

## Segmentation of Depth Data in Piece-wise Smooth Parametric Surfaces

Aitor Aldoma, Thomas Mörwald, Johann Prankl and Markus Vincze  
Vision4Robotics, ACIN, Vienna University of Technology  
{aldoma, moerwald, prankl, vincze}@acin.tuwien.ac.at

**Abstract.** *This paper describes an efficient algorithm to extract piece-wise smooth surfaces from depth images. The algorithm is based on the Mumford-Shah (MS) functional. A solution is obtained by means of a multi-model and multi-scale region merging strategy that does not require to define the number of regions in advance. Our current formulation allows smooth regions to be modeled either as planar or B-splines surfaces and thus provides a parametric representation of the scene upon convergence. Additionally, we propose a final refinement step that corrects initial region boundaries obtained by means of supervoxel segmentation. This final stage results in smooth boundaries (due to the boundary length penalization in the MS) that better separate different regions in the scene. We demonstrate the performance of the proposed algorithm in indoor scenes, acquired with RGB-D sensors, showcasing man-made objects and structures.*

### 1. Introduction

Segmentation of images into meaningful structures is a major research area in the field of computer vision. Even though segmentation has been predominantly investigated for intensity and color images, the recent appearance of RGB-D sensors has sparked a renewed interest among roboticists. Clearly, the availability of depth data in conjunction with color images provides additional cues to aid in segmentation. Segments cannot only be assessed by their similarity in color space, but also by their continuity and smoothness in Euclidean space. Nevertheless, while computer vision scientists have adopted energy minimization techniques (which in some cases consider the whole extend of the

image as well as interaction among segments) to address the challenges present in segmentation, recent approaches making use of depth information still rely strongly on local heuristics (in particular during the initial stages of the segmentation pipeline) to determine the extend of individual regions in an image. While these algorithms perform well in the envisioned situations, their strong dependence on local properties of the data results in an undesired lack of robustness to local perturbations. This results in complex pipelines that are difficult to adapt to novel situations or slightly different sensors.

Because of the aforementioned caveats and inspired by recent trends in the segmentation of intensity images, this paper formalizes the segmentation of depth images into piece-wise smooth surfaces within the Mumford-Shah framework (see Section 3). We propose an algorithm (based on Koepfler *et al.* [6]) to obtain an approximated solution of the functional that upon convergence results in a parametric surface representation of the input data (see Section 4). We demonstrate the performance of the proposed approach in Section 5 on two datasets acquired with RGB-D sensors but with different characteristics vouching for the generalization capabilities of the proposed framework.

### 2. Related work

Various approaches to segment images into larger patches exist. Most of them are based on simple color and edge features [4, 17, 21, 22, 2, 19], some include depth information [7, 10, 20] and others rely on the estimation of shape primitives [9, 5] or combine 3d-shape with color information [16, 11]. In the following paragraph we review aspects of these approaches starting with

algorithms relying on appearance cues.

Many approaches formulate image segmentation as energy minimization with a MRF [17, 21, 22]. In addition to an appearance model computed from color and texture Werlberger et al. [22] introduce a shape prior which is modeled as a Geodesic Active Contour energy. In [2] and [19] the objective function is formalized with the Mumford-Shah functional [12]. Bernard et al. [2] introduce a continuous parametric function using B-splines to model a contour energy term. Strelakovski and Cremers [19] rewrite the proximal operator in a primal-dual algorithm using Moreau’s identity to achieve real-time performance.

A graph cut is also used in [7, 10]. While Kootstra et al. [7] include the disparity deviation of pixels to the dominant plane and solve an MRF-formulation using  $\alpha/\beta$  swap [3], Mishra et al. [10] use fixation points and a shortest path in a log polar transformed edge image. Ückermann et al. [20] propose a model-free algorithm which subsequently combines smooth surface patches, directly computed in depth images, to form object hypothesis. The approach by Hager et al. [5] is able to segment objects from cluttered scenes in point clouds by using a strong prior 3d model. Hence, it is limited to parametric models such as boxes and cylinders. The problem of fitting higher order surfaces to point clouds was addressed by Leonardis et al. [9]. They segment range images by estimating piecewise linear surfaces, modeled with bivariate polynomials. A Model Selection framework based on the Minimum Description Length (MDL) principle is used to find the best interpretation of the scene. MDL for Model Selection is also used in [11]. Instead of piecewise linear surfaces Mörwald et al. use planes and B-spline surfaces.

