3D Piecewise Planar Object Model for Robotics Manipulation

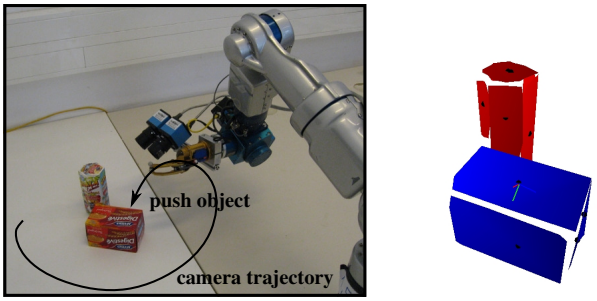
Johann Prankl, Michael Zillich, and Markus Vincze

***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 in 2D 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 robots 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 sufficiently accurately 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 Figure 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 splitting 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 analysis of the shape related to affordances. 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] and [4]).



(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 [5], [6], [7]. In [6] Aloimonos et. al stressed that an active observer can solve basic vision problems in a much more efficient way. They

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

introduce a general methodology, in which they believe low-level vision problems should be addressed. Metta et al. [8] developed an active strategy for a robot to acquire visual experience through simple experimental manipulation. The experiments are oriented towards determining what 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 [9], to sequentially fit the model to a data set and then to remove inliers. To generate plane hypotheses Vincent et al. [10] 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. [11] 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 [12], [13], [14] 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. [15], Fouhey et al. [16] and Chin et al. [17], 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. [18], 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 [19], [20] 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 [21], [22], [23], which exploit algebraic constraints satisfied by all scene objects, even though they move relative to each other, and non-algebraic methods [24], [25], which essentially combine rigid SfM with segmentation. Most related to our system are the methods proposed by Schindler [1] and by Ozden [26]. 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 to 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 Figure 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.

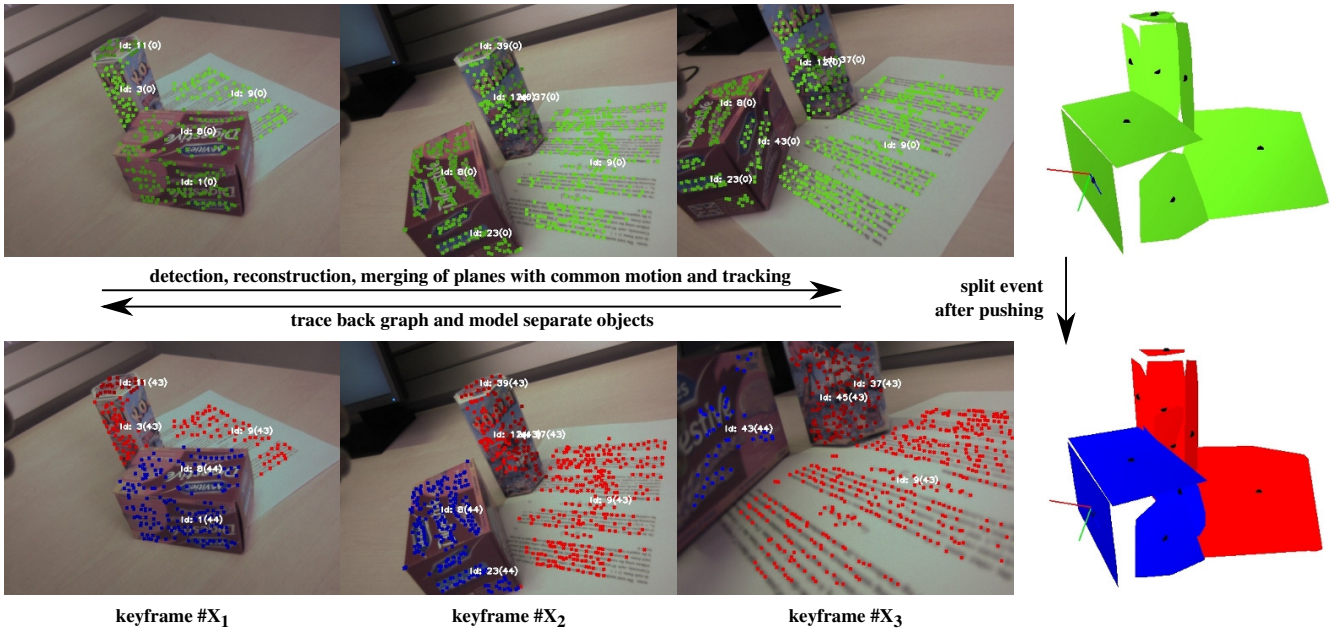


Fig. 2. The upper row shows three keyframes of sequence 1 (897 frames) with detected planes in green which are merged because of common 3D motion. The brightness of the interest points indicates the assignment to different planes. After the gripper (two black dots on the left image border) pushes an object the keyframe-graph is traced back and two different objects are modelled (lower image row).

