissimilarity, the Hellinger distance is an adequate indicator for feature relevance [19]. We average Hellinger distances across all pairs of locations (from the dataset's set of locations \mathcal{L}_D) and define

$$\bar{H}^D(F) := \binom{|\mathcal{L}_D|}{2}^{-1} \sum_{L_i \in \mathcal{L}_D} \sum_{L_j \in \mathcal{L}_D, i < j} H_F^D(L_i, L_j) \quad (6)$$

VII. EVALUATION

We tested our classifiers using two different sets of features on both of our datasets. We first describe our experimental setup and then present and discuss the results.

A. Experimental Setup

We performed experiments for the datasets presented in Section V using the eight classifiers defined in Section VI-B. For each classifier and dataset, we performed two experiments, one using a feature vector containing just the *maxWup* and *avgWup* features and one using a feature vector containing all the features in an effort to determine the power of the WUP similarity when applied to various types of classifiers. In each experiment, we performed leave-one-out cross-validation, i.e. for each class of objects, we removed all of the objects belonging to the class from the kitchen for training and

used our classifiers to infer the location at which the object should be stored. A classification result is considered correct if the predicted location is the one at which the object was originally located. If a location contained only objects of a single class, we skipped this class during leave-one-out crossvalidation as our classifiers would have no training samples for that particular location.

B. Results and Discussion

We list the mean and standard deviation of the percentage of correctly classified objects (accuracy) for our dataset of mockup kitchens and the accuracy values and their mean for our real kitchen datasets in Table I.

From these results, we conclude that the features based on WUP similarity are, indeed, highly discriminative. Using the maximum *maxWup* similarity alone yields an accuracy of 88% and 72% for the mock-up and real kitchens respectively. This coincides with the human intuition that placing an object at the location where the most similar object is located would be a reasonable strategy.

In our dataset containing the ten kitchen mockups, adding additional features and using more sophisticated classifiers like SVMs, (boosted) decision trees and naive Bayes yields only a small improvement, if any, with boosted decision trees using all features yielding the best results (90% accuracy).

In the more complex scenario of the real kitchen dataset, however, (boosted) decision trees and SVMs are able to improve upon the simple *maxWup* classifier by up to 8%. It is interesting to note that we get the best results with decision trees using only the WUP similarities. The semantic similarity measure is sufficiently powerful to obtain a correct classification rate of 79% in a real kitchen. In many ways, the real-world environment is a more complex scenario, because object placement may be influenced by additional spatial restrictions and convenience considerations (e.g. highly similar objects are not always placed at a single location because no container is large enough to hold them all). Such constraints were not considered in our mockup experiments.

It is important to realize that a classification rate close to 100% may not be possible in practice. First, a limited subset of objects in real-world environments may indeed have been placed arbitrarily (for reasons such as lack of time or laziness), inducing noise in the data. The underlying principles would therefore be neither possible nor desirable for us to model. Second, several objects of one class could reasonably be placed at more than one location, yet our

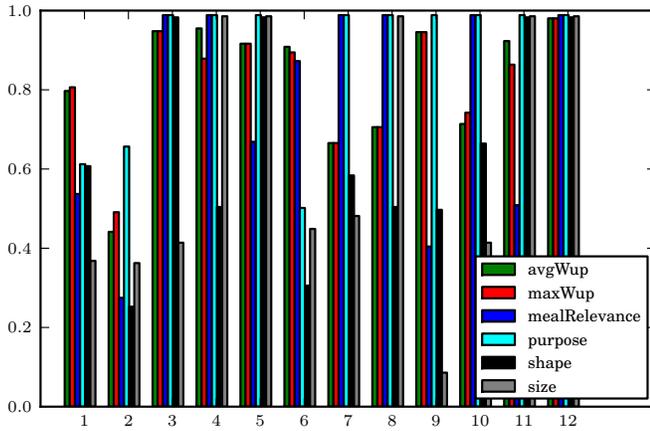


Fig. 7. Average feature importance ($I_F^D(L)$) for the 12 different locations (x-axis) in the mockup kitchen dataset. High values indicate structure at a particular location with respect to a particular group of features.

evaluation considers only one of them as correct. Given probabilistic models like our naive Bayesian classifiers, we could have considered locations that received a probability higher than a certain threshold as correct, but we did not follow this approach because the results would hardly have been comparable to the other classifiers.