Like Mörwald et al. the approach in this paper uses basic surface models, such as planes and B-splines. Instead of using Model Selection and MDL where the complexity for each model needs to be defined with respect to their number of parameters, we integrate these surface models into the Mumford-Shah functional [12] and model complexity is implicitly encoded by the curvature of the regional surfaces.

### 3. Piece-wise Smooth Segmentation

This section briefly reviews the Mumford-Shah framework for image segmentation. Then, we propose an adaptation of the functional for the segmentation of depth images into piece-wise smooth parametric surfaces.

#### 3.1. Mumford-Shah framework

In a nutshell, the celebrated Mumford-Shah functional [12] is used to establish an optimality criterion to segment an image into a disjoint set of sub-regions. The aim of the functional is to find an approximation  $I$  of an input image  $I_o$  such that (i)  $I$  is similar to  $I_o$ , (ii)  $I$  is smooth within the different sub-regions and (iii) the boundaries between regions are of minimal length. In the continuous setting, the functional is formulated as

$$E(I, C_i) = \int_{\Omega} \|I - I_0\|^2 dx + \beta \int_{\Omega \setminus C_i} \|\nabla I\|^2 dx + \alpha \int_{C_i} ds, \tag{1}$$

where  $\Omega$  is the image domain and  $C_i$  represents the boundaries of the different sub-regions in the image.  $\alpha$  and  $\beta$  are parameters ( $\geq 0$ ) penalizing lack of smoothness within regions and boundary length, respectively. Of special interest is the piecewise constant Mumford-Shah model when  $\beta \rightarrow \infty$  enforcing the different regions in the image approximation,  $I$ , to be constant.

#### 3.2. Multi-model MS for depth images

This section proposes a set of modifications to the MS framework in order to extract piece-wise smooth parametric surfaces from a depth image. *Multi-model* refers to the availability of different parametric surface models (with increasing expressiveness and potentially decreasing smoothness) to approximate piece-wise smooth sub-regions in the input data. In our current formulation, surfaces can be represented by planar or B-splines (with 3x3 control points) surfaces. Please note that these two parametric models of surfaces are by construction smooth and differentiable. Thus, (1) becomes in our setting:

$$\begin{aligned}
E(D, C_i) = & \int_{\Omega} \|D - D_0\|^2 dx + \beta \int_{\Omega \setminus C_i} \kappa^2 dx \\
& + \alpha \int_{C_i} ds.
\end{aligned} \tag{2}$$

where  $D_0$  represents the input depth image and  $D$  represents an approximation of the input depth and is composed by different piecewise smooth regions parametrically modeled either as planar or B-spline surfaces. Note that in our specific setting, the second term of (2) penalizes the curvature  $\kappa$  of the approximating surface instead of  $\|\nabla D\|$ . This formulation allows on one hand to overcome the problem of favoring fronto-parallel planar surfaces (with  $\|\nabla D\| = 0$ ) over equally planar but slanted surfaces (with  $\|\nabla D\| > 0$ ) [15, 8]. On the other hand, it favors regional models with less expressiveness (e.g. planar surfaces) over richer models (e.g. B-splines) presenting higher complexity. Intuitively, we would prefer segmentations that use simpler parametric models unless there is a good reason to increase the model complexity, such as low regional data fidelity and/or reduction of the boundary length.

## 4. Implementation

This section revolves around the implementation of the piecewise smooth surface segmentation framework proposed in the previous section. In particular, we address the problem of minimizing the functional in (2).

### 4.1. Overview

Provided with a depth image of a scene,  $D_0$ , the algorithm starts by computing an oversegmentation of the scene in terms of supervoxels. These small regions are the basis to minimize (2) and provide as well the initial boundaries between different sub-regions. In particular, a solution is obtained by incrementally merging pairs of adjacent regions (i.e., sharing a boundary) that improve the functional energy. Our multi-model scheme is introduced here by first trying to reduce the energy by fitting planes to neighboring regions. Once the energy cannot be further reduced, the model expressiveness is increased and the merging process is restarted by fitting B-splines surfaces to connected regions. Upon convergence, a refinement stage is performed that

swaps the associated region at pixels located at the boundary between two regions. This final stage aims at improving the initial boundaries provided by the over-segmentation in the scene in situations where they do not adhere properly to the actual boundaries of the smooth surfaces in the scene. The outcome of the proposed algorithm at different stages is depicted in Figure 1.

