Fusion of Geometry and Color Information for Scene Segmentation

(Article begins on next page)

Fusion of Geometry and Color Information for Scene Segmentation

Carlo Dal Mutto, *Student Member, IEEE,* Pietro Zanuttigh, *Member, IEEE,* and Guido M. Cortelazzo, *Member, IEEE*

Abstract—Scene segmentation is a well-known problem in computer vision traditionally tackled by exploiting only the color information from a single scene view. Recent hardware and software developments allow to estimate in real-time scene geometry and open the way for new scene segmentation approaches based on the fusion of both color and depth data. This paper follows this rationale and proposes a novel segmentation scheme where multi-dimensional vectors are used to jointly represent color and depth data and normalized cuts spectral clustering is applied to them in order to segment the scene. The critical issue of how to balance the two sources of information is solved by an automatic procedure based on an unsupervised metric for the segmentation quality. An extension of the proposed approach based on the exploitation of both images in stereo vision systems is also proposed. Different acquisition setups, like Time-of-Flight cameras, the Microsoft Kinect device and stereo vision systems have been used for the experimental validation. A comparison of the effectiveness of the different depth imaging systems for segmentation purposes is also presented. Experimental results show how the proposed algorithm outperforms scene segmentation algorithms based on geometry or color data alone and also other approaches that exploit both clues.

Index Terms—Segmentation, Clustering, Sensor Fusion, Unsupervised Metric, Depth Map, Stereo Vision, Time-of-Flight, Kinect

I. INTRODUCTION

Scene segmentation is the well-known problem of identifying the different elements of a scene. Images are the most common way of representing scenes, therefore it is not surprising that scene segmentation by way of images has attracted a lot of attention. Unfortunately scene segmentation by images is an ill-posed problem, and, despite a huge amount of research, it is still a very challenging task. Many segmentation techniques based on different insights have been developed, such as methods based on graph theory [1], methods based on clustering algorithms, (e.g. [2] and [3]), and also other methods based on region merging, level sets, watershed transforms and many other techniques [4]. The main drawback of image segmentation, independently from the deployed technique, is that the information carried by a single image may not suffice to completely understand the scene structure (consider for instance the simple case of an object and a background of the same color). Current technology allows to acquire scene descriptions beyond simple images: indeed geometrical scenes representations can be simply obtained in various ways. Binocular and multi-view stereo vision systems have been

extensively studied and their capabilities have been proved (an extensive review can be found in [5]). The market currently offers also more expensive and accurate active methods such as structured light systems and laser scanners. Lately matricial Time-of-Flight range cameras (e.g., Mesa Imaging SR4K [6]) and structured-light cameras (e.g., Microsoft Kinect [7]) have reached the market and are gaining popularity. Finally unstructured scene reconstruction tools like Microsoft Photosynth [8] can also provide the geometrical representation of a scene from a collection of pictures taken from random positions. The fusion of depth information acquired by any of these tools together with the color information coming from a standard color camera allows to obtain scene descriptions accounting for both geometry and color, i.e., representations where each sample has both geometry and color information associated to it. In this context, scene segmentation can be approached within a sensor fusion framework by algorithms exploiting both clues together and not just color as in standard segmentation algorithms. Within this perspective the segmentation problem can be formulated as the search for effective ways of meaningfully partitioning a set of samples featuring color and geometry information. Note how the proposed approach is close to 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 inside a scene together with prior knowledge and many other features extracted from the data acquired by the human visual system.

While the literature about scene segmentation based on color information is extremely vast, the number of works addressing scene segmentation by way of color and geometry information is still rather limited. A first possible solution is to perform two independent segmentations, one on the color image and one on the depth data, and then join the two results, as proposed in [9]. Many approaches, like [10] and [11], consider the special case of the recognition of the foreground from the background rather than the general scene segmentation case. In [12] two likelihood functions, one built on the basis of depth information and the other on the basis of color data, are combined together in order to assign samples to the background or to the foreground. Two different approaches for the segmentation of binocular stereo video sequences are presented in [13]: one, based on Layered Dynamic Programming, explicitly extracts depth information while the other one, based on Layered Graph Cuts, uses stereo correspondences without explicitly computing depth. Some other recent works try to jointly solve the segmentation and stereo disparity estimation problems. Ladicky et al. [14] exploit a probabilistic framework

based on Conditional Random Fields. This approach uses some heuristics about the scene structure that limit it to a particular scene setting (i.e., urban streets). A more general approach, also based on a probabilistic framework has been presented in [15]. Clustering techniques has been widely used in image segmentation and are well-suited to be extended in order to include different spatial and color features as shown in [16]. They can be exploited for joint depth and color segmentation by adding also the depth component to the vectors that are then clustered. Bleiweiss et Werman [17] follow this approach and apply *mean shift* clustering to vectors containing both the color and depth information. In [18] superparamagnetic clustering and channel representations are instead exploited to segment plant scenes from the color and depth data acquired by a Microsoft Kinect camera. In [19] we proposed a segmentation scheme for stereoscopic data that exploits different stereo vision algorithms in order to extract depth information which is then used to assist image segmentation based on clustering techniques. The approach of [19] was limited to stereo vision setups and required a supervised adjustment of a parameter weighting the relevance of geometry against color, an issue common also to other joint depth and color segmentation schemes such as [17] and [18].

This paper instead proposes a novel general scene segmentation scheme, based on the normalized cuts spectral clustering algorithm [3], that exploits the fusion of geometry and color information in a parameterless framework. This paper, differently from [19], introduces a completely general approach that can be applied in a completely automated way (i.e. it does not require any supervision for the choice of the balancing parameter between depth and color) regardless of the acquisition device and data type. Furthermore this work introduces an interesting improvement w.r.t. to [19] also for the stereoscopic image case that allows to exploit both color images of the stereo setup (see Section V) and it explicitly handles samples without a valid depth value due to occlusions or to the depth estimation algorithm.

The paper is organized as follows: Section II formalizes the adopted scene representation fusing both color and geometry. Section III introduces the proposed scene segmentation algorithm based on the normalized cuts spectral clustering algorithm. In Section IV an algorithm for the automatic balancing of the weight between geometry and color is proposed. It is based on a novel unsupervised metric for scene segmentation quality assessment. Section V proposes an extension of the segmentation algorithm tailored to the important case of stereoscopic data that besides geometry exploits the color of both images of a stereo pair. Section VI reports the experimental results and demonstrates how the fusion of geometry and color within the proposed method outperforms segmentation algorithms based on either geometry or color information only, or on the fusion of the two clues. In Section VII the results of the segmentation of the same scene acquired with different depth imaging techniques are presented and the performance of the different acquisition systems for segmentation purposes are discussed. Finally, Section VIII draws the conclusions.

II. JOINT REPRESENTATION OF GEOMETRY AND COLOR INFORMATION

Fig. 1 shows an overview of the proposed scene segmentation algorithm. The procedure can be subdivided into two main stages. In the first stage, a unified 6-dimensional representation of the scene points is built in order to fuse geometry and color information in a fully automatic way. In the second stage the obtained point set is segmented by means of spectral clustering.
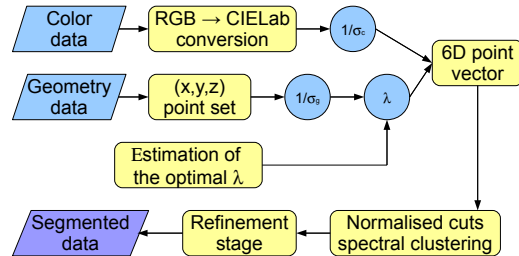

Fig. 1.   Architecture of the proposed segmentation scheme

This section addresses the construction of the unified representation for the fusion of geometry and color information. The description assumes the availability of a generic scene $\mathcal{S}$ described by a set of N points $p_i, i = 1, ..., N$ featuring both geometry and color information. Let us stress that for our purposes, the specific characteristics of the used 3D acquisition system are irrelevant and the acquired scene can be represented both by an image with the corresponding depth map or by a colored sparse point-cloud independently of the acquisition system. Such independence from the acquisition equipment is of major practical relevance since it allows to apply the proposed segmentation method with total generality to any type of color and geometry data describing a scene. In Appendix A it is also shown how to handle samples without a valid depth value due to the limitations of the employed acquisition system.

