STAG: Smart Tools & Apps for Graphics (2015)
Silvia Biasotti and Marco Tarini and Andrea Giachetti (Editors)

Scene segmentation based on NURBS surface fitting metrics

G. Pagnutti and P. Zanuttigh

Department of Information Engineering, University of Padova, Italy

Abstract

This paper proposes a segmentation scheme jointly exploiting color and depth data within a recursive region splitting framework. A set of multi-dimensional vectors is built from color and depth data and the scene is segmented in two parts using normalized cuts spectral clustering. Then a NURBS model is fitted on each of the two parts and various metrics based on the surface fitting results are used to measure the plausibility that each segment represents a single surface or object. Segments that do not represent a single surface are recursively split in a tree-structured procedure until the final segmentation is obtained. Different metrics based on the fitting error and on the curvature of the fitted surfaces are presented and tested inside this framework. Experimental results show how a reliable scene segmentation can be obtained from this procedure.

1. Introduction

Many scene segmentation schemes based on color data have been proposed, based on graph theory [FH04], on various clustering schemes [CM02, SM00], on region splitting and merging, and on many other techniques, but none of them is able to provide a completely satisfactory solution in all situations, due to the many challenging issues associated with this ill-posed problem. The recent introduction of consumer depth cameras (e.g., Microsoft Kinect or Intel RealSense cameras) has led to the development of several approaches based on the joint usage of color and depth data (see Section 2). The 3D spatial information provides a very informative description of the scene that allows to solve many critical situations which color alone could not disambiguate. In particular the joint usage of color and depth data resembles what happens inside the human brain where the disparity between the images seen by the two eyes is one of the clues used to separate the different objects in a scene together with prior knowledge and other features extracted from the color data acquired by our eyes.

This paper proposes a novel scene segmentation scheme that extends the approach of [PZ14]. The input data is firstly represented as a set of 6-dimensional vectors containing both color and geometry information for each sample. Then the scene is segmented in two parts using normalized cuts spectral clustering. A NURBS parametric surface is fitted on each of the two parts and various metrics based on the surface fitting error and on the curvature of the fitted surfaces are used to measure the plausibility that each segment represents a single surface or object. This idea is exploited inside an iterative scheme where the segmentation is progressively refined by recursively splitting the segments that do not represent a single surface in the 3D space. In the experimental results the impact of the usage of different surface fitting metrics has been evaluated and compared with state-of-the-art segmentation methods.

Compared to [PZ14] there are two main contributions. The first is the evaluation of different criteria in order to decide which segments need to be iteratively split. While in [PZ14] only the MSE of the surface fitting was considered, in this work several different metrics have been used. They belong to two main families, one considering metrics evaluating the accuracy of the fitting and the other instead based on the analysis of the curvature of the fitted surfaces. The second improvement is the use of a more refined surface fitting algorithm, in particular the new algorithm adaptively changes the number of control points, thus avoiding the bias towards smaller segments of the previous approach.

The paper is organized in the following way: after reviewing the related work in Section 2, Section 3 presents the work-flow of the segmentation algorithm. More in detail, subsection 3.1 briefly recalls the employed joint color and depth segmentation scheme, while subsection 3.2 presents the employed surface fitting algorithm and subsection 3.3 shows how these elements are combined into the proposed approach. The different metrics used to evaluate the surface fitting are presented in Section 4. The results are presented in Section 5 and Section 6 draws the conclusions.

2. Related work

Even if it is a recent research field, various works addressing scene segmentation by means of color and depth information have been published. A first possible solution is to perform two independent segmentations from the color image and the depth data, and then join the two results [CM09]. In [WZYZ10] two likelihood functions, based on color and depth data, are combined together in order to segment the background from the foreground. Two different approaches for the segmentation of binocular stereo video sequences are presented in [KCB*05]: one, based on Layered Dynamic Programming and the other based on Layered Graph Cuts. Some recent works try to jointly solve the segmentation and stereo disparity estimation problems, e.g., [LSR*10] and [BRK*11]. Clustering techniques can be exploited for joint depth and color segmentation as in [BW09] and [WFFD11]. In [DMZC12] a segmentation scheme based on spectral clustering that is able to automatically balance the relevance of the two clues is presented. The approach of [PZ14] exploits spectral clustering inside a recursive approach where a surface fitting scheme is used to recognize if each segment needs to be further split. In [EPD12] superpixels produced by an over-segmentation of the scene are combined together in segments corresponding to planar surfaces using an approach based on Rao Blackwellized Monte Carlo Markov Chain. An extension to the segmentation of multiple depth maps has been also proposed [SD14]. The approach of [GAGM14] exploits instead hierarchical segmentation based on the output of contour extraction. A combined approach for segmentation and object recognition has been presented in [NSF12], that also exploits a hierarchical scheme starting from an initial over-segmentation. Finally the approach of [RBF12] exploits a MRF superpixel segmentation and a tree-structured scheme.

3. Proposed segmentation scheme

The proposed approach exploits a pre-processing step followed by a tree-structured iterative segmentation algorithm. In the first step a set of multi-dimensional vectors (one for each pixel) is built from the color image and the depth map. Normalized cuts spectral clustering is then used to recursively segment the scene into two parts. At each iteration a NURBS model is fitted over each of the two segments and a set of different metrics related to the fitting operation is computed. These metrics are compared to the results obtained in the previous steps for the same cluster (except for the first iteration). If the segmentation has allowed to obtain a better fitting according to the selected criterion (the criteria are described in Section 4), the process is iterated by recursively splitting the two segments, otherwise it is stopped on this branch. The procedure continues until it is not possible to further subdivide any of the produced segments. A schematic representation of the approach is shown in Figure 1.