### 4.2. Oversegmentation

Over-segmentation of an image into regions of similar pixels, known as superpixels, is a widely used preprocessing step in order to reduce the amount of data for subsequent computationally expensive algorithms. We use the method of Papon et al. [13], which is able to cluster a set of points using color and the 3D information. The main idea is to select spatially uniform distributed (in Euclidean space instead of image space) seed points and to iteratively cluster neighboring points enforcing spatial connectivity and smoothness. In contrast to traditional superpixel algorithms working on image space [1], this results in supervoxels which do not flow across boundaries in 3D space and are smooth by considering surface normals. The implementation used in this paper is the one provided by the original authors within the Point Cloud Library. The supervoxel extraction is governed in our case by two parameters indicating spatial compactness and smoothness. Please note that supervoxels provide on one hand an initial reduction of the number of regions and on the other hand, pixels get grouped together in larger regions that allow the extraction of parametric surface models.

### 4.3. Multi-scale and multi-model region merging

The previous stage results in an oversegmentation of the image domain into a disjoint set of regions  $\Omega = \{R_1 \cup R_2 \cup \dots \cup R_i \cup \dots \cup R_n\}$ .  $C_i \in R_i$  is defined as the boundary between  $R_i$  and adjacent regions. Provided with this initial set of regions and boundaries, this section describes the algorithm to minimize the functional in (2). To this end, we propose an adaptation of the multi-scale algorithm by Koepfler *et al.* [6] that is reviewed in the following for completeness. They minimize the piece-wise constant Mumford-Shah model for an intensity image  $I_0$

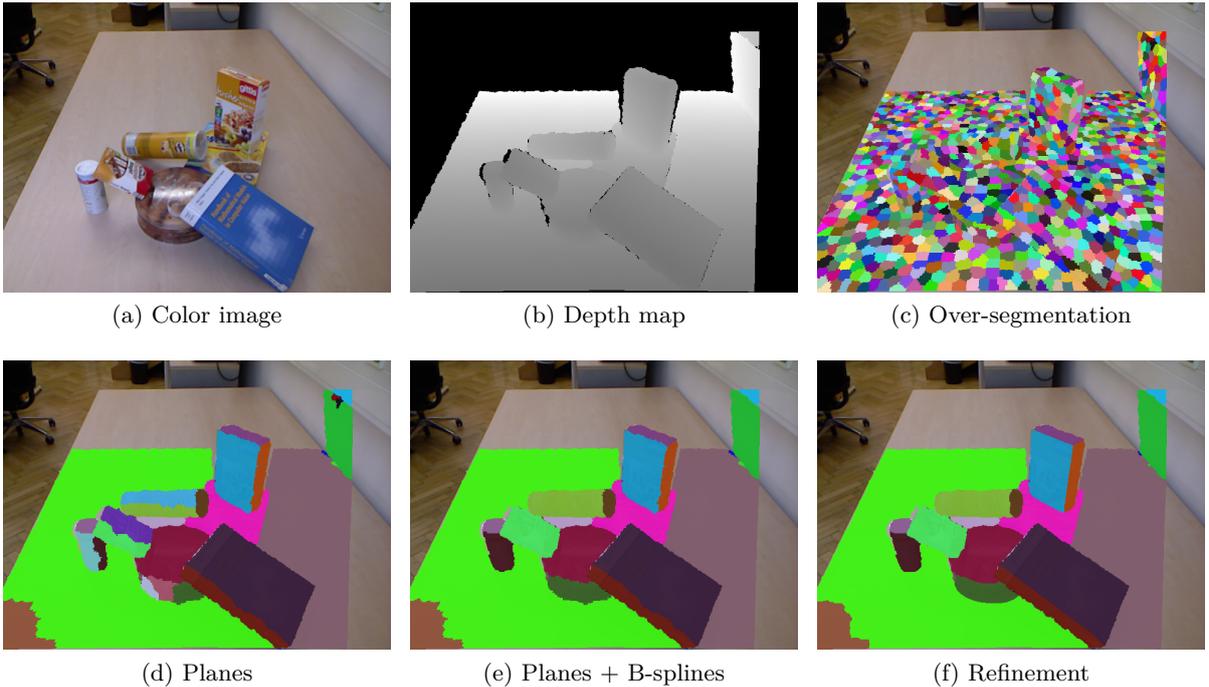


Figure 1: Overview of the different stages of the method

$$E(I, C_i) = \int_{\Omega} (I - I_0)^2 dx + \alpha \int_{C_i} ds, \quad (3)$$

where the smoothness term has been dropped by letting  $\beta \rightarrow \infty$  in (1). The algorithm in [6] proceeds by iteratively merging adjacent regions whereby the different regions composing the piece-wise constant approximation  $I = \{R_1 \cup \dots \cup R_n\}$  are modeled by the average intensity of all pixels within each region. In a nutshell, at each iteration, the algorithm selects the merging move with minimal  $\hat{\alpha}_k$ . The  $\hat{\alpha}_k$  of a certain move  $k$  representing the merging of two regions  $R_i$  and  $R_j$  is defined as:

$$\hat{\alpha}_k = \frac{-\Delta \mathcal{E}_{region}}{\Delta \mathcal{E}_{length}} = \frac{-(\mathcal{E}_{R_i} + \mathcal{E}_{R_j} - \mathcal{E}_{R_i \cup R_j})}{|C_i| + |C_j| - |C_{ij}|} \quad (4)$$

where  $C_{ij}$  represent the boundary length obtained by merging both regions and  $\mathcal{E}_{\{R_i, R_j, R_i \cup R_j\}}$  represent the regional error for a piece-wise constant region. The algorithm terminates when all possible merging moves in the current state have an  $\hat{\alpha}_k$  larger than the user

parameter  $\alpha$ , indicating the lack of energetically favorable moves. The multi-scale attribute arises from the fact that as the algorithm proceeds, the boundary length penalizer  $\hat{\alpha}_k$  is incrementally increased. Therefore, it is possible to obtain different segmentations at different scales.

In contrast to [6], we propose a modification that is very similar to the original algorithm but with two main differences:

- 1) We minimize the piece-wise smooth MS instead of the piece-wise constant model by allowing regions to be modeled as parametric smooth surfaces, and
- 2) We incrementally increase the model complexity representing piece-wise smooth regions once the energy cannot be further reduced by simpler models.

Therefore, (4) becomes:

$$\hat{\alpha}_k = \frac{-\Delta \mathcal{E}_{region} - \beta \Delta \mathcal{E}_{smooth}}{\Delta \mathcal{E}_{length}}, \quad (5)$$

and due to 2), the proposed algorithm works not only at multiple scales but also with different model complexities. In our current implementation, with two piece-wise smooth models (i.e.

planar and B-splines surfaces), our algorithm can be considered a two-pass version of the algorithm of Koepfler (see Algorithm 1). Using the appropriate data structures as well as exploiting incremental computation properties of surface parametric models (see Section 4.4), merge moves can be efficiently implemented.

---

**Algorithm 1** Multi-scale and multi-model region merging

---

```

Input:  $\alpha, \beta$ 
Models = {PLANE, BSPLINE_3x3}
 $m \leftarrow 0$ 
 $\mathcal{C} = \{R_i, R_j, \hat{\alpha}_k\}$  //sorted merging candidates
converged  $\leftarrow$  false
while not converged do
   $c = \{R_i, R_j, \hat{\alpha}_k\} \leftarrow \text{pop}(\mathcal{C})$ 
  if  $\hat{\alpha}_k > \alpha$  then
    if  $m \geq \text{length}(\text{Models})$  then
      converged  $\leftarrow$  true
    else
      //increase model type
       $m \leftarrow m + 1$ 
      //merging candidates with current model type
       $\mathcal{C} \leftarrow \text{comp\_candidates}(\text{Models}[m], \mathcal{C})$ 
      continue
    end if
  end if
  //apply merge and update structures
  {new_cands, affected}  $\leftarrow$  merge( $c$ )
   $\mathcal{C} \leftarrow \text{remove\_candidates}(\text{affected}, \mathcal{C})$ 
  new_cands  $\leftarrow$  comp_candidates(Models[ $m$ ], new_cands)
  insert_sorted( $\mathcal{C}$ , new_cands)
end while

```

---

## 4.4. Model fitting

The algorithm proposed in the previous section relies on the ability to fit planar and B-splines models to regions in the scene that build up the piece-wise smooth approximation ( $D$ ) of the input data  $D_0$ . To this end, this section focuses on how to incrementally (whenever possible) and efficiently extract the parametric representation of regions as the algorithm iterates.

### 4.4.1 Planar surfaces

Planar surfaces are a good initial choice to parametrically approximate unknown surface data:

- 1) Locally, planar models can approximate almost any structure.
- 2) Planar surfaces are a recurrent structure in man-made environments.
- 3) They can be efficiently estimated by first- and second-order moments of the underlying data followed by Eigenvector analysis of the resulting 3x3 covariance matrix.

