2012 IEEE/RSJ International Conference on
Intelligent Robots and Systems
October 7-12, 2012. Vilamoura, Algarve, Portugal

Segmentation of Unknown Objects in Indoor Environments

Andreas Richtsfeld, Thomas Mörwald, Johann Prankl, Michael Zillich and Markus Vincze

***Abstract*— We present a framework for segmenting unknown objects in RGB-D images suitable for robotics tasks such as object search, grasping and manipulation. While handling single objects on a table is solved, handling complex scenes poses considerable problems due to clutter and occlusion. After pre-segmentation of the input image based on surface normals, surface patches are estimated using a mixture of planes and NURBS (non-uniform rational B-splines) and model selection is employed to find the best representation for the given data. We then construct a graph from surface patches and relations between pairs of patches and perform graph cut to arrive at object hypotheses segmented from the scene. The energy terms for patch relations are learned from user annotated training data, where support vector machines (SVM) are trained to classify a relation as being indicative of two patches belonging to the same object. We show evaluation of the relations and results on a database of different test sets, demonstrating that the approach can segment objects of various shapes in cluttered table top scenes.**

I. INTRODUCTION

Segmenting unknown objects from generic scenes is an enabler in many robotics tasks, such as object search, grasping and manipulation, but remains one of the ambitious and elusive goals of computer vision and is in general a very ill defined problem. With the recent introduction of cheap and powerful 3D sensors (such as the Microsoft Kinect or Asus XtionPRO) which deliver a dense point cloud plus color for almost any indoor scene, a renewed interest in 3D methods holds the promise to push the envelope slightly further.

In this work we aim at segmenting unknown objects of arbitrary (but reasonably compact) shape from table top scenes, where objects need not stand isolated but can be jumbled in heaps. An example for such a scene is shown in Fig. 1. Moreover we want a compact and accurate representation of object shapes, suitable in a robotics domain for various tasks.

The dense and reliable point cloud delivered by the mentioned sensors allows, after pre-segmentation based on surface normals, to robustly fit planar surface patches to parts of the point cloud. Planes are fast to compute and capture a good range of typical man made objects. In order to also model curved objects with high accuracy we fit NURBS (non-uniform rational B-splines) and replace planes whenever NURBS provide a better fit. We use model selection [11] to find the combination of planes and NURBS optimally explaining the point cloud data.

Segmenting objects from a scene then amounts to identifying groups of surface patches that are likely to belong to the same objects. I.e. we perform perceptual grouping,

Vision for robotics group (v4r), Automation and Control Institute (ACIN), Vienna University of Technology, 1040 Vienna, Austria
{ari, tm, jp, mz, mv}@acin.tuwien.ac.at

Fig. 1. Segmented objects from a cluttered table top scene.

but not as is more traditionally done in 2D using e.g. edges and junctions, but using 3D surface relations. We define several pairwise relations, based on grouping principles such as proximity, similarity or continuation and create a relation feature vector for each pair of surface patches. We distinguish two types of feature vectors, one for neighbouring and one for non-neighbouring surface patches, each containing relations that are applicable to neighbouring or non-neighbouring surfaces respectively. Each of these relations indicates to a certain degree that the respective surface patches are likely to belong to the same object, with e.g. closeness being a very good indicator and color similarity being far weaker. To arrive at a single value of relatedness of two surface patches from such a vector of very different individual relations we use a learning approach. Human-annotated ground truth data is used to train a SVM for each type of feature vector. The SVMs categorize a relation feature vector as either indicating same or different object for a given pair of patches. We then construct a graph from all the surface patches and pairwise relations and use the output of the SVMs as the pairwise energy term in a graph cut based segmentation. The resulting segmentation is able to detect many typical objects as they arise in service robotics tasks (books, boxes, product packaging, cups, etc.), provided that single surfaces are big enough to be captured in sufficient detail by the sensor and that enough training data is provided to the SVMs to capture all arising surface relations.

The key novelty in our approach lies in a 3D perceptual grouping approach of plane and NURBS representations based on learned relations, suitable for segmenting objects of reasonable size and compactness, even if they are stacked into heaps or are partially occluded.

The paper is structured as follows. The next section sets the presented work into context with related work in this research field. Sec. III shows the structure of the proposed framework, while Sec. IV and Sec. V explains more detailed the components of the framework. Evaluation and results are shown in Sec. VI, before the work ends with a conclusion and an outlook in Sec. VII.