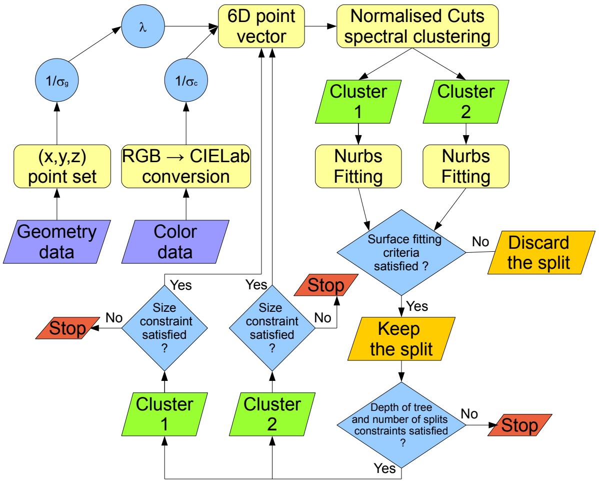

Figure 1: *Overview of the proposed approach.*

3.1. Joint segmentation of color and depth data

This stage exploits an approach derived from [DMZC12] and [DZC11] that firstly represents the scene as a set of 6-dimensional vectors built from the geometry and color data and then apply spectral clustering. The depth and color cameras are firstly jointly calibrated. With the calibration data it is possible to compute the 3D coordinates x,y and z of each 3D scene point (i.e., each pixel in the depth map) and to associate to it a vector representing its color in the R, G, B color space. In order to properly segment this representation geometry and color need to be represented in a consistent way. To this purpose color values are converted to the CIELab perceptually uniform space in order to give a perceptual significance to the distance between colors that will be used in the clustering algorithm. The color of each sample p_i is thus represented by the vector

$$\mathbf{p_i^c} = [L(p_i), a(p_i), b(p_i)]^T \quad , i = 1, ..., N \tag{1}$$

Geometry is simply represented by the 3D spatial position $x(p_i)$, $y(p_i)$, and $z(p_i)$, i.e., as:

$$\mathbf{p_i^g} = [x(p_i), y(p_i), z(p_i)] \quad , i = 1, ..., N \tag{2}$$

The segmentation procedure should allow to compare geometry and color distances independently on the size of the scene or on the light conditions. For this reason the x,y and z components are normalized by the average σ_g of the standard deviations of the point coordinates obtaining the vectors $[\bar{x}(p_i), \bar{y}(p_i), \bar{z}(p_i)]$. Following the same rationale, the color information vectors $[\bar{L}(p_i), \bar{a}(p_i), \bar{b}(p_i)]$ are obtained by normalizing color data with the average σ_c of the standard deviations of the L, a and b components. Using the normalized geometry and color information, each point is finally represented as:

$$\mathbf{p}_i^f = \begin{bmatrix} \bar{L}(p_i) \\ \bar{a}(p_i) \\ \bar{b}(p_i) \\ \lambda\bar{x}(p_i) \\ \lambda\bar{y}(p_i) \\ \lambda\bar{z}(p_i) \end{bmatrix}, i = 1,...,N \tag{3}$$

the parameter λ controls the relative contribution of color and geometry (i.e., the higher λ is, the more relevant is geometry information). An approach for the automatic tuning of this parameter in a similar framework has been presented in [DMZC12].

Various clustering techniques can be used to segment the 6D vectors built in this way. For this work normalized cuts spectral clustering [SM00] has been used since it provides very good performances. The main drawback of this algorithm is that it is very computationally expensive. For this reason we used the method based on the integral eigenvalue problem proposed in [FBCM04]. This approach allows to compute a first solution on a randomly subsampled subset of the points and then propagate the solution to the whole points set by a specific technique called Nyström method. A very good approximation of the initial solution with limited computation and memory resources can be obtained with this method. Finally, in order to avoid the creation of very small regions in proximity of edges and other artifacts due to noise a final refinement stage removing regions smaller than a pre-defined threshold T_p is applied.

3.2. Surface fitting on the segmented data

NURBS (Non-Uniform Rational B-Splines) are piecewise rational polynomial functions expressed in terms of proper bases (a complete overview of this topic can be found in [PT97]). They allow to represent freeform parametric curves and surfaces in a concise way, by means of control points. A parametric NURBS surface is defined as

$$\mathbf{S}(u,v) = \frac{\sum_{i=0}^{n}\sum_{j=0}^{m} N_{i,p}(u)N_{j,q}(v)w_{i,j}\mathbf{P}_{i,j}}{\sum_{i=0}^{n}\sum_{j=0}^{m} N_{i,p}(u)N_{j,q}(v)w_{i,j}} \tag{4}$$

where the $\mathbf{P}_{i,j}$ are the control points, the $w_{i,j}$ are the corresponding weights, the $N_{i,p}$ are the univariate B-spline basis functions, and p,q are the degrees in the u,v parametric directions respectively.

In our experiments, we set the degrees in the u and v directions equal to three. We set the weights all equal to one, thus our fitted surfaces are non-rational (i.e., spline). The points to fit are a subset of the rectangular grid given by the sensor pixel arrangement, and we exploit this by setting the corresponding (u_k, v_l) surface parameter values as 2D locations on the image plane of the camera. Since the number of surface control points gives the degrees of freedom in our model, we set it adaptively depending on the number of input samples. To achieve this, we consider the horizontal and vertical extents of the segment to fit. We set 20 as maximum number of control points to use in a parametric direction for a segment extending over the whole image, while for smaller ones we determine the number proportionally to the segment extents. This choice of parameters provides enough degrees of freedom to represent the shape of any common object, and the adaptive scheme at the same time prevents the fitting to always be more accurate for smaller segments, independently on how the segmentation algorithm was successful in detecting the objects in the scene. Notice that this adaptivity is an improvement over the fitting scheme used in [PZ14], since in the previous work a fixed number of control points was always used, favoring small segments and then possibly leading to over-segmentation, as underlined in the experimental results section.

Once determined the (u_k, v_l) parameter values corresponding to the points to fit, the surface degrees and the number of control points in the u, v parametric directions, we consequently obtain the NURBS knots (needed for the definition of the $N_{i,p}$ basis functions) as in Chapter 9 of [PT97]. Finally, by considering Eq. 4 evaluated at (u_k, v_l) and equated to the points to fit, we obtain an over-determined system of linear equations. We solve it in the least-squares sense by means of singular value decomposition (SVD), thus obtaining the surface control points.

