EU FP7 CogX
ICT-215181
May 1 2008 (52months)

DR 5.4:
Active learning of cross-modal concepts

Danijel Skočaj, Matej Kristan, Alen Vrečko, Barry Ridge, Aleš Leonardis, Michael Zillich, Johann Prankl, Markus Vincze, Sergio Roa, and Geert-Jan Kruijff

University of Ljubljana, TU Vienna, DFKI Saarbrücken
⟨danijel.skocaj@fri.uni-lj.si⟩

Due date of deliverable:	31 July 2011
Actual submission date:	31 July 2011
Lead partner:	UL
Revision:	final
Dissemination level:	PU

An important characteristic of a robot that operates in a real-life environment is the ability to expand its current knowledge continuously, autonomously and in an interaction with the environment. It has to understand what it does not know and it should act appropriately to obtain the missing information ant to update its knowledge accordingly. This self-understandig/self-extension cycle is implemented by the active learning paradigm, which is the central topic of this deliverable. We address the problem of active learning on different levels of cross-modal learning and propose several approaches that facilitate active learning. We evaluate different active learning strategies, address the problem of active vision, and present the methods that we have developed for self-supervised learning of object affordances, as well the extension of the approach to cross-modal binding and learning.

Executive Summary

Active learning is an important characteristic of a system that is supposed to be capable of self-extension based on self-understanding. Such a system should be able to understand what it does and does not know, therefore it should be able to detect the gaps in its knowledge and determine what kind of information would be needed to fill these gaps. In order to self-extend, it should first be able to plan a sequence of actions that would produce the required information and, finally, it should be able to update its knowledge using the newly acquired information. In this deliverable we address active learning from several directions on different levels of cross-modal learning. We propose several methods and approaches; some of them address the full active learning cycle, while some of them only a part of it.

We addressed the problem of active learning of conceptual knowledge and developed a framework that implements the full active learning cycle and enables thorough evaluation of different active learning strategies. The experimental results show that the active learning approach outperforms the passive one and that the adaptation of the learning process to the learners knowledge significantly improves the learning peformance. The framework is based on the odKDE methodology we initially developed in Year 2 and significantly improved in Year 3.

One important research line was also active vision. We use an active approach to segment the scene into independently moving objects and subsequently learn their models. We also developed a principled methodology for detecting the incompleteness of the learned object models, and to determine what can be done to complete the model and what the benefits of doing so are.

We have also continued our work on learning object affordances. We redefined and extended our low-level self-supervised cross-modal learning algorithm and developed an online relevance learning vector quantisation method that enables more efficient and effective learning of object affordance classes.

Finally, we also revised and refined the problem definition of binding and cross-modal learning from Year 2, reformulated and adapted its formulation in Markov Logic Networks, and applied it to a cognitive system architecture enabling active learning on the system level.

Some of the work presented in this deliverable is a continuation of the work performed in Year 2 and mostly presented in deliverables DR.5.2. *Continuous learning of cross modal concepts* and DR.5.3 *Representations of gaps in categorical knowledge*, as well as in deliverable DR.2.2 *Active Vision, learning and manipulation.* This deliverable, however, also presents work that has been initiated in Year 3. In both cases, the novelty and the value added in Year 3 are clearly exposed in the sections below. The work has been mainly performed as envisioned in the workplan and forms a solid basis for

further research and extensions in the direction of more general interactive learning of cross-modal concepts.

Role of Active learning of cross modal concepts in CogX

In the process of active cross-modal learning, the system tries to understand what it does know and what it does not. Based on this it actively plans and executes corresponding actions to obtain the missing information and then updates the current knowledge accordingly. Therefore, the main research topic addressed in this deliverable is central in the project that aims to develop autonomous systems that self-understand and self-extend.

Contribution to the CogX scenarios and prototypes

In order to monitor and show progress on active and interactive continuous learning, we have designed the George scenario (Interactive cross-modal learning scenario) [55]. This scenario has been designed as a use case for guiding and testing system-wide research and for demonstrating methods developed in WP 5 (and also some other workpackages) in a working system. Therefore, many of the methods presented in this deliverable have been integrated into the overall system, which is used in the George scenario.

1 Tasks, objectives, results

1.1 Planned work

This deliverable mainly tackles the problems addressed in Task 5.3 of Workpackage 5:

> *Task 5.3: Active learning of cross-modal concepts. Increase the systems autonomy to enable continuous detection of ignorance, and active planning and execution of knowledge producing actions enabling autonomous continuous self-extension.*

Therefore, the main goal was to develop the theory and methods to be integrated into a robot capable of active continuous learning of cross-modal concepts.

Before we begin with the description of the planned and performed work, let us first elaborate the active learning paradigm in the context of artificial cognitive systems. As depicted in Fig. 1, we envision one active learning cycle (a planned incremental update of the current knowledge) comprising of four main steps:

1. **Detection of knowledge gaps.** The system should first self-understand, it should understand what it does and what it does not know. It should detect using its internal modal representations what information is missing.

2. **Generation of gap completion proposal.** Based on the detected ignorance the learner should determine (plan) what information is needed to fill the gap in the knowledge and issue a request (a desire, a motive) to the overall system about **what** information it would like to obtain (e.g., a view from the opposite side is needed, more similar objects are needed, a push from another direction is needed, etc.).

3. **Planning and execution of actions.** The system should then plan the sequence of actions that would lead to the state that would reveal information asked by the particular modal learner. It would thus determine (plan) **how** and **when** to obtain a particular piece of information (e.g., the robot would move to get a novel view-point, or it would grasp and rotate the object, or it would ask a human to rotate the object, or it would push the object, or it would initiate a dialogue with the tutor, etc.).

4. **Updating the current knowledge.** After the action has been executed the modal learner will gather novel information and use it for updating the current internal representations.

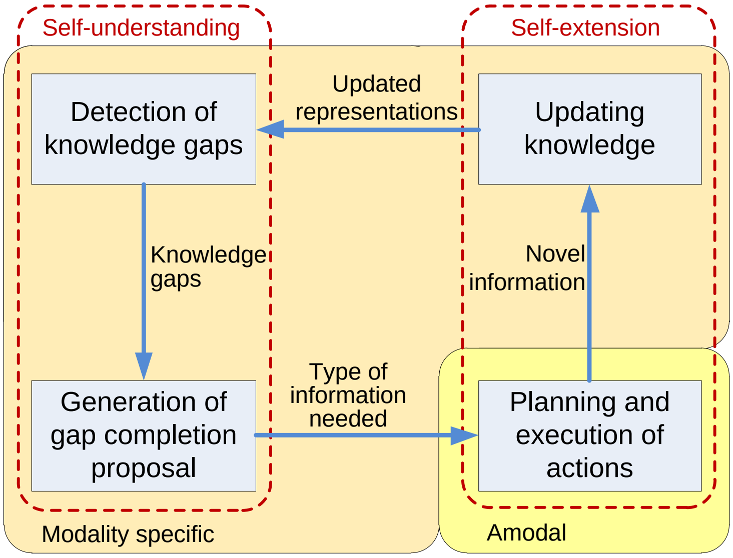


Figure 1: Active learning cycle.

In a multimodal system engaged in active learning, each modality keeps its internal modal representation of the world and concepts that are being learned. These internal representations have to meet certain requirements to enable active learning. They should enable detection of knowledge gaps (Step 1) and determination of what kind of information is needed to fill these gaps (Step 2). After this information is obtained they also have to allow for efficient update (Step 4). It should also be possible to abstract this modal information into an amodal one, that can be used by higher-level cognitive processes to plan the necessary actions (Step 3).

The approaches that we have developed in WP 5 are dealing with these requirements. Some of the approaches that will be described in this deliverable address all four steps listed above, therefore implementing the full active learning cycle. Some of the methods we will present address only a subset of these steps and they can be seen as enabling technologies for active learning.

In Year 2 deliverable DR.5.2 we identified different types of cross-modal learning. This year we discuss these different types of learning within the active learning paradigm. In weakly-coupled cross-modal learning (*Low-level uni-modal learning with high-level cross-modal supervision*) the models are built within individual modalities, and the other modalities only supervise learning, by, e.g., providing a label or reinforcement signal. In this deliverable two research lines related to this type of learning are presented: **Active**

learning of categorical knowledge and **Active vision** approaches. In closely-coupled cross-modal learning (*Self-supervised low-level cross-modal learning*), learning processes are more intertwined. A model is learnt by combining information from different modalities into a common level of representation, and then using this level as a starting point to build a common cross-modal classifier or predictor. Here, we present our work on **Self-supervised learning of object affordances**. When referring to modality specific representations in the active learning cycle (Fig. 1) we have in mind representations obtained in both of these two types of cross-modal learning (the ones that reside in a single modality, as well as the ones that span across modalities). We identified also a third type of cross-modal learning that is performed on a higher level of abstraction (*High-level cross-modal learning*). Here, a model is acquired that connects modal conceptual structures from different modalities by learning associations between them. This type refers to the (amodal) third step in the active learning cycle, and actually enables active learning in an integrated cross-modal system. We implemented this type of learning in our work on **Cross-modal binding and learning**.

We will structure this deliverable according to this division. First, let us look at our plans and goals that we had set:

- **Active learning of categorical knowledge.**

  Our first goal was to address the problem of active learning of conceptual knowledge. We planned to develop a framework, which would implement all four steps in the active learning cycle and would enable analysis of different active learning approaches. We wanted to determine what are the factors that influence the performance of active learning, how the learning process can be sped up, and what are the requirements of the learning algorithms that alow this to happen. We planned to build the active learning framework on the top of the knowledge gaps methodology that we developed in Year 2 and presented in the deliverable DR.5.3.

  This methodology is based on generative models called the online Kernel Density Estimators (oKDE). In Year 2 we have performed a thorough experimental analysis of the oKDE which was published in a pattern recognition journal [26]. That analysis layed out the basis for extension of the oKDE to models that explicitly address the discrimination capacity of the discriminative models. As a proof-of-concept we have derived a first discriminative extension of the oKDE in [25]. While [25] was a solid proof-of-concept, several unanswered questions remained. Namely, the method was computationally slower than oKDE, it exceeded oKDE in producing simpler models, but we could not conclude with absolute certainty that the models were also better in terms of classification capabilities. This year we therefore planned to perform the additional extensive analysis, which would lead to final reformulation of the online discriminative Kernel Density Estimators (odKDE), which would ideally outperform the oKDE by all criteria in a discriminative setting.

- **Active vision.**

  Our previous work on object recognition [39, 32] reported in DR.2.2 addressed Step 4 (Updating the current knowledge) of the active learning cycle, by providing mechanisms to incrementally learn 3D visual object models online. While this allows for very simple learning strategies such as driving around the table once, the generated models did not represent their completeness and vice versa knowledge gaps explicitly to support Steps 1-3 of the active learning cycle. The planned work for this year was therefore to augment object representations with meaningful, quantitative measures of completeness that tie in with the overall goal and motive management framework.

- **Self-supervised learning of object affordances.**

  In our work on low-level cross-modal object affordance learning, we sought to make a distinction between object shape properties as one modality and object effects under action as another modality and aimed to form percepts within each of those modalities and cross-modal associations between them. In continuation of our work in [42], we aimed to build a more theoretically robust formulation of our self-supervised cross-modal learning algorithm, whilst also incorporating additional mechanisms as necessary with the goal of ensuring that the algorithm is more capable of rapid learning over short training periods.

- **Cross-modal binding and learning.**

  The main goal for this year was to apply on the overall system the principles from the Year 2 prototype binding system. How should a binding system based on Markov Logic Networks be integrated in our cognitive architecture? On a system level, this competence enables abstraction and integration of modal information required by Step 3 in the active learning cycle. The plan was also to design a flexible on-line learner based on Markov Logic Networks that would not only provide the cross-modal learning functionality to the integrated binding system, but possibly act as a general multi-purpose learner on higher cognitive level.

1.2 Actual work performed

In this section we briefly describe the main achievements related to the topic of this deliverable. For detailed descriptions of the work performed the reader is referred to the papers attached in the annex of this deliverable.