Since the organization of an environment such as a kitchen is likely to change over time, it can be necessary to retrain the classifiers that actually require a learning phase. This is, however, not an issue, as in all cases, learning times were negligible.

We now analyze the degree to which there is discernable organizational structure at the various locations in a kitchen. In Figure 7, we present a plot showing the importance measure $I_F^D(L)$ defined in Section VI-C for one of the kitchens from our mockup dataset. A photograph of that kitchen layout is shown in Figure 4. We reduced the importance values for our twelve *avgWup*, twelve *maxWup*, four *purpose* and five *purposeMeal* features to four average values, one for each set of features. The plot shows that all features except size are very prevalent at location 3. In our dataset, this seems reasonable, because the location contains cooking pots and pans, which are similar with respect to most aspects with the exception of size. At location 2, feature importance values are low, because it represents the fridge, which contains the greatest variety of different products. Looking at the prevalent feature values at locations with high importance gives us some human-readable indication of the organization principles, e.g. “FoodVessel, PhysicalDevice, PrincipalMeal, shape: other” for location 3 or “FoodOrDrink, Breakfast” for location 4, which contains breakfast cereals.

In order to analyze the discriminative power of the various features, we present in Figure 8 the Hellinger distances of feature distributions averaged across all pairs of locations in our mockup kitchen dataset. (The graph for the real kitchen dataset is very similar and was omitted for brevity).

We observe that, as expected, the WUP similarities have the highest average Hellinger distances, which means that among the features we considered, they are most well-suited for a discrimination between places.

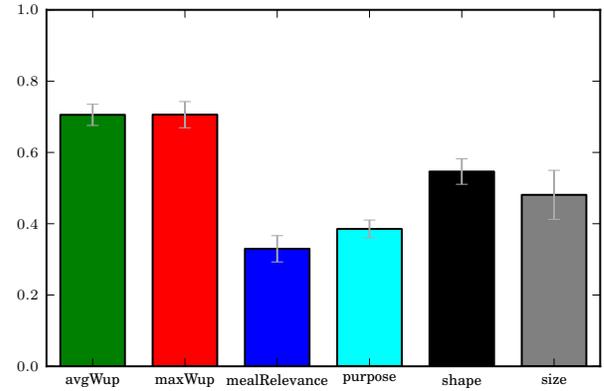


Fig. 8. Mean and standard deviation of the Hellinger distance-based importance measure \overline{H}^D (defined for feature groups by taking the average) for the ten kitchens in our mockup kitchen dataset

VIII. DEMONSTRATION

In order to show the applicability to a real robot system, we integrated our methods into the KNOWROB knowledge processing system³. In Figure 9 we show the system overview. The high-level control layer of the robot system can use a Prolog-based tell-ask interface to query the knowledge base and, in particular, to infer suitable storage locations for items, given information about the environment and the storage locations of objects that were previously perceived. All the necessary background knowledge is represented in the KNOWROB kitchen ontology, which was augmented with additional data from the website *germandeli.com*, as explained in Section IV.

Our methods were used in a demonstration performed during the CoTeSys Spring Workshop 2011, where the PR2 robot TUM-James was given the task to store newly acquired items in appropriate locations. In a simulated shopping sequence, TUM-James first placed items in a shopping

³We provide an open source implementation of our algorithm to solve the object allocation task as an extension to the KNOWROB system [1] at http://code.in.tum.de/pubsvn/knowrob/tags/latest/comp_orgprinciples/

