Künstl Intell (2015) 29:95–99
DOI 10.1007/s13218-014-0339-7

Object Detection for Robotic Applications Using Perceptual Organization in 3D

Andreas Richtsfeld · Michael Zillich · Markus Vincze

Published online: 23 December 2014

Abstract Object segmentation of unknown objects with arbitrary shape in cluttered scenes is still a challenging task in computer vision. A framework is introduced to segment RGB-D images where data is processed in a hierarchical fashion. After pre-segmentation and parametrization of surface patches, support vector machines are used to learn the importance of relations between these patches. The relations are derived from perceptual grouping principles. The proposed framework is able to segment objects, even if they are stacked or jumbled in cluttered scenes. Furthermore, the problem of segmenting partially occluded objects is tackled.

Keywords Object segmentation · Perceptual organization

1 Introduction and Related Work

Wertheimer [22, 23], Köhler [7], Koffka [6] and Metzger [9] were the pioneers of studying Gestalt psychology. *Gestalt principles* (also called *Gestalt laws*) aim to formulate the regularities according to which the perceptual input is organized into unitary forms, also referred to as wholes, groups, or Gestalten. The principles are much like heuristics, which are mental short-cuts for solving problems. *Perceptual organization* can be defined as the ability to impose structural organization on sensory data, so as to group sensory primitives arising from a common underlying cause [2]. In computer vision this is more often called *perceptual grouping*.

Perceptual grouping has a long tradition in computer vision, but many especially of the earlier approaches suffered from susceptibility to scene complexity. Accordingly scenes tended to be “clean” or the methods required an unwieldy number of tunable parameters and heuristics to tackle scene complexity. A classificatory structure and a list of representative work for perceptual grouping methods in computer vision was introduced by Sarkar and Boyer [2, 18].

Generic object segmentation from 3D-data or from RGB-D images was in the past less popular, but a few methods exist [1, 3, 8, 19, 20]. Recently two methods have been published: The method of Mishra et al. [10] is an attention-driven active segmentation algorithm, designed to extract boundaries of (freestanding) simple objects, and the method by Ückermann et al. [21], which is an edge-based segmentation approach using pre-defined heuristics to end up with object hypotheses.

This article summarizes the work of Richtsfeld et.al. published in [12, 14–17]. Compared to other image segmentation work, a hierarchical grouping process over several levels of data abstraction is proposed using the structure of Sarkar and Boyer. Input data is organized in bottom-up fashion, stratified by layers of abstraction: signal, primitive, structural and assembly level, see Fig. 1. Raw sensor data is grouped in the signal level to surface patches, before the primitive level produces parametric surfaces and associated boundaries. Perceptual grouping principles are learned in the structural and assembly level to infer a value representing the connectivity between patches. Finally, a globally optimal segmentation is achieved using Graph-Cut on a graph consisting of the surface patches and the connectivity values between these patches.

The main contribution of the work is the combination of perceptual grouping with SVM learning following a

A. Richtsfeld (✉) · M. Zillich M. Vincze
Vienna University of Technology, Vienna, Austria
e-mail: andreas.richtsfeld@gmail.com

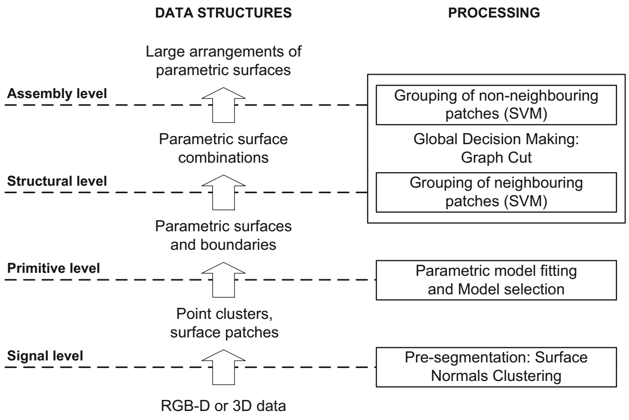

DATA STRUCTURES
PROCESSING
Large arrangements of parametric surfaces
Assembly level
Grouping of non-neighbouring patches (SVM)
Global Decision Making: Graph Cut
Parametric surface combinations
Structural level
Grouping of neighbouring patches (SVM)
Parametric surfaces and boundaries
Primitive level
Parametric model fitting and Model selection
Point clusters, surface patches
Signal level
Pre-segmentation: Surface Normals Clustering
RGB-D or 3D data