1.2.1 Active learning of categorical knowledge

In general settings, novel training samples may arrive at any time, and the representations of the learned models have to be able to adapt to new situations. The representations should allow for incremental or online learning. The representations are being build incrementally; their reliability is being increased through time resulting in the continuous improvement of the classification performance. A natural goal of an online learner is to speed up the learning process, therefore to utilize the learned models as soon as possible (e.g., for recognition of the learned concepts). As we know from our everyday experience, a good teaching material facilitates learning, as well as a good teachers does. Similar question also arise in the case of online machine learning: which training samples to present to the learner and in what order? This is the main question we address in our work presented in Annex 2.1 [54].

We constrain our analysis by considering a case of learning conceptual knowledge. We assume that we have a teacher that teaches the learner about certain concepts (e.g., object properties, spatial relations, object categories, etc.). The teacher provides labels that are used by the learner to update its representations. We discuss different ways of how this knowledge is transferred from the teacher to the learner, ranging from purely *passive approach* where the teacher completely drives the learning to the *active learning* approaches where the learner takes the initiative and actively influences the learning processes. We start with completely passive *teacher-driven* learning, where the teacher presents new training samples by treating all training samples equally without considering the learner's knowledge. Then we proceed by differentiating between training samples and giving a higher priority to the samples that are expected to hold a significant discriminative information. Then the teacher also takes the learner's current knowledge into account when selecting training samples. We also evaluated active *learner-driven* approaches, where the learner inspects its internal knowledge and detects the ignorance. Different ways of generating knowledge gap completion proposals are discussed, i.e., by referring only to the current training sample or also to the previously observed samples, by referring to the most ambiguous concept, or by generating the actual training sample, which is estimated to improve the current representation most. Detailed description of these approaches and the results of the experimental evaluation are given in Annex 2.1 [54]. We analyzed the learning curve with respect to different levels of the influence the learner has to the learning process. The experimental results show that the active learning approach outperforms the passive one and that the adaptation of the learning process to the learner's knowledge significantly facilitates and speeds up the learning.

The learning mechanisms that we used are based on the underlying KDE methodology we have developed. We have performed an extensive analysis of how to capitalize on discriminative information in the framework of online Kernel Density Estimators, leading to the final version of the online discriminative Kernel Density Estimators (odKDE). The theory about the new odKDE and its experimental analysis has now been submitted to a journal as a follow-up paper to the oKDE [26] that was developed in Year 2. We have revised the theory on the odKDE, which led to a simpler formulation of the algorithm. We have shown that the previously proposed oKDE can be treated as a general framework for online construction of probability density functions from streaming data, that applies compression to the models to keep complexity low. We have proposed a formulation of a cost function that measures discrimination loss during compression. A principal novelty in this function was the fact that it can be directly plugged in the existing compression optimization algorithm used in oKDE. While the computation of the cost function is very slow in its original form, we have derived a simplification that decomposes the cost function and allows much faster computation. In contrast to the oKDE, which is purely reconstructive, the odKDE implements a principle of constrained generalization, rather than maximizing some discriminative cost function (as it is usually done in discriminative models). While both the oKDE and the odKDE involve reconstructive updates by new data-points, the odKDE then gradually generalizes the model while constraining the discrimination loss.

In our study that was performed during this project, we have shown that a crucial point in an online model is that it has to be sufficiently complex to properly adapt its structure when the new data-point arrives. It appears now that the presented paradigm of reconstructive updating combined with discriminatively-constrained generalization maintains excellent balance and sufficient complexity required for online updating and generalization (smoothing) that improves classification properties while keeping the model simple. We performed extensive comparison with a batch support vector machine, oKDE and batch state-of-the-art KDEs on several standard publicly available machine learning datasets to allow deeper comparison to the competing methods and to experimentally show that the odKDE now outperforms the oKDE in better recognition capability and in generating simpler models. The analysis and the new methodology is described in Annex 2.2 [24].

1.2.2 Active vision

One aspect of cross-modal learning encountered in many robotics tasks is to learn the association between visual object appearance and linguistic object labels, such as putting an object on a table and saying “This is a coke can”. Actively learning these object models is essentially comprised of three parts. 1) Segmenting from the scene what is considered to be an object 2) Completing object models from partially acquired models, possibly including actions such as view point changes or clarification questions to the tutor 3) And linking the acquired visual model to other modalities, most notably language.

For complex scenes where object boundaries are not immediately clear static segmentation approaches such as plane pop-out [66, 65] are not sufficient. To disambiguate between possible interpretations in these cases, the work presented in Annex 2.3 [40] follows an active approach to segment the scene into independently moving objects. We extended previous work on learning planar patch based object models to include motion cues to group sets of patches that show consistent motion after a push is applied to a part of the scene.

Once one view of an object has been identified and learned, the robot needs a principled methodology to extend its partial knowledge. I.e., it needs to identify where the model is incomplete, what can be done to complete it and what are the benefits of doing so. In the work presented in Annex 2.4 [68] we extend previous work on incremental online learning of object models with learned probabilistic measures for observed detection success, predicted detection success and model completeness. This allows the robot to quantify its current knowledge and the predicted increase in knowledge for a given action (i.e. change of view point).

1.2.3 Self-supervised learning of object affordances

Another line of research focuses on affordance learning. The main goal is to learn to predict what will happen with an object that has been pushed in terms of classification (what the resulting affordance class will be).

Continuing our work in [42], we have refined our low-level self-supervised cross-modal learning algorithm such that it is placed on a more theoretically sound footing, incorporates additional mechanisms that make it more effective over short-term training periods, and we have performed more extensive experiments both on simulated datasets and with real object affordance learning data in order to demonstrate its effectiveness.

From a theoretical standpoint, we have shown how, given the cross-modal structure of codebook layers of prototypes, we may derive learning rules based on the learning vector quantization paradigm that can employ class probabilities instead of actual class labels during training, thus allowing us to bootstrap the self-supervised learning process in an online manner even when the categories are not yet fully known. This is an important consideration at the lower-level where data of lower-order features co-occur in an online manner across multiple modalities and higher-level concepts ought to be formed dynamically.

In addition, given the sparsity of training data in the autonomous robotics setting, as well as the expense of gathering additional data, and the necessity to learn online and update models as soon as training data arrives, it is often not feasible to train over large datasets or long training periods of multiple epochs where the training data are recycled. To address these issues, we developed additional mechanisms to augment the base algorithm and ensure that it can achieve effective results rapidly. The first such mechanism is a form of feature relevance determination where the prototypes in the classifying codebook layer are analysed using the Fisher criterion to weigh each of the feature dimensions with respect to their discriminative relevance. The second mechanism is that of ineffective prototype culling, where prototypes that do not contribute to, or indeed even inhibit, the classification process are culled from consideration whenever the network is tasked with classifying a sample. We demonstrate the effectiveness of these techniques in experiments on both simulated and real data in [41] (c.f. Annex 2.5).

With regard to active learning in the low-level self-supervised cross-modal setting, the algorithm we have developed as referenced in [41] (c.f. Annex 2.5) may also be employed within such a context. Following the self-understanding workflow of Figure 1, the algorithm can account for the detection of knowledge gaps in two key ways. Firstly, the algorithm dynamically detects novel affordance categories as data become available. It could potentially start out with one or two categories based on clusters in the data, but as new data are gathered in the effect modality, representing the effects of actions on objects, it can detect if new clusters are forming that could potentially be novel affordance categories. Secondly, based on the available training objects and the potential actions that can be applied to them, the algorithm can, based on its current knowledge, provide posterior probabilities for different possible affordance class predictions. It can therefore indicate which action/object combinations are the most ambiguous in terms of their affordance class predictions, thus potentially indicating gaps in knowledge. Again following Figure 1, with regard to generating gap completion proposals, once novel clusters have been detected in the effect modality, the algorithm forms novel affordance categories in an unsupervised manner, thus implicitly filling such knowledge gaps. In the case where multiple training objects are available, the algorithm can propose performing an action on an object that provides the most ambiguous posterior class probabilities for affordance class prediction. When it comes to updating knowledge, the algorithm fits naturally here because it can learn online, forming updated cross-modal representations of affordances using novel data from object interactions. Taken together, these three aspects of the algorithm enable the planning and execution of actions for active learning. As knowledge gaps are detected, gap completion proposals may be generated, actions may be planned and executed in order to gather novel data to fill the gap, and finally the algorithm may update its representations.

To test whether or not the algorithm functions effectively under an active learning assumption, we performed some preliminary experiments where, using previously gathered object affordance data [42], we assumed that the entire training set was available to the algorithm and actively selected the samples at each online training step by choosing the most ambiguous sample in terms of the algorithm's class prediction. This was made possible because the algorithm can provide posterior class probabilities as well as class label predictions. To explain this in brief, unsupervised online hyper-clustering of prototypes is performed in both an object property modality and an object effect modality based on a stream of data co-occurring in each, while a cross-modal co-occurrence mapping between prototypes in each modality is also constructed. Crucially, the prototypes in the object effect modality are meta-clustered to form category clusters that may be projected onto the object property modality, either in terms of category cluster labels or category cluster probabilities using the information provided by the co-occurrence mapping. This process is explained in detail in [41] (c.f. Annex 2.5). The results of our initial experiments have been inconclusive. While the active learning approach appears to generally perform better than regular online learning over partially-ordered data, it does not appear to perform as well as online learning over randomly ordered data on this particular dataset. We suspect that the small training set size may have a significant influence on this performance, but this warrants further investigation which we aim to attempt in due course.

We have also considered the problem of affordance learning by using a different approach, continuing our work in [45, 46]. We follow the idea presented above, but we focus here on the problem of how a robot can predict the results of interaction with an object, in terms of predicting a concrete object movement (trajectory) as well as a categorization of the movements. We initially approached the problem of trajectory prediction using regression techniques, such as Recurrent Neural Networks (RNNs) trained with offline and active learning strategies for samples selection [45], as described in the Year 2 deliverable DR.5.2. RNNs can predict sequences with long-term dependencies, making them useful for predicting many steps ahead. However, since learning in RNNs is based on gradient descent methods, learning requires a lot of cycles. Furthermore, predictions need not always be entirely accurate, due to convergence to local minima.

As an alternative, we have explored Bayesian learning and vector quantization algorithms for solving this prediction problem. Our goal is to extract probabilistic finite state machines which model and discretize the behavior of a dynamical system learnt from robot-object interactions. Preliminary experiments were presented in [46], where stochastic finite state machines were extracted in a simulated scenario. In order to tackle both classification and prediction problems, we can use the output functions in a probabilistic finite state machine either to predict a continuous series, or to classify a temporal pattern. This yields two different ways in which we can combine the acquisition of probabilistic (discrete) machines, with the learning of dynamic (continuous) systems. We use offline learning methods for learning probabilistic finite state machines. We have performed several series of experiments with artificial datasets, to show the potential of these algorithms. Vector quantization is explored by using some datasets with Gaussian noise distributions in the presence of outliers, where we apply an incremental algorithm which suggests a life-long learning procedure. Extraction of probabilistic machines is explored by integrating Vector quantization of input, state and output spaces of the dynamical system and Bayesian learning and conditional entropy for inferring the transitions between states and its probabilities. An experiment where a probabilistic finite state machine is inferred from data generated from a noisy automaton shows the strength of this approach. We compared our results to other approaches found in the literature, where we show better performance (Annex 2.7 [44]). In future work, the use of information-theoretical approaches to calculate transitions might also be useful to explore active selection of samples via an information-gain measure.

1.2.4 Cross-modal binding and learning

We revised and refined the problem definition of binding and cross-modal learning from Year 2 and reformulated and adapted its formulation in Markov Logic Networks (MLNs). The main issue this year was how to apply the principles from the problem definition and the specifics of MLNs to a cognitive architecture. This work is described in detail in Annex 2.6 [61].

In a cognitive system the sensory information is usually first processed by the *perceptual layer*. While processing on the perceptual layer is inherently *intra-modal*, high-level cognition, which includes many processes that are critical for active learning (e. g. , motivation, planning), usually assumes *a-modal* information. High-level cross-modal learning and binding play a crucial role in overcoming the semantic gap between the two representation categories, assuring that the high-level representations are grounded in multiple modalities. As such, its ability to associate between various modal and a-modal representations plays an important role in active learning in general, since more often than not active learning involves multiple modalities (including a-modal representations, as depicted in Fig. 1).