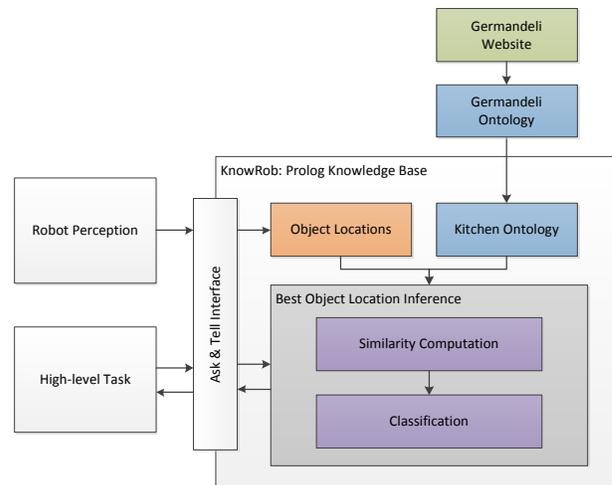


Fig. 9. System overview: Integration of our methods into the KNOWROB knowledge processing system

basket and then emptied the basket on a tabletop. It subsequently identified the objects based on visual models it had previously acquired from product images, determined the ontology concepts to which the items belong and subsequently inferred storage locations, given knowledge about the locations of other objects and their attributes, using the techniques we described above. For the demonstration, we chose the simple maximum *maxWup* inference because it requires no training and still performs well. Excerpts of this demonstration can be seen in the accompanying video and in the image sequence shown in Figure 10.

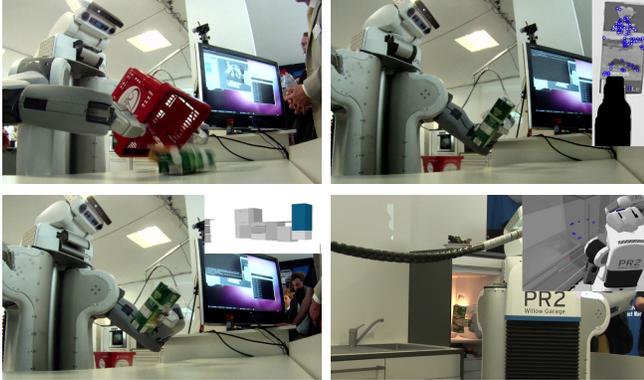


Fig. 10. Pictures of the live demonstration: The robot empties the shopping basket and identifies the categories to which the items belong. It then infers a suitable storage location given its knowledge of the environment and places the object accordingly.

IX. CONCLUSION AND FUTURE WORK

In this work, we have addressed the important task of identifying structure in human living environments – a task that will become increasingly relevant as robots begin to assume their role as household assistants under real-world conditions. We considered the task of allocating objects to likely storage locations and demonstrated its practical implementation in a real robot system.

The allocation task can reasonably be viewed as a classification task, and we thus analyzed the suitability of various classification schemes. Our thesis that organizational principles can be viewed as manifestations of clusterings that are governed by similarity was confirmed in our experiments, and the semantic similarity measure that we proposed based on *WUP similarity* proved to be highly informative. Average classification rates of at least 79% could be reached even in real-world scenarios, and standard classifiers such as support vector machines and (boosted) decision trees proved to be adequate. Naive Bayesian classifiers performed less reliably, but we could exploit their probabilistic semantics to gain additional information about the structure of the problem.

In future work, we intend to include more features that will consider, for example, different states of objects, spatial relations between the containers in an environment and a more fine-grained representation of locations, such that we can model the concrete spatial configuration within specific locations. Furthermore, we plan to address the problem of transferring our approach to the learning of organizational principles in other everyday environments.

ACKNOWLEDGMENTS

We thank our colleagues Dejan Pangercic, Máthé Koppány, Zoltan-Csaba Marton, Lucian C. Goron, Monica S. Opris and Thomas Rühr for their contributions to the live demonstration and all those who contributed to the data collection process. Furthermore, we thank Prof. Charles C. Kemp for many helpful discussions.

This work was supported in part within the DFG cluster of excellence *Cognition for Technical Systems (CoTeSys)*.