Fig. 1 Hierarchical perceptual grouping over four levels of data abstraction and associated processing methods

designated hierarchical structure. The learning approach of the framework enables segmentation for a wide variety of different objects in cluttered scenes, even if objects are partially occluded. Figure 2 shows segmentation of a complex scene, processed with the proposed framework.

2 Pre-Segmentation

3D cameras, such as Microsoft's Kinect or Asus' Xtion provide RGB-D data, consisting of a color image and the associated depth information for each pixel. The task of the pre-segmentation module is twofold: First the sensor characteristics are modelled and considered during normal calculation. Second, neighbouring pixels are clustered to uniform patches without discontinuities using the estimated normals.

A classic way to calculate the normals of a point-cloud is to locally fit planes to neighbouring 3D points. RGB-D sensors deliver organized point-clouds and a kernel is used to define the neighbourhood of a certain point. There are two parameters that influence the normal calculation: the kernel radius k_r and the inlier distance d_{in} to account for high deviations that would distort the local plane. The former one defines the number of points used and thus the smoothing of the normals, the latter the maximum allowed euclidean distance of the points within the window to the centre point to be considered for the normal calculation.

Recursive clustering of normals is controlled by the maximum allowed angle between normals γ_{cl} and by the maximum allowed normal distance of points d_{cl} to a hypothesized plane, defined by the mean of the already clustered point normals and the mean position.

3 Parametrization and Model Selection

In the last section uniform patches without discontinuities were extracted from RGB-D data. Parametrization of these patches to certain surface models reduces data size and leads to more meaningful abstractions. Two parametric models are chosen, a plane model to represent simple planar patches and B-spline surfaces which can model free-form surfaces allowing representation of difficult surfaces. B-splines could also represent planes makine explicit plane superfluous. But B-splines are more expensive in terms of data size, computation and and especially for further processing. More details of the used B-spline fitting approach can be found in [11].

To reduce the number of patches after parametrization neighbouring patches get merged after parametrization, if a joint parametric model fits better than the two individual models. To come to a decision, model selection with minimum description length (MDL) [8] is used.

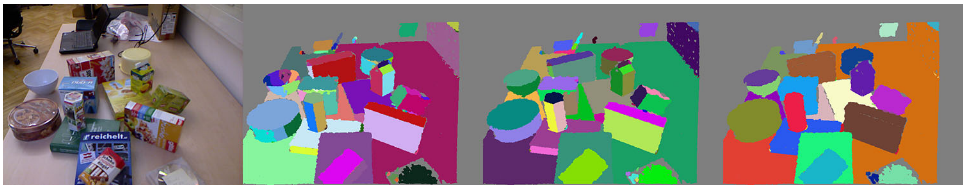
4 Parametric Surface Grouping

After the first two levels parametric surfaces are available for further processing in the structural and in the assembly level. A crucial task for surface grouping is to find relations between surface patches, indicating that they belong to the same object and to define them in a way that relations are valid for a wide variety of different objects.