Color data require a 3D vector, in order to account for the R, G and B color components and another 3D vector is required for geometry information in order to describe the x, y and z coordinates of a point with respect to a given reference system (such a reference system can be obtained from the calibration data and the depth-maps produced by many acquisition systems). First of all geometry and color information need to be unified in a meaningful way. We choose to represent the color values in a perceptually uniform space in order to give a perceptual significance to the Euclidean distance between colors. This helps keeping consistent with the perceived color difference the distances used in the clustering process of Section III. Note also that a uniform color space ensures that the distances in each of the 3 color components are comparable, thus simplifying the clustering of the 3D vector associated to color information. The CIELab space was selected for color representation, i.e., the color information of

each scene point $p_i, i = 1, ..., N \in \mathcal{S}$, is the 3D vector:

$$\mathbf{p_i^c} = \begin{bmatrix} L(p_i) \\ a(p_i) \\ b(p_i) \end{bmatrix}, i = 1, ..., N \quad (1)$$

Geometry can be simply represented by the 3D coordinates $x(p_i)$, $y(p_i)$, and $z(p_i)$ of each point $p_i \in \mathcal{S}$, i.e. as:

$$\mathbf{p_i^g} = \begin{bmatrix} x(p_i) \\ y(p_i) \\ z(p_i) \end{bmatrix}, i = 1, ..., N \quad (2)$$

The ideal scene segmentation algorithm should be insensitive to the relative scaling of the point-cloud geometry since not all the scene acquisition systems are able to provide geometrical descriptions with respect to an absolute scale system (e.g. meters). For instance, tools like Photosynth [8] are only able to reconstruct the scene geometry up to an arbitrary scale factor. Therefore, in order to be independent with respect to scaling, all the components of $\mathbf{p_i^g}, i = 1, ..., N$ are normalized w.r.t. the average σ_g of the standard deviations of the point coordinates. To be more precise, let σ_x, σ_y and σ_z be the standard deviations of sets $x(p_i)$, $y(p_i)$ and $z(p_i), i = 1, ..., N$ respectively. The average standard deviation is then defined as $\sigma_g = (\sigma_x + \sigma_y + \sigma_z)/3$ and the adopted geometry representation is vector:

$$\begin{bmatrix} \bar{x}(p_i) \\ \bar{y}(p_i) \\ \bar{z}(p_i) \end{bmatrix} = \frac{3}{\sigma_x + \sigma_y + \sigma_z} \begin{bmatrix} x(p_i) \\ y(p_i) \\ z(p_i) \end{bmatrix} = \frac{1}{\sigma_g} \begin{bmatrix} x(p_i) \\ y(p_i) \\ z(p_i) \end{bmatrix} \quad (3)$$

It is worth noting that since the proposed segmentation algorithm is based on relative points distances and the overall distances are normalized, segmentation based on (3) besides scaling will also be insensitive to the choice of the reference frame. Furthermore by using the coordinates of the point in the 3D space it is ensured that all the 3 spatial dimensions refer to the same space and that they are consistent, differently from other approaches like [17] where the 2D coordinates in image space are used together with depth data, which lies in a different space.

In order to balance the relevance of the two kinds of information (color and geometry) in the merging process, the color information vectors $\mathbf{p}_i^c, i = 1, ..., N$ are normalized as well by the average σ_c of the standard deviations σ_L, σ_a and σ_b of their L, the a and the b components respectively. The final color representation therefore is:

$$\begin{bmatrix} \bar{L}(p_i) \\ \bar{a}(p_i) \\ \bar{b}(p_i) \end{bmatrix} = \frac{3}{\sigma_L + \sigma_a + \sigma_b} \begin{bmatrix} L(p_i) \\ a(p_i) \\ b(p_i) \end{bmatrix} = \frac{1}{\sigma_c} \begin{bmatrix} L(p_i) \\ a(p_i) \\ b(p_i) \end{bmatrix} \quad (4)$$

From the above normalized geometry and color information

vectors, each scene point $p_i^f, i = 1, ..., N$ is represented as:

$$\mathbf{p}_i^f = \begin{bmatrix} \bar{L}(p_i) \\ \bar{a}(p_i) \\ \bar{b}(p_i) \\ \lambda \bar{x}(p_i) \\ \lambda \bar{y}(p_i) \\ \lambda \bar{z}(p_i) \end{bmatrix}, i = 1, ..., N \quad (5)$$

where λ is a parameter balancing the contribution of color and geometry. High values of λ increase the relevance of geometry, while low values of λ increase the relevance of color information. Fig. 2 shows an example of the relevance of λ in the segmentation of the *plant* scene, which is a 3D model obtained by Microsoft Photosynth. For low values of λ (e.g., $\lambda = 0.001$) the segmentation is dominated by the color clue, thus leading to some artifacts due to the noise on the color data. For higher value of λ (e.g., $\lambda = 5$), the segmentation is dominated by the geometry clue, and the entire plant is segmented into three parts that do not take in account color, denying as well a meaningful segmentation. For intermediate values of λ (e.g., in this case $\lambda = 1$), geometry and color information in this case are well balanced providing correct segmentation results by the proposed method. Note that the value of λ leading to the best segmentation results depends on the specific scene data. While in [19] λ was selected in a supervised way, this paper shows how to automatically select it by the method that will be introduced in Section IV.

III. SEGMENTATION BY MEANS OF SPECTRAL CLUSTERING AND NYSTRÖM METHOD

The scene representation introduced in the previous section produces a set P_c formed by the 6D vectors $\mathbf{p}_i^f, i = 1, ..., N$ which represents in a intuitive and consistent way the geometry and color information of the scene points $p_i, i = 1, ..., N$. Vectors $\mathbf{p}_i^f$ are well suited to be clustered by one of the various clustering techniques. Central grouping algorithms, such as k-means and mean-shift clustering, are fast and effective, but have the main drawback of assuming Gaussian the distributions of the points in P_c. Since this assumption is not generally verified in the considered application, this family of methods applied to the set P_c gives poor results. Fig. 3 shows an example of the results of k-means clustering and of mean-shift clustering on set P_c of points relative to the *baby* scene. The methods based on pairwise affinity measures computed between all the possible couples of points in P_c operate somehow within a philosophy opposite to that of central grouping. They are more flexible, because they do not assume a Gaussian model for the distribution of the points, and consequently their results in practical segmentation situations are more accurate and robust. The main drawback of the pairwise affinity methods is that they need to compare all the possible pairs of points in P_c. Computing and storing all the possible affinities forces a tremendous amount of processing, very expensive in terms of both CPU and memory resources. The normalized cuts spectral clustering method presented in [3] is an effective example of this family. In this method a graph is firstly built from all the points (vertices) and their pairs (edges), and then partitioned according to spectral graph theory
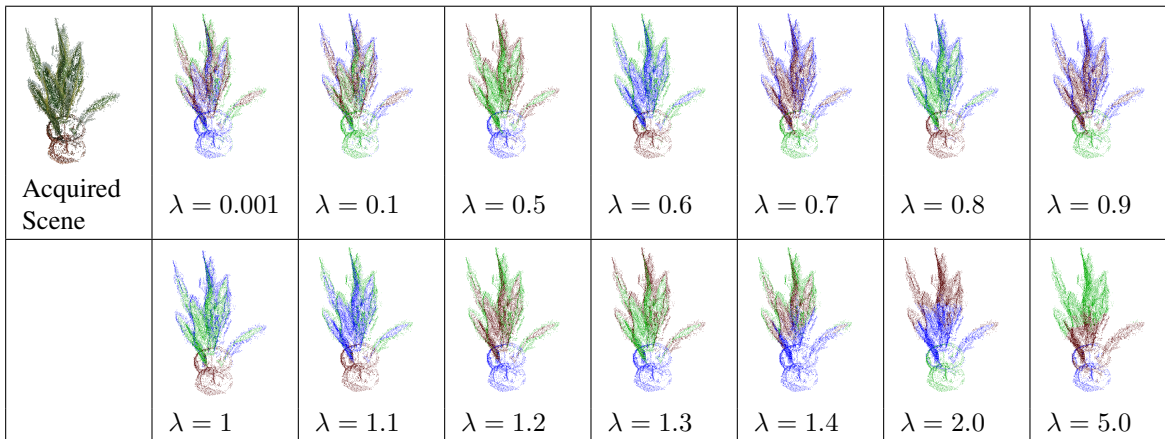
Acquired Scene	$\lambda = 0.001$	$\lambda = 0.1$	$\lambda = 0.5$	$\lambda = 0.6$	$\lambda = 0.7$	$\lambda = 0.8$	$\lambda = 0.9$
	$\lambda = 1$	$\lambda = 1.1$	$\lambda = 1.2$	$\lambda = 1.3$	$\lambda = 1.4$	$\lambda = 2.0$	$\lambda = 5.0$