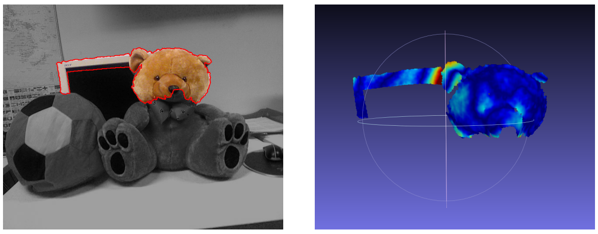

Figure 2: *A 3D NURBS surface fitted over two clusters originated by segmentation of one of the test scenes. The red areas correspond to larger MSE values. Notice how the large fit error between the teddy head and the monitor portion reveals that the two segments do not actually belong to the same object. (Best viewed in color)*

3.3. Iterative tree structured fitting and segmentation

After presenting the main building blocks the iterative segmentation procedure is now presented. The recursive tree-structured approach of [PZ14] has been extended by using the surface fitting metrics and criteria presented in the next section, and improved by adaptively setting the number of control points of the fitted surfaces dependently on the extents of the segments to fit. The 6-dimensional representation of Eq. 3 is used as input. The complete 6D point cloud P is firstly segmented into two parts P_0 and P_1 using normalized cuts spectral clustering as described in subsection 3.1. A NURBS surface is then fitted on each of the two segments, obtaining the two surfaces S_0 and S_1. At this point the values of the various metrics presented in Section 4 are also computed for the two segments S_0 and S_1. In order to proceed to the next step a set of conditions must be satisfied, i.e.,:

- The size of P_0 and P_1 must be bigger than $2T_p$. If one of the two segments does not satisfy the constraint it is kept as part of the final solution and it is not split anymore. This is consistent with the choice of not allowing segments smaller than T_p made in subsection 3.1 since the split would produce at least one segment smaller than T_p.
- A maximum number T_d of recursive splits on each segment is set, when the segmentation tree of Fig. 3 reaches the maximum allowed depth the procedure is stopped on the corresponding branch.
- A maximum number of splits (i.e., segments) T_s is also set. Again when it is reached the procedure is stopped.

However at this first iteration the stop conditions are very unlikely to be reached and the procedure continues recursively by splitting the two point clouds P_0 and P_1 into two parts obtaining the sets P_{00}, P_{01} and P_{10}, P_{11} respectively. The various metrics are also computed on this newly obtained segments. Note that the point clouds are sorted on the basis of the selected fitting accuracy metric (i.e., if the MSE is being used at each step the point cloud with the maximum MSE is processed). In order to describe the general case let us assume that the segment P_i is considered for splitting (e.g., $i = 0$ or $i = 1$ at the first iteration): the segment is split into two sub-segments P_{i0} and P_{i1} as before, the two NURBS approximations S_{i0} and S_{i1} and the various fitting metric values v_{i0} and v_{i1} are computed. At this point the weighted average of the considered metric on the two sub-segments is compared with the one of the original segment P_i:

$$\frac{v_{i0}|P_{i0}| + v_{i1}|P_{i1}|}{v_i|P_i|} < T \tag{5}$$

where v_i can be v_i^{MSE}, v_i^{MAE} or any other of the measures presented in Section 4 depending on the selected fitting metric. The weights are the cardinalities of the two sets, while the impact of the setting of T is discussed in the experimental results section. If the constraint of Eq. 5 is satisfied it means that the segmentation has improved the accuracy of the scene representation by recognizing the different surfaces (i.e., objects) in the scene. In this case it must be kept and the two sub-segments P_{i0} and P_{i1} are further subdivided with the same procedure. If the constraint is not satisfied the segmentation is discarded, the segment P_i is kept as a single object and no further processing is done on this branch of the tree. Before proceeding the previously introduced conditions are also checked on each segment before splitting, i.e.:

$$(|P_i| > 2T_p) \wedge (Depth(T_i) < T_d) \wedge (count(i) < T_s) \tag{6}$$

where $Depth(T_i)$ is the depth of the $i-th$ node and $count(i)$ is the number of splits made until the current iteration. The same procedure is applied recursively to all the sub-segments generated during the computation until the condition of Eq. 5 is satisfied and none of the stopping conditions is violated leading to a tree structure similar to the one of Figure 3.

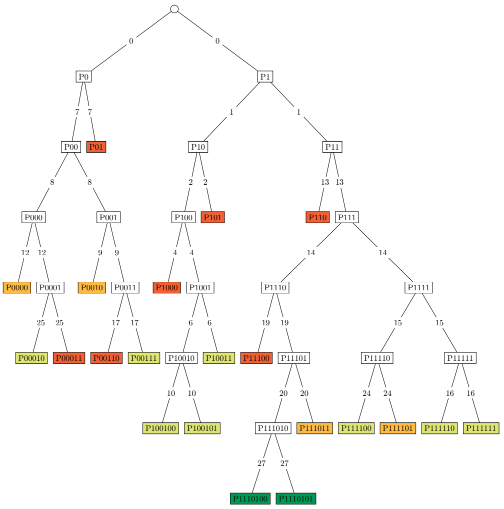

Figure 3: *Tree structure for the segmentation of a sample scene. The colored nodes correspond to the final segments. Red segments: further segmentation was attempted but rejected since not satisfying the surface fitting criteria (this example refers to the MSE metric). Orange: the segmentation was rejected since one of the resulting sub-segments would be smaller than T_p. Light green: not split since smaller than $2T_p$. Green: stopped since the maximum tree depth T_d was reached. (Best viewed in color)*

4. Surface fitting metrics

The key idea exploited in this work is that if the segment correctly represents a single object or surface in the scene it should be accurately fitted by a smooth surface computed with the approach of subsection 3.2, while if the segment contains multiple objects (and so should be recursively split) at different depths or edges between different surfaces these issues will affect the fitting. Two main insights can be exploited for this purpose. The first is to analyze the difference between the position of the samples on the fitted NURBS surface and on the original data. For this work 4 different metrics based on the accuracy of the fitting have been considered, i.e., the Mean Squared Error (MSE), the Mean Absolute Error (MAE), the Variance of the Error (VE) and the Number of points with a Large Error (NLE). The second idea consists in analyzing the curvature of the fitted surface instead, since large curvature values or variations should correspond to edges or jumps in the depth values. The considered metrics are the Variance of the MaX Curvature absolute value (VMXC), the Variance of the MEan Curvature (VMEC), the Variance of the Gaussian Curvature (VGC), the Mean of the MaX Curvature absolute value (MMXC) and the Number of points with a Large Curvature (NLC).