Inspired by the already discussed Gestalt principles, the following relations between neighbouring surface patches are introduced, which will be used for the structural level:

- r_{co} ... Similarity of patch colour
- r_{rs} ... Similarity of patch size
- r_{tr} ... Similarity of texture quantity
- r_{ga} ... Similarity of texture: gabor filter
- r_{fo} ... Similarity of texture: fourier filter

Fig. 2 Original image, pixel clusters, parametric surface patches, segmented scene. (more examples in [12, 16])



- r_{co3} ... Similarity of color on patch border
- r_{cu3} ... Mean curvature on patch border
- r_{cv3} ... Curvature variance on patch border
- r_{di2} ... Mean depth on patch border
- r_{vd2} ... Variance of depth on patch border
- r_{2d3} ... 3D-2D boundary ratio

The assembly level is the last level of grouping and is responsible to group spatially separated surface groupings. Similar to the structural level, relations between patches are introduced. The first five relations are equal to the relations used at the structural level and the following are added:

- r_{md} ... Minimum distance between patches
- r_{nm} ... Similarity of mean of surface normals direction
- r_{nv} ... Similarity of variance of surface normals direction
- r_{ac} ... Diff. of normals direction of nearest contour points
- r_{dn} ... Mean (normal) distance of nearest contour points
- r_{cs} ... Collinearity continuity
- r_{oc} ... Mean collinearity occlusion
- r_{ls} ... Closure line support
- r_{as} ... Closure area support
- r_{lg} ... Closure lines to gaps

A feature vector r_{st} for the structural level is defined, containing all relations between neighbouring patches and a feature vector r_{as} for the assembly level, containing all relations between non-neighbouring patches.

Now one has to decide, whether two surface patches belong together or not. This decision is based on the relations of the feature vector. Setting thresholds for classification is getting more complex the more relations are used and would not be manually adjustable any more. A solution to this problem lies in learning of the grouping rules using a learning method which classifies feature vectors to single decision values.

In this approach we use a support vector machine (SVM) to learn to classify the given feature vectors. SVMs are maximum margin classifiers, i.e. they try to find a separating hyperplane between different classes in the data with the maximum margin. SVMs support non-linear classification by using a kernel, mapping input data from a general set S into an inner product space V, which is of higher dimension than the input space. This is done in the hope that the data will gain meaningful linear structure.

For the offline training and online testing phase the freely available *libsvm package* [4] is used. After training the SVM is not only capable to provide a binary decision *same* or *notsame* for each feature vector $\mathbf{r}$, but also a probability value $p(same\,|\,\mathbf{r})$ for each decision, based on the theory introduced by Wu et al. [24]. As solver we use C-support vector classification (C-SVC) with $C = 1$, $\gamma = 1/n$ and $n = 9$ and as kernel the radial basis function (RBF):

$$\mathbf{K}(\mathbf{x_i}, \mathbf{x_j}) = \mathbf{e}^{\gamma||\mathbf{x_i}-\mathbf{x_j}||^2} \tag{1}$$

Hand-annotated ground truth segmentation from a set of RGB-D images is used with the estimated feature vectors r_{st} and r_{as} to train a SVM for each level during an offline training phase. Feature vectors of patch pairs from the same object represent positive training examples and vectors of pairs from different objects or objects and background represent negative examples. With this strategy, not only the affiliation of patches to the same object, but also the disparity of object patches to other objects or background is learned.

After the learning phase the SVMs are able to classify the feature vectors, delivering a probability value for each patch pair. When using the estimated probabilities, groups of neighbouring surface patches could be formed by applying a threshold [e.g., $p(same\,|\,\mathbf{r}) = \mathbf{0.5}$]. With this strategy, a single wrong decision of the SVM (e.g., $p = 0.51$) would probably lead to wrongly connected objects. Hence, an optimal object hypotheses can not be created by simply thresholding these values. Instead, a globally optimal solution can be found by building a graph and performing Graph-Cut segmentation.

5 Global Decision Making

After SVM classification in the structural and assembly level some probability estimates may contradict when trying to form object hypotheses. A globally optimal solution has to be found to overcome vague or wrong local predictions from the SVMs at the structural and assembly level. To this end a graph is defined, where surface patches represent nodes and edges are represented by the probability values of the SVMs. Graph-cut segmentation method introduced by Felzenszwalb et al. [5] is emplyed, using the probability values as the pairwise energy terms to find a global optimum for object segmentation.