Fig. 2. Different segmentation results on the *plant* scene for different values of λ. *(Best viewed in colors)*

a)   b)
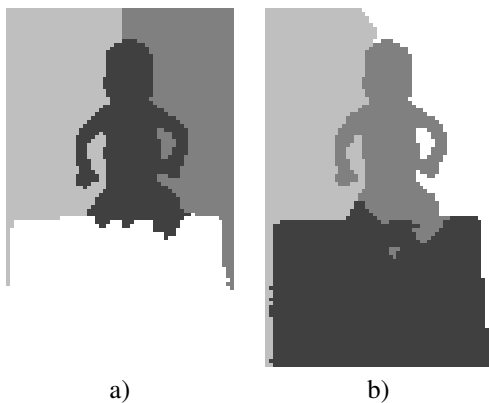
Fig. 3. Segmentation of the *baby* scene applying a) k-means clustering and b) mean-shift clustering

criteria. Normalized cuts is the minimization criterion adopted for the graph cut in this case in order to account both for the similarity between the pixels inside the same segment and the dissimilarity between the pixels in different segments. The minimization by the normalized cut criterion can be regarded and solved as a generalized eigenvalue problem. A variety of methods have been proposed for the efficient approximation of the graph associated to the set of points in order to overcome the computational and memory burden. A possible solution is imposing that not all the points are connected, but that the non negligible connections only concern small sets of points. This assumption practically leads to oversegmentation, and implicitly imposes some models to the point distributions. In the method based on the integral eigenvalue problem proposed in [20] the set of points is firstly randomly subsampled (a set of n points is randomly extracted from the whole set of N points); this subset of n points is then partitioned by the method proposed in [3], and the solution is propagated to the whole N points set by a specific technique called Nyström method. As shown in [20], the results of this method are comparable to the ones of the normalized spectral clustering algorithm, but at computation and memory costs comparable with those of the central grouping algorithm. For this reason the Nyström method approach to the normalized cut spectral clustering (briefly denoted with NNCSC) was selected for our

scene segmentation application. The fact that NNCSC does not assume any model for the distribution of the points in P_c is a rather important feature. Indeed color distribution, as already pointed out, is usually not Gaussian and it is even more unlikely that the geometry distribution is Gaussian (just consider, for instance, that a Gaussian distribution cannot represent well the surface of a human body in 3D). Moreover, since the two components of the point-cloud feature vectors are usually not Gaussian, it is far more unlikely that their joint distribution is Gaussian. In some way, NNCSC provides a nice framework to incorporate the fact that P_c is partitioned into subsets where color and geometry are homogeneous, without imposing an overall model, which for the distributions of the points in P_c would be very hard to derive and in any case it would be quite unlikely Gaussian. For a detailed explanation of normalized cuts spectral clustering, the interested reader is referred to [3], and for Nyström method to [20]. A drawback of normalized cuts, shared with other clustering algorithms like k-means, is that the number of clusters K in which the point-cloud is partitioned needs to be known a priori. This issue can be overcome by the use of an automatic selector of the number of clusters K, such as the one proposed in [21]. The Nyström method approximation leads to a very fast algorithm, hence suitable for real time applications. It will be shown that the clusters found by NNSC applied to P_c represent rather well the different scene regions.

In order to avoid small regions due to noise we also included an optional refinement stage for samples arranged on regular grids (i.e., when the input data are images and depth maps) where regions with extension smaller than a threshold are removed and their points are assigned to the cluster corresponding to the mode of the points closer to the region. Such a refinement was instead not used in the results of Fig. 15, 16 and 17 where the data are not aligned on regular grids.

IV. AUTOMATIC WEIGHTING OF COLOR AND DEPTH INFORMATION

The optimal value of the λ parameter, i.e. the relative weight between depth and color information, depends on the color and geometry properties of the scene and it turns out to be a key issue in the proposed segmentation scheme. Given that a

single optimal value of λ does not exist, this section proposes an effective method for the automatic setting of λ, based on an unsupervised metric for segmentation quality assessment. This approach allows to obtain a parameterless segmentation method that does not rely on manual tuning of the weighting coefficient λ.

A number of unsupervised metrics for the evaluation of image segmentation quality have been proposed in the last decades (a comprehensive taxonomy of them is given in [22]). Among the various metrics of the literature, the FRC metric of [23] has proven to be at the same time very reliable and computationally fast. This method, as proposed by the authors, takes as input a color image and a segmentation map and returns as output a measure of the segmentation quality. Our context is slightly different, because our input is threefold, namely a color image I, a depth-map D (with the geometry information) and a segmentation map S (where the image has been divided in a set of K segmented regions $S_i, i = 1, .., K$) and we are forced to introduce a novel segmentation metric that considers together both color and geometry. In the case of unstructured data representations (i.e. point clouds), each point has an associated 3-dimensional color vector and I is simply the set of all the color vectors associated to the 3D points. The depth map D is instead replaced by a set of 3-dimensional vectors with the (x, y, z) coordinates. The segmentation map simply associates each point to one of the clusters. Both color and geometry data are firstly normalized as follows:

- The three color channels (red, green and blue) of I, i.e., I_R, I_G and I_B are normalized in order to obtain a color representation $\tilde{I}$ with values in the range $[0, 1]$.
- Depth map D is also normalized to depth map $\tilde{D}$ with values in $[0, 1]$. In the case of unstructured data D is also shifted and normalized in order to have all the coordinates in the range $[0, 1]$. More precisely, for unstructured data, the chosen normalization factor is the maximum of the sides of the bounding box including the point cloud. The same normalization factor is used for all the 3 dimensions in order to avoid "stretching" the point cloud.

Following the approach presented in [23], a "good" segmentation should have two fundamental properties, namely:

- inside a single segmented region the image should have uniform properties (i.e., a constant color or some repeating pattern or texture).
- each couple of different segments should have different properties (this ensures that there is no over-segmentation of the image).

In the considered situation the above criteria should be satisfied with respect both to the color image and to the depth map. Firstly we consider the segmentation map S and the normalized color image $\tilde{I}$: the evaluation of the first property is quite simple for regions of constant color, where it is usually associated to the standard deviation of the data inside the segmented region, but it is quite difficult for heavily textured regions. This issue in [23] and other works on segmentation evaluation is approached by computing various texture or color distribution descriptors. Unfortunately such descriptors are not always reliable. Indeed heavily textured regions with complex

color patterns are where both state-of-the-art segmentation techniques and evaluation metrics usually either have major problems or completely fail. Since in our application also depth information is available, we decided to give more importance to the color component of the metric in regions with limited texture and less importance in heavily textured regions where depth data can be more reliable. The idea adopted to obtain this result is to subtract from the standard deviation of the data of a segmented region the standard deviation due to the amount of texture inside the region. More precisely it is assumed that the amount of texture of a segmented region S_i, denoted as $\sigma_t(S_i)$, is proportional to the average local standard deviation of the samples internal to segment S_i, namely:

$$\sigma_t(S_i) = \frac{\sum_{j \in S_i^*} \sigma_w(j)}{|S_i^*|} \qquad (6)$$

where $\sigma_w(j)$ is the local standard deviation computed on a small window (for the experimental results a 3×3 window has been used) centered on pixel j. S_i^* is the set of the internal pixels of segment S_i, i.e., the ones for which window $w(j)$ lies completely inside the segment. $|S_i^*|$ is instead the cardinality of S_i^*. Note that this reasoning assumes that the scene color information is represented by way of an image. If the scene is represented by a sparse colored point cloud the window can be replaced by the set of the points with distance from j lower than a threshold t. In the case of point clouds this approach is however computationally expensive. It can be made faster by avoiding the subtraction of the texture standard deviation at the price of a loss in the metric performances. A measure of the internal disparity D_{intra}^i of the ith segment S_i can be computed as follows:

$$D_i^{intra} = \max(\sigma(S_i) - \sigma_t(S_i), 0) \frac{|S_i|}{N} \qquad (7)$$

where $\sigma(S_i)$ is the global standard deviation of the color data inside the segmented region, $|S_i|$ is the cardinality of the points in the ith region S_i and N is the total number of points in P_c. As previously said the idea is to consider the standard deviation due to the clustering accuracy and not to the complexity of the texture pattern inside the segmented region. The average local standard deviation is therefore subtracted to the global standard deviation of the color inside the region (in the case that the local standard deviation is greater than $\sigma(S_i)$, D_i^{intra} will be set to 0). Expression (7) reduces the weight of highly texturized regions, which is quite reasonable in light of the fact that for these regions depth data offer more reliable indications. This is particularly true if depth information is computed by stereo vision techniques since their performance, as well known, is more reliable in textured regions. In any case it seems rather reasonable to use depth in heavily texturized regions and color information in regions with uniform or limited texture which are easy to segment by color information and usually correspond to areas where depth is poorly estimated due to the lack of features to be matched. Finally the segments are also weighted on the basis of their size.

The D^{intra} measure for the whole image is computed as

the sum of the D_i^{intra} values of each segmented region:

$$D^{intra} = \sum_i D_i^{intra} \tag{8}$$

The disparity between the different segmented regions is instead computed as the distances between the centroids of pairs of clusters (note that here a cluster corresponds to a segmented region) as in the FRC metric introduced in [23]:

$$D_{i,j}^{inter} = |E(S_i) - E(S_j)| \tag{9}$$

These disparities are then averaged on all the segment pairs:

$$D^{inter} = \frac{\sum_{i,j(i \neq j)} D_{i,j}^{inter}}{K(K-1)} \tag{10}$$

and the final metric for color data is computed as the difference between the disparity between different regions and the internal disparity divided by 2, i.e., as:

$$Q^{color}(\tilde{I}, S) = \frac{D^{inter} - D^{intra}}{2} \tag{11}$$

The metric for geometry information is computed in the same way but without considering the local standard deviations, namely:

$$D_i^{Dintra} = \sigma^D(S_i)\frac{|S_i|}{N} \tag{12}$$

$$D^{Dintra} = \sum_i D_i^{intra} \tag{13}$$

$$D_{i,j}^{Dinter} = |E^D(S_i) - E^D(S_j)| \tag{14}$$

$$D^{Dinter} = \frac{\sum_{i,j(i \neq j)} D_{i,j}^{Dinter}}{K(K-1)} \tag{15}$$

$$Q^{depth}(\tilde{D}, S) = \frac{D^{Dinter} - D^{Dintra}}{2} \tag{16}$$

where $\sigma^D(S_i)$ is the standard deviation of the geometry values in region S_i and D^{Dinter} is also computed with respect to geometry data. Note how D is a set of scalar values in the case of depth maps and a set of 3-dimensional vectors in the case of point clouds, i.e. in the unstructured data case D has the same structure of color data with x, y and z in place of the three color channels. Finally the combined segmentation quality metric is computed as follows:

$$Q(\tilde{I}, \tilde{D}, S) = Q^{color}(\tilde{I}, S) + n_f * Q^{depth}(\tilde{D}, S) \tag{17}$$

$$\text{with} \quad n_f = \begin{cases} 1 & \text{for unstructured data} \\ 3 & \text{for depth maps} \end{cases}$$

In the case of depth maps depth relevance is multiplied by 3 in order to assign the same total weight to the 3 color channels together and to the depth data. In the unstructured data case both representations have 3 components and the multiplication by 3 is not needed.