4.1. Mean Squared Error (MSE)

This metric is the Mean Squared Error (MSE) between the depth samples in segment P_i and the points obtained by sampling the NURBS approximation S_i at the locations corresponding to the points in P_i, i.e,:

$$v_i^{MSE} = \frac{\sum_{\mathbf{P_j} \in P_i} \left(\mathbf{P_j} - \mathbf{S_j}\right)^2}{|P_i|} \quad (7)$$

where $\mathbf{S_j}$ is the value of the NURBS approximation S_i at the location corresponding to $\mathbf{P_j}$. Eq. 7 measures the accuracy of the surface fitting and the resulting value will be denoted with v_i^{MSE}. If this metric is selected the idea is that properly segmented regions should have a low MSE since they contain a single surface that can be accurately fitted, while segments containing multiple surfaces at different depths can not be accurately fitted and can lead to higher MSE values. An example of this can be seen in Figure 2. For this reason the accuracy of the fitting before and after the split of a segment P_i into two parts P_{i0} and P_{i1} is compared and if the ratio is below a threshold T the segmentation is accepted, i.e.:

$$\frac{v_{i0}^{MSE}|P_{i0}| + v_{i1}^{MSE}|P_{i1}|}{v_i^{MSE}|P_i|} < T \quad (8)$$

where the weights are the cardinalities of the two sets. In our results we have tested for T three different values, 0.8, 0.9 and 1. Notice that with the latter the segmentation is accepted if there is any improvement in the accuracy, independently on how large the improvement is.

4.2. Mean Absolute Error (MAE)

This metric is the Mean Absolute Error (MAE) between the depth samples in segment P_i and the points obtained by sampling the NURBS approximation S_i at the locations corresponding to the points in P_i. It works exactly as the MSE, except for the fact that the absolute values are used instead of the squared values, i.e,:

$$v_i^{MAE} = \frac{\sum_{\mathbf{P_j} \in P_i} |\mathbf{P_j} - \mathbf{S_j}|}{|P_i|} \quad (9)$$

Notice that MSE tends to give more importance to large errors due to the square operation, while this metric gives a more uniform weight to the fitting errors. The criterion to evaluate the segmentation works in the same way of the MSE one, i.e.,:

$$\frac{v_{i0}^{MAE}|P_{i0}| + v_{i1}^{MAE}|P_{i1}|}{v_i^{MAE}|P_i|} < T \quad (10)$$

the threshold T has been set to 0.8 , 0.9 and 1 as in the previous case.

4.3. Variance of the Error (VE)

This metric measures the Variance of the Error (VE) between the depth samples in segment P_i and the points obtained by sampling the NURBS approximation S_i at the locations corresponding to the points in P_i, i.e.:

$$v_i^{VE} = \frac{\sum_{\mathbf{P_j} \in P_i} \left(|\mathbf{P_j} - \mathbf{S_j}| - v_i^{MAE}\right)^2}{|P_i|} \quad (11)$$

This metric instead of considering the absolute error value considers its deviation from the mean, accounting for the idea that the presence of discontinuities should produce large errors in some restricted areas in contrast with large areas with small errors. The criterion is the ratio between the variances before and after the split as for the other metrics:

$$\frac{v_{i0}^{VE}|P_{i0}| + v_{i1}^{VE}|P_{i1}|}{v_i^{VE}|P_i|} < T \quad (12)$$

where T is as above.

4.4. Number of points with a Large Error (NLE)

Following the same rationale, i.e., the presence of some localized regions with large fitting errors, it is possible to measure the number of points with an associate absolute error greater than a threshold T_l. The idea behind this metric is that the samples corresponding to an edge or close to the jumps between two objects in the same segment should have an error larger than the threshold, while the others should have an associated error below the threshold. A drawback of this metric is that there is an additional threshold and its setting is quite critical to obtain optimal results. The value of T_l depends on the amount of noise on the depth camera data, for the results we set it to 0.1. After a proper splitting of a segment the sum of the number of samples with a large error on the two parts $v_{i0}^{NLE} + v_{i1}^{NLE}$ must be smaller than the number of points with a large error on the original segment v_i^{NLE}, i.e.:

$$\frac{v_{i0}^{NLE} + v_{i1}^{NLE}}{v_i^{NLE}} < T \quad (13)$$

where T is as above.

4.5. Variance of the MaX Curvature absolute value (VMXC)

For this metric, we consider the two principal curvatures κ_1 and κ_2 of the NURBS approximating surface at the locations corresponding to the points in P_i. They are well defined since our fitted surfaces are bivariate cubic splines with single knots, thus C^2 everywhere. We expect that for segments containing multiple objects, the fitted surface would show high oscillations caused by the depth jumps. Therefore, we take the maximum of the absolute values of the principal curvatures, $\kappa^{max} = \max(|\kappa_1|, |\kappa_2|)$, and we consider as error metric its variance over the points of the segment P_i, i.e.,

$$v_i^{VMXC} = \frac{\sum_{\mathbf{P_j} \in P_i} \left(\kappa_j^{max} - v_i^{MMXC}\right)^2}{|P_i|} \tag{14}$$

where

$$v_i^{MMXC} = \frac{\sum_{\mathbf{P_j} \in P_i} \kappa_j^{max}}{|P_i|} \tag{15}$$

is the mean of κ^{max} over P_i (notice that we denote by κ_j^{max} the value of κ^{max} for the NURBS approximation S_i at the location corresponding to $\mathbf{P_j}$). The criterion to accept the segmentation of P_i into segments P_{i0} and P_{i1} is then

$$\frac{v_{i0}^{VMXC}|P_{i0}| + v_{i1}^{VMXC}|P_{i1}|}{v_i^{VMXC}|P_i|} < T \tag{16}$$

where T is as above.

4.6. Variance of the MEan Curvature (VMEC)

Similarly as above, we consider as error metric the variance v_i^{VMEC} of the mean curvature, $H = \frac{1}{2}(\kappa_1 + \kappa_2)$. That is, considering $v_i^{MMEC} = \frac{\sum_{\mathbf{P_j} \in P_i} H_j}{|P_i|}$ which is the mean of H over P_i, it is

$$v_i^{VMEC} = \frac{\sum_{\mathbf{P_j} \in P_i} \left(H_j - v_i^{MMEC}\right)^2}{|P_i|} \tag{17}$$

where H_j is the value of H for the NURBS approximation S_i at the location corresponding to $\mathbf{P_j}$. This is a variation of metric VMXC, based on the idea that the maximum curvature absolute value could be very high because of just one of the two principal curvatures, while considering the mean curvature could be more adequate for some shapes. Moreover, by taking the absolute values it is possible that information about large variations between positive and negative curvature values gets lost, while it is taken into account if the mean curvature is used instead. The criterion for the segmentation is then

$$\frac{v_{i0}^{VMEC}|P_{i0}| + v_{i1}^{VMEC}|P_{i1}|}{v_i^{VMEC}|P_i|} < T \tag{18}$$

where T is as above.