978-1-4673-1736-8/12/S31.00 ©2012 IEEE

II. RELATED WORK

Various approaches to segment objects either in 2D images or in point clouds exist. Early approaches aimed to formulate generic Gestalt principles to organise 2D scenes into objects. For an overview of this early work in perceptual organisation we want to refer to Boyer and Sarkar [2]. More recently Zillich [23] proposed an any-time perceptual grouping framework to segment convex parts in images. Gestalt principles are also used by Kootstra et al. in [10] and [9]. They developed a symmetry detector to initialize segmentation based on a Markov Random Field (MRF). Furthermore Koostra et al. developed a quality measure based on Gestalt principles to rank segmentation results.

Many state-of-the-art approaches formulate image segmentation as energy minimization with a MRF [1], [19], [3], [17]. In addition to an appearance model computed from colour and texture, which is commonly used to better distinguish foreground from background, Bergstrom et al. [1] formulate an objective function where it is possible to incrementally add constraints generated through human-robot interaction. In [20] Werlberger et al. propose a variational model for interactive segmentation using a shape prior. This method is based on minimizing the Geodesic Active Contour energy.

Active segmentation is proposed in Mishra et al. [12] and [13] where an image point is fixated and the shortest path in a log polar transformed edge image is computed. The authors propose to use the depth image from stereo cameras in addition to the edges computed from colour and texture information to improve segmentation.

The approach by Hager et al. [8] is able to segment objects from cluttered scenes in point clouds generated from stereo by using a strong prior 3D model of the scene and explicitly modelling physical constraints such as support and handles dynamic changes such as object appearance/disappearance. It is however limited to parametric models (boxes, cylinders).

The problem of fitting higher order surfaces to point clouds was already addressed by the framework of Leonardis et al. [11]. They segment range images by estimating piecewise linear surfaces, modelled with bivariate polynomials. Furthermore they developed a Model Selection framework, which is used to find the best interpretation of the range data in terms of Minimum Description Length (MDL). Instead of using bivariate polynomials we first pre-segment the scene, describe the scene with plane or NURBS models, and then decide which combination of planes and NURBS approximates the point cloud better in terms of MDL. Additionally, we cluster surface patches to objects depending on learned patch relations.

III. SYSTEM OVERVIEW

The proposed system consists of two major parts, namely *Data Abstraction* and *Object Segmentation*. Figure 2 shows the processing chain for those parts in detail. Data abstraction starts with pre-segmentation, based on surface normals, before model selection decides whether a plane or a NURBS surface patch fits better to the given pre-segmented point cloud data. The second major part has to decide, which sets of surface patches together form objects. First, relations between surface patches are calculated and feature vectors are built. Support vector machines are used to obtain a meaningful energy term from the feature vector, which is used by a graph cut algorithm to globally optimize the splitting of a graph consisting of surface patches as nodes and relations as edges with energy terms, estimated by the SVMs.

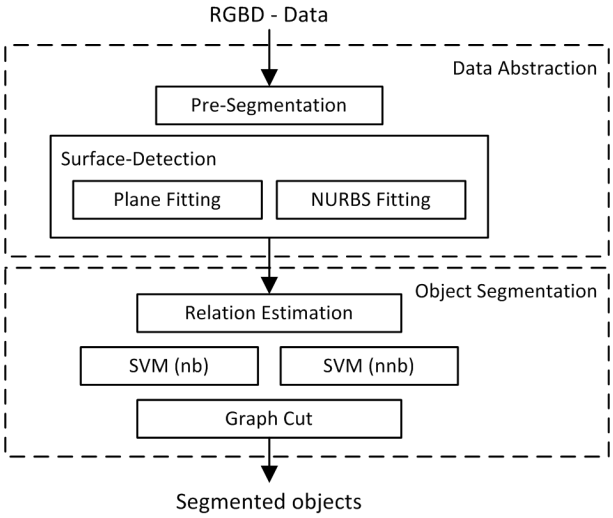

Fig. 2. System overview: RGBD data abstraction into a best plane/NURBS surface model representation and object segmentation using Graph-Cut based on relations trained offline using support vector machines (SVMs).