The optimal λ can be automatically selected as the value that maximizes the $Q(\tilde{I}, \tilde{D}, S)$ value in (17). Different values of λ correspond to different segmentation maps S that in turn correspond to different values of $Q(\tilde{I}, \tilde{D}, S)$. The value of λ that maximizes (17) is the value that provides the best

segmentation with respect to the Q metric. This approach was experimentally found to be very effective, indeed in all the experimental examples it always gave the value of λ providing the best segmentation. An example of this fact is reported in Fig. 4 where the maximum of Q (obtained for $\lambda = 4$) corresponds to the best segmentation. Indeed only for $\lambda = 4$ even the part of the box between the legs of the baby is correctly associated to the box segment. The plot of Q versus λ clearly shows how the correspondence between the values of λ and the changes in segmentation quality are well reflected by changes of the $Q(\tilde{I}, \tilde{D}, S)$ value. Fig. 5 shows the behaviour of metric Q versus λ on a different scene, while Fig. 6 refers to the computation of the metric on a point cloud representation instead of a color image and a depth map as in the other two cases. It is worth noting that, although the plots are quite different, in all the 3 cases the maximum of Q corresponds to the value of λ delivering the best segmentation result. It is finally worth noting that in spite this method requires to compute several segmentations, it can be easily managed within reasonable computation times by coarse to fine approaches. For instance a set of segmentations can be firstly performed on a subsampled dataset and then, once the optimal λ value is selected, the full resolution segmentation can be computed only for that value of λ. Furthermore in the case of video segmentation, since the optimal λ depends on the general scene properties, it could be computed on the first frame and then propagated to a set of subsequent frames.

V. SEGMENTATION OF STEREO IMAGE PAIRS

Stereo vision algorithms are rather attractive for various reasons: there is a copious literature about them [5], they require an inexpensive setup and they use only a pair of images as input data, hence representing the next step in terms of acquisition complexity with respect to segmentation based on single images. Stereo vision data therefore represent a situation of special interest for the proposed segmentation approach. In this section an ad-hoc extension of the proposed method for this kind of data is proposed. It is worth noting though that the segmentation scheme introduced so far can already provide very good performance without the further extension of this section. This optional refinement allows to improve performance in the cases where two images and a depth map are available.

As it is well known, a stereo vision system is constituted by two standard cameras that acquire two slightly different views of the same scene. If the stereo vision is calibrated, depth information can be estimated from the two views by one of the many stereo vision algorithms (see [24] for a comparison of state-of-the-art algorithms in this field). The segmentation method introduced so far can already be applied to the depth map obtained from stereo vision and to one of the two images. However since in this case a second image of the same scene is also available, this section introduces a way to exploit it in order to further improve the segmentation results.

Lets denote with $\mathcal{L}(p_i)$ and $\mathcal{R}(p_i)$ the pair of rectified images and with $\mathcal{D}(p_i)$ the disparity map estimated from them (relative to the left view). Without loss of generality assume
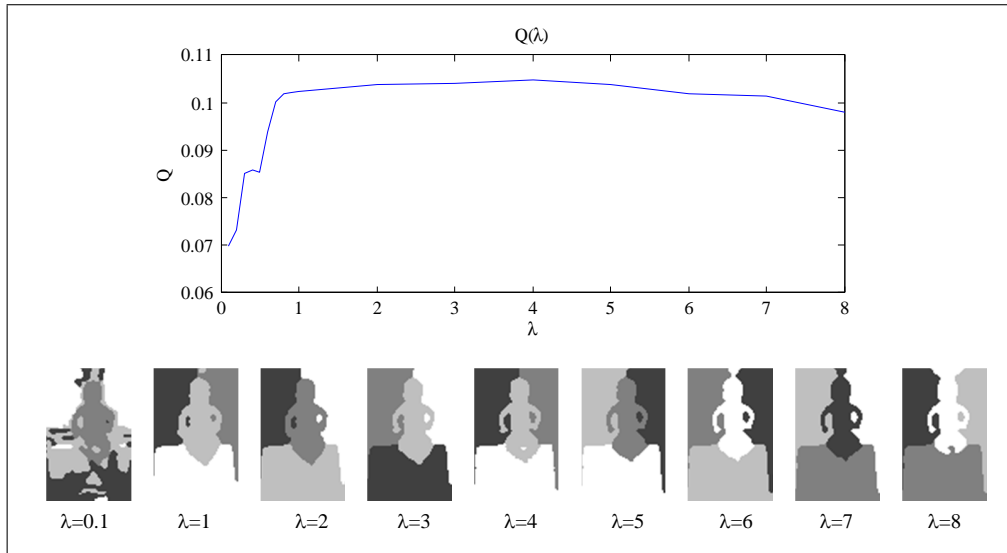
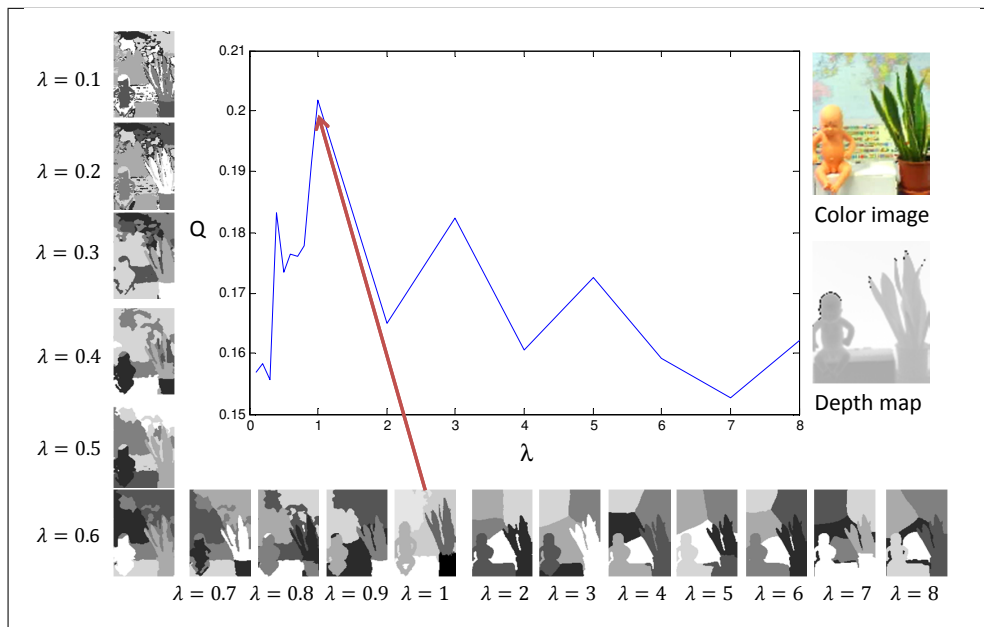Fig. 4. Values of Q versus λ and segmentation of the *baby* scene for different values of λ

Fig. 5. Values of Q versus λ and segmentation of the *baby and plant* scene for different values of λ
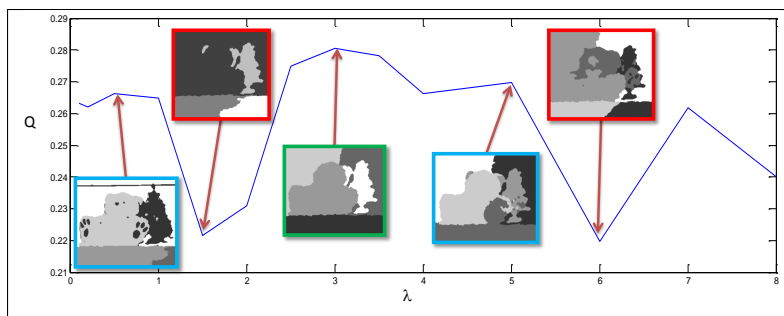

Fig. 6. Values of Q versus λ and segmentation of the point cloud of the third row of Figure 18 for different values of λ. Note how the best segmentation (shown in green) is correctly recognized, good segmentations (in blue) correspond to high Q values and the bad segmentations (in red) to low Q values.