4.7. Variance of the Gaussian Curvature (VGC)

We obtain another variation of metric VMXC by considering the variance of the Gaussian curvature, $K = \kappa_1\kappa_2$. That is, the metric is

$$v_i^{VGC} = \frac{\sum_{\mathbf{P_j} \in P_i} \left(K_j - v_i^{MGC}\right)^2}{|P_i|} \tag{19}$$

where $v_i^{MGC} = \frac{\sum_{\mathbf{P_j} \in P_i} K_j}{|P_i|}$ is the mean of K over P_i (and K_j is the value of K for the NURBS approximation S_i at the location corresponding to $\mathbf{P_j}$). Notice that the previous considerations about using the mean curvature instead of the maximum curvature absolute value hold for the Gaussian curvature too. The corresponding criterion for the segmentation is

$$\frac{v_{i0}^{VGC}|P_{i0}| + v_{i1}^{VGC}|P_{i1}|}{v_i^{VGC}|P_i|} < T \tag{20}$$

where T is as above.

4.8. Mean of the MaX Curvature absolute value (MMXC)

In addition to the curvature variance as an indicator of surface oscillations, we consider large values of the curvatures themselves as an index of poor segmentation, since they are expected to correspond to sharp edges, or gaps between separate objects. Following this rationale, we consider v_i^{MMXC} defined in Eq. 15, i.e., the mean of κ^{max} maximum of the absolute values of the principal curvatures over the points in P_i, as a further error metric. The corresponding criterion for the segmentation is

$$\frac{v_{i0}^{MMXC}|P_{i0}| + v_{i1}^{MMXC}|P_{i1}|}{v_i^{MMXC}|P_i|} < T \tag{21}$$

where T is as above.

4.9. Number of points with a Large Curvature (NLC)

To investigate the presence of local regions with large curvature values, we also take into account the number of points for which the maximum of the absolute values of the principal curvatures κ^{max} is greater than a threshold T_c. This is roughly equivalent to counting the points that correspond to edges or gaps between the objects. Clearly it is not straightforward how to set the threshold T_c, since an optimal value would depend both on the noise in the data and on the shape of the objects. For our results we set it to 10, corresponding to a radius of curvature of 0.1. Then, the criterion we obtain for the segmentation is

$$\frac{v_{i0}^{NLC} + v_{i1}^{NLC}}{v_i^{NLC}} < T \tag{22}$$

where v_i^{NLC} is the number of points in the segment P_i for which κ^{max} is greater than T_c, and T is as for the previous metrics.

5. Experimental results

In order to evaluate the performances of the proposed approach and the impact of the different surface estimation metrics on the segmentation algorithm

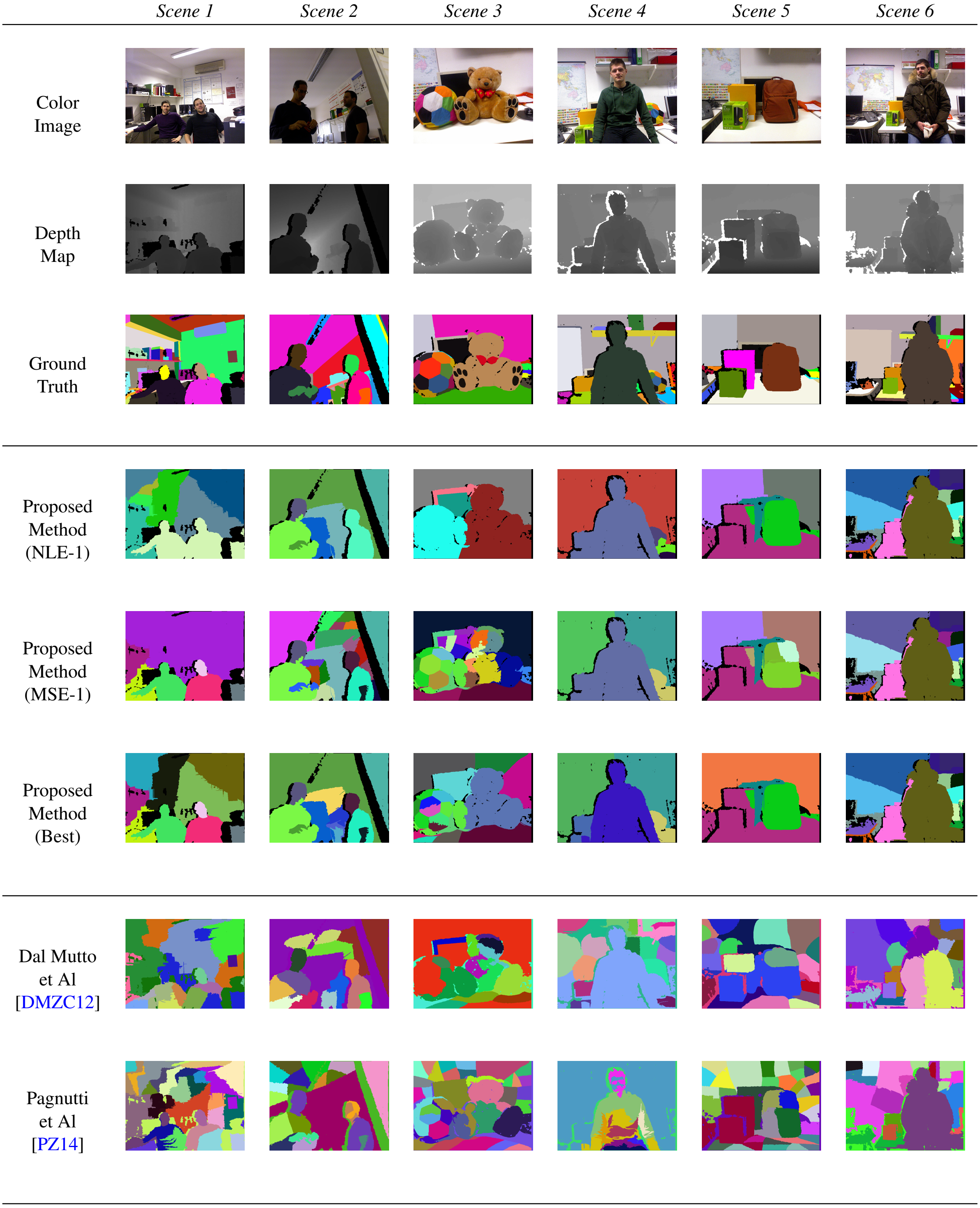

Figure 4: *Segmentation of some sample scenes. For each scene the color image, depth map and ground truth segmentation are shown. Then 3 different results for the proposed approach are presented, corresponding to VE (with $T = 0,9$), to NLE (with $T = 1$) and to the choice of the best measure for each scene. The last two rows show the results of two competing approaches ([DMZC12] and [PZ14]).*