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

1) Algorithm: Therefore 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 ensure that interest points are assigned to the best current 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. 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. In the first iteration P is initialized with tracked planes of the last image. 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 plane 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 and that the model cannot be fitted sequentially. 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 [27] and adapted in [28]. To select the best model, the savings for each hypothesis h are expressed as

$$S_h = S_{data} - \kappa_1 S_{model} - \kappa_2 S_{error} \quad (1)$$

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. One model (the homography of a plane) is used 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 and thus the merit term of a model results in

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

where $p(f_k|h)$ is the likelihood that an interest point belongs to the plane hypothesis h. Details for the derivation of Equation 2 can be found in [18]. An interest point can only be assigned to one model. Hence, overlapping models compete for interest points which can be represented by interaction costs

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

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

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

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. This iterative method keeps the number of hypotheses tractable. Furthermore experiments have shown that a greedy approximation 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], [26]. In [26] Ozden et al. defined the following requirements:

1) Determine the number of independently moving objects at the beginning of a sequence, and whenever that number changes
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 2D 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.

To reconstruct planes which are not assigned to a 3D motion model we use a standard SfM pipeline similar to Nister et al. [29] and Klein et al. [30]. Therefore the nonlinear refined homography is directly decomposed to initialize the first camera pose (cp. [31]). In the following frames the relative motion from C^{-1} to C is estimated using RANSAC [9] and a direct least squares solution between the two point sets (cp. Haralick et al. [32]). A sparse bundle adjustment implementation by Lourakis [33] 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 colour and the 3D location of interest points is introduced and planes are assigned to the object with a higher probability. Analogous to Equation 2 the formulation

$$\begin{aligned} p_{ij} &= -\epsilon_1 + \frac{1}{N} \sum_{k=1}^{N} \left((1-\epsilon_2) + \epsilon_2 p(f_{i,k}^{proj}|H_j) \right) \\ &+ \epsilon_3 p^*(a_i|A_j) \end{aligned} \quad (5)$$

is used to assign the plane i to the object j with the higher likelihood p_{ij}. In Equation 5 $p(f_{i,k}^{proj}|H_j)$ is the probability that an interest point of a plane i belongs to the 3D object H_j. Likewise in Equation 2, this is modelled using a Gaussian error model. Therefore 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. ϵ denote constants to weight the different factors, where ϵ_1 is an offset which must be reached to be considered as moving together and ϵ_3 is a weighting factor to reduce the influence of the appearance model and primarily merge depending on the motion.

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. The likelihood

$$\begin{aligned} p^*(a_i|A_j) &= \frac{1}{N} \sum_{k=1}^{N} \left((1-\epsilon_4) + \epsilon_4 p(f_{i,k}^{3D}|H_j) \right) \\ &+ \log(p(c_i|C_j)) \end{aligned} \quad (6)$$

[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.

combines these factors in a probabilistic manner. The first term describes a probabilistic voting scheme. Therefore 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. Therefore 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 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.

D. Separating planes in case of different motions

Motivated by Palmer [34] – who stated that although the vast majority of objects in ordinary environments are stationary the vast majority of the time, ones that move are important – we trigger our object modelling if an object separates, that is, planes start moving differently. Therefore in Equation 5, which is used to continuously test if planes start moving separately, ϵ_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 Equation 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. Figure 2 depicts an event chain where planes are merged because of common motion, start moving separate 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.

[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 of an event occurs.

III. EXPERIMENTS

For all experiments we use Shi-Tomasi interest points [35] and a KLT-tracker [36]. In [37] 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* [38].

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 Figure 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.

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. Figure 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. Figure 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 Figure 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}

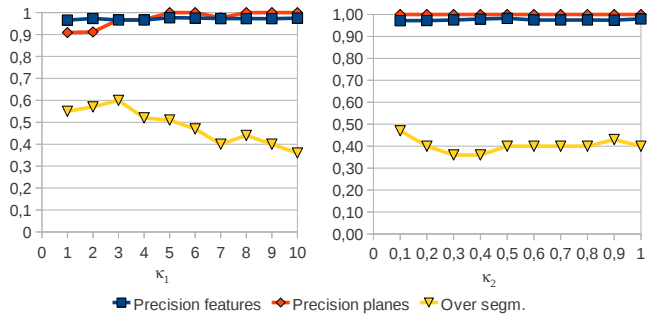


Fig. 3. Parameter optimisation

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.