In addition, because first- and second-order moments can be incrementally computed, planar models become a very efficient model for region merging strategies. In other words, the planar fit of two regions that are to be merged can be efficiently computed by reusing the previously computed statistics of the individual regions.

**Relation to (2):** The regional fit of a region,  $R_i$ , modeled as a planar surface is computed as the squared depth error of the underlying pixels to the model. Regarding the smoothness term, planar surfaces do not present any curvature and thus, the smoothness term has no effect in the energy for any region modeled as a planar surface.

### 4.4.2 B-spline surfaces

Modeling curved surface areas is a well studied problem and there are many mathematical solutions such as superquadrics, wavelets and bivariate polynomials to name a few. We choose B-splines due to their beneficial properties:

- 1) They are very flexible w.r.t. the degrees of freedom we wish to model.
- 2) Derivatives and curvature may be computed explicitly at any point of the surface.
- 3) The mathematical formulation of fitting a B-spline to a point-cloud or depth map becomes solving a linear system of equations.

A B-spline surface is defined as the sum of weighted basis functions

$$S(\xi, \eta) = \sum_{j=1}^m \varphi_{j,p}(\xi, \eta) \mathbf{b}_j \quad (6)$$

where  $(\xi, \eta) \in R$  and  $\varphi_j(\xi, \eta)$  is a bivariate basis function which can be efficiently evaluated by the *Cox-de-Boor* algorithm. They define the influence of the weights, also called control points  $\mathbf{b}_j$ . The polynomial order of the basis functions is denoted by  $p$ . A full explanation of B-splines is available in the book of Piegl et al. [14]. Note that we embed the B-spline surface into the domain of the depth map thus becoming a function  $S : \mathbb{R}^2 \rightarrow \mathbb{R}$ .

Fitting to a depth image  $D : \mathbb{R}^2 \rightarrow \mathbb{R}$  is the problem of finding control points such that the distance between  $D$  and  $S$  is minimized. Since we aim for piecewise smooth regions, a least squares optimization w.r.t. the control points is

a sufficiently accurate approximation.

$$\min_{\mathbf{b}} \int_R \|D(\xi, \eta) - S(\xi, \eta, \mathbf{b})\|^2 \quad (7)$$

where  $\mathbf{b}$  denotes a vector collecting the control points. This is equivalent to the first term of Eq. (2). We define the B-spline domain to match the index space of the depth image. This allows to conveniently query surface points and its derivatives (up to order  $p - 1$ ) at any image location  $(\xi, \eta)$ .

**Relation to (2):** The attentive reader might already have noticed that the functional we minimize in Eq. (7) is equivalent to the first term of Eq. (2). By using the Greville abscissae and re-projecting into  $\mathbb{R}^3$  we obtain the B-spline control points and therefore the surface in Euclidean space ( $\mathbf{S} \in \mathbb{R}^3$ ). We then explicitly evaluate the mean curvature  $\kappa$  for computing the second term as

$$\kappa = \left\langle \frac{\partial^2 \mathbf{S}}{\partial^2 \xi}, \mathbf{n} \right\rangle + \left\langle \frac{\partial^2 \mathbf{S}}{\partial^2 \eta}, \mathbf{n} \right\rangle \quad (8)$$

with  $\mathbf{n}$  being the normal surface of the B-spline

$$\mathbf{n} = \frac{\partial \mathbf{S}}{\partial \xi} \times \frac{\partial \mathbf{S}}{\partial \eta}. \quad (9)$$

#### 4.5. Refinement stage

So far, we have focused on the minimization of the functional in (2) by merging neighboring regions as described in sections 4.3 and 4.4. While being a successful strategy, it suffers from the inability to change the location of initial region boundaries resulting from the over-segmentation stage. Therefore, if supervoxels flow across object boundaries, the merging moves will not be able to correct these artifacts. Aiming at further minimizing the energy cost, we propose a refinement stage that includes another variety of moves. In particular, the refinement stage aims at swapping the region association of pixels at the boundary between regions provided that this swap minimizes the functional. This simple strategy results in the removal of wiggly boundaries due to a reduction of the overall boundary length as well as a better pixel-wise association due to a reduction of the data error term. By applying this refinement stage after the functional cannot be further minimized by means of merging moves, the overall cost of this stage is computationally acceptable (since the number of

boundary pixels has in general been greatly reduced prior to this stage by merging operations).