REFERENCES

- [1] M. Tenorth and M. Beetz, “KnowRob – Knowledge Processing for Autonomous Personal Robots,” in *IEEE/RSJ International Conference on Intelligent Robots and Systems*, 2009, pp. 4261–4266.
- [2] C. Galindo, A. Saffiotti, S. Coradeschi, P. Buschka, J. Fernández-Madriral, and J. González, “Multi-hierarchical semantic maps for mobile robotics,” in *Proc. of the IEEE/RSJ Intl. Conf. on Intelligent Robots and Systems (IROS)*, Edmonton, CA, 2005, pp. 3492–3497.
- [3] K. Sjö, H. Zender, P. Jensfelt, G.-J. M. Kruijff, A. Pronobis, N. Hawes, and M. Brenner, “The explorer system,” in *Cognitive Systems*, H. I. Christensen, G.-J. M. Kruijff, and J. L. Wyatt, Eds. Springer, 2010.
- [4] A. Bouguerra, L. Karlsson, and A. Saffiotti, “Handling uncertainty in semantic-knowledge based execution monitoring,” in *IROS*. IEEE, 2007, pp. 437–443.
- [5] P. Viswanathan, D. Meger, T. Southey, J. Little, and A. Mackworth, “Automated spatial-semantic modeling with applications to place labeling and informed search,” in *Computer and Robot Vision, 2009*. IEEE, 2009, pp. 284–291.
- [6] H. Zender, O. Martínez Mozos, P. Jensfelt, G. Kruijff, and W. Burgard, “Conceptual spatial representations for indoor mobile robots,” *Robotics and Autonomous Systems*, vol. 56, no. 6, pp. 493–502, 2008.
- [7] S. Vasudevan, S. Gächter, V. Nguyen, and R. Siegwart, “Cognitive maps for mobile robots - an object based approach,” *Robotics and Autonomous Systems*, vol. 55, no. 5, pp. 359–371, 2007.
- [8] S. Vasudevan and R. Siegwart, “Bayesian space conceptualization and place classification for semantic maps in mobile robotics,” *Robotics and Autonomous Systems*, vol. 56, no. 6, pp. 522–537, 2008.
- [9] D. D. Fu, K. J. Hammond, and M. J. Swain, “Action and perception in man-made environments,” in *Proc. of the 14th IJCAI*, Montreal, Canada, 1995, pp. 464–469.
- [10] C. Matuszek, J. Cabral, M. Witbrock, and J. DeOliveira, “An introduction to the syntax and content of Cyc,” *Proceedings of the 2006 AAAI Spring Symposium on Formalizing and Compiling Background Knowledge and Its Applications to Knowledge Representation and Question Answering*, pp. 44–49, 2006.
- [11] M. Tenorth, U. Klank, D. Pangercic, and M. Beetz, “Web-enabled Robots – Robots that Use the Web as an Information Resource,” *Robotics & Automation Magazine*, vol. 18, no. 2, pp. 58–68, 2011.
- [12] Z. Wu and M. S. Palmer, “Verb semantics and lexical selection,” in *ACL*, 1994, pp. 133–138.
- [13] M. Hall, E. Frank, G. Holmes, B. Pfahringer, P. Reutemann, and I. H. Witten, “The WEKA data mining software: an update,” *SIGKDD Explorations*, vol. 11, no. 1, pp. 10–18, 2009.
- [14] R. R. Bouckaert, “Naive bayes classifiers that perform well with continuous variables,” in *Australian Conference on Artificial Intelligence*, ser. Lecture Notes in Computer Science, G. I. Webb and X. Yu, Eds., vol. 3339. Springer, 2004, pp. 1089–1094.
- [15] I. Rish, “An empirical study of the naive bayes classifier,” in *IJCAI-01 workshop on “Empirical Methods in AI”*.
- [16] D. Jain and M. Beetz, “Soft Evidential Update via Markov Chain Monte Carlo Inference,” in *KI 2010: Advances in Artificial Intelligence, 33rd Annual German Conference on AI*, ser. Lecture Notes in Computer Science, vol. 6359. Springer, 2010, pp. 280–290.
- [17] G. H. John and P. Langley, “Estimating continuous distributions in Bayesian classifiers,” in *Proc. 11th Conference on Uncertainty in Artificial Intelligence*. Morgan Kaufmann, 1995, pp. 338–345.
- [18] L. L. Cam and G. L. Yang, *Asymptotics in Statistics – Some Basic Concepts*. Berlin: Springer, 1990.
- [19] W. Li and X. Jia, “Feature selection algorithm based on hellinger distance,” *Journal of Computer Applications*, vol. 30, no. 6, pp. 1530–1532, 2010.