Fig. 2 illustrates a possible application of our binding and cross-modal learning system to a cognitive architecture. We can see that the information from the perceptual layer is used by three distinct processes:

- The process of *concept grounding* uses modal concepts produced by the learning process in modal learners (e. g. various color and shape types) to ground the binding rules.
- The process of *instance grounding* relies on the ability of the perceptual layer to quickly present (usually relying on one modality only)

quantitative estimates about the entities (instances) the cognitive system is currently sensing. While the multi-modal representations of perceived entities are quantitatively and qualitatively finalized by the binding process itself (union configuration), these initial approximate representations can be considered to be placeholders for potential objects (i. e. possible percept unions). They are devoid of any features or other kind of attributes.

- The recognition process in modal learners results in the percept configuration, which represents the input to the process of *binding inference*.

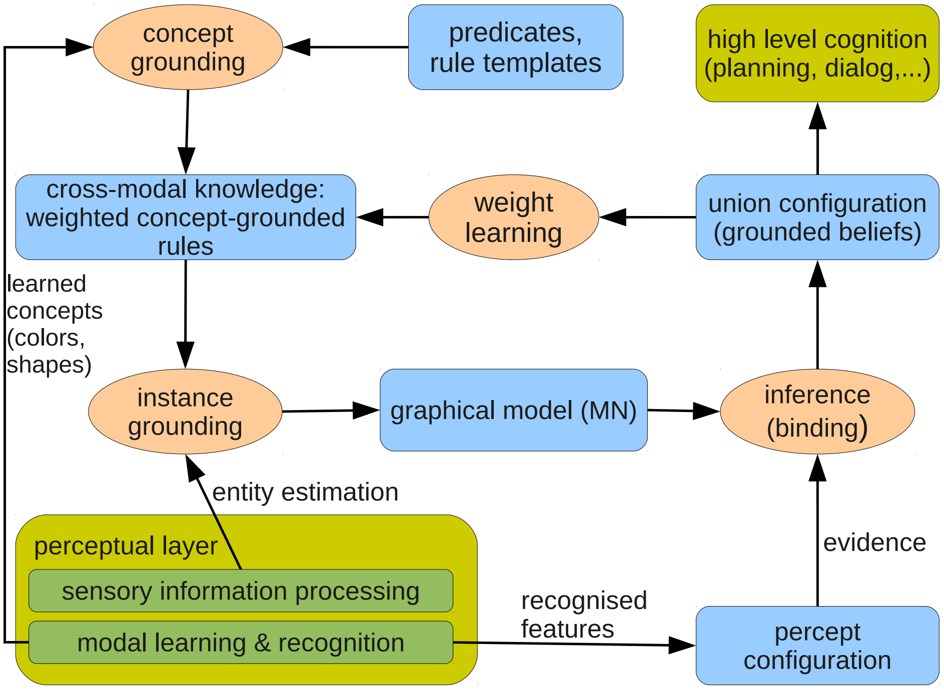

concept grounding
predicates, rule templates
high level cognition (planning, dialog,...)
cross-modal knowledge: weighted concept-grounded rules
weight learning
union configuration (grounded beliefs)
learned concepts (colors, shapes)
instance grounding
graphical model (MN)
inference (binding)
entity estimation
perceptual layer
sensory information processing
modal learning & recognition
evidence
recognised features
percept configuration

Figure 2: Cross-modal learning and binding as part of a cognitive system.

The final product of binding — the union configuration is used both as the basis for a-modal representations in high-level cognition and as a source of learning samples for the *weight learning*.

The processes of *instance grounding*, *binding inference* and *weight learning* form the *inner binding loop*, which exploits the perceptive abilities of modal learners and recognizers to improve its cross-modal associative power. On the other hand the process of *concept grounding* exploits the concept forming ability of modal learners to produce new cross-modal concepts, which are eventually evaluated within the existing cross-modal knowledge by the inner binding loop.

To demostrate the above principles in practice, we developed a CAST component with online MLN functionality — the MLN learner. In the current system the MLN learner is used as a reference resolution tool (a form of binding) by the dialogue subsystem. Because of its inherent flexibility (MLNs), it can have many other applications that require learning in the high-level cognition.

1.3 Relation to the state-of-the-art

In this section we discuss how our work is related to, and goes beyond the current state-of-the-art.

1.3.1 Active learning of categorical knowledge

Active learning of categorical knowledge has been often addressed in the literature. The proposed approaches mainly focus on estimating classifiers using minimal amount of data. They are motivated by the fact that there are many situations in which large quantities of unlabeled data are relatively easily obtained, however, the cost of labeling each sample can be high. Depending on how the data is accessed, we can divide the approaches to active learning into two major groups: (i) pool-based approaches to learning, and (ii) learning from streaming data. In pool-based learning, all data is available in advance, a selection procedure determines the learning points, queries an oracle (a teacher) for labels of these points, and uses these points to construct a classifier. Here, an important issue is which data-points to chose for querying. A plethora of papers have been published on this topic proposing numerous approaches [8, 52, 51, 35, 1, 64, 28, 19, 2, 23] using different kinds of classifiers and committees of classifiers, as well as probabilistic rules for selecting the next best sample for querying. In the stream-based learning, the data comes sequentially, and possibly indefinitely. Here the challenge is to constantly adapt the classifier to the possibly changing properties of the data and identify in the observed sequence the potentially informative data-points for querying the oracle. Although the samples are introduced sequentially, most of the learning algorithms for streaming data process the data in small batches [67, 12, 20]. For situations in which a teacher is sequentially presenting training samples to the learner, the pool-based approaches are not applicable, since they assume that the learner would have access to all observed samples. In that respect, the traditional streaming-data-based active learning approaches are also not applicable, since they assume a batch of data-points to be available for constructing the classifier. In real-life situations, it is desirable that the learner detects good candidates for querying on the fly and updates its classifiers accordingly, while requiring minimal involvement of the teacher. Several learning strategies have been proposed in this social learning context [4, 53, 10]. We also analyze the problem of active learning of categorical knowledge in the light of these requirements. In our earlier work [53], we addressed this problem in the constrained stream-based interactive settings. In the current work [54] we take more general approach. We discuss different ways of conveying the categorical knowledge from the teacher to the learner and analyze different knowledge gaps completion methods. We experimentally show that the active learning approach and the adaptation of the learning process to the learners knowledge significantly facilitate and speed up learning.

Our experiments were performed with the learner and classifier based on the discriminative reconstructive representations learned by odKDE. The nature of the purely reconstructive models is that they retain only the information which is necessary for an approximate reconstruction of the data – and this information is also crucial for proper online updating [26]. But since the retained reconstructive information does not necessarily encompass the discriminative information, such approximations often lead to reduced discrimination performance of the reconstructive models on specific classification tasks. On the other hand, purely discriminative models disregard the reconstructive information, which may lead to reduction of their robustness, e.g, [13]. Some early work on incrementalization of a linear discriminant analysis [58] suggests that accounting for the reconstructive information is indeed crucial for robust incremental updates of discriminative models. The leading hypothesis of our present work can be stated as: if we gradually generalize a generative model while constraining its loss of discriminative information, we will arrive at a model that has better discriminative properties than the original reconstructive model, but at a same time we will retain sufficient reconstructive information to allow efficient updating from new data-points. Our analysis in [24] has confirmed that this hypothesis indeed holds. When applied to the state-of-the-art oKDE [26], we have arrived at a discriminative online Kernel Density Estimator (odKDE). Experimental results in [24] show that, compared to the batch state-of-the-art KDE approaches [33, 15, 17] and the state-of-the-art online KDE [26], the odKDE produces significantly simpler models with on average better classification performance.

1.3.2 Active vision

Active strategies for segmentation of objects in robotic scenarios have been explored in the past [30]. Our work differs in that we additionally employ well known structure from motion (SfM) techniques to acquire detailed 3D object models. While classical SfM moving through a static scene is essentially solved in a coherent theory, recent work focuses on dynamic scenes composed of rigidly moving objects. The solutions available so far can be broadly classified into algebraic methods [60, 9], which exploit algebraic constraints satisfied by all scene objects, even though they move relative to each other, and non-algebraic methods [57, 14], which essentially combine rigid SfM with segmentation. Most related to our system are [50, 36] which use interleaved segmentation and 3D reconstruction of tracked features into independent objects. Our work differs in that we introduce planar patches as intermediate structure between tracked interest points and full 3D objects, as the simpler plane models can be extracted more robustly. These planar patches are then grouped into 3D objects. Finally, instead of a sparse point cloud we get a dense representation in terms of planes, which is more suitable for robotic manipulation.

A robot that aims to continually extend its knowledge can not rely only on models trained offline. Instead partial objects acquired e.g. via the method mentioned above should be extended online in interaction with the environment. Various methods have been proposed to overcome the offline learning vs. online recognition division [59, 37, 47, 16, 43, 38]. While the work presented in this report uses fairly standard techniques for recognition, it focuses more on a representation of the object model that allows the system to reason about model completeness and further knowledge gathering actions.

1.3.3 Self-supervised learning of object affordances

Also self-supervised cross-modal learning has been addressed many times in the literature in varying forms. An early example of a cross-modal neural network similar in structure to that of our algorithm is provided by [31], where self-organizing maps acting in separate spaces were connected together via a Hebbian mapping and labeled samples were presented to the network for cross-modal clustering. One of the more inspirational papers for our own work, that of de Sa and Ballard in [11], used a multi-modal neural network to study the McGurk effect [29] using co-occurring visual and audio data of utterances from human speakers. The authors also employed learning vector quantization in their multimodal framework in the traditional form where category clusters were used as class labels. Coen [5, 6, 7] also addressed the idea of cross-modal learning by clustering in separated, but interconnected modalities, though without online learning. Both multimodal learning and the McGurk effect have more recently been addressed in the deep learning community [34] where deep autoencoder networks were used in both audio and visual modalities to learn features cross-modally.

1.3.4 Cross-modal binding and learning

Many of the past attempts at binding information within cognitive systems were restricted to associating linguistic information to lower level perceptual information. Roy et al. tried to ground the linguistic descriptions of objects and actions in visual and sound perceptions and to generate descriptions of previously unseen scenes based on the previously accumulated knowledge [48, 49]. This is essentially a *symbol grounding problem* first defined by Harnad [18]. Chella et al. proposed a three-layered cognitive architecture around the visual system with the middle, *conceptual layer* bridging the gap between linguistic and sub-symbolic (visual) layers [3]. Related problems were also often addressed by Steels [56].

Jacobsson et al. approached the binding problem in a more general way [22] [21] developing a cross-modal binding system that could form associations between multiple modalities and could be part of a wider cognitive architecture. The cross-modal knowledge was represented as a set of binary functions comparing binding attributes in pair-wise fashion. A cognitive architecture using this system for linguistic reference resolution was presented in [62]. This system was capable of learning visual concepts in interaction with a human tutor. A probabilistic binding system was developed within the same group that encodes cross-modal knowledge into a Bayesian graphical model [63]. In [27] a framework for constructing high-level cognitive representations of the environment, called beliefs, was presented. Markov logic was used as the main framework for various types of inference over beliefs, including perceptual grouping, which comes very close to our definition of binding. All these systems ([22] – [27]) assumed static cross-modal knowledge.

Our goal is to design a flexible binding system, capable to continuously adapt the probabilistic representation of cross-modal knowledge to the challenges of a dynamic environment. These requirements lead us in the direction of Markov graphical models as a powerful and flexible platform for probabilistic problem formulation. We base our work, however, on a formal definition of the binding problem, which is still general enough to accommodate other possible approaches to binding.

2 Annexes

2.1 Skočaj et. al "About different active learning approaches for acquiring categorical knowledge"

Bibliography D. Skočaj, M. Kristan and A. Leonardis: "About different active learning approaches for acquiring categorical knowledge" Submitted to Twentieth International Electrotechnical and Computer Science Conference, Portorož, Slovenia, 29–21 September 2011.