that the target is the segmentation of the scene as seen from the left image $\mathcal{L}(p_i)$. The disparity map can be used to locate for each pixel of the left image the corresponding one in the right image, except for the pixels that are visible only in the left view (because of occlusions or because they are out of the right frame) or the pixels without a disparity value because of the limitations of the adopted stereo vision algorithm. Hence it is worth defining image $\mathcal{R}_w$ as follows:

$$\mathcal{R}_w(p_i) = \begin{cases} \mathcal{R}(p_i - \mathcal{D}(p_i)) & \text{if } \mathcal{D}(p_i) \text{ exists} \\ \mathcal{L}(p_i) & \text{if } \mathcal{D}(p_i) \text{ does not exist} \end{cases} \tag{18}$$

Image $\mathcal{R}_w(p_i)$ represents the right image warped to the viewpoint of the left one except for the points of the left image not visible in the right one. For these points the corresponding value in the left image is simply copied onto $\mathcal{R}_w(p_i)$. Fig. 7d shows an example of such an image. The disparity map is related to the depth map $\mathcal{Z}(p_i)$ through the well-known equation $\mathcal{Z}(p_i) = (bf)/\mathcal{D}(p_i)$ where b is the baseline of the stereo vision setup and f focal length of the two cameras. The depth map can then be used together with calibration information in order to compute the positions of the scene points in the 3D space. Therefore in the stereo case for each scene point p there is available:

- its color value in the left view $\mathcal{L}(p_i) = [L_l(p_i), a_l(p_i), b_l(p_i)]$
- its color value in the right view $\mathcal{R}_w(p_i) = [L_r(p_i), a_r(p_i), b_r(p_i)]$ (as previously said replaced by a copy of $\mathcal{L}(p_i)$ for the points not visible in the right view)
- its position in the 3D space $(x(p_i), y(p_i), z(p_i))$

As in Section II all the various components can be normalized by the corresponding standard deviations obtaining the three normalized vectors:

$$\begin{bmatrix} \bar{L}_l(p_i) \\ \bar{a}_l(p_i) \\ \bar{b}_l(p_i) \end{bmatrix} = \frac{1}{\sigma_{cl}} \begin{bmatrix} L_l(p_i) \\ a_l(p_i) \\ b_l(p_i) \end{bmatrix}$$

$$\begin{bmatrix} \bar{L}_r(p_i) \\ \bar{a}_r(p_i) \\ \bar{b}_r(p_i) \end{bmatrix} = \frac{1}{\sigma_{cr}} \begin{bmatrix} L_r(p_i) \\ a_r(p_i) \\ b_r(p_i) \end{bmatrix}$$

$$\begin{bmatrix} \bar{x}(p_i) \\ \bar{y}(p_i) \\ \bar{z}(p_i) \end{bmatrix} = \frac{1}{\sigma_g} \begin{bmatrix} x(p_i) \\ y(p_i) \\ z(p_i) \end{bmatrix}$$

where the standard deviations σ_{Ll}, σ_{al} and σ_{bl} refer to the left view and σ_{Lr}, σ_{ar} and σ_{br} to the right one. Let $\sigma_{cl} = (\sigma_{Ll} + \sigma_{al} + \sigma_{bl})/3$ and $\sigma_{cr} = (\sigma_{Lr} + \sigma_{ar} + \sigma_{br})/3$ be the average standard deviations of color data for the left and right image respectively. The standard deviation of the geometry data is defined as in Section II. From the above normalized geometry and color information vectors each scene point $p_i, i = 1, ..., N$ can be represented by a 9-dimensional vector representing its 3D position and its color in the two views naturally extending

the representation of Section II:

$$\mathbf{p}_i^f = \begin{bmatrix} \bar{L}_l(p_i) \\ \bar{a}_l(p_i) \\ \bar{b}_l(p_i) \\ \bar{L}_r(p_i) \\ \bar{a}_r(p_i) \\ \bar{b}_r(p_i) \\ \lambda\bar{x}(p_i) \\ \lambda\bar{y}(p_i) \\ \lambda\bar{z}(p_i) \end{bmatrix} , i = 1, ..., N \tag{19}$$

This 9-dimensional vector can be used as input to the spectral clustering algorithm of Section III and used to segment the scene seen from the left image. In case the segmentation of both views was needed the same approach can be clearly adopted with the disparity map relative to the right view and by swapping the left and right images in the previous discussion. The advantage of the 9-dimensional representation will be clear from the experimental results in Section VI-B.
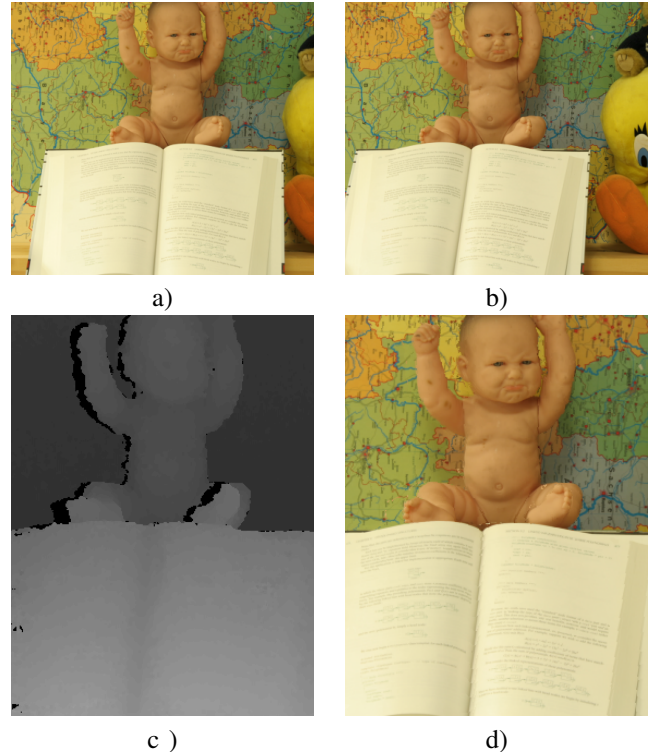
Fig. 7. Input data for the segmentation of stereoscopic pairs: a) left view; b) right view; c) disparity relative to the left view (disparity values have been stretched in order to improve the readability of the printed paper); d) detail of the right view warped to the left viewpoint. Note how occlusions in the warped view were filled by copying data from the left view. Some small artifacts noticeable in the figure are due to the errors in the disparity estimation (in this case estimated by the method of [25]).

VI. EXPERIMENTAL RESULTS

The performances of the proposed scene segmentation algorithm is verified on datasets representing different scenes, acquired with different technologies. This is purposely done in order to assess the effectiveness of the joint usage of color and geometry for scene segmentation, independently of the specific

3D data types and of the used acquisition tools. In particular the considered scenes are acquired by: a trinocular system made by one matricial Time-of-Flight range camera and two standard cameras; a standard 2-views stereo vision system; a Microsoft Kinect sensor [7] and by Microsoft Phothosynth [8], i.e., an unstructured scene reconstruction system.

A. Results on the trinocular system data

A matricial Time-of-Flight range camera and two standard RGB cameras can be used as a single system for acquiring both geometry and color information [26]. The input data is obtained by taking all the 3D points acquired by the matricial Time-of-Flight and by appending to them the color information of the corresponding pixels obtained from the images of the two cameras. It is preferable to deploy two RGB cameras rather than only one in order to alleviate the occlusion problems. The used system features a Mesa Imaging SR4000 matricial Time-of-Flight range camera and 2 high resolution RGB cameras. It is calibrated by the procedure described in [26]. The proposed segmentation algorithm is tested on several scenes and compared with scene segmentation based on geometry or color information only obtained both by using our method and two state-of-the-art segmentation algorithms (i.e., the graph-based method of Felzenszwalb et al. [1] and the mean-shift algorithm of [27]). The results of Fig. 8 clearly show the effectiveness of the proposed method. The scenes shown in the figure contain good examples of common issues making non-trivial scene segmentation, namely issues due to the background color articulation and to the complexity of the scene geometry (as in the case of the plant of the second and third rows of the figure). The first two columns of Fig. 8 show the color and geometry information relative to three different scenes (one for each row). These data have been used as input for three different segmentation methods (namely NNCSC, [1] and [27]) using either color information only or geometry information only and the corresponding results are shown in rows from 3 to 8. Finally the rightmost column shows the results of the proposed segmentation technique based on the fusion of color and geometry information. Color based segmentation exhibits various problems, e.g., the space between the arms is not so clearly recognizable in the color segmentation results of the first row of Fig. 8. In the scene of the first row of Fig. 8 segmentation based on geometry information only gives better results, although not completely satisfactory (e.g. [27] provides the best results, indeed it is the only method that recognizes the two regions but the separation is not as accurate as for the proposed method). The proposed technique fusing color and geometry clearly performs better than the compared state-of-the-art algorithms. For instance in the case of the scene of the first row of Fig. 8 it is the only method that accurately separates the baby from the white box behind it. The second and third rows of Fig. 8 confirm that the proposed scene segmentation method allows for a very good segmentation of both the plant and the vase which are very difficult subjects to segment on the basis of either color or geometry only (e.g., the proposed method is the only one capable to correctly extract the complete baby shape in the third row experiment).