## 5. Experimental results

This section provides an initial qualitative evaluation of the proposed method. Figure 2 and 3 show the resulting segmentation for three scenes from the OSD0.2 [16] and three from the NYU-depth (v2) [18] dataset respectively. Both datasets have been acquired indoor using RGB-D cameras but as it can be seen from the images they showcase different scenarios. In particular, OSD focuses on the segmentation of household objects in table-top scenarios. On the other hand, the NYU dataset includes thousands of scenes from domestic environments and its focus lies on larger objects (e.g., furniture, room structure, etc). A major distinctive trait among both datasets is the depth range covered by the datasets. While most of the objects in OSD are to be found not farther away than  $1.5m$  from the sensor, the NYU dataset depth range is much larger. It is a well known fact that the quality of RGB-D data degrades rapidly after  $2m$  and therefore, segmentation of meaningful structures on the NYU dataset is much more challenging, specially for algorithms like the one proposed in this paper relying solely on depth information.

These differential traits have required a different parameter setting for both datasets (see captions of Figure 2 and 3 for specific values). Overall, we can see that the scenes get segmented into meaningful structures vouching for the efficiency of the proposed method. However, in Figure 3 one can observe how segmentation quality degrades due to the noise in the data as the distance to the camera increases. Figure 4 shows the reconstructed point cloud from the depth data obtained after minimizing the proposed functional. As expected, noise in the data gets smoothed by using smooth parametric surface models. Finally, Figure 5 shows the effect of the boundary regularizer on a scene from the OSD dataset. As boundaries become more costly, larger structures arise.

## 6. Conclusions and future work

This paper has presented a formulation based on the Mumford-Shah functional to segment depth data into smooth surfaces. Our prelimi-

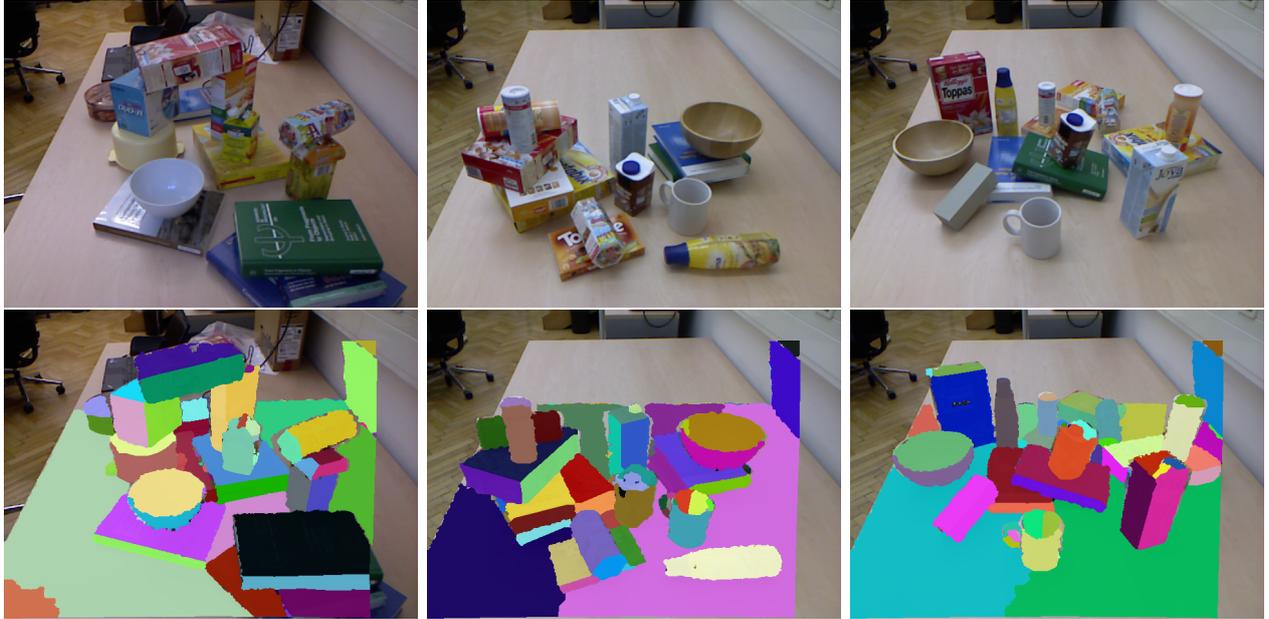


Figure 2: Qualitative results for three scenes in the OSD dataset.  $\alpha = 1.5^{-4}$ ,  $\beta = 1$ .