Abstract In this paper we address the problem of acquiring categorical knowledge from the active learning perspective. We describe and implement several teacher and learner-driven approaches that require different levels of teacher competencies and consider different types of knowledge for selection of training samples. The experimental results show that the active learning approach outperforms the passive one and that the adaptation of the learning process to the learners knowledge significantly improves the learning performance.

Relation to WP Self-understanding and self-extension, which are addressed in the active learning framework, are the main topics of the project, and active learning is the main topic of Task 5.3 in WP 5, therefore the paper tackles the central issues of this workpackage and the project as whole.

2.2 Kristan and Leonardis "Online Discriminative Kernel Density Estimation With Gaussian Kernels"

Bibliography M. Kristan and A. Leonardis: "Online Discriminative Kernel Density Estimation With Gaussian Kernels". Submitted for journal publication, 2011.

Abstract We propose an approach for a supervised online estimation of probabilistic discriminative models. The method is based on the recently proposed online Kernel Density Estimation (oKDE) framework which produces Gaussian mixture models and allows adaptation using only a single data point at a time. The oKDE builds reconstructive models from the data and maintains its complexity low by compressing the models from time to time. We propose a new cost function that measures loss of interclass discrimination during compression, thus guiding the compression towards simpler models that still retain discriminative properties. We call the resulting method an online discriminative Kernel Density Estimator (odKDE).We compare the odKDE to oKDE, batch state-of-the-art KDEs and support vector machine (SVM) on standard publicly-available datasets. The odKDE achieves comparable classification performance to that of best batch KDEs and SVM, while allowing online adaptation, and produces models of lower complexity than the oKDE.

Relation to WP This paper proposes the underlying methodology which is used in this workpackage for online construction of mutually-exclusive concepts such as particular colors and spatial relations. The proposed method constructs discriminative models for maximal classification performance, but at the same time also keeps the models in their generative form. The fact that the models are generative is crucial for knowledge revision during active learning, since the generative nature allows the robot to revise its knowledge models, perform hallucination and detect gaps and uncertainties in the knowledge.

2.3 Prankl et al. "3D Piecewise Planar Object Model for Robotics Manipulation"

Bibliography J. Prankl, M. Zillich, M. Vincze: "3D Piecewise Planar Object Model for Robotics Manipulation", Proc. Int. Conf. Robotics and Automation (ICRA), pages 1784–1790, 2011

Abstract Man made environments are abundant with planar surfaces which have attractive properties for robotics manipulation tasks and are a prerequisite for a variety of vision tasks. This work presents automatic online 3D object model acquisition assuming a robot to manipulate the object. Objects are represented with piecewise planar surfaces in a spatio-temporal graph. Planes once detected as homographies are tracked and serve as priors in subsequent images. After reconstruction of the planes the 3D motion is analyzed and initial object hypotheses are created. In case planes start moving independently a split event is triggered, the spatio-temporal object graph is traced back and visible planes as well as occluded planes are assigned to the most probable split object. The novelty of this framework is to formalize Multibody Structure-and-Motion (MSaM), that is, to segment interest point tracks into different rigid objects and compute the multiple view geometry of each object, with Minimal Description Length (MDL) based on model selection of planes in an incremental manner. Thus, object models are built from planes, which directly can be used for robotic manipulation.

Relation to WP Identifying which parts of the scene make up individual objects is the first step in autonomously extending the system's knowledge about objects. While plane pop-out as attentional operator is sufficient for table top scenes of limited complexity, only interacting with the environment allows the system to segment objects in more generic complex scenes. The work presented here allows to actively learn new object models. Objects are represented as sets of rigidly connected planar patches, which are segmented from the scene background by pushing against surfaces and observing the resulting motion. While pushing of objects was still done by hand in this work, next steps will be using the actual robot and include planning of pushes which result in an optimal information gain.

2.4 Zillich et al. "Knowing Your Limits - Self-Evaluation and Prediction in Object Recognition"

Bibliography M. Zillich, J. Prankl, T. Mörwald, M. Vincze: "Knowing Your Limits - Self-Evaluation and Prediction in Object Recognition", Proc. Int. Conf. on Intelligent Robots and Systems (IROS), 2011

Abstract Allowing a robot to acquire 3D object models autonomously not only requires robust feature detection and learning methods but also mechanisms for guiding learning and assessing learning progress. In this paper we present probabilistic measures for observed detection success, predicted detection success and the completeness of learned models, where learning is incremental and online. This allows the robot to decide when to add a new keyframe to its view-based object model, where to look next in order to complete the model, predicting the probability of successful object detection given the model trained so far as well as knowing when to stop learning.

Relation to WP Representing not only knowledge but also where knowledge is missing and how it can be completed is one of the central themes in CogX. The above work extends previous methods of online incremental learning of object models with measures to quantify observed detection success, predicted detection success and model completeness. This allows the system to actively plan for knowledge gathering actions in order to complete partial object models. In that respect this work is also relevant for WP4, specifically Task 4.2: General planning of information gathering and dialogue actions (as reported in DR.4.3), and Task 4.3: Planning for active learning.

2.5 Ridge et al. “Self-supervised Cross-Modal Relevance Learning Vector Quantization”

Bibliography B. Ridge, D. Skočaj, and A. Leonardis: “Self-supervised Cross-Modal Relevance Learning Vector Quantization”. To be submitted for journal publication, 2011.

Abstract Given the absence of category labels or the expense of acquiring them in many applications, self-supervised learning algorithms that discover and exploit categories autonomously are an important consideration in machine learning. It is not always obvious how such self-supervision should proceed, particularly in the online learning setting, though in the context of cross-modal learning, where data samples co-occur across multiple modalities or views, the idea becomes more tractable. In this paper, we present a self-supervised cross-modal learning framework that employs codebook layers of prototypes to cluster in separate modalities that are connected via a cross-modal co-occurrence mapping. As data co-occur in multiple modalities online, we demonstrate how the category clusters that emerge in the data may be used to drive self-supervised learning. In our setting, we employ a modified form of learning vector quantization that may be trained using class probabilities instead of class labels, an effective means of bootstrapping self-supervised learning when the category clusters are not yet fully-formed. We also employ both a feature relevance determination mechanism and a means of culling ineffective prototypes, thus facilitating rapid learning over few training epochs. We demonstrate the effectiveness of our approach in experiments on both simulated data and data gathered from a cognitive robotics object affordance learning setting.

Relation to WP This paper addresses how cross-modal learning can be performed at a low-level where features are streaming from multiple sensory modalities online and co-occurring across these modalities. It demonstrates how not only can higher-level concepts be formed in this setting, but also how they can be used to drive self-supervised learning at this level. This is commensurate with the goals of WP5, which strives to analyse the problems of low-level cross-modal learning and online learning.

2.6 Vrečko et al. "Modeling Binding and Cross-modal Learning in Markov Logic Networks"

Bibliography A. Vrečko, A. Leonardis and D. Skočaj: "Modeling Binding and Cross-modal Learning in Markov Logic Networks". Submitted for journal publication, 2011

Abstract Binding – the ability to combine two or more modal representations of the same entity into a single shared representation is vital for every cognitive system operating in a complex environment. In order to successfully adapt to changes in an dynamic environment the binding mechanism has to be supplemented with cross-modal learning. In this paper we define the problems of high-level binding and cross-modal learning. By these definitions we model a binding mechanism in a Markov logic network and define its role in a cognitive architecture. We evaluate a prototype binding system online, using three different inference methods.

Relation to WP The paper addresses the problems of high-level cross-modal binding and learning, as defined in WP 5. It defines both problems and shows, using the Markov logic networks, how can high-level cross-modal associations be learned in this framework.

2.7 Roa et al. "Robust Vector Quantization for Inference of Substochastic Sequential Machines"

Bibliography S. Roa and G.-J. Kruijff: "Robust Vector Quantization for Inference of Substochastic Sequential Machines". Technical Report, 2011.

Abstract The article explores the problem of discretizing the continuous evolution of a dynamical system. A probabilistic discrete state, input and output space representation of the system, together with probabilistic transition functions, can be learned by the proposed algorithm. The method is based on the CrySSMEx algorithm for extracting substochastic finite state machines and a new Vector quantization algorithm. Experiments on Vector quantization were performed with artificial data generated by using gaussian noise distributions. Noisy automata were also used to test the algorithm and corresponding probabilistic finite state machines were extracted.

Relation to WP The paper is related to Tasks 5.3 and 5.4. The algorithm described can be used either for categorical learning or prediction learning. This is a preliminary work that can potentially be extended for active learning.

References

[1] C. Campbell, N. Cristianini, and A. Smola. Query learning with large margin classifiers. In *Proceedings of the Seventeenth International Conference on Machine Learning*, page 111118, 2000.

[2] N. Cebron and M. R Berthold. Active learning for object classification: from exploration to exploitation. *Data Mining and Knowledge Discovery*, 18(2):283299, 2009.

[3] A. Chella, M. Frixione, and S. Gaglio. A cognitive architecture for artificial vision. *Artif. Intell.*, 89(1-2):73–111, 1997.

[4] S. Chernova and M. Veloso. Interactive policy learning through confidence-based autonomy. *Journal of Artificial Intelligence Research*, 34:1–25, 2009.

[5] M. H. Coen. Cross-Modal clustering. In *Proceedings Of The National Conference On Artificial Intelligence*, volume 20, page 932. Menlo Park, CA; Cambridge, MA; London; AAAI Press; MIT Press; 1999, 2005.

[6] M. H. Coen. *Multimodal dynamics: self-supervised learning in perceptual and motor systems.* PhD thesis, Massachusetts Institute of Technology, Dept. of Electrical Engineering and Computer Science, 2006.

[7] M. H. Coen. Self-Supervised acquisition of vowels in american english. In *Proceedings Of The National Conference On Artificial Intelligence*, volume 21, page 1451. Menlo Park, CA; Cambridge, MA; London; AAAI Press; MIT Press; 1999, 2006.

[8] D. Cohn, L. Atlas, and R. Lander. Improving generalization with active learning. *Machine Learning*, 15(2):201221, 1994.

[9] J. Costeira and T. Kanade. A multi-body factorization method for motion analysis. In *5th Int. Conf. on Computer Vision*, pages 1071–1076, 1995.

[10] J. de Greeff, F. Delaunay, and Belpaeme T. Human-robot interaction in concept acquisition: a computational model. In *Proceedings of the 2009 IEEE 8th International Conference on Development and Learning*, pages 1–6, Washington, DC, USA, 2009. IEEE Computer Society.

[11] V. R. de Sa and D. H. Ballard. Category learning through multimodality sensing. *Neural Computation*, 10:1097–1117, 1998.

[12] W. Fan, Y. Huang, H. Wang, and P. S Yu. Active mining of data streams. In *Proceedings of the Fourth SIAM International Conference on Data Mining*, pages 457–461, 2004.

[13] Sanja Fidler, Danijel Skočaj, and Aleš Leonardis. Combining reconstructive and discriminative subspace methods for robust classification and regression by subsampling. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 28:337–350, March 2006.

[14] Andrew W. Fitzgibbon and Andrew Zisserman. Multibody structure and motion: 3-d reconstruction of independently moving objects. In *European Conference on Computer Vision*, pages 891–906. Springer-Verlag, 2000.

[15] M. Girolami and C. He. Probability density estimation from optimally condensed data samples. *ieeepami*, 25(10):1253–1264, 2003.

[16] Michael Grabner, Helmut Grabner, and Horst Bischof. Learning Features for Tracking / Tracking via Discriminative Online Learning of Local Features. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR'07)*, 2007.

[17] P. Hall, S. J. Sheater, M. C. Jones, and J. S. Marron. On optimal data-based bandwidth selection in kernel density estimation. *Biometrika*, 78(2):263–269, 1991.

[18] S. Harnad. The symbol grounding problem. *Physica D: Nonlinear Phenomena*, 42:335–346, 1990.

[19] A.D. Holub, P. Perona., and M. Burl. Entropy-based active learning for object recognition. In *Workshop on Online Learning for Classification, in conjunction with Conf. Comp. Vis. Pattern Recognition*, pages 1–8, 2008.

[20] S. Huang and Y. Dong. An active learning system for mining time-changing data streams. *Intelligent Data Analysis*, 11(4):401419, 2007.