It is fair to recall that the proposed technique incorporates NNCSC as clustering method. The usage of either k-means or of mean-shift as clustering method would give poorer results as shows the comparison of the results of the first row of Fig. 8 with the ones of Fig. 3. Fig. 9 refers to the *baby and plant* scene (the one in the last row of Fig. 8) and offers an extensive comparison between the results of different clustering techniques, namely it compares the proposed method based either on NNCSC, k-means or mean-shift and the techniques of [1] and [27]. Each row corresponds to a different method, while the different columns show the results on color only, on geometry only, and on the fusion of color and geometry. The results of row 4 and 5, obtained by the state-of-the-art image segmentation methods of [1] and [27] on either color only or geometry only information, demonstrate the effectiveness of the fusion of color and geometry by the proposed method. It is also worth noting how the proposed approach implemented with simpler clustering schemes would have a performance inferior to the one obtained by using NNCSC even if applied to color and geometry together.

Fig. 10 refers instead to the segmentation of a person. It can be seen that the human shape is perfectly identified by the proposed method (Fig. 10e), in contrast to the very bad result obtained by color information only, and to the one obtained by geometry only, that presents artefacts in the lower part of the body (e.g., feet). This is a good example of a typical issue of segmentation based on geometry only. Geometry information turns out well suited to separate objects and people from the background, but not to separate different objects in touch with each other. At the same time color segmentation is prone to be mislead by complex texture patterns, such as the texture on the person's shirt. By suitably fusing the two clues it is possible to solve both issues at the same time.

The execution time of the current MATLAB implementation of the proposed segmentation algorithm was less than 0.5 seconds on all the analysed scenes.

B. Results on stereo vision data

The proposed scene segmentation method was also tested on data obtained from a stereo vision system (for these results geometry was recovered using the method of [25]). Our segmentation algorithm was tested on data from the Middlebury [24] stereo vision repository which is a very commonly used benchmark for stereo vision. Fig. 11a and Fig. 11b show the input data of the *aloe* scene of [24]. This is a quite challenging scene due to the heavily texturized background and to the complex shape of the plant. Fig. 11c shows the result of the segmentation by the proposed method applied to one of the two views together with depth data. The results are already quite good: most of the leaves are recognized and the vase is correctly separated from the plant. However some artifacts are still present, e.g., the artifacts on the right side of the vase due to the dark background or the ones on the upper right leaf. Fig. 11d shows the benefits of the approach described in Section V that exploits also the second color view. Segmentation accuracy is improved (e.g., the upper right leaf is correctly detected and the artefact on the right of the vase

Color	Depth	Color Segmentation			Geometry segmentation			Fusion
		NNCSC	[1]	[27]	NNCSC	[1]	[27]	Proposed

Fig. 8. Segmentation of datasets acquired by the trinocular setup: (first row) *baby* scene, (second row) *plant* scene, (third row) *baby and plant* scene. The figure shows the results of the proposed method (rightmost column) using only color, only depth and by fusing the two clues. The figure also reports the results of two state-of-the-art methods (i.e. [1] and [27]) applied to color or geometry only.

disappears). However some artifacts due to missing values in the depth data computed by [25] are still visible (e.g. on the side of some leafs). Fig. 11e shows the results obtained by also applying the occlusion handling scheme of Appendix A, note how the artifacts due to missing depth data disappear. Fig. 11f shows the results of [17], that also jointly exploits depth and color, while the Figures from 11g to 11l show the results of state-of-the-art segmentation algorithms working on either color only or geometry only. The proposed method (the results of the complete scheme are the ones of Fig. 11e) clearly outperforms the other approaches.

Fig. 12 refers instead to the *baby2* scene of the Middlebury repository. Again the proposed approach (Fig. 12e) outperforms the other approaches shown in the Figures from 12f to 12l. In this case the results of the proposed approach are already very good with a single color view, however the exploitation of the second color view allows to get rid of a couple of minor remaining artefacts.

The performances of the proposed approach are also compared with other recent segmentation schemes jointly exploiting color and depth information. Fig. 13 shows a comparison[1] between the proposed scheme and the methods of [9] and [17] on two scenes from the Middlebury dataset. The proposed method is the only one that in both situations correctly recognizes all the three main regions of the scene (i.e. vase, plant and background in the first and baby, box and background in the second). The method of [9] can correctly recognize the foreground region shape but it cannot divide the objects on the basis of color information (it appears a bit biased towards depth data), while the method of [17] produces some artifacts (e.g. on the left side of the baby or close to the plant leaves), even if it is able to distinguish the baby from the box. Furthermore note how the proposed method allows to automatically balance the two clues, while the method of [17]

---

[1]The figures with the results of [9] have been taken from their paper while the method of [17] has been implemented following the description on the paper.

requires a manual parameter tuning in order to obtain a good segmentation.

C. Results on Kinect data

Nowadays, scene descriptions accounting for both geometry and color can be readily and inexpensively obtained also by cheap mass market devices such as the Microsoft Kinect [7]. In fact, the Kinect sensor includes both an active system that captures a real time description of the scene geometry and a color camera. The wide availability and low cost of such sensors open a wide application scenario to the proposed segmentation framework since it eliminates the need of expensive 3D acquisition devices or of computationally complex state-of-the-art stereo algorithms.

In order to take advantage of both the geometry and the color acquired by the Kinect in a unique framework, it is firstly necessary to calibrate its depth sensor with the color camera. A first possibility is to perform a standard stereo camera calibration with OpenCV [28] on the color images acquired by the color camera and on the amplitude image acquired by the depth camera (with the IR projector obscured). The proposed segmentation algorithm can then be applied to the Kinect data as shown by the results of Fig. 14. It is worth noting that the overall scene segmentation is correct, but there are some errors near depth discontinuities. Such errors are due to the artefacts present in the depth data acquired by the Kinect sensor (i.e., the acquired depth and color edges are not precisely aligned as clearly visible in Fig. 14a and Fig. 14b).

A second possibility offered by the freely available OpenNI [29] framework is to directly acquire a colored point cloud. Fig. 15 and Fig. 16 show a couple of point clouds acquired in this way and the corresponding segmentations. Again the results are very good and the objects are correctly separated from the background (even the part of the teddy bear that touches the table is correctly separated from the table itself).

Method	Color image	Depth map	Color segm.	Geom. segm.	Fusion
Proposed method (Spectral Clust.)					
K-means clustering					
Mean-shift clustering					
Felzenszwalb et Al.[1]					
Edison [27]					

Fig. 9. Segmentation of the *baby and plant* scene using different segmentation algorithms on color, geometry and the fusion of color and geometry by the proposed approach.