we used the dataset associated to [PZ14] and available at the address http://lttm.dei.unipd.it/downloads/segmentation/. This dataset contains 6 different scenes acquired with either the Kinect or the Asus Xtion consumer depth cameras. The dataset contains the color views, the depth data and the calibration information (color and depth data are shown in the first two columns of Figure 4). Calibration data allow to align the color and depth data and to build the representation of subsection 3.1 that is used as input for the proposed algorithm. In order to perform an accurate evaluation of the segmentation a manually segmented ground truth has been added to the dataset for the purposes of this work. Ground truth data has been compared with the segmentation results (the ground truth is shown in the third row of Figure 4) in order to obtain the numerical results. Notice that the points without a valid depth value (i.e., the ones that the depth camera has not been able to acquire) have been set as invalid points also in the ground truth in order to avoid considering them in the comparisons of the segmentations.

The proposed method has been applied to the scenes in the dataset using the various metrics and criteria presented in Section 4 in order to decide which segments need to be split. The results have been evaluated by comparing the ground truth with the obtained segmentations using 3 different error metrics, i.e., Rand Index (RI), Ground Truth Region Covering (GTRC) and Variation of Information (VoI). For the detailed description of these error metrics please refer to [AMFM11], however notice as an higher value correspond to better results for the first two metrics, while lower values are better in the last one.

Table 1 shows the accuracy of the obtained segmentations according to the RI metric. A first basic result is that on average error-based metrics produce better results than curvature-based approaches. However this is not true for all the metrics and all the scenes. Also notice how no single metric is better on all the considered scenes, on different scenes different approaches can lead to better results. However, according to this metric, the number of points with a large error (NLE) metric on average seems to be the best error metric. It is the best one only on scene 6 but it has been able to provide good results on all the considered scenes. Also the MSE and MAE metrics have provided good results, as expected they behave similarly and have good but not exceptional performances on all the considered scenes. Another good metric is the Error Variance (VE), this metric has a more erratic behavior, being the best one on 2 scenes, but leading to not completely satisfactory results in others. Curvature-based metrics on average have slightly lower performances, among them the best one is the NLC (number of points with a large curvature) metric, that is the best on scene 4 and has average performances similar to the MSE and MAE. The other curvature metrics have lower scores according to this metric, but not too far from the error-based ones. Finally notice that according to the RI measure setting $T = 1$, i.e., simply ensuring that the metric values improve even by a very small amount, is the best option for all metrics (except VMEC).

Metric	T	Scene						Mean
		1	2	3	4	5	6	
MSE	**1**	0.70	0.90	0.89	0.84	0.87	0.91	0.85
	0.9	0.67	0.90	0.60	0.78	0.83	0.90	0.78
	0.8	0.67	0.84	0.60	0.75	0.84	0.86	0.76
MAE	**1**	0.67	0.90	**0.89**	0.84	0.88	0.92	0.85
	0.9	0.67	0.84	0.60	0.77	0.82	0.85	0.76
	0.8	0.67	0.79	0.60	0.77	0.82	0.81	0.75
VE	**1**	**0.90**	0.91	0.60	0.84	**0.89**	0.85	0.83
	0.9	0.70	**0.91**	0.60	0.75	0.83	0.90	0.78
	0.8	0.67	0.90	0.60	0.75	0.84	0.86	0.77
NLE	**1**	0.87	0.90	0.82	0.78	0.87	**0.92**	**0.86**
	0.9	0.67	0.89	0.60	0.78	0.84	0.91	0.78
	0.8	0.67	0.89	0.60	0.78	0.84	0.81	0.77
VMXC	**1**	0.79	0.79	0.86	0.86	0.72	0.86	0.81
	0.9	0.79	0.79	0.87	0.86	0.70	0.86	0.81
	0.8	0.79	0.79	0.84	0.86	0.70	0.85	0.81
VMEC	1	0.79	0.79	0.89	0.87	0.72	0.87	0.82
	0.9	0.79	0.79	0.88	0.86	0.74	0.87	0.82
	0.8	0.63	0.79	0.88	0.86	0.74	0.87	0.79
VGC	**1**	0.79	0.85	0.85	0.87	0.70	0.85	0.82
	0.9	0.79	0.85	0.85	0.87	0.70	0.85	0.82
	0.8	0.79	0.85	0.85	0.86	0.70	0.85	0.82
MMXC	**1**	0.79	0.79	0.89	0.87	0.72	0.81	0.81
	0.9	0.63	0.79	0.60	0.78	0.73	0.82	0.72
	0.8	0.63	0.79	0.60	0.78	0.70	0.81	0.72
NLC	**1**	0.86	0.79	0.87	**0.87**	0.88	0.81	0.85
	0.9	0.70	0.79	0.86	0.87	0.75	0.82	0.80
	0.8	0.70	0.79	0.60	0.78	0.70	0.70	0.71
[DMZC12]		0.87	0.86	0.74	0.81	0.86	0.88	0.84
[PZ14]		**0.91**	0.83	0.86	0.63	0.86	0.90	0.83

Table 1: *Segmentation evaluation according to the RI metric (higher is better). The table shows the segmentation performances for the various measures considering 3 different threshold parameter setting for each measure. The best setting for the threshold in each measure and the best results on each scene and on average are underlined in bold.*

The results according to the GTRC metric (Table 2) are similar, even if there are some small differences. Also in this case error-based metrics provide better results and the performance gap between the two families of approaches is even larger according to this metric. According to GTRC the best error measure is the Mean Squared Error (MSE), in particular with the threshold parameter set to $T = 1$. The other error-based metrics (MSE, MAE and NLE) have all good performances, in particular NLE. Also according to this metric the best curvature-based option is NLC, while the other curvature-based approaches have lower results. For