[21] H. Jacobsson, N. Hawes, G-J. Kruijff, and J. Wyatt. Crossmodal content binding in information-processing architectures. In *Proc. of the 3rd ACM/IEEE International Conference on Human-Robot Interaction*, Amsterdam, March 2008.

[22] H. Jacobsson, N. Hawes, D. Skočaj, and G-J. Kruijff. Interactive learning and cross-modal binding - a combined approach. In *Symposium on Language and Robots*, Aveiro, Portugal, 2007.

[23] A. Kapoor, K. Grauman, R. Urtasun, and T. Darrell. Active learning with gaussian processes for object categorization. In *International Conference on Computer Vision*, 2007.

[24] M. Kristan and A. Leonardis. Online Discriminative Kernel Density Estimation With Gaussian Kernels. Submitted for journal publication.

[25] M. Kristan and A. Leonardis. Online discriminative kernel density estimation. In *International Conference on Pattern Recognition*, 2010.

[26] M. Kristan, A. Leonardis, and D. Skočaj. Multivariate Online Kernel Density Estimation with Gaussian Kernels. *Journal of Pattern Recognition*, 44(10-11):2630–2642, 2011.

[27] P. Lison, C. Ehrler, and G.-J. Kruijff. Belief modelling for situation awareness in human-robot interaction. In *Proceedings of the 19th IEEE International Symposium in Robot and Human Interactive Communication.* IEEE, 2010.

[28] A. McCallum and K. Nigam. Employing EM in pool-based active learning for text classification. In *Proceedings of the fifteenth international conference on machine learning*, pages 350–358, 1998.

[29] H. McGurk and J. MacDonald. Hearing lips and seeing voices. 1976.

[30] Giorgio Metta and Paul Fitzpatrick. Better vision through manipulation. *Adaptive Behavior*, 11:109–128, 2003.

[31] R. Miikkulainen. Dyslexic and Category-Specific aphasic impairments in a Self-Organizing feature map model of the lexicon. *Brain and Language*, 59(2):334–366, 1997.

[32] T. Mörwald, J. Prankl, A. Richtsfeld, M. Zillich, and M. Vincze. BLORT - The Blocks World Robotic Vision Toolbox. In *Best Practice in 3D Perception and Modeling for Mobile Manipulation (in conjunction with ICRA 2010)*, 2010.

[33] J. M. L. Murillo and A. A. Rodriguez. Algorithms for gaussian bandwidth selection in kernel density estimators. In *NIPS*, 2008.

[34] J. Ngiam, A. Khosla, M. Kim, J. Nam, H. Lee, and A. Y Ng. Multimodal deep learning. In *NIPS 2010 Workshop on Deep Learning and Unsupervised Feature Learning*, 2010.

[35] H. T Nguyen and A. Smeulders. Active learning using pre-clustering. In *Proceedings of the twenty-first international conference on Machine learning*, page 79, 2004.

[36] Kemal Egemen Ozden, Konrad Schindler, and Luc Van Gool. Multibody structure-from-motion in practice. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 32:1134–1141, 2010.

[37] M. Özuysal, V. Lepetit, F. Fleuret, and P. Fua. Feature Harvesting for Tracking-by-Detection. In *European Conference on Computer Vision*, volume 3953, pages 592–605, 2006.

[38] Qi Pan, Gerhard Reitmayr, and Tom Drummond. ProFORMA: Probabilistic Feature-based On-line Rapid Model Acquisition. In *Proc. British Machine Vision Conference (BMVC)*, 2009.

[39] J. Prankl, M. Zillich, B Leibe, and M. Vincze. Incremental Model Selection for Detection and Tracking of Planar Surfaces. In *BMVC*, pages 1784–1790, 2010.

[40] J. Prankl, M. Zillich, and M. Vincze. 3D Piecewise Planar Object Model for Robotics Manipulation. In *Proceedings of the IEEE International Conference on Robotics and Automation (ICRA)*, 2011.

[41] B. Ridge, A. Leonardis, and D. Skočaj. Self-Supervised Cross-Modal relevance learning vector quantization. To be submitted for journal publication., 2011.

[42] B. Ridge, D. Skočaj, and A. Leonardis. Self-supervised cross-modal online learning of basic object affordances for developmental robotic systems. In *Proceedings of the IEEE International Conference on Robotics and Automation (ICRA)*, Anchorage, USA, May 2010.

[43] Hayko Riemenschneider, Michael Donoser, and Horst Bischof. Robust Online Object Learning and Recognition by MSER Tracking. In *Proc. 13th Computer Vision Winter Workshop (CVWW)*, 2007.

[44] S. Roa and G.-J. Kruijff. Robust Vector Quantization for Inference of Substochastic Sequential Machines. Technical report, 2011.

[45] S. Roa and G.-J.M. Kruijff. Offline and active gradient-based learning strategies in a pushing scenario. In *International Workshop on Evolutionary and Reinforcement Learning for Autonomous Robot Systems 2010. ERLARS 2010*, pages 29–34, Lisboa, Portugal, 2010.

[46] S. Roa and G.-J.M. Kruijff. On the automatic entropy-based construction of probabilistic automata in a learning robotic scenario. In *Robotics: Science and Systems 2010 Workshop: Towards Closing the Loop: Active Learning for Robotics*, 2010.

[47] Peter M. Roth, Michael Donoser, and Horst Bischof. Tracking for Learning an Object Representation from Unlabeled Data. In *Proc. 11th Computer Vision Winter Workshop (CVWW)*, pages 46–51, 2006.

[48] D. Roy. Learning visually-grounded words and syntax for a scene description task. *Computer Speech and Language*, 16(3-4):353–385, 2002.

[49] D. Roy. Grounding words in perception and action: computational insights. *TRENDS in Cognitive Sciences*, 9(8):389–396, 2005.

[50] Konrad Schindler, David Suter, and Hanzi Wang. A model-selection framework for multibody structure-and-motion of image sequences. *Int. J. Comput. Vision*, 79(2):159–177, 2008.

[51] G. Schohn and D. Cohn. Less is more: Active learning with support vector machines. page 839846, 2000.

[52] H. S. Seung, M. Opper, and H. Sompolinsky. Query by committee. In *Proceedings of the fifth annual workshop on Computational learning theory*, page 287294, 1992.

[53] D. Skočaj, M. Kristan, and A. Leonardis. Formalization of different learning strategies in a continuous learning framework. In *Proceedings of the Ninth International Conference on Epigenetic Robotics; Modeling Cognitive Development in Robotic Systems*, pages 153–160, Venice, Italy, November 12-14 2009.

[54] Danijel Skočaj, Matej Kristan, and Aleš Leonardis. About different active learning approaches for acquiring categorical knowledge. In *Proceedings Of The Twentieth International Electrotechnical and Computer Science Conference*, 2011.

[55] Danijel Skočaj, Matej Kristan, Alen Vrečko, Marko Mahnič, Miroslav Janíček, Geert-Jan M. Kruijff, Marc Hanheide, Nick Hawes, Thomas Keller, Michael Zillich, and Kai Zhou. A system for interactive learning in dialogue with a tutor. 2011. Accepted for publication at IROS 2011.

[56] L. Steels. *The Talking Heads Experiment. Volume 1. Words and Meanings.* Laboratorium, Antwerpen, 1999.

[57] P. H. S. Torr. Geometric motion segmentation and model selection. *Phil. Trans. Royal Society of London A*, 356:1321–1340, 1998.

[58] M. Uray, D. Skočaj, P. Roth, H. Bischof, and A. Leonardis. Incremental LDA learning by combining reconstructive and discriminative approaches. In *British machine vision conference 2007*, pages 272–281, 2007.

[59] L. Vacchetti, V. Lepetit, and P. Fua. Stable Real-Time 3D Tracking using Online and Offline Information. *PAMI*, 2004.

[60] R. Vidal and Y. Ma. A unified algebraic approach to 2-D and 3-D motion segmentation. In *European Conference on Computer Vision (ECCV)*, pages 1–15, 2004.

[61] A. Vrečko, A. Leonardis, and D. Skočaj. Modeling binding and cross-modal learning in Markov Logic Networks. 2011. Submitted for journal publication.

[62] A. Vrečko, D. Skočaj, N. Hawes, and A. Leonardis. A computer vision integration model for a multi-modal cognitive system. In *Proc. of the 2009 IEEE/RSJ Int. Conf. on Intelligent RObots and Systems*, pages 3140–3147, St. Louis, Oct. 2009.

[63] J. L. Wyatt, A. Aydemir, M. Brenner, M. Hanheide, N. Hawes, P. Jensfelt, M. Kristan, G.-J. M. Kruijff, P. Lison, A. Pronobis, K. Sjöö, D. Skočaj, A. Vrečko, H. Zender, and M. Zillich. Self-understanding & self-extension: A systems and representational approach. *IEEE Transactions on autonomous mental development*, 2010. Accepted for publication.

[64] Z. Xu, K. Yu, V. Tresp, X. Xu, and J. Wang. Representative sampling for text classification using support vector machines. In *Proceedings of the 25th European conference on IR research*, pages 393–407, 2003.

[65] Kai Zhou, Andreas Richtsfeld, Michael Zillich, and Markus Vincze. From Holistic Scene Understanding to Semantic Visual Perception: A Vision System for Mobile Robot. In *ICRA 2011 Workshop: Semantic Perception, Mapping and Exploration (SPME)*, Shanghai, 2011.

[66] Kai Zhou, Andreas Richtsfeld, Michael Zillich, Markus Vincze, Alen Vrečko, and Danijel Skočaj. Visual Information Abstraction for Interactive Robot Learning. In *The 15th International Conference on Advanced Robotics (ICAR)*, Tallinn, Estonia, 2011.

[67] X. Zhu, P. Zhang, X. Lin, and Y. Shi. Active learning from stream data using optimal weight classifier ensemble. *Systems, Man, and Cybernetics, Part B: Cybernetics, IEEE Transactions on*, (99):1–15, 2010.

[68] Michael Zillich, Johann Prankl, Thomas Mörwald, and Markus Vincze. Knowing Your Limits - Self-evaluation and Prediction in Object Recognition. In *IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 2011.

3D Piecewise Planar Object Model for Robotics Manipulation

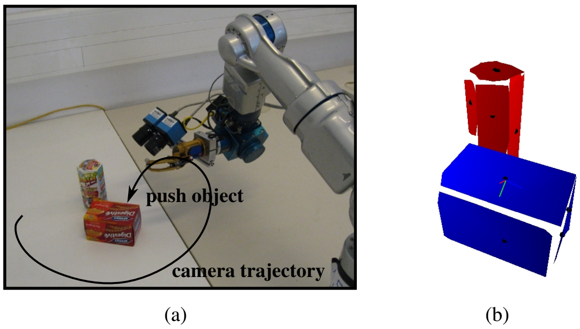
Johann Prankl, Michael Zillich, and Markus Vincze

***Abstract*— Man-made environments are abundant with planar surfaces which have attractive properties for robotics manipulation tasks and are a prerequisite for a variety of vision tasks. This work presents automatic on-line 3D object model acquisition assuming a robot to manipulate the object. Objects are represented with piecewise planar surfaces in a spatio-temporal graph. Planes once detected as homographies are tracked and serve as priors in subsequent images. After reconstruction of the planes the 3D motion is analyzed and initial object hypotheses are created. In case planes start moving independently a split event is triggered, the spatio-temporal object graph is traced back and visible planes as well as occluded planes are assigned to the most probable split object. The novelty of this framework is to formalize Multi-body Structure-and-Motion (MSaM), that is, to segment interest point tracks into different rigid objects and compute the multiple-view geometry of each object, with Minimal Description Length (MDL) based on model selection of planes in an incremental manner. Thus, object models are built from planes, which directly can be used for robotic manipulation.**

I. INTRODUCTION

Increasing interest in enabling robot manipulators to operate in everyday environments leads to the problem of how to acquire object models for manipulation. One does not want to specify all objects and possible obstacles in advance but allow the robot to actively acquire its own models, using the robot's ability to change view points and to interact with the scene. Many objects in man-made environments consist of planar surfaces, such as tables, shelves or box-shaped packaging. Also curved surfaces can be approximated with sufficient accuracy for most robotics tasks with piecewise planar surfaces, as is common in modelling for computer graphics. Planar surface patches support reasoning about object properties important for manipulation, such as contact points and friction cones, in contrast to models based on distinctive interest points, which typically lead to sparse point sets and are more suitable for recognition.