IV. DATA ABSTRACTION

Object segmentation based on non-abstracted sensor data is a complex task since reasoning on raw sensor data is more difficult than on abstracted data. We propose to abstract real-world sensor data into meaningful pieces before we do object segmentation. Many everyday objects which can be found in households are of compact size and have smooth surfaces, which are sometimes planar, but also free-form surfaces. Representation of such objects requires a simple and computationally efficient planar model as well as a more complex one, such as NURBS (Non-Uniform Rational B-Splines). Our framework pre-segments RGBD data into planar patches, before a model selection algorithm decides whether NURBS fitted to neighbouring patches, or the individual planar patches better represent the underlying data.

A. Pre-Segmentation

For assembling surface patches to object hypotheses, we first need to segment the point cloud into individual patches and estimate the surface model parameters. We use planes and NURBS, thus ideal pre-segmentation amounts to cluster points located on a smooth surface. One possibility would be to compute the surface normals for the point cloud and cluster neighbouring points with similar normals. Estimated normals from point cloud data are noisy or tend to smooth edges and simple clustering of normals would lead to under-segmentation. Hence, we use a stronger plane model to

over-segment the point cloud and try to merge neighbouring planes to smooth surfaces in a second step (described in the next Section IV-B).

Plane detection is done by recursively clustering normals in a hierarchical fashion. Therefore a pyramid is created by sub-sampling the point cloud and planes detected in adjacent levels have to compete for points. To select the best subset based on a Minimum Description Length (MDL) criterion we adapted the Model Selection approach proposed by Prankl et al. [15] for plane detection in image pairs. In contrast to Prankl et al. who implemented a QBP approach, we use the model selection criterion in an greedy approach to substitute planes detected at the coarser level with overlapping more detailed ones at the detailed level.

B. Surface Detection

Estimation of surface models can be done with the robust estimation method RANSAC by sequentially fitting models and filtering the inliers. The plane model has only three parameters, therefore random sampling is an appropriate approach. For NURBS, where the number of parameters is three times the number of control points, an intractable number of random samples would be necessary to select inliers of a surface patch. For this reason we greedily merge the pre-segmented planar patches and substitute them with NURBS by using Model Selection and Minimum Description Length (MDL) in a similar way as indicated in Section IV-A.

The idea of Model Selection is that the same data point can not belong to more than one surface model. Accordingly, the savings

$$S_H = \frac{N}{A_m} - \kappa_1 S_m - \frac{\kappa_2}{A_m} \sum_{i=1}^{N} (1 - p(f_i|H)), \qquad (1)$$

for models of neighbouring patches S_i and S_j are compared to the savings of a model fitted to a merged patch S_{ij} and in case S_{ij} is larger the individual patches are substituted. In Eq. 1 N is the number of data points explained by the hypothesis H, S_m stands for the cost of coding different models, $p(f_i|H)$ is the probability (modelled with a Gaussian error model), that a data point f_i belongs to H and A_m is a normalization value representing the size of merged patches. κ_1 and κ_2 are constants to weight the different terms. In the following paragraphs estimation of the model parameter for planes and NURBS is described.

1) Plane Fitting: Even if planes could in principle be modelled as simply specific cases of NURBS, planes have an exceptional role in our framework for reasons of computational efficiency. For the least squares estimation of the plane parameter we use a SVD of the covariance matrix of the data points. On modern CPU/GPU hardware this allows to compute the surface normals of a point cloud with a *sliding window* approach in real time.

2) NURBS Fitting: For representing free-form surfaces there are a number of geometric models available. Most widely used in industry are NURBS (non-uniform rational NURBSs). The reasons for their popularity are the convenient manipulation and the ability to represent all conic sections, i.e. circles, cylinders, ellipsoids, spheres and so forth. The possibility for refinement through knot insertion allows for adaption to local irregularities, while selecting a certain polynomial degree gives reason about the curvature of the measured surface we want to fit to.

A good overview of the properties and advantages of NURBS can be found in Chapter 1.1.2 in [6]. The mathematical concept of NURBS would go far beyond the scope of this paper. So for those who are interested let us refer to the well known book by Piegl et al. [14], which explains the fitting technique used in Chapter 9.4. Initialisation is done by performing a principle component analysis (PCA) on the point-cloud of interest and defining the control points of the NURBS according to the eigenvectors of the PCA.