a)   b)   c)   d)   e)   f)   g)   h)   i)
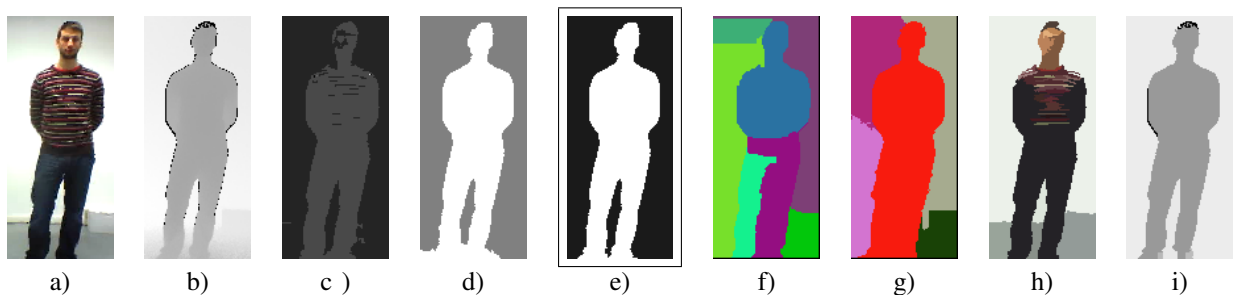
Fig. 10. Segmentation of the datasets acquired by the trinocular setup on a person scene. The figure shows: a) color image; b) corresponding depth-map; c) segmentation on the basis of color information only; d) segmentation on the basis of geometry only; e) segmentation based on the proposed method, fusing geometry and color; f) segmentation obtained by applying [1] to color information; g) segmentation obtained by applying [1] to geometry information; h) segmentation obtained by applying [27] to color information; i) segmentation obtained by applying [27] to geometry information;
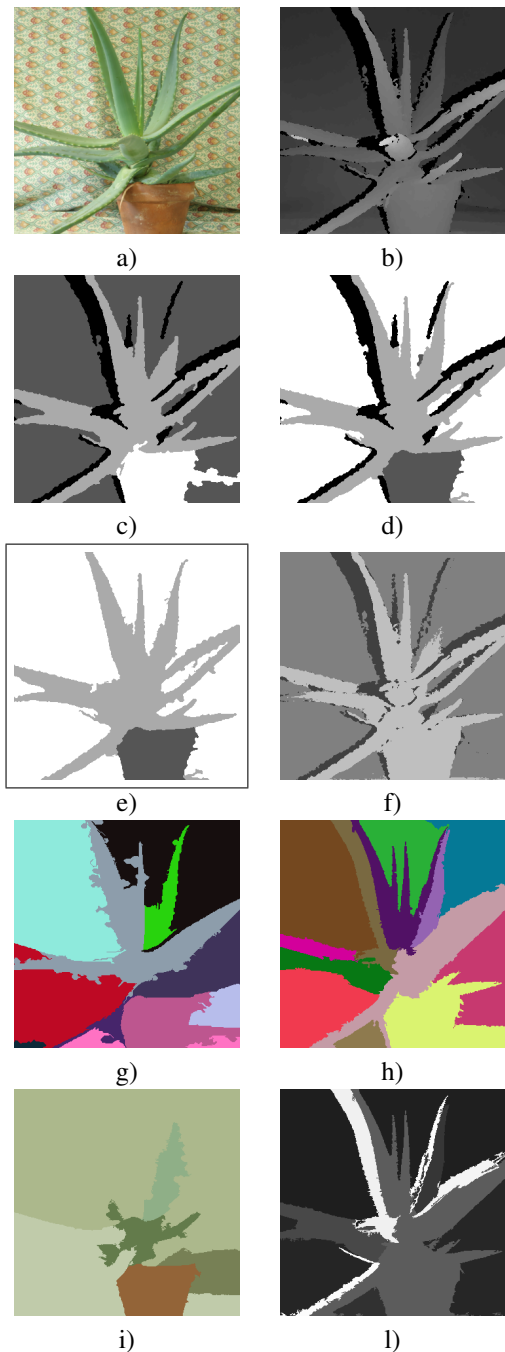
Fig. 11. Segmentation of the *aloe* scene from the Middlebury dataset: a) color image; b) corresponding disparity map (disparity values have been stretched in order to improve the readability of the printed picture); c) segmentation based on the proposed method exploiting geometry and one of the color views; d) segmentation based on the proposed method exploiting both color views and geometry as described in Section V; e) segmentation based on the proposed method exploiting both color views and geometry and also the occlusion handling scheme of Appendix A; f) segmentation performed by [17] that jointly exploits color and depth data; g) segmentation performed by [1] on the basis of color information only; h) segmentation performed by [1] on the basis of depth information only; i) segmentation performed by [27] on the basis of color information only; l) segmentation performed by [27] on the basis of depth information only.
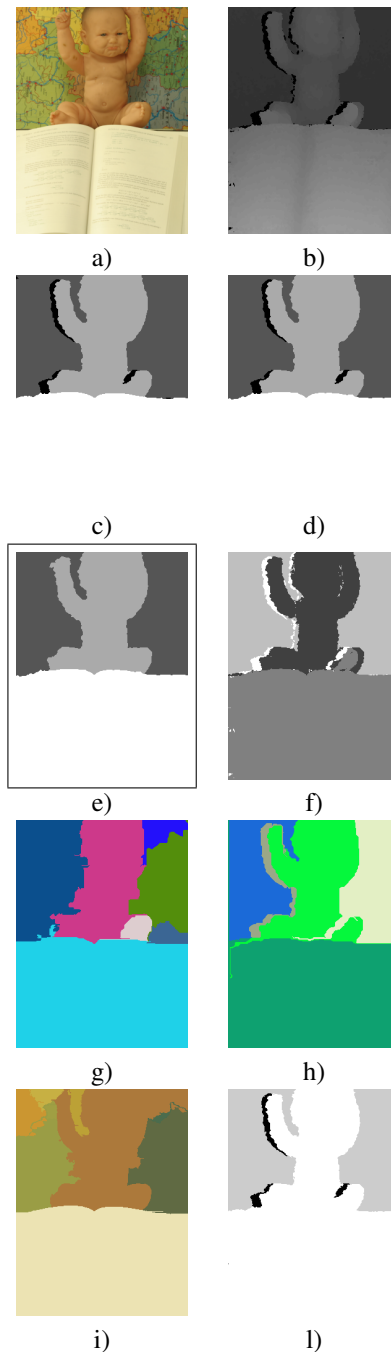
Fig. 12. Segmentation of the *baby2* scene from the Middlebury dataset: a) color image; b) corresponding disparity map (disparity values have been stretched in order to improve the readability of the printed picture); c) segmentation based on the proposed method exploiting geometry and only one of the color views; d) segmentation based on the proposed method exploiting both color views and geometry as described in Section V; e) segmentation based on the proposed method exploiting both color views and geometry and also the occlusion handling scheme of Appendix A; f) segmentation performed by [17] that jointly exploits color and depth data; g) segmentation performed by [1] on the basis of color information only; h) segmentation performed by [1] on the basis of depth information only; i) segmentation performed by [27] on the basis of color information only; l) segmentation performed by [27] on the basis of depth information only.

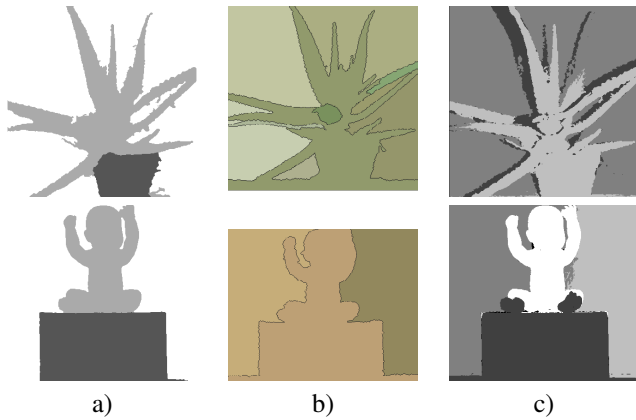Fig. 13.   Comparison of different segmentation methods based on the joint use of depth and color information on the *aloe* scene (*first row*) and on the *baby1* scene (*second row*): a) Proposed method; b) Calderero and Marques [9]; c) Bleiweiss and Werman [17].
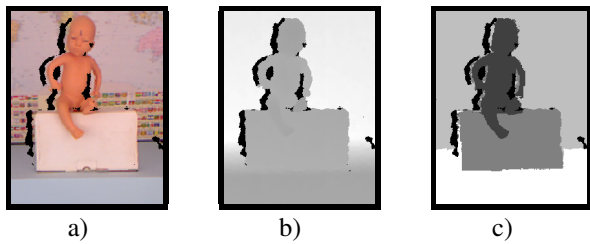

Fig. 14.   Segmentation of the *baby* scene acquired with a Kinect sensor: a) color image, b) depth image, c) segmented image.
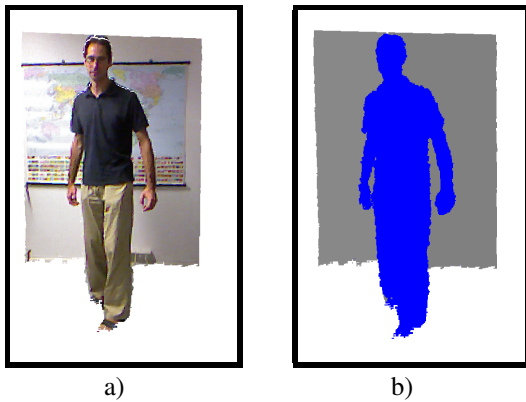

Fig. 15.   Segmentation of a person scene acquired with a Kinect sensor: a) point cloud acquired by the Kinect sensor, b) segmentation of the point cloud.
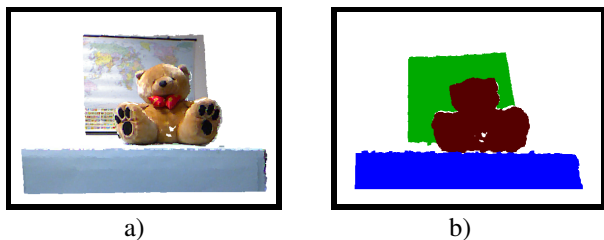

Fig. 16.   Segmentation of a teddy bear acquired with a Kinect sensor: a) point cloud acquired by the Kinect sensor, b) segmentation of the point cloud.