Our overall goal is to build a cognitive robotic experimentation framework. The rationale behind our system is to enable human tutor driven learning-by-showing as well as completely automatic on-line model acquisition by the robot (see Fig. 1). Schindler et al. [1] use a model selection framework for multibody Structure-from-Motion estimation of image sequences. In contrast we use model selection to detect piecewise planar surfaces. We describe plane hypotheses using the 2D projective transformation (homography) computed from four interest point pairs in two uncalibrated images. In the first step our model is simpler than that of Schindler, but it enables the robot to interact in more realistic environments. After 3D reconstruction of the planes the motion is analyzed and initial object hypotheses are created. In case planes start moving independently a split event is triggered and current visible planes as well as already occluded planes, stored in a temporal object hypotheses graph are assigned to the most plausible split object model. For assignment of the planes a Minimal Description Length (MDL) criterion formalizing the colour distribution and the distance of planes within an object is used. Hence, at each timestamp piecewise planar object models of the current scene are available, which directly can be used for robotic manipulation. In case an interest point descriptor, such as the popular SIFT proposed by Lowe [2] is computed this model can directly be used for object recognition and full pose registration from a single image (see [3]).



(a) (b)

Fig. 1. Example scenario we used to test our system, where a camera moves around objects and pushes them. The image shows a stereo setup, from which we use only a single camera.

After a review of the related work, we give an overview of the system in Section II and its core parts, namely the plane detection (Section II-A), Structure-from-Motion (Section II-B), merging of planes (Section II-C) and splitting of piecewise planar object models (Section II-D). Finally, results of the experiments are shown in Section III.

A. Related work

Although this work focuses on a framework for modelling objects we first want to mention some literature from the field of active vision, which is the motivation for our experiments shown later on and then tackle related work our system is based on. The early attempts on Active Vision, that is an active observer whose purpose is to improve the quality of the perceptual results, goes back to [4], [5]. In [5] Aloimonos et. al stressed that an active observer can solve basic vision problems in a much more efficient way. They introduce a general methodology, in which they believe low-level vision problems should be addressed. Metta et al. [6]

J. Prankl, M. Zillich, and M. Vincze are with the Automation and Control Institute, Vienna University of Technology, Austria {prankl,zillich,vincze}@acin.tuwien.ac.at

developed an active strategy for a robot to acquire visual experience through simple experimental manipulation. The experiments are oriented towards determining which parts of the environment are physically coherent, that is, which parts will move together, and which are more or less independent. Our experiments are similar, but in contrast to Meta, who studies the causal chains of events we focus on learning a 3D piecewise planar object model triggered by motion events.

The basic parts of our object model are planes. Detecting planes in uncalibrated image sequences is well studied. Most approaches use a hypothesize-and-test framework. A popular method for detecting multiple models is to use the robust estimation method RANSAC [7], to sequentially fit the model to a data set and then to remove inliers. To generate plane hypotheses Vincent et al. [8] use groups of four points which are likely to be coplanar to compute the homography. To increase the likelihood that the points belong to the same plane they select points lying on two different lines in an image. In contrast Kanazawa et al. [9] define a probability for feature points to belong to the same plane using the Euclidean distance between the points. Both approaches use a RANSAC scheme, iteratively detect the dominant plane, remove the inliers and precede with the remaining interest points. The success of the plane computation depends on the coplanarity of four matched points. In [10], [11] different strategies are proposed to sequentially reduce the set of points/lines to three pairs. More recent approaches, such as proposed by Toldo et al. [12] and Chin et al. [13], concentrate on robust estimation of multiple structures to treat hypotheses equally and do not favour planes detected first over subsequent planes by greedily consuming features. These approaches have to create plane hypotheses independently of each other and thus it is not possible to restrict the search space, which leads to higher computational complexity. Our method is most similar to the approach by Prankl et al. [14], who propose incremental model selection based on the MDL principle to overcome these drawbacks.

The planes, represented by homographies, are the basic entities for 3D reconstruction and for merging/splitting to create the final object model. While classical Structure-from-Motion moving through a static scene is essentially solved in a coherent theory [15] and several robust systems exist, in recent years, researchers focused on dynamic scenes composed of rigidly moving objects. The solutions available so far can be broadly classified into algebraic methods [16], [17], which exploit algebraic constraints satisfied by all scene objects, even though they move relative to each other, and non-algebraic methods [18], [19], which essentially combine rigid SfM with segmentation. Most related to our system are the methods proposed by Schindler [1] and by Ozden [20]. They use interleaved segmentation and 3D reconstruction of tracked features into independent objects. Instead of directly sampling features and generating 3D object hypotheses, we incrementally cluster features to planes in 2D using homographies and then reconstruct and merge/split planes into independently moving objects in 3D. Thus in the first step we use a simpler model to more robustly cluster tracked features to planes, followed by a second step, reconstruct, merge/split planes and create the final object model. Finally, instead of a sparse point cloud we get a dense representation with planes, which directly can be used for robotic manipulation.

Algorithm 1 Piecewise planar object modelling pipeline

```
1) Instantiate new interest points (IPs)
2) Track interest points
3) Track planes modelled by homographies and try to
   estimate 3D motion for existing objects
   **if** plane does not support 3D motion **then**
     • trigger split event and create new objects from
       current and past keyframes
   **end if**
   **if** average displacement of the IPs $< d$ pixels **then**
     • **goto** step 1
   **else**
     • init a new keyframe and continue
   **end if**
4) Detect and renew planes
5) Merge and reconstruct planes greedily
   **if** new plane supports active object motion model **then**
     • insert plane
   **else**
     • create new 3D object and motion model (SfM)
   **else if**
6) Refine objects using incremental bundle adjustment
7) **goto** step 1
```

II. SYSTEM

We developed a method to create piecewise planar object models from an uncalibrated image sequence on the fly. The idea is to use a simple model for clustering interest points to planes, which is combined with tracking in an interleaved way and then reconstruct and merge planes to create object hypotheses. In case planes start moving independently a split event is triggered and the history of that object hypothesis is reviewed to assign current visible planes as well as already occluded planes to the best split hypothesis. Hence, we can handle more complex scenes and additionally we get a structural model of planes instead of a sparse point cloud. Algorithm 1 gives a detailed outline of the piecewise planar object modelling pipeline and Fig. 2 depicts the events, that is detection, tracking, merging and splitting of planes.

A. Plane detection using homographies

The idea is to cluster interest points at image level using the 2D projective transformation (homography). Interest points of a plane cluster belong to the same object with a high probability and thus build a reliable part for the following 3D reconstruction.

1) Algorithm: We embedded Minimal Description Length (MDL) based model selection in an iterative scheme. Existing planes, tracked from the last images or created in the last iteration compete with newly created hypotheses to

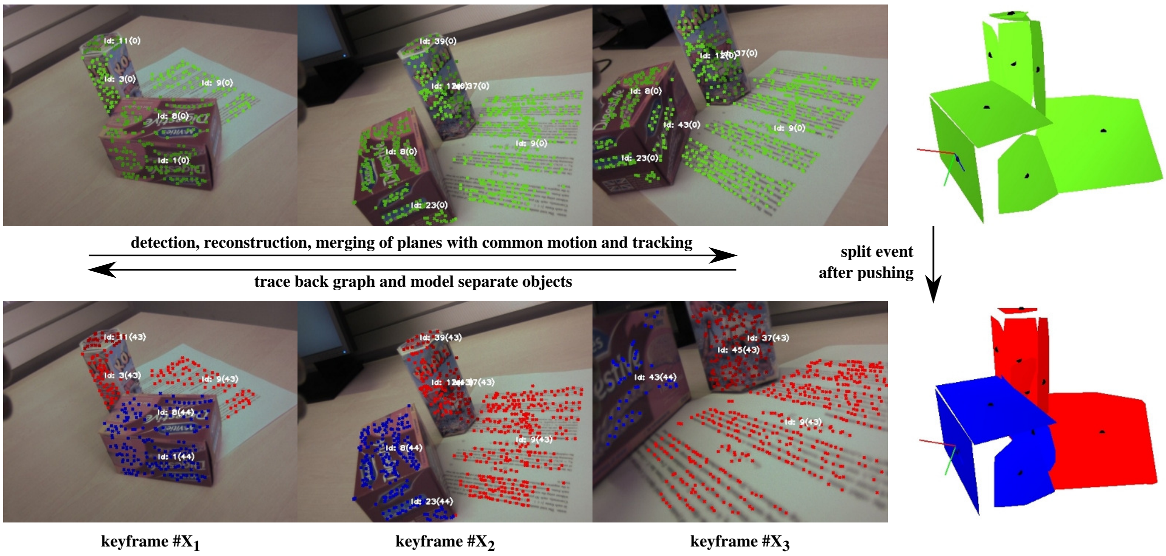


Fig. 2. The upper row shows three keyframes of sequence 1 (897 frames) with detected planes in green which are merged because of common 3D motion. The brightness of the interest points indicates the assignment to different planes. After the gripper (two black dots on the left image border) pushes the plane 43 the keyframe-graph is traced back and the object model (44) and the background object (43) are created (lower image row). Changing plane id's of the top surface of the hexagonal object indicate that planes represented by homographies are substituted with better explanations.

Algorithm 2 Plane detection and tracking

```
$P \leftarrow P_{tracked}$, $P' \leftarrow 0$
$k \leftarrow 0$, $\epsilon \leftarrow M/N$, $S \leftarrow 0$
while $\eta = (1 - \epsilon^M)^k \geq \eta_0$ do
    $P' \leftarrow P$
    Add $Z$ random plane hypotheses to $P'$
    Select plane hypotheses from $P'$ and store in $P$
    Count number of explained interest points (inliers) $I$ for $P$
    if $I > I_{max}$ then
        $I_{max} \leftarrow I$
        $\epsilon \leftarrow I_{max}/N$
    end if
    $k \leftarrow k + 1$
end while
```

ensure that interest points are assigned to the best currently available hypothesis. Additionally hypothesis generation is guided to unexplained regions. This method avoids the bias towards dominant planes typical for iterative methods, and it limits the search space which leads to a faster explanation of the entire image in terms of piecewise planar surfaces. Algorithm 2 shows the proposed method for plane detection and tracking. P is initialized with tracked planes of the last image. Then in each iteration a small number Z of new plane hypotheses P' is computed which have to compete with the selected hypotheses P of the last iteration. The termination criterion is based on the true inlier ratio ϵ and the number of samples M which are necessary to compute the homographies. As long as we do not know these values we use the best estimate available up to now. For ϵ that is the ratio of the number of explained interest points I_{max} of the current best plane hypotheses and the number of matched interest points N to explain. Accordingly M is the number of plane hypotheses currently selected multiplied with the minimal set of interest points $m = 4$ to compute one homography. Furthermore in Algorithm 2 k is the number of iterations, η stands for the probability that no correct set of hypotheses is found and η_0 is the desired failure rate. Due to the incremental scheme it is possible to guide the computation of new hypotheses to unexplained regions.