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 subdivide planes if a better explanatation 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

Figures 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. Figure 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 Figures 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. Figure 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. In the case shown in Figure 5 this results in a much too big top surface of the red object which covers a part of the heart-shaped box. Figures 7 and 8 show the limits of

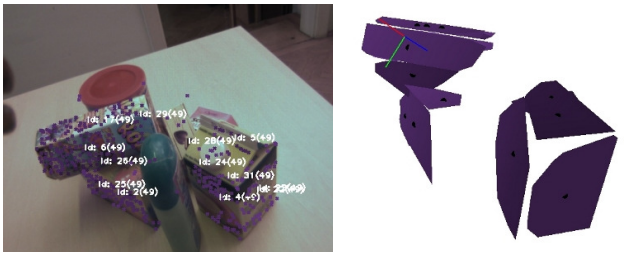


Fig. 8. Example image and reconstruction of a small more complex sequence which shows the limits of our system. Planes of the three dominant objects at the front are reconstructed, while the object at the centre of the image and the objects at the background are not detected because of low texture and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

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 Figure 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 Figure 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.

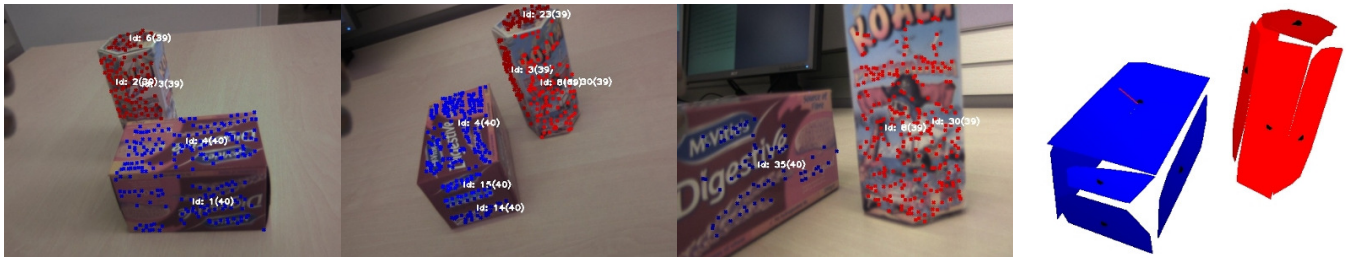


Fig. 4. Sequence 2 (715 frames).

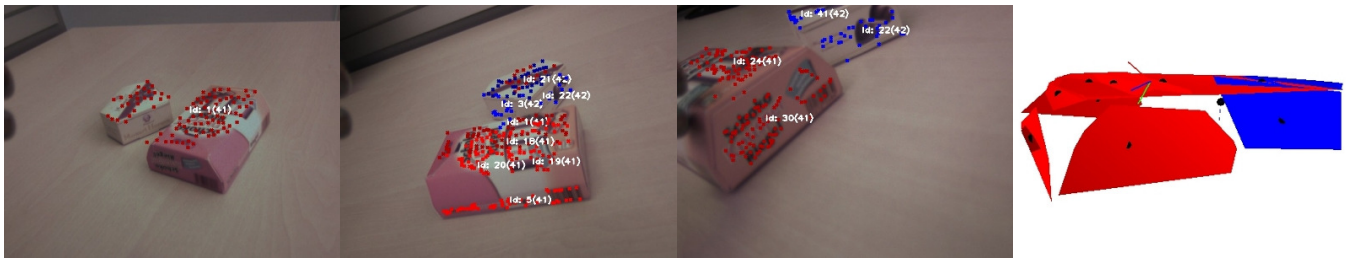


Fig. 5. Sequence 3 (543 frames).

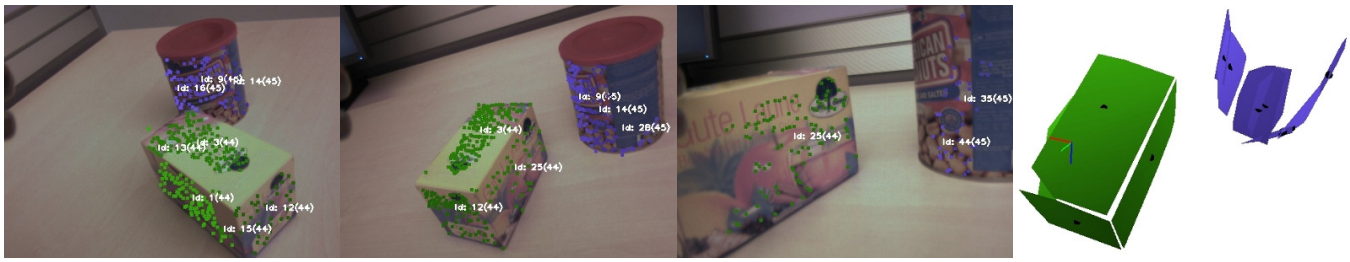


Fig. 6. Sequence 4 (870 frames).