V. OBJECT SEGMENTATION

The previous section explained how to find the best representation of a point cloud, but now we are interested in grouping these patches into object hypotheses. First we introduce relations between surface patches which contribute to distinguish between patches which belong together or not and create a feature vector of relations. We propose to use two different feature vectors, one for neighbouring and another one for non-neighbouring patches, consisting of different relations. This two step approach follows quite naturally, because different types of relations hold for neighbouring patches e.g. touching along a common boundary, vs. patches separated due to occlusion. Based on the estimated vectors, we train one SVM for each type of vector to predict later the probability of affiliation of surface patches. We build a graph, using the surface patches as nodes and the estimated prediction values from the SVM as edges. Finally, a Graph-Cut algorithm decides globally which surface patches belong together to form a common object hypothesis based on energy minimization.

A. Relation Estimation

The major task of object segmentation is to find relations between surface patches which indicate that they belong to one object. Inspired by Gestalt principles [21], such as proximity, similarity and continuation, which were widely used in cognitive computer vision, we investigated the relevance of several relations, inferred and adapted to 3D-data. By experimental investigations of those relations, we quickly experienced the difference of relevance of features, if patches are neighbours in 3D space or not. We propose to use different relations for neighbouring and non-neighbouring surface patches and introduce two different feature vectors, based on the following relations:

- r_{cu} ... mean curvature along 2D patch border
- r_{di} ... mean distance along 2D patch border
- r_{vdi} ... variance of distance along 2D patch border
- r_{cb} ... difference of color along 2D patch border
- r_{ch} ... difference of patch colour
- r_{tr} ... difference of patch texture
- r_{ga} ... gabor filter match
- r_{fou} ... fourier filter match

- r_{cu3} ... curvature along 3D patch border
- r_{di3} ... distance along 3D patch border
- r_{nm} ... difference of mean surface normals direction
- r_{nv} ... difference of variance of normals direction
- r_{ac} ... mean angle of normals of nearest contour points
- r_{dn} ... mean normal distance of nearest contour points
- r_{md} ... minimum distance between patch borders
- r_{rs} ... relative patch size difference

The feature vector of neighbouring (r_{nb}) and non-neighbouring surface patches (r_{nnb}) are then constructed in the following way:

$$\mathbf{r_{nb}} = \{r_{cu}, r_{di}, r_{vdi}, r_{cb}, r_{ch}, r_{tr}, r_{ga}, r_{fou}, r_{cu3}, r_{di3}\} \quad (2)$$

$$\mathbf{r_{nnb}} = \{r_{ch}, r_{tr}, r_{ga}, r_{fou}, r_{nm}, r_{nv}, r_{ac}, r_{dn}, r_{md}, r_{rs}\} \quad (3)$$

The first four relations are merely suitable for feature vectors of neighbouring patches, because 2D neighbourhood is required to estimate these values. The curvature relation r_{cu} expresses the mean angle between surface normals of neighbouring points along patch borders. r_{di} describes the mean distance between neighbouring patch border points and r_{vdi} is the variance of the distance along the border. Colour difference along the border r_{cb} and color difference of the patch r_{ch} is calculated as the Fidelity distance a.k.a. Bhattacharyya coefficient of the UV components in the YUV color space, because they are less susceptible to brightness changes in the scene than a comparison in the RGB color space. The texture rate of a surface patch is the rate between the number of canny edge pixels on the surface and the sum of all surface pixels. The difference of texture r_{tr} is then defined as difference of the texture rate between two surface patches. The gabor and fourier filter are implemented as proposed in [18]. For the gabor filter six directions (in 30° steps) with five different kernel sizes are used. A feature vector with 60 values is built from the mean and the standard deviation. The gabor filter match r_{ga} is the minimum difference between these two vectors ($d = \sum_{i,j} \sqrt{(\mu_i - \mu_j)^2 + (\sigma_i - \sigma_j)^2}$), when one feature vector gets shifted so that different orientations of the gabor features are matched. This guarantees roughly scale and rotation invariance for the gabor filter. The fourier filter match r_{fou} is again calculated as Fidelity distance of five histograms consisting of 8 bins and filled with the normalised absolute values of the first five coefficients from the DFT. The curvature r_{cu3} and the distance r_{di3} along the 3D borders is compared to the related 2D relations merely calculated if the neighbouring points in the 2D image space are also neighbours in 3D space. It turned out that a good distance for measuring of neighbourhood in 3D is 0.015m. r_{nm} and r_{nv} are the difference of the mean and variance from all normals of a surface patch.