2) Minimal Description Length based model selection: In each iteration selected homographies of the last iteration have to compete with newly sampled hypotheses. For the selection, the idea is that the same feature cannot belong to more than one plane. Thus an over-complete set of homographies is generated and the best subset in terms of a Minimum Description Length criterion is chosen. The basic mathematical tool for this is introduced in [21] and adapted in [22]. To select the best model, the savings for each hypothesis h are expressed as

$$S_h = S_{data} - \kappa_1 S_{model} - \kappa_2 S_{error} \tag{1}$$

where in our case S_{data} is the number of interest points N explained by h and S_{model} stands for the cost of coding the model itself. In our case we only have one model (the homography of a plane) and thus $S_{model} = 1$. S_{error} describes the cost for the error added, which we express with the log-likelihood over all interest points f_k of the plane hypothesis h. Experiments have shown that the Gaussian error model in conjunction with an approximation of the log-likelihood comply with the expectations. κ_1 and κ_2 are constants to weight the different factors. Finally the merit term of a model results in

$$s_{ii} = S_h = -\kappa_1 + \sum_{k=1}^{N} \left((1 - \kappa_2) + \kappa_2 p(f_k|h)\right), \tag{2}$$

where $p(f_k|h)$ is the likelihood that an interest point belongs to the plane hypothesis h. Details for the derivation of (2) can be found in [14]. An interest point can only be assigned to one model. Hence, overlapping models compete for interest points which can be represented by interaction costs

$$s_{ij} = -\frac{1}{2} \sum_{f_k \in h_i \cap h_j} ((1-\kappa_2) + \kappa_2 \min\{p(f_k|h_i), p(f_k|h_j)\}) \,. \tag{3}$$

Finding the optimal possible set of homographies for the current iteration leads to a Quadratic Boolean Problem (QBP)[1]

$$\max_n \mathbf{n}^T S \mathbf{n} \;,\; S = \begin{bmatrix} s_{11} & \cdots & s_{1N} \\ \vdots & \ddots & \vdots \\ s_{N1} & \cdots & s_{NN} \end{bmatrix} \tag{4}$$

where $\mathbf{n}$= $[n_1, n_2, \cdots, n_N]$ stands for the indicator vector with $n_i = 1$ if a plane hypothesis is selected and $n_i = 0$ otherwise. Because of iteratively adding plane hypotheses the number of planes leading to the QBP is tractable. Furthermore experiments have shown that s greedy approximation of the QBP gives good results and thus the solution can be found very fast.

B. Structure from Motion (SfM)

The final results of our system are 3D models of objects. Approaching this goal from object reconstruction our system is strongly related to the dynamic SfM frameworks [1], [20]. In [20] Ozden et al. defined the following requirements:

1) Determine the number of independently moving objects of a sequence
2) Segment the feature tracks into different moving objects in each frame
3) Compute their 3D structure and the camera motion for the frame
4) Resolve geometric ambiguities
5) Robustness to short feature tracks due to occlusion, motion blur, etc.
6) Scale to realistic recording times

They propose interleaved segmentation and 3D reconstruction of the feature tracks into independent objects. Instead of directly sampling features and generating 3D object hypotheses we incrementally cluster features to planes in projective space and track them. Thus the first two items as well as the third are approached more robustly with a simpler model in 2D followed by reconstruction, clustering and splitting of planes to objects in 3D.

For reconstruction of the planes we use a standard SfM pipeline similar to Nister et al. [23]. The nonlinear refined homography is directly decomposed to initialize the first camera pose (see [24]). In the following frames the relative motion from C^{-1} to C is estimated using RANSAC [7] and a direct least squares solution between the two point sets (cp. Haralick et al. [25]). A sparse bundle adjustment implementation by Lourakis [26] over the last N frames is used to refine camera pose and 3D points of the plane. Once a plane is reconstructed our algorithm tries to incorporate planes greedily in case of consistent motion.

C. Merging of planes with consistent motion

Merging of planes amounts to checking whether the motion of a new plane is consistent with the motion of an existing object. In contrast to Schindler et al. [1] we aim at building individual object models and thus, once an object is split we do not merge them again if they start moving together. Hence, it is possible that several objects with the same motion are tracked at the same time and a new plane moves consistent with more than one object. If merging would be done only because of consistent motion this plane would be assigned to one of the objects just by chance. Therefore a pseudo-likelihood depending on motion, colour and the 3D interest point adjacency is introduced and planes are assigned to the object with a higher probability. Analogous to (2) the formulation

$$\begin{aligned} p_{ij} &= -\nu_1 + \frac{1}{N}\sum_{k=1}^{N} \Big((1-\nu_2) + \nu_2 p(f_{i,k}^{proj}|H_j)\Big) \\ &+ \nu_3 p^*(a_i|A_j) \end{aligned} \tag{5}$$

is used to assign the plane i to the object j with the higher likelihood p_{ij}, where $p(f_{i,k}^{proj}|H_j)$ is the probability that an interest point of a plane i belongs to the 3D object H_j. This is modelled using a Gaussian error model. The camera pose of object j is used to compute 3D points for plane i and the projections are compared to the corresponding tracked image points. ν denote constants to weight the different factors, where ν_1 is an offset which must be reached to be considered as moving together and ν_3 is a weighting factor to reduce the influence of the appearance model $p^*(a_i|A_j)$ and primarily merge depending on the motion. The appearance model

$$\begin{aligned} p^*(a_i|A_j) &= \frac{1}{N}\sum_{k=1}^{N} \big((1-\nu_4) + \nu_4 p(f_{i,k}^{3D}|H_j)\big) \\ &+ \log(p(c_i|C_j)) \end{aligned} \tag{6}$$

combines interest point adjacency and the colour in a probabilistic manner. Interest point adjacency (first term) is based on a probabilistic voting scheme. For this a neighbourhood graph of all currently available 3D points is constructed. This graph is used to compute the mean μ and the standard deviation σ of the length of edges which connect points of the same plane. Then μ and σ are used to compute Gaussian votes $p(f_{i,k}^{3D}|H_j)$, where each 3D point of a target plane votes for the nearest object and thus the object which is close to the plane accumulates more votes and gets a higher probability that the plane belongs to that object. The second term models the colour distribution of the objects. For this we build the $8 \times 8 \times 8$ colour histogram c_i of the target plane i and the histogram C_j of the object j to which the plane should be assigned. We use normalized rgb colours to

[1] QBP assumes pairwise interaction, which in our case can be violated. But this is still a good approximation because interaction always increases cost, yielding a desirable bias against weak hypotheses.

be insensitive to brightness differences of object planes. The border of the plane is approximated by the convex hull of the interest points. For comparison of colour models we use the Bhattacharyya coefficient

$$p(c_i|C_j) \sim \sum_q \sqrt{c_i(q)C_j(q)}. \tag{7}$$

Hence, the probability of a plane i which has to be assigned to an object j consists of a probabilistic vote of each interest point to the nearest object and a probability describing the colour similarity. Being aware that merging of planes based on colour and 3D interest point adjacency is a critical point, experiments have shown that for our scenarios, where only a few objects are modelled simultaneously, this is a good second merging criterion next to motion.

D. *Separating planes in case of different motions*

We trigger object modelling if an object separates, that is, planes start moving differently. Therefore in (5), which is used to continuously test if planes start moving separately, ν_3 is set to zero and first visible planes are separated only because of motion without using colour and shape. Then past observations, where the planes had a common motion are examined. If the camera moves around an object and planes could not be tracked because of (self-)occlusion (6) is used to assign them to the new object with the higher probability. Therefore we represent objects in a keyframe[2] based graph structure. Each observation of an object is assigned to a keyframe and linked to an observation in the previous as well as in the next keyframe. Thus the object itself is stored distributed within the graph structure and each observation holds the current pose to the reference frame and the appearance modelled with interest points and the colour histogram. Fig. 2 depicts an event chain where planes are merged because of common motion, start moving separately and thus the object is split and new object models are built by tracing back the graph and assigning occluded planes to the object with the higher probability.

III. EXPERIMENTS

For all experiments we use a KLT-tracker [27]. In [28] it has been shown that a sub-pixel refinement essentially improves pose estimation. Hence, we use the affine refined location of the interest points with sub-pixel accuracy and finally compute a non-linear optimized homography using *homest* [29].

To test our system we use five videos each with about 800 frames. Motivated by our cognitive robotic scenarios the sequences show packaging of arbitrary shapes typically found in a supermarket (see Fig. 2). We placed two different objects on a table and manually moved camera and gripper around them in a way that one half of the objects is already occluded before the gripper pushes one object. The goal of the experiments is that our system detects the planes, reconstructs, tracks and merges them depending on common motion and finally, after pushing one object, creates two separate piecewise planar object models.

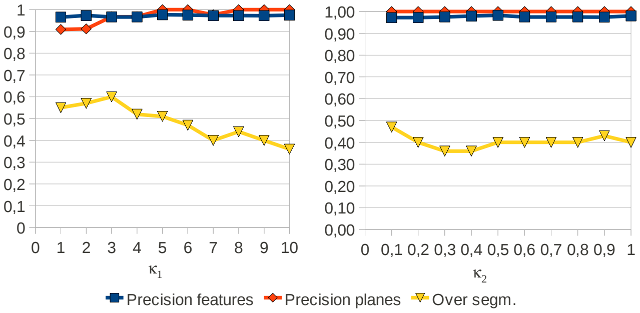


Fig. 3. Parameter optimisation

Three numbers are computed to compare the results, that is the feature based precision

$$p_{f,pr} = \frac{n_{f,tp}}{n_{f,tp} + n_{f,fp}} \tag{8}$$

which is the ratio of the number of inliers $n_{f,tp}$ correctly located on a ground truth plane and the total number of features per detected plane $n_{f,tp}+n_{f,fp}$. The second number is the over-segmentation-rate

$$p_{ov} = \frac{n_{p,fp}}{n_{p,tp} + n_{p,fp}} \tag{9}$$

per plane which indicates how often a plane is replaced during tracking. $n_{p,fp}$ the number of false positives is the number of detected planes minus the number of correctly detected planes $n_{p,tp}$. Furthermore we computed the plane based accuracy

$$p_{pl,pr} = \frac{n_{p,tp}}{n_{p,tp} + n_{p,fp}} \tag{10}$$

which describes the ratio of the correctly detected planes $n_{p,tp}$ and the total number of detected planes $n_{p,tp} + n_{p,fp}$.

A. *Plane detection*

To test the plane detection we selected 30 keyframes and manually marked a total number of about 150 planes. With the first video sequence we tested the behaviour of the parameters of our algorithm. Fig. 3 shows our performance measures for the parameter $\kappa_1 = [1...10]$ and $\kappa_2 = [0...1.]$. It can be seen that our algorithm is quite robust against variation of the parameters. Fig. 3 (left) shows, that the Parameter κ_1 mostly influences the over-segmentation-rate while the plane based precision slightly increases. The feature based precision $p_{f,pr}$ and the plane based precision $p_{pl,pr}$ are almost constant in Fig. 3 (right) and the over-segmentation-rate has a minimum for $\kappa_2 = 0.3$.

The results for all five videos are shown in Table I. It can be seen that our algorithm did not detect a totally wrong plane ($p_{pl,pr} = 1$) while in some cases interest points match a plane by chance ($p_{f,pr} \approx 0.97$). The over-segmentation-rate p_{ov} would be zero, if in the final 3D object model each manually marked plane is reconstructed by exactly one plane. Our plane detection/tracking algorithm is designed to

[2] In our system keyframes are a subset of frames of the whole video sequence, which are automatically selected for plane detection or in case a split event occurs.

sequence	$p_{f,pr}$	$p_{pl,pr}$	p_{ov}	time per frame [s]
1	0.97	1.0	0.40	0.25
2	0.98	1.0	0.25	0.18
3	0.93	1.0	0.43	0.15
4	0.99	1.0	NA	0.19
5	0.99	1.0	0.60	0.26

TABLE I

RESULTS OF OUR FIVE VIDEO SEQUENCES.

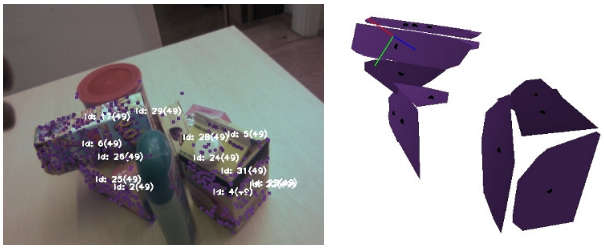


Fig. 8. Example image and reconstruction of a small more complex sequence which shows the limits of our system. Planes of the three dominant objects at the front are reconstructed, while the object at the centre of the image and the objects at the background are not detected because of low texture and too few features.

subdivide planes if a better explanation can be obtained in terms of smaller planes. The final object model consists of all these planes and thus is $p_{ov} \approx 0.4$. Furthermore the p_{ov} is not zero because sometimes the manually marked planes are indeed not flat but a little bit curved.