Metric	T	Scene						Mean
		1	2	3	4	5	6	
MSE	**1**	0.42	0.62	0.51	0.53	0.55	0.61	**0.54**
	0.9	00.29	0.67	0.35	0.53	0.48	0.60	0.49
	0.8	0.29	0.62	0.35	0.49	0.52	0.54	0.47
MAE	**1**	0.29	0.63	0.53	0.53	0.45	0.61	0.51
	0.9	0.29	0.62	0.34	0.53	0.47	0.54	0.47
	0.8	0.29	0.48	0.35	0.53	0.49	0.48	0.44
VE	1	**0.49**	0.62	0.35	0.53	**0.60**	0.41	0.50
	0.9	0.42	**0.68**	0.35	0.49	0.48	0.60	0.50
	0.8	0.29	0.67	0.35	0.49	0.52	0.54	0.48
NLE	**1**	0.36	0.62	0.46	0.53	0.60	**0.62**	0.53
	0.9	0.29	0.62	0.35	0.53	0.52	0.61	0.49
	0.8	0.29	0.62	0.35	0.53	0.52	0.48	0.46
VMXC	1	0.34	0.48	0.41	**0.58**	0.32	0.45	0.43
	0.9	0.34	0.48	0.39	**0.58**	0.39	0.45	0.44
	0.8	0.34	0.48	0.39	**0.58**	0.39	0.53	0.45
VMEC	1	0.34	0.48	0.52	0.54	0.33	0.45	0.44
	0.9	0.34	0.48	0.50	0.54	0.41	0.45	0.45
	0.8	0.26	0.48	0.50	0.58	0.41	0.51	0.46
VGC	1	0.34	0.52	0.42	0.55	0.39	0.44	0.44
	0.9	0.34	0.52	0.42	0.55	0.39	0.44	0.44
	0.8	0.34	0.52	0.42	0.54	0.39	0.46	0.45
MMXC	**1**	0.34	0.48	**0.54**	0.55	0.33	0.36	0.43
	0.9	0.26	0.48	0.34	0.53	0.36	0.41	0.40
	0.8	0.26	0.48	0.34	0.53	0.39	0.49	0.41
NLC	**1**	0.47	0.49	0.48	0.56	0.49	0.37	0.48
	0.9	0.39	0.49	0.51	0.55	0.44	0.41	0.47
	0.8	0.39	0.49	0.34	0.53	0.39	0.40	0.43
[DMZC12]		0.34	0.52	0.28	0.46	0.35	0.48	0.40
[PZ14]		0.36	0.42	0.32	0.34	0.36	0.54	0.39

Table 2: *Segmentation evaluation according to the GTRC metric (higher is better). The table shows the segmentation performances for the various measures considering 3 different threshold parameter setting for each measure. The best setting for the threshold in each measure and the best results on each scene and on average are underlined in bold.*

Metric	T	Scene						Mean
		1	2	3	4	5	6	
MSE	**1**	2.42	1.79	1.99	1.73	1.68	1.77	**1.90**
	0.9	2.79	1.39	2.58	1.60	1.67	1.79	1.97
	0.8	2.79	1.33	2.58	1.80	1.53	1.98	2.00
MAE	1	2.79	1.66	**1.88**	1.72	2.69	1.68	2.07
	0.9	2.79	**1.30**	2.68	1.61	1.76	1.82	1.99
	0.8	2.79	1.72	2.58	1.61	1.58	2.02	2.05
VE	1	2.39	1.79	2.58	1.73	1.61	2.64	2.12
	0.9	2.42	1.40	2.58	1.80	1.65	1.78	1.94
	0.8	2.79	1.39	2.58	1.80	1.53	1.98	2.01
NLE	**1**	2.84	1.55	2.23	1.60	1.54	**1.67**	1.91
	0.9	2.79	1.51	2.58	1.60	**1.53**	1.75	1.96
	0.8	2.79	1.51	2.58	1.60	**1.53**	2.02	2.00
VMXC	1	2.73	1.72	2.60	**1.59**	2.49	2.49	2.27
	0.9	2.73	1.72	2.52	**1.59**	1.91	2.46	2.16
	0.8	2.73	1.72	2.58	**1.59**	1.91	2.03	2.09
VMEC	1	2.73	1.72	2.13	1.78	2.38	2.43	2.19
	0.9	2.73	1.72	2.18	1.71	2.12	2.43	2.15
	0.8	3.03	1.72	2.10	1.59	2.12	2.14	2.12
VGC	1	2.73	1.85	2.53	1.79	1.92	2.56	2.23
	0.9	2.73	1.85	2.53	1.79	1.92	2.57	2.23
	0.8	2.73	1.85	2.53	1.71	1.92	2.37	2.19
MMXC	**1**	2.73	1.72	2.06	1.78	2.38	2.26	2.15
	0.9	3.03	1.72	2.68	1.63	2.29	2.20	2.26
	0.8	3.03	1.72	2.68	1.63	1.92	1.98	2.16
NLC	**1**	**2.19**	1.67	2.20	1.75	2.04	2.35	2.03
	0.9	2.49	1.67	2.19	1.78	2.04	2.20	2.06
	0.8	2.49	1.67	2.68	1.63	1.91	2.37	2.13
[DMZC12]		3.06	2.02	3.08	2.67	2.43	2.11	2.56
[PZ14]		3.03	2.40	3.06	2.73	2.77	2.14	2.69

Table 3: *Segmentation evaluation according to the VoI metric (lower is better). The table shows the segmentation performances for the various measures considering 3 different threshold parameter setting for each measure. The best setting for the threshold in each measure and the best results on each scene and on average are underlined in bold.*

error-based approaches $T = 1$ still seems a very good option (except for VE), while for some curvature-based measures the best results correspond to $T = 0.8$ (VMEC, VMXC and VGC).

Finally Table 3 shows the results according to the VoI metric. Results are very similar to GTRC: MSE with $T = 1$ is the best option, even if NLE gets very close to the MSE score. MAE and VE also achieve good results. Again curvature-based approaches have low performances, with the exception of NLC, that is the only curvature based approach able to get results close to the ones of error based approaches. Probably the biggest difference in the results of this metric is the impact of the threshold parameters, according to this metric different values of the parameter are optimal for different measures.

The results have also been compared with the ones of two recent competing approaches, i.e., the methods of [DMZC12] and [PZ14]. The proposed approach outperforms both compared approaches in almost all situations and the average errors are better according to all the considered metrics. Only in Scene 1 according to RI, [PZ14] is able to outperform the proposed method.