For the next two features the 20% of nearest points between two patches are calculated. r_{ac} and r_{dn} compare then the mean angle between the surface normals and the mean distance in normal direction. Relation r_{md} measures the minimum distance between the patch borders and the last relation r_{rs} describes the relative difference of the patch size of two patches.

B. Support Vector Machine Learning

The introduced relation vectors r_{nb} and r_{nnb} are used in an offline phase to train the two SVM using a hand-annotated set of depth images. Feature vectors of patch pairs from the same object represent positive training examples and vectors from pairs between two different objects and objects and background represent negative examples. With this strategy, not only the affiliation of patches to objects, but also the disparity of object patches to other objects or background will be learned.

For the training, as well as prediction and evaluation we use the libsvm package [4], a free SVM software package. As kernel is the radial basis function (RBF) used:

$$\mathbf{K}(x_i, x_j) = e^{\gamma ||x_i - x_j||^2} \quad (4)$$

In the online phase, the SVM is capable to provide not only a binary decision *same* or *notsame* for each relation vector $\mathbf{r}$, but also a probability value $p(same\,|\,\mathbf{r})$ for each decision, based on the theory introduced by Wu et al. [22].

C. Graph Cut Segmentation

After SVM prediction some estimates may contradict when forming object hypotheses. A globally optimal solution has to be found to overcome vague or wrong local predictions from the two SVMs. To this end we define a graph, where patches represent nodes and relations represent edges. The graph is fully connected, as we defined relations between all surface neighbors, as well as all non-neighbouring surface patches. We then employ graph-cut segmentation, introduced by Felzenszwalb et al. [7], using the probability values from the SVMs as the pairwise energy terms.

VI. RESULTS

Each learning approach is only as good as its training data, in our case training images for the SVM, containing objects of different shape complexity in several assemblies. The training images should be complex enough to distinguish later between objects in the scene. On the other side, images should contain simple examples to learn the typical relations between surface patches from the same object.

TABLE I
NUMBER OF IMAGES AND RELATIONS FROM THE DATABASE.

	learn	r_{nb}	r_{nnb}	test	r_{nb}	r_{nnb}
Boxes	17	157	48	16	393	2489
Stacked Boxes	12	101	58	12	129	1873
Occluded Obj.	8	73	124	7	58	518
Cylindric Obj.	8	104	250	8	84	650
Mixed Objects				12	407	4553
Complex Scene				11	754	28364
TOTAL	45	435	480	66	1825	38447

Evaluation of the proposed system was done on the object segmentation dataset proposed in [16]. The dataset consists of 111 images in six subsets as shown in Tab. I. For the whole dataset the annotation of objects in the RGBD images is provided to enable learning and also evaluation on these sets.