B. Reconstruction

Fig. 2, 4, 5, 6 and 7 show the qualitative results of our system. Planes merged to one object are drawn with the same colour, whereas the brightness of interest points indicates the assignment to different planes. In each figure the third image of each row shows the perspective of the camera shortly before/after the object is pushed and the last one depicts the reconstructed objects. Fig. 2 shows the whole event chain, that is, detection, reconstruction and merging of planes with a common motion coloured green and separating planes as they start moving independently (indicated in red and blue). In the Sequences 1, 2, 4 and 5, shown in Fig. 2, 4, 6 and 7 object modelling was successful and accurate as expected. The 3D reconstruction (right image of each row) shows that sometimes parts of an object, which we intuitively would mark as one plane are split. That is on the one hand, because these planes are indeed not flat but a little bit curved and on the other hand model selection within our plane detection algorithm replaces a plane in the following keyframes if a better, more complete/accurate plane is found. Fig. 5 shows one of the failures which might occur. These two objects have approximately the same height and thus one joined explanation was favoured instead of two separate. Fig. 7 and 8 show the limits of our system. Our reconstruction relies on planes modelled by homographies and thus for one plane a theoretical minimum number of five interest points are necessary ($4 + 1$ which supports the homography). Because of reliability issues we used a threshold of 10. Hence, in Fig. 7 even though a small plane is detected (shown in the middle image, plane with $id = 17$) the top of the cleaner bottle is completely lost. In Fig. 8 the object in the middle, which has hardly any texture and the finer scene details at the background are invisible for our system whereas the three prominent objects are nicely recovered.

IV. CONCLUSION AND FURTHER WORK

We explored how robot motion can be used to learn more about unknown objects in a home or service robot task. Using our approach it is possible to model the object surface from pushing the parts. If accidentally several objects are pushed, different motion will occur and they will be modelled as two different items. We formalize model selection with Minimal Description Length (MDL) to incrementally cluster features to planes in 2D using homographies and then reconstruct and merge/split planes into independently moving objects in 3D. Merging as well as splitting is triggered based on a probability which combines 3D motion, structure and colour information of the planes. Consistent with plane detection this is formalized with MDL. Instead of a sparse point cloud, which is typical for Multi-body Structure-and-Motion, we get a dense representation with planes, which directly can be used for robotic manipulation. For future work we want to introduce more complex object models where parts are linked with joints, e.g., scissors.

ACKNOWLEDGEMENT

The research leading to these results has received funding from the European Community's Seventh Framework Programme [FP7/2007-2013] under grant agreement No. 215181, CogX and No. 215821, GRASP.

REFERENCES

[1] K. Schindler, D. Suter, and H. Wang, "A model-selection framework for multibody structure-and-motion of image sequences," *Int. J. Comput. Vision*, vol. 79, no. 2, pp. 159–177, 2008.

[2] D. G. Lowe, "Distinctive image features from scale-invariant keypoints," *Int. J. Comput. Vision*, vol. 60, no. 2, pp. 91–110, 2004.

[3] T. Mörwald, J. Prankl, A. Richtsfeld, M. Zillich, and M. Vincze, "Blort - the blocks world robotic vision toolbox," in *Best Practice in 3D Perception and Modeling for Mobile Manipulation at ICRA 2010*, Anchorage, Alaska, 2010.

[4] R. Bajcsy, "Active perception," *Proceedings of the IEEE*, vol. 76, no. 8, pp. 966 –1005, aug. 1988.

[5] J. Aloimonos, I. Weiss, and A. Bandyopadhyay, "Active vision," *Int. J. of Computer Vision*, vol. 1, pp. 333–356, 1988.

[6] G. Metta and P. Fitzpatrick, "Better vision through manipulation," *Adaptive Behavior*, vol. 11, pp. 109–128, 2003.

[7] M. A. Fischler and R. C. Bolles, "Random sample consensus: a paradigm for model fitting with applications to image analysis and automated cartography," *Commun. ACM*, vol. 24, no. 6, pp. 381–395, 1981.

[8] E. Vincent and R. Laganiere, "Detecting planar homographies in an image pair," in *Image and Signal Processing and Analysis (ISPA).*, 2001, pp. 182 –187.

[9] Y. Kanazawa and H. Kawakami, "Detection of planar regions with uncalibrated stereo using distributions of feature points," in *British Machine Vision Conference (BMVC)*, 2004.

[10] M. Lourakis, A. Argyros, and S. Orphanoudakis, "Detecting planes in an uncalibrated image pair," in *British Machine Vision Conference (BMVC)*, 2002.

[11] J. Piazzi and D. Prattichizzo, "Plane detection with stereo images," in *IEEE International Conference on Robotics and Automation (ICRA)*, May 2006, pp. 922 –927.

[12] R. Toldo and A. Fusiello, "Robust multiple structures estimation with j-linkage," in *European Conference on Computer Vision (ECCV)*. Berlin, Heidelberg: Springer-Verlag, 2008, pp. 537–547.

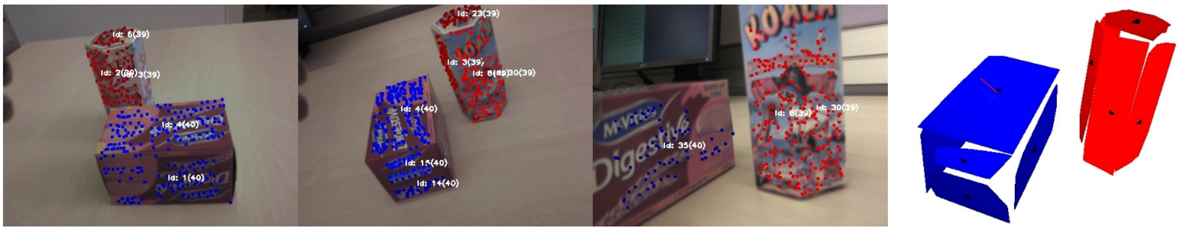


Fig. 4. Sequence 2 (715 frames).

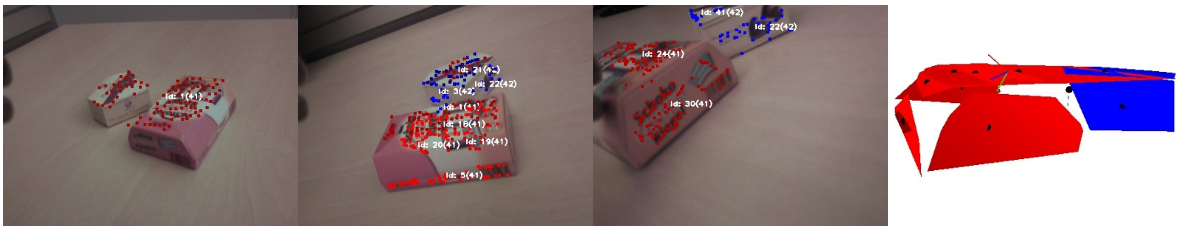


Fig. 5. Sequence 3 (543 frames).

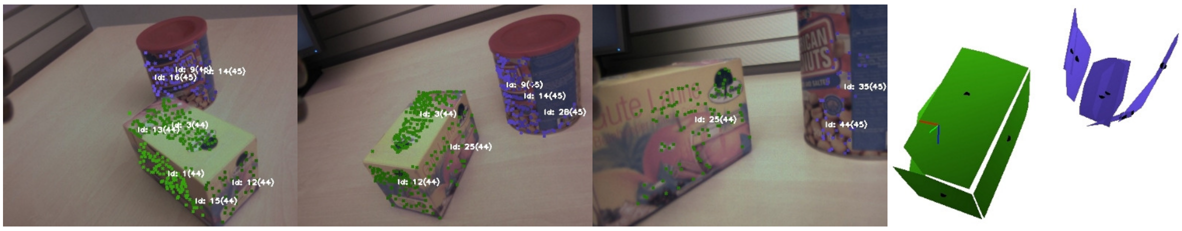


Fig. 6. Sequence 4 (870 frames).

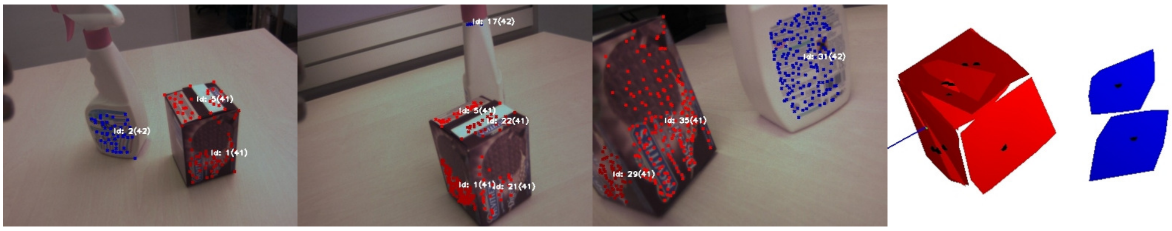


Fig. 7. Sequence 5 (811 frames).

[13] T.-J. Chin, H. Wang, and D. Suter., "Robust fitting of multiple structures: The statistical learning approach," in *IEEE International Conference on Computer Vision (ICCV)*, 2009.

[14] J. Prankl, M. Zillich, B. Leibe, and M. Vincze, "Incremental Model Selection for Detection and Tracking of Planar Surfaces," in *British Machine Vision Conference (BMVC)*, 2010, pp. 87.1–87.12.

[15] A. Hartley, R.; Zisserman, *Multiple View Geometry in computer vision*. Cambridge University Press, 2008.

[16] R. Vidal and Y. Ma, "A unified algebraic approach to 2-d and 3-d motion segmentation," in *European Conference on Computer Vision (ECCV)*. Springer Berlin / Heidelberg, 2004, vol. 3021, pp. 1–15.

[17] J. Costeira and T. Kanade, "A multi-body factorization method for motion analysis," jun. 1995, pp. 1071 –1076.

[18] P. H. S. Torr, "Geometric motion segmentation and model selection," *Phil. Trans. Royal Society of London A*, vol. 356, pp. 1321–1340, 1998.

[19] A. W. Fitzgibbon and A. Zisserman, "Multibody structure and motion: 3-d reconstruction of independently moving objects," in *European Conference on Computer Vision (ECCV)*. Springer-Verlag, 2000, pp. 891–906.

[20] K. E. Ozden, K. Schindler, and L. V. Gool, "Multibody structure-from-motion in practice," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 32, pp. 1134–1141, 2010.

[21] A. Leonardis, A. Gupta, and R. Bajcsy, "Segmentation of range images as the search for geometric parametric models," *Int. J. Comput. Vision*, vol. 14, no. 3, pp. 253–277, 1995.

[22] B. Leibe, A. Leonardis, and B. Schiele, "Robust object detection with interleaved categorization and segmentation," *Int. J. Comput. Vision*, vol. 77, no. 1-3, pp. 259–289, 2008.

[23] D. Nister, O. Naroditsky, and J. Bergen, "Visual odometry," vol. 1, 2004, pp. I–652 – I–659 Vol.1.

[24] S. K. J. S. S. Ma, Y.; Soatto, *An Invitation to 3-D Vision - From Images to Geometric Models*, J. S. L. W. S. Antman, S.S.; Marsden, Ed. Springer, 2004.

[25] R. Haralick, H. Joo, C. Lee, X. Zhuang, V. Vaidya, and M. Kim, "Pose estimation from corresponding point data," *Systems, Man and Cybernetics, IEEE Transactions on*, vol. 19, no. 6, pp. 1426 –1446, nov. 1989.

[26] M. A. Lourakis and A. Argyros, "Sba: A software package for generic sparse bundle adjustment," *ACM Trans. Math. Software*, vol. 36, no. 1, pp. 1–30, 2009.

[27] C. Tomasi and T. Kanade, "Detection and tracking of point features," Carnegie Mellon University, Tech. Rep. CMU-CS-91-132, April 1991.

[28] C. Mei, G. Sibley, M. Cummins, P. Newman, and I. Reid, "A constant-time efficient stereo slam system," in *British Machine Vision Conference (BMVC)*, 2009.

[29] M. Lourakis, "homest: A c/c++ library for robust, non-linear homography estimation," [web page] http://www.ics.forth.gr/ lourakis/homest/, Jul. 2006, [Accessed on 20 Jul. 2006.].