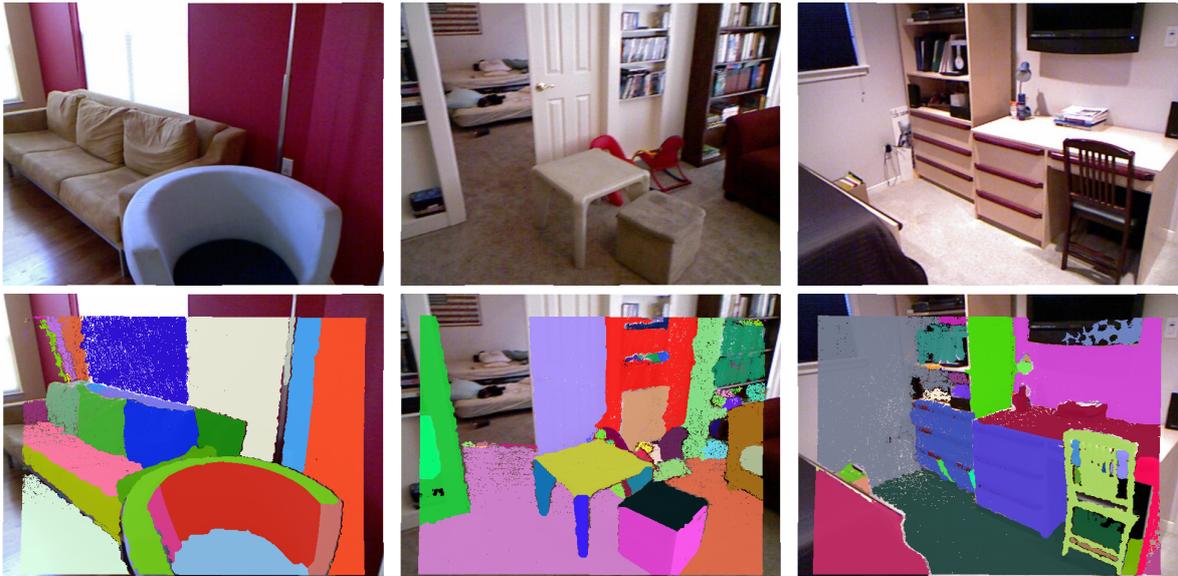


Figure 3: Qualitative results for three scenes in the NYU dataset.  $\alpha = 1^{-2}$ ,  $\beta = 2$ . Depth information beyond 3.5m is ignored.

nary results show that the method is an effective and elegant alternative for the task at hand. In the future, we plan to extend our current model to more complex models being able to represent simple objects (i.e. by means of superquadrics) as well as the addition of appearance information to improve segmentation in situations where depth data becomes unreliable. The addition of shape priors based on the knowledge of recurrent

objects is another interesting research direction.

### Acknowledgements

The research leading to these results has received funding from the European Community's Seventh Framework Programme FP7/2007-2013 under grant agreement No. 600623, STRANDS and from the Austrian Science Foun-

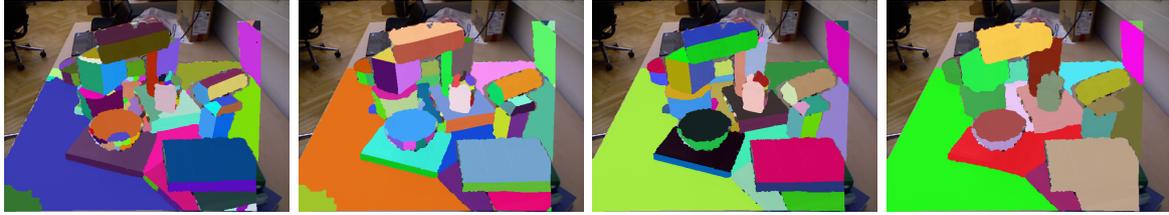


Figure 5: Effects of boundary regularizer ( $\alpha = 5^{-5}, 1^{-4}, 2^{-4}, 3^{-4}$ )

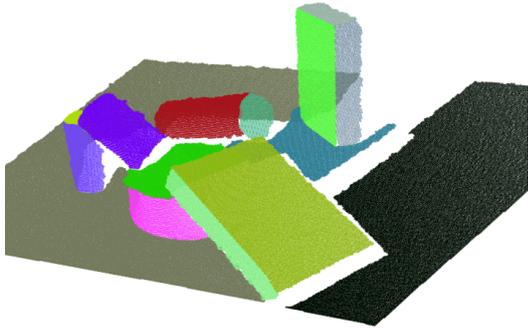


Figure 4: Resulting point cloud after segmentation using the proposed method. The depth of the points has been corrected to lie on the underlying parametric model surface.

ation (FWF) under grant agreement No. I513-N23, vision@home.