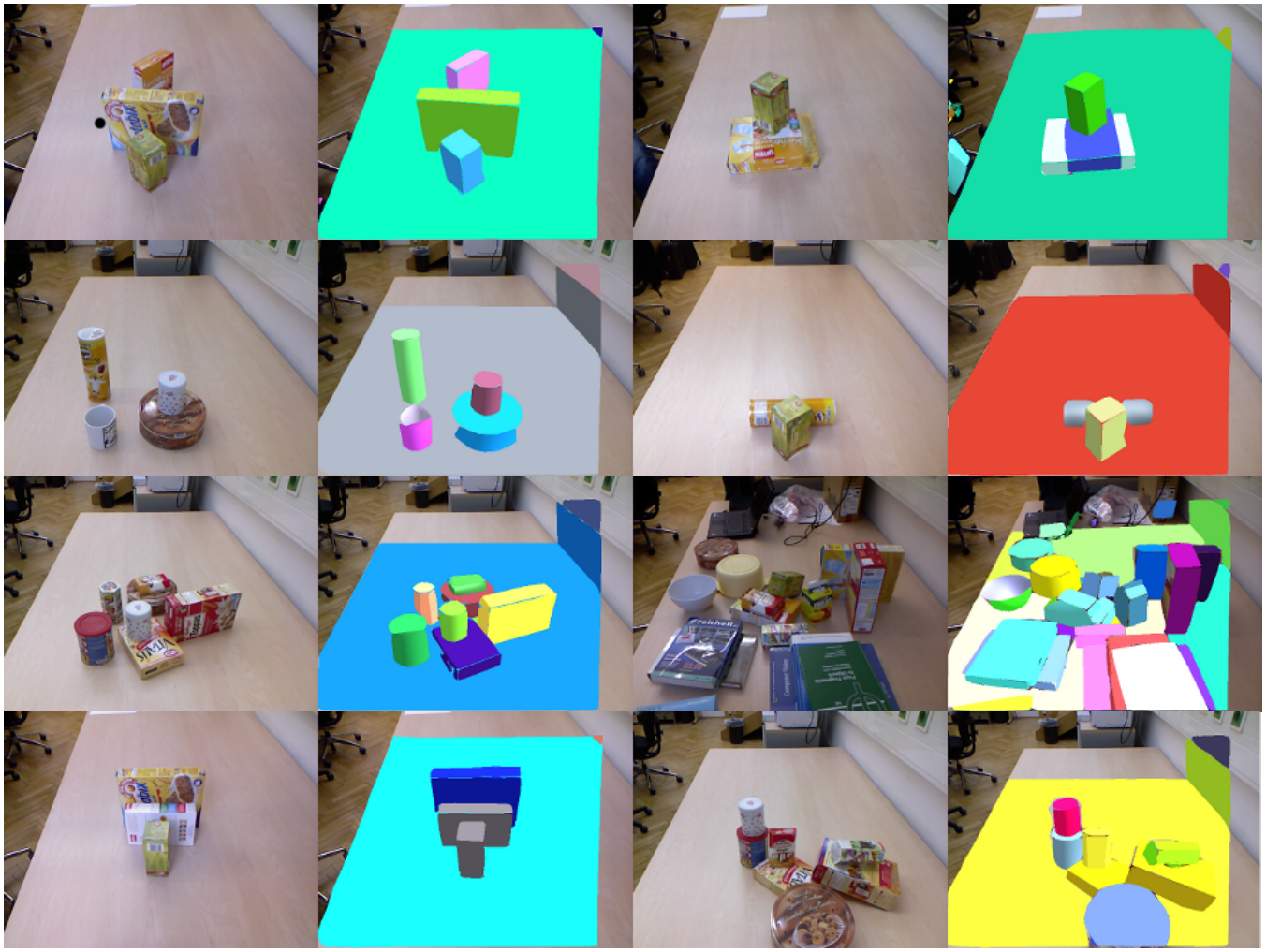

Fig. 3. Selected examples of segmented objects with the proposed approach (objects randomly coloured). The first six examples showing results from each dataset, the last two examples are showing under-segmentation caused by wrong predictions of the SVM_{nnb}.

Chen [5] introduced the F-score for evaluation of features for SVM classification. F-score is a technique which measures the discrimination of two sets of real numbers. Given training vectors r_k, $k = 1,...,m$, if the number of positive and negative instances are n_+ and n_-, respectively, then the F-score of the i-th feature is defined as:

$$F(i) = \frac{(\bar{r}_i^{(+)} - \bar{r}_i)^2 + (\bar{r}_i^{(-)} - \bar{r}_i)^2}{\frac{1}{n_+ - 1}\sum_{k=1}^{n_+}(r_{k,i}^{(+)} - \bar{r}_i^{(+)})^2 + \frac{1}{n_- - 1}\sum_{k=1}^{n_-}(r_{k,i}^{(-)} - \bar{r}_i^{(-)})^2} \quad (5)$$

where $\bar{r}$, $\bar{r}^{(+)}$ and $\bar{r}^{(+)}$ are the average of the i-th feature of the whole, positive and negative data sets, respectively, $r_{k,i}^{(-)}$ is the i-th feature of the k-th negative instance. The numerator indicates the discrimination between the positive and negative sets, and the denominator indicates the one within each of the two sets. The larger the F-score, the more likely this feature is more discriminative. Figure 4 shows the F-score of the learning set from 435 feature vectors for the SVM_{nb} and 480 for the SVM_{nnb}. Relations for the SVM_{nb} are far stronger than for SVM_{nnb} when considering the logarithmic scale. Since the F-score does not reveal mutual