6 Evaluation

After all parts of the framework are introduced, evaluation of the proposed object segmentation method is shown. Because of the limited space only a part of the evaluation of [12] can be presented. The proposed object segmentation method is compared with the method of Mishra et al. [10] and the method by Ückermann et al. [21].

Table 1 Precision and recall on the OSD and Willow Garage dataset for the approach by Mishra et al [10], Ückermann et al. [21] and for the proposed approach, when using the SVM of the structural level SVM_{st} and when using both data abstraction levels SVM_{st+as}

	Mishra		Ückermann		SVM_{st}		SVM_{st+as}	
	$P(\%)$	$R(\%)$	$P(\%)$	$R(\%)$	$P(\%)$	$R(\%)$	$P(\%)$	$R(\%)$
OSD: boxes	76.87	75.86	97.12	94.72	96.47	97.91	96.47	97.91
OSD: stacked objects	70.57	74.61	95.61	93.26	86.70	96.23	86.72	97.54
OSD: occluded objects	67.37	55.81	94.53	74.76	94.18	78.23	94.00	91.62
OSD: cylindric objects	69.81	87.38	96.47	92.50	96.21	97.11	87.35	97.71
OSD: mixed objects	62.99	76.29	95.27	93.42	91.21	95.90	91.21	95.90
OSD: complex scenes	61.06	54.61	93.14	83.49	87.50	91.49	86.78	92.09
Complete OSD dataset	66.10	67.91	94.91	88.79	90.85	93.88	89.95	95.00
Willow dataset	77.51	83.82	98.69	98.83	98.11	98.82	98.10	98.81

Evaluation is done on the object segmentation database (OSD) [13] as well as on the Willow Garage dataset [1]

Table 1 shows *PrecisionP* and *RecallR* of segmentation from the OSD for the algorithms of Mishra and Ückermann and for both methods, when using just the support vector machine of the structural level SVM_{st} or when using also the SVM of the assembly level SVM_{st+as} to build relations between estimated patches. The SVMs are trained with the four learning sets of the OSD for all experiments, even for the evaluation of the Willow Garage dataset. This shows the generalization of the presented approach with respect to other objects and scenes during training.

The results in Table 1 show that the presented method works significantly better than the approach by Mishra for all sets of the OSD as well as for the Willow Garage dataset. In contrast the results of Ückermann are almost similar to this method. A closer look on the values shows a higher precision P but at the same time a lower recall R. This indicates that their approach avoids wrong assignments of surfaces, but at the cost of sometimes over-segmenting the objects.

The benefit of using the assembly level can be seen for the occluded objects set of the OSD. Recall is much higher when additionally using the assembly level, while precision remains almost constant on a high level. This demonstrates that occluded parts have been connected without wrongly assigning surface patches.

Evaluation of the method by Mishra on the Willow Garage dataset shows better performance compared to the OSD dataset, because of the reduced complexity of scenes. Objects in the dataset are mainly free-standing on a ground plane and there are no occluded objects. Segmentation with the proposed approach performs also well on such examples, but the benefit when using the assembly level is not evident any more in this case. But when considering that the SVMs have been trained with data of the OSD, this can be interpreted as an evidence that perceptual grouping rules act in a generic manner and are portable into different situations with different types of objects.

7 Conclusion

A framework was presented for segmenting unknown objects of reasonably compact shape in cluttered table top scenes of RGBD-images. Raw input data is abstracted in a hierarchical framework. Instead of matching geometric object models, more general perceptual grouping rules are learned with support vector machines (SVMs) to group parametric surfaces. Experiments have shown the generic manner of the learned rules on different datasets with different objects and a comparison with state of the art methods show the good performance compared to other methods.