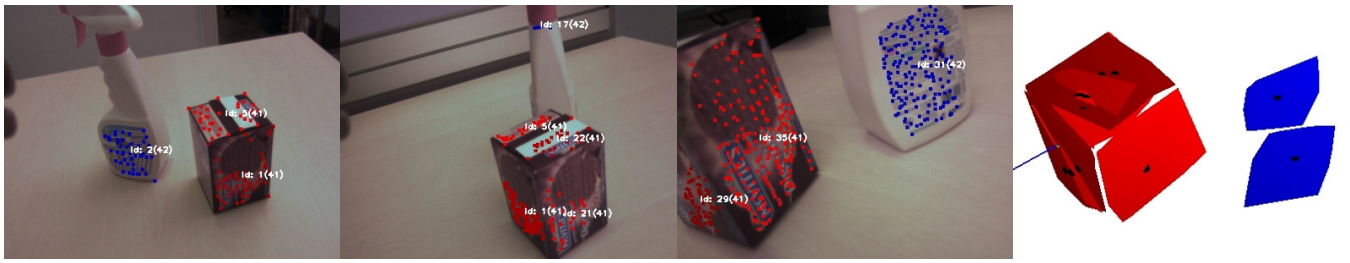


Fig. 7. Sequence 5 (811 frames).

[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] A. C. Romea, D. Berenson, S. Srinivasa, and D. Ferguson, "Object recognition and full pose registration from a single image for robotic manipulation," in *IEEE International Conference on Robotics and Automation (ICRA '09)*, May 2009.

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

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

[7] D. H. Ballard, "Animate vision," *Artif. Intell.*, vol. 48, no. 1, pp. 57–86, 1991.

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

[9] 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.

[10] E. Vincent and R. Laganiere, "Detecting planar homographies in an image pair," in *Image and Signal Processing and Analysis, 2001. ISPA 2001. Proceedings of the 2nd International Symposium on*, 2001, pp. 182 –187.

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

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

[13] G. Lopez Nicolas, J. Guerrero, O. Pellejero, and C. Sagues, "Computing homographies from three lines or points in an image pair," in *CIAP 2005*, 2005, pp. 446–453.

[14] J. Piazzi and D. Prattichizzo, "Plane detection with stereo images," in *Robotics and Automation, 2006. ICRA 2006. Proceedings 2006 IEEE International Conference*, may 2006, pp. 922 –927.

[15] R. Toldo and A. Fusiello, "Robust multiple structures estimation with j-linkage," in *ECCV '08: Proceedings of the 10th European Conference on Computer Vision*. Berlin, Heidelberg: Springer-Verlag, 2008, pp. 537–547.

[16] D. Fouhey, D. Scharstein, and A. Briggs, "Multiple plane detection in image pairs using j-linkage," in *20th International Conference on Pattern Recognition (ICPR 2010)*, To appear 2010.

[17] 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.

[18] J. Prankl, M. Zillich, B. Leibe, and M. Vincze, "Incremental model selection for detection and tracking of planar surfaces," in *Proceedings of the British Machine Vision Conference*. BMVA Press, 2010, pp. 87.1–87.12.

[19] O. Faugeras, Q.-T. Luong, and T. Papadopoulou, *The Geometry of Multiple Images: The Laws That Govern The Formation of Images of A Scene and Some of Their Applications*. Cambridge, MA, USA: MIT Press, 2001.

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

[21] R. Vidal and Y. Ma, "A unified algebraic approach to 2-d and 3-d motion segmentation," in *Computer Vision - ECCV 2004*, ser. Lecture Notes in Computer Science, T. Pajdla and J. Matas, Eds. Springer Berlin / Heidelberg, 2004, vol. 3021, pp. 1–15.

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

[23] R. Vidal and R. Hartley, "Motion segmentation with missing data using powerfactorization and gpca," vol. 2, jun. 2004, pp. II–310 – II–316 Vol.2.

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

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

[26] 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.

[27] 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.

[28] 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.

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

[30] G. Klein and D. Murray, "Improving the agility of keyframe-based SLAM," in *Proc. 10th European Conference on Computer Vision (ECCV'08)*, Marseille, October 2008, pp. 802–815.

[31] 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.

[32] 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.

[33] 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.

[34] S. Palmer, *Vision Sciene - Photons to Phenomenology*. The MIT Press, 1999.

[35] J. Shi and C. Tomasi, "Good features to track," jun. 1994, pp. 593 –600.

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

[37] 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.

[38] 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.].