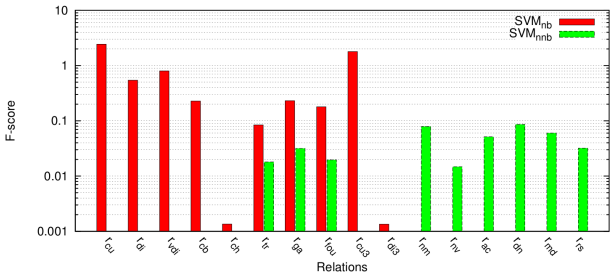

Fig. 4. F-score for relations between surface patches.

information, this does not inevitably mean that all relations together predict utterly precise. The evaluation results from the six different trainings sets are shown in Tab. II. The first column presents the accuracy of the SVM predictions for relations between neighbouring surface patches, followed by the over- and under-segmentation for the case of using just one SVM for graph building. The next three columns show the accuracy of the SVM for non-neighbouring surface patches and the over- and under-segmentation for operation with both SVMs. Over-segmentation F_{os} is defined using

TABLE II

RESULTS OF OBJECT SEGMENTATION. SVM ACCURACY, OVER- AND UNDER-SEGMENTATION WHEN USING ONE OR TWO SVM'S.

	SVM_{nb}	F_{os}	F_{us}	SVM_{nnb}	F_{os}	F_{us}
Boxes	88.55%	1.8%	0.2%	98.19%	0.2%	17.2%
St. Boxes	89.15%	1.3%	7.1%	98.99%	0.0%	28.2%
Occl. Obj.	87.93%	16.6%	0.1%	99.23%	0.0%	0.2%
Cyl. Obj.	91.66%	2.6%	0.3%	96.77%	2.6%	3.5%
Mixed Obj.	91.04%	1.9%	19.7%	94.97%	1.3%	39.2%
Complex S.	84.61%	7.0%	8.0%	98.97%	5.4%	146%
TOTAL	87.72%	4.5%	7.9%	98.41%	2.7%	69.5%

the number of correctly assigned object pixels divided by the number of all object pixels and under-segmentation F_{us} using the number of incorrectly assigned pixels divided by the number of all object pixels:

$$F_{os} = 1 - \frac{N_{true}}{N_{all}} \qquad F_{us} = \frac{N_{false}}{N_{all}} \tag{6}$$

The results of the segmentation evaluation in Tab. II show the difference of using one or two SVMs and the associated trade-off between over- and under-segmentation. When using both SVMs, the framework tends to higher under-segmentation, because only one wrongly connected non-neighbouring surface can connect two objects, which can be seen in the last two examples in Fig. 3. On the other hand the second SVM enables handling of occlusions, which can be seen for the results of the occluded objects dataset. For complex scenes is the advantage in over-segmentation not in relation with the disadvantage in under-segmentation. Even when around 98% of the predictions of the SVM_{nnb} are correct, the wrong ones will highly affect the under-segmentation.

VII. CONCLUSION AND FURTHER WORK

We presented a framework for segmenting unknown objects in RGBD-images of cluttered table top scenes, by first approximating surfaces with a combination of planes and NURBS and then segmenting the scene based on learned relations between surface patches. With the proposed approach of using two SVMs for prediction of connectedness between patches, we tackle also the problem of segmenting occluded objects.

One of the problems we are still facing is the weakness of the relations for predictions with the SVM for non-neighbouring surface patches. We intend to address this by further investigations of grouping principles suitable for RGBD-data as well as exploration of different database setups for training of the SVMs.

The presented results of the proposed framework are showing that the approach is promising and has the ability of usage in several indoor robotic tasks where identifying unknown objects or grasping plays a role.

VIII. ACKNOWLEDGMENTS

The research leading to these results has received funding from the European Communitys Seventh Framework Programme [FP7/2007-2013] under grant agreement No. 215181, CogX and by the Austrian Science Foundation under grant agreement No. I513-N23.

REFERENCES

[1] N. Bergström, M. Björkman, and Danica Kragic. Generating object hypotheses in natural scenes through human-robot interaction. In *Intelligent Robots and Systems (IROS), 2011 IEEE/RSJ International Conference on*, pages 827–833. IEEE, 2011.