References

1. Boyer KL, Mirza MJ, Ganguly G (1994) The robust sequential estimator : a general approach and its application to surface organization in range data. IEEE Trans Pattern Anal Mach Intell (PAMI) 16(10):987–1001
2. Boyer KL, Sarkar S (1999) Perceptual organization in computer vision: status, challenges, and potential. Comput Vision Image Underst 76(1):1–5
3. Campbell N, Vogiatzis G, Hernández C, Cipolla R (2007) Automatic 3D object segmentation in multiple views using volumetric graph-cuts. Br Mach Vision Conf 28:530–539
4. Chang CC, Lin CJ (2011) LIBSVM : a library for support vector machines. ACM Trans Intell Syst Technol 2(3):1–27
5. Felzenszwalb PF, Huttenlocher DP (2004) Efficient graph-based image segmentation. Int J Comput Vision 59(2):167–181
6. Koffka K (1935) Principles of Gestalt psychology, international library of psychology, philosophy, and scientific method, vol 20. Harcourt, Brace and World
7. Köhler W (1959) Gestalt psychology today. Am Psychol 14(12):727–734
8. Leonardis A, Gupta A, Bajcsy R (1995) Segmentation of range images as the search for geometric parametric models. Int J Comput Vis 14(3):253–277
9. Metzger W (1936) Laws of seeing. The MIT Press

[1] http://www.acin.tuwien.ac.at/index.php?id=312.

10. Mishra AK, Shrivastava A, Aloimonos Y (2012) Segmenting simple objects using RGB-D. In: International conference on robotics and automation (ICRA), pp 4406–4413
11. Mörwald T (2013) Object modelling for cognitive robotics. Ph.D. thesis, Vienna University of Technology
12. Richsfeld A (2013) Robust object detection for robotics using oerceptual organization in 2D and 3D. Ph.D. thesis
13. Richtsfeld A (2012) The object segmentation database (OSD). http://www.acin.tuwien.ac.at/?id=289
14. Richtsfeld A, Mörwald T, Prankl J, Balzer J, Zillich M, Vincze M (2012) Towards scene understanding object segmentation using RGBD-images. In: Proceedings of the 2012 computer vision winter workshop (CVWW). Mala Nedelja, Slovenia
15. Richtsfeld A, Mörwald T, Prankl J, Zillich M, Vincze M (2012) Segmentation of unknown objects in indoor environments. In: IEEE/RSJ international conference on intelligent robots and systems, pp 4791–4796
16. Richtsfeld A, Mörwald T, Prankl J, Zillich M, Vincze M (2013) Learning of perceptual grouping for object segmentation on RGB-D data. J Vis Commun Image Represent
17. Richtsfeld A, Zillich M, Vincze M (2012) Implementation of Gestalt principles for object segmentation. In: 21st international conference on pattern recognition (ICPR). Tsukuba, Japan
18. Sarkar S, Boyer KL (1993) Perceptual organization in computer vision—a review and a proposal for a classificatory structure. IEEE Trans Syst Man Cybern 23(2):382–399
19. Sedlacek D, Zara J (2009) Graph cut based point-cloud segmentation for polygonal reconstruction. In: 7th international conference on computer vision systems, pp 218–227
20. Strom J, Richardson A, Olson E (2010) Graph-based segmentation for colored 3D laser point clouds. In: IEEE/RSJ international conference on intelligent robots and systems (IROS), pp 2131–2136
21. Ückermann A, Haschke R, Ritter H (2012) Real-time 3D segmentation of cluttered scenes for robot grasping. In: 12th IEEE-RAS international conference on humanoid robots
22. Wertheimer M (1923) Untersuchungen zur Lehre von der Gestalt II. Psychol Res 4(1):301–350
23. Wertheimer M (1958) Principles of perceptual organization. In: a source book of Gestalt psychology, pp 115–135
24. Wu TF, Lin CJ (2004) Probability estimates for multi-class classification by pairwise coupling. J Mach Learn Res 5:975–1005