## References

- [1] R. Achanta, A. Shaji, K. Smith, A. Lucchi, P. Fua, and S. Suesstrunk. Slic superpixels compared to state-of-the-art superpixel methods. *TPAMI*, 2012. 3
- [2] O. Bernard, D. Friboulet, P. Thevenaz, and M. Unser. Variational b-spline level-set method for fast image segmentation. In *Biomedical Imaging: From Nano to Macro 5th IEEE International Symposium on*, 2008. 1, 2
- [3] Y. Boykov, O. Veksler, and R. Zabih. Fast approximate energy minimization via graph cuts. *TPAMI*, 2001. 2
- [4] P. F. Felzenszwalb and D. P. Huttenlocher. Efficient Graph-Based Image Segmentation. *IJCV*, 2004. 1
- [5] G. D. Hager and B. Wegbreit. Scene parsing using a prior world model. *The International Journal of Robotics Research*, 2011. 1, 2
- [6] G. Koepfler, C. Lopez, and J. Morel. A multi-scale algorithm for image segmentation by variational method., 1994. 1, 3, 4
- [7] G. Kootstra, N. Bergström, and D. Kragic. Fast and Automatic Detection and Segmentation of Unknown Objects. In *Humanoids*, Bled, 2011. 1, 2
- [8] G. Kusch and D. Cremers. Fast and accurate large-scale stereo reconstruction using variational methods. In *ICCVW*, 2013. 3
- [9] A. Leonardis, A. Gupta, and R. Bajcsy. Segmentation of range images as the search for geometric parametric models. *IJCV*, 1995. 1, 2
- [10] A. K. Mishra, Y. Aloimonos, L. F. Cheong, and A. Kassim. Active Segmentation. *TPAMI*, 2011. 1, 2
- [11] T. Moerwald, A. Richtsfeld, J. Prankl, M. Zillich, and M. Vincze. Geometric data abstraction using b-splines for range image segmentation. In *ICRA*, 2013. 1, 2
- [12] D. Mumford and J. Shah. Optimal approximations by piecewise smooth functions and associated variational problems. *Communications on Pure and Applied Mathematics*, 42(5):577–685, 1989. 2
- [13] J. Papon, A. Abramov, M. Schoeler, and F. Wörgötter. Voxel cloud connectivity segmentation - supervoxels for point clouds. In *CVPR*, 2013. 3
- [14] L. Piegl and W. Tiller. *The NURBS book*. Monographs in visual communication. Springer, 1996. 5
- [15] R. Ranftl, S. Gehrig, T. Pock, and H. Bischof. Pushing the limits of stereo using variational stereo estimation. In *IV*, 2012. 3
- [16] A. Richtsfeld, T. Mörwald, J. Prankl, M. Zillich, and M. Vincze. Segmentation of unknown objects in indoor environments. In *IROS*, 2012. 1, 6
- [17] C. Rother, V. Kolmogorov, and A. Blake. "GrabCut": interactive foreground extraction using iterated graph cuts. *ACM Trans. Graph.*, 23(3):309–314, Aug. 2004. 1, 2
- [18] N. Silberman, D. Hoiem, P. Kohli, and R.ergus. Indoor segmentation and support inference from rgb-d images. In *ECCV*, 2012. 6

- [19] E. Strelakovsky and D. Cremers. Real-time minimization of the piecewise smooth mumford-shah functional. In *ECCV*, 2014. 1, 2
- [20] A. Ückermann, R. Haschke, and H. Ritter. Real-time 3d segmentation of cluttered scenes for robot grasping. 2012. 1, 2
- [21] S. Vicente, V. Kolmogorov, and C. Rother. Joint optimization of segmentation and appearance models. *ICCV*, 2009. 1, 2
- [22] M. Werlberger, T. Pock, M. Unger, and H. Bischof. A Variational Model for Interactive Shape Prior Segmentation and Real-Time Tracking. In *SSVM*, 2009. 1, 2