[2] Kim L Boyer and Sudeep Sarkar. Perceptual organization in computer vision: status, challenges, and potential. *Comput. Vis. Image Underst.*, 76(1):1–5, 1999.

[3] Y Y Boykov and M.-P. Jolly. Interactive graph cuts for optimal boundary amp; region segmentation of objects in N-D images. In *Computer Vision, 2001. ICCV 2001. Proceedings. Eighth IEEE International Conference on*, volume 1, pages 105 –112 vol.1, 2001.

[4] Chih-chung Chang and Chih-jen Lin. LIBSVM : A Library for Support Vector Machines. *ACM Transactions on Intelligent Systems and Technology*, 2(3):27:1—-27:27, 2011.

[5] Yi-wei Chen and Chih-jen Lin. Combining SVMs with Various Feature Selection Strategies. *Strategies*, 324(1):1–10, 2006.

[6] J A Cottrell, T J R Hughes, and Y Bazilevs. Isogeometric Analysis. *Continuum*, page 355, 2010.

[7] Pedro F. Felzenszwalb and Daniel P. Huttenlocher. Efficient Graph-Based Image Segmentation. *International Journal of Computer Vision*, 59(2):167–181, September 2004.

[8] Gregory D Hager and Ben Wegbreit. Scene parsing using a prior world model. *The International Journal of Robotics Research*, 2011.

[9] Gert Kootstra, Niklas Bergström, and Danica Kragic. Fast and Automatic Detection and Segmentation of Unknown Objects. In *Humanoids*, Bled, 2011.

[10] Gert Kootstra, Niklas Bergström, and Danica Kragic. Gestalt Principles for Attention and Segmentation in Natural and Artificial Vision Systems. In *SPME*, Shanghai, 2011.

[11] Aleš Leonardis, Alok Gupta, and Ruzena Bajcsy. Segmentation of range images as the search for geometric parametric models. *International Journal of Computer Vision*, 14(3):253–277, April 1995.

[12] Ajay K Mishra and Yiannis Aloimonos. Active Segmentation. *I. J. Humanoid Robotics*, 6(3):361–386, 2009.

[13] Ajay K Mishra, Yiannis Aloimonos, Loong Fah Cheong, and Ashraf Kassim. Active Segmentation. *IEEE transactions on pattern analysis and machine intelligence*, 6(3):361–386, August 2011.

[14] Les Piegl and Wayne Tiller. *The NURBS book*. Monographs in visual communication. Springer, 1996.

[15] Johann Prankl, Michael Zillich, Bastian Leibe, and Markus Vincze. Incremental Model Selection for Detection and Tracking of Planar Surfaces. In *Proceedings of the British Machine Vision Conference*, pages 87.1—-87.12. BMVA Press, 2010.

[16] Andreas Richtsfeld. The Object Segmentation Database (OSD), http://www.acin.tuwien.ac.at/?id=289, 2012.

[17] Carsten Rother, Vladimir Kolmogorov, and Andrew Blake. "GrabCut": interactive foreground extraction using iterated graph cuts. *ACM Trans. Graph.*, 23(3):309–314, August 2004.

[18] Ahsan Ahmad Ursani, Kidiyo Kpalma, and Joseph Ronsin. Texture features based on Fourier transform and Gabor filters: an empirical comparison. In *2007 International Conference on Machine Vision*, pages 67–72. Ieee, December 2007.

[19] S Vicente, V Kolmogorov, and C Rother. Joint optimization of segmentation and appearance models. *2009 IEEE 12th International Conference on Computer Vision*, (Iccv):755–762, 2009.

[20] Manuel Werlberger, Thomas Pock, Markus Unger, and Horst Bischof. A Variational Model for Interactive Shape Prior Segmentation and Real-Time Tracking. In *International Conference on Scale Space and Variational Methods in Computer Vision (SSVM)*, Voss, Norway, 2009.

[21] M. Wertheimer. Untersuchungen zur Lehre von der Gestalt. II. *Psychological Research*, 4(1):301–350, 1923.

[22] TF Wu and CJ Lin. Probability estimates for multi-class classification by pairwise coupling. *The Journal of Machine Learning Research*, 5:975–1005, 2004.

[23] Michael Zillich. Incremental Indexing for Parameter-Free Perceptual Grouping. In *31st Workshop of the Austrian Association for Pattern Recognition*, pages 25–32, 2007.