Some visual results are shown in Figure 4. The figure shows the results for the NLE measure with $T = 1$ (that is the best option according to RI) and for the MSE metric with $T = 1$ (the best solution according to the other two metrics). Both options provide very good results on the considered scenes since the different objects get almost always recognized. At the same time the scenes are not over-segmented and the objects and background are not split in several pieces. By comparing the obtained results with the ones of [DMZC12] and [PZ14], shown in the last two rows, it is clear how the proposed approach is able to obtain better results and avoids the over-segmentation issues of the two competing approaches. The comparison of the MSE metric results with [PZ14], that exploits this measure, allows also to notice the improvement due to the adaptive surface fitting scheme.

6. Conclusions

In this paper a segmentation scheme jointly exploiting color and depth information has been presented and evaluated. The proposed approach is based on a recursive tree-structured region splitting method that exploits a surface fitting scheme to determine if the segmentation has correctly divided the 3D surfaces present in each segment. Different measures based on the fitting error and on the curvature of the fitted surfaces have been considered in order to evaluate if a segment needs to be further split and their impact on the final results has been evaluated. Experimental results demonstrate that error based metrics, in particular the mean squared error and the number of samples with a large error, have on average better performances. The results also confirm the effectiveness of the proposed approach. It has been able to properly divide the main objects in the scene and at the same time to avoid over-segmentation issues thus outperforming the compared methods. Further research will be devoted to the exploitation of surface orientation information and to the inclusion of region merging approaches into the considered framework.

References

[AMFM11] Arbelaez P., Maire M., Fowlkes C., Malik J.: Contour detection and hierarchical image segmentation. *Pattern Analysis and Machine Intelligence, IEEE Transactions on 33*, 5 (May 2011), 898–916. 8

[BRK*11] Bleyer M., Rother C., Kohli P., Scharstein D., Sinha S.: Object stereo- joint stereo matching and object segmentation. In *IEEE conf. on Computer Vision and Pattern Recognition (CVPR)* (June 2011). 2

[BW09] Bleiweiss A., Werman M.: Fusing time-of-flight depth and color for real-time segmentation and tracking. In *Proceedings of DAGM 2009 Workshop on Dynamic 3D Imaging* (2009), pp. 58–69. 2

[CM02] Comaniciu D., Meer P.: Mean shift: a robust approach toward feature space analysis. *IEEE Transactions on Pattern Analysis and Machine Intelligence 24*, 5 (2002), 603–619. 1

[CM09] Calderero F., Marques F.: Hierarchical fusion of color and depth information at partition level by cooperative region merging. In *Proc. IEEE Int. Conf. Acoustics, Speech and Signal Processing ICASSP 2009* (2009), pp. 973–976. 2

[DMZC12] Dal Mutto C., Zanuttigh P., Cortelazzo G.: Fusion of geometry and color information for scene segmentation. *Selected Topics in Signal Processing, IEEE Journal of 6*, 5 (2012), 505–521. 2, 3, 7, 8, 9, 10

[DZC11] Dal Mutto C., Zanuttigh P., Cortelazzo G.: Scene segmentation assisted by stereo vision. In *Proceedings of 3DIMPVT 2011* (Hangzhou, China, May 2011). 2

[EPD12] Erdogan C., Paluri M., Dellaert F.: Planar segmentation of rgbd images using fast linear fitting and markov chain monte carlo. In *Proc. of 9th Conference on Computer and Robot Vision (CRV)* (May 2012), pp. 32–39. 2

[FBCM04] Fowlkes C., Belongie S., Chung F., Malik J.: Spectral grouping using the nyström method. *IEEE Transactions on Pattern Analysis and Machine Intelligence 26*, 2 (2004), 214–225. 3

[FH04] Felzenszwalb P., Huttenlocher D.: Efficient graph-based image segmentation. *International Journal of Computer Vision 59*, 2 (Sept. 2004), 167–181. 1

[GAGM14] Gupta S., Arbeláez P., Girshick R., Malik J.: Indoor scene understanding with rgb-d images: Bottom-up segmentation, object detection and semantic segmentation. *International Journal of Computer Vision* (2014), 1–17. 2

[KCB*05] Kolmogorov V., Criminisi A., Blake A., Cross G., Rother C.: Bi-layer segmentation of binocular stereo video. In *IEEE conf. on Computer Vision and Pattern Recognition (CVPR)* (june 2005), vol. 2, p. 1186 vol. 2. 2

[LSR*10] Ladicky L., Sturgess P., Russell C., Sengupta S., Bastanlar Y., Clocksin W., Torr P.: Joint optimisation for object class segmentation and dense stereo reconstruction. In *Proceedings of the British Machine Vision Conference* (2010). 2

[NSF12] Nathan Silberman Derek Hoiem P. K., Fergus R.: Indoor segmentation and support inference from rgbd images. In *Proceedings of ECCV* (2012). 2

[PT97] Piegl L., Tiller W.: *The NURBS Book (2Nd Ed.)*. Springer-Verlag New York, Inc., New York, NY, USA, 1997. 3

[PZ14] Pagnutti G., Zanuttigh P.: Scene segmentation from depth and color data driven by surface fitting. In *IEEE International Conference on Image Processing (ICIP)* (2014), IEEE, pp. 4407–4411. 1, 2, 3, 7, 8, 9, 10

[RBF12] Ren X., Bo L., Fox D.: Rgb-(d) scene labeling: Features and algorithms. In *IEEE conference on Computer Vision and Pattern Recognition (CVPR)* (2012), pp. 2759–2766. 2

[SD14] Srinivasan N., Dellaert F.: A rao-blackwellized mcmc algorithm for recovering piecewise planar 3d model from multiple view rgbd images. In *IEEE International Conference on Image Processing (ICIP)* (2014). 2

[SM00] Shi J., Malik J.: Normalized cuts and image segmentation. *IEEE Transactions on Pattern Analysis and Machine Intelligence 22*, 8 (2000), 888–905. 1, 3

[WFFD11] Wallenberg M., Felsberg M., Forssén P.-E., Dellen B.: Channel coding for joint colour and depth segmentation. In *Proceedings of Pattern Recognition 33rd DAGM Symposium* (September 2011), vol. 6835 of *Lecture Notes in Computer Science*, pp. 306–315. 2

[WZYZ10] Wang L., Zhang C., Yang R., Zhang C.: Tofcut: Towards robust real-time foreground extraction using a time-of-flight camera. In *3DPVT* (2010). 2