D. Results from Photosynth data

The acquisition systems of sections VI-A and VI-B are classical tools capable to acquire dense representations of both geometry and color of a scene in terms of an image and the corresponding depth-map. An unstructured 3D scene reconstruction tool like Microsoft Photosynth [8] is rather attractive not only because it is a free tool but also because it just requires to shoot a number of uncalibrated standard pictures of the scene. Photosynth can now be used even on mobile phones and is probably the only way today available for obtaining 3D data by mobile phones. The major limitation of Photosynth is that it is only able to provide a sparse representation of the scene geometry and color since the geometry is estimated only for salient features-point. Color information can be associated to such salient points. The main characteristic of a salient region is that it is markedly different from the rest of the scene. Therefore, grouping a set of salient points means grouping points that by construction and assumption are significantly different from each other. This characteristic of the acquisition system is by itself rather problematic. Another challenge for the segmentation is given by the sparsity of the obtained point cloud. Another important characteristic of the data is that the estimated scene geometry is defined up to an arbitrary scale factor. We tested our algorithm on the scene of Fig. 17, obtained by Photosynth. Fig. 17b shows the resulting segmentation (each color in the image corresponds to a scene segment). In light of the complexity of the point-cloud, and of the difficulties inherent to this type of data as observed above the results can be considered remarkably good.
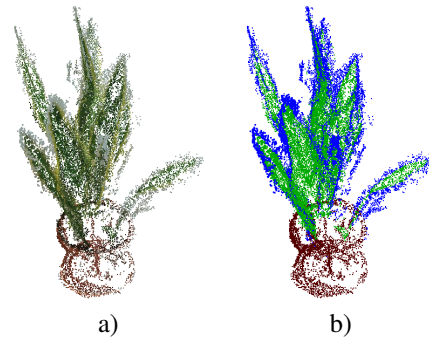
Fig. 17.   Segmentation of the scene *plant* acquired with Photosynth [8]: a) acquired scene, b) scene segmented by the proposed method jointly exploiting geometry and color. *(Best viewed in colors)*

VII.   COMPARISON OF THE DIFFERENT CONSIDERED IMAGING SYSTEMS FOR SCENE SEGMENTATION PURPOSES

As shown in the experimental results the proposed segmentation scheme can be applied to the data coming from different 3D acquisition systems. Two interesting questions that may arise at this point concern how the segmentation accuracy depends on the employed acquisition system and which is the best imaging system for segmentation purposes. In order to give a first answer to these questions a set of different scenes is acquired with 3 different imaging systems, i.e. the trinocular system described in Section VI-A, a Kinect camera and a stereo vision system exploiting the algorithm of [25]. The

acquired data are segmented exploiting the method proposed in this paper and Fig. 18 shows the obtained segmentations. Each of the five rows of the Fig. 18 corresponds to a different scene (shown in the first column), while each of the last three columns corresponds to a different acquisition system. It is clear that the trinocular setup (column b) gives the best results. This is mostly due to two reasons: firstly there are not occluded areas due to the fact that the ToF camera does not suffer from this issue, secondly the depth data are more accurate than the data produced by the other acquisition devices. Note in particular that edge localization is more precise than the ones of the other devices. Unfortunately it is also the most expensive of the three systems. In spite the Kinect is a much cheaper solution, it can be effectively exploited for joint color and depth segmentation. Even from the data of this cheap device it is possible to recognize all the main objects in the framed scenes (as shown by the images in column c). Probably the biggest limit of the Kinect data is the edge localization. It suffers both from the edge artifacts typical of the depth data acquired by the Kinect and from the limited accuracy of the calibration between the color and the depth camera. Note how we used the internal calibration provided by the Kinect that is not as precise as the one we performed for the trinocular setup. The higher spatial resolution of the Kinect with respect to that of ToF cameras is of little use for segmentation purposes because of its poor edge localization. Stereo vision (column d) gives the worse results mostly because of the artifacts in the provided depth data and of the missing depth samples due to occlusions. Even though the filling algorithm presented in Appendix A allows to assign the samples without a depth value to one of the segmented regions, the segmentation performances in these areas are reduced. This is an issue also in the case of Kinect, but the number of samples without a depth value is much smaller in this case than in the case of stereo vision systems. Artifacts in the computed depth maps due to uniformly textured regions also affects the segmentation, in particular on the background of the considered scenes. The results of stereo systems shown in column d are also not so good as the ones of Section VI-B. This is due to the fact that the used stereo vision algorithm (but it is a problem common to many stereo techniques) performs very well on heavily textured scenes built *ad-hoc* for stereo vision testing, e.g. the ones of the Middlebury dataset, but not as well with real scenes. However stereo setups are also very inexpensive and do not require active lighting. They can also be used for the acquisition of large-scale and outdoor scenes while both the Kinect and the ToF camera can only measure distances up to a few meters and essentially cannot work outdoor since they are heavily affected by sunlight. As summarized by Table I, each of the considered acquisition systems has its own advantages and disadvantages and the choice of the proper setup should be done on the basis of the target application.

VIII. CONCLUSIONS

Recent technology advancements make possible the acquisition of geometry and color data in many ways ranging from active sensors of various kind and price to free software tools

	Trinocular Setup (ToF + cameras)	Microsoft Kinect	Stereo vision
Edge localization	Good	Poor	Poor
Resolution	Low	Medium	High
Missing depth values	Very few	A few	Yes
Outdoor scenes	No	No	Yes
Cost	High	Low	Low

TABLE I
COMPARISON OF THE DIFFERENT ACQUISITION SETUPS.

like Microsoft's Photosynth (available also on mobile devices). These current practical possibilities motivate casting scene segmentation as a sensor fusion problem that combines color and geometry. The application scenario opened by the rationale of this paper features several points of interest worth explicit mention.

A significant contribution of this paper is the introduction of a novel scene segmentation technique fusing both geometry and color information that outperforms scene segmentation based on color only or geometry only. The proposed segmentation method adopts an original 6-dimensional representation of the scene fusing geometry and color in a way meaningful for clustering.

The clustering algorithm is an essential ingredient of the proposed segmentation technique and the normalized cuts spectral clustering algorithm was selected for its outstanding performance in this application. Indeed experimental results prove that for the considered scene segmentation problem normalized cut spectral clustering outperforms other clustering methods such as k-means and mean-shift. Furthermore the Nyström method has been applied in order to reduce memory and CPU requirements.

An interesting contribution of this paper is a novel unsupervised metric for the assessment of scene segmentation quality used for the automatic selection of a weight balancing the mutual relevance of geometry versus color.

Another intriguing point made by this paper concerns the segmentation results obtained from stereo data. This may be regarded as the advantage brought to segmentation in going to the next level of complexity with respect the the single image case, i.e., the usage of two images. Segmentation brings a new perspective to the evaluation of stereo algorithms, i.e., their effectiveness for scene segmentation rather than their 3D reconstruction accuracy.

The capability to handle different 3D data representations from different acquisition sensors represents a major conceptual and practical advantage. Indeed it is shown that the fusion of geometry and color within the proposed technique is always effective no matter what the data types are and how they were acquired. This is practically rather relevant in front of the many 3D acquisition tools currently available. In particular data produced by very low cost tools like Microsoft Kinect or unstructured scene reconstruction algorithms freely available on the web such as Photosynth have a special interest. Both Kinect data and unstructured scene reconstruction algorithms are of great interest for the computer vision community. The availability of such data is a rather recent event and their segmentation is a new topic rich of possible developments and

| Color image | Segmentation from ToF and cameras data | Segmentation from Kinect data | Segmentation from stereo vision data |

a) b) c) d)

Fig. 18. Segmentation of some samples scenes exploiting depth data coming from different acquisition systems: a) color image of the scene; b) segmentation from the ToF camera data and the color images provided by the trinocular setup; c) segmentation from the Kinect data; d) Segmentation from the stereo vision data.

in Section VII a discussion of the performance and issues of the various acquisition devices for segmentation purposes has been presented. The experimental results of this paper, available at *http://lttm.dei.unipd.it/downloads/segmentation*, offer a performance benchmark for other algorithms considering scene segmentation on the basis of the fusion of geometry and color information.

Once proven the usefulness of fusing of geometry and color for scene segmentation it is fair to say that in this connection there are several issues worth further investigation. Among them an intriguing research direction concerns the relationship between stereo algorithms and segmentation, with special focus on the joint solution of the segmentation and disparity estimation. The critical role of clustering within the considered approach makes sensible to monitor the improvement opportunities offered by alternate new clustering approaches. It is finally important to note that the proposed method is very efficient and its real-time implementation will be carefully considered in the near future.

APPENDIX
HANDLING MISSING DEPTH VALUES

The proposed segmentation technique assumes that both geometry and color information are simultaneously available for all the scene points. In practice this may not always be the case because of:
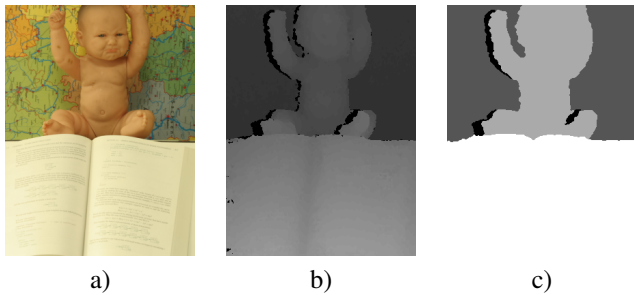
Fig. 19. Color image (a), disparity map (b) and segmentation (c) of a scene in which for some points only color information is present.

1) Occlusions, especially in either the case of stereo vision or in the Kinect case;
2) ToF camera or Kinect range camera missing measurements due to low scene reflectivity of to saturation
3) Stereo correspondences discarded by left-right check or confidence-based rejection approaches.

In particular it may happen that, for scenes represented by a depth map and a color image, some points may only be associated to color and not to depth information. Let us consider the case of a scene with a set of points for which both geometry and color information are available. These points can be segmented by the proposed method together with a set of points that have not been segmented (for which only color information is available). It is desirable to have the possibility to propagate the segmentation to the points characterized by the presence of color information only (which have not been segmented). An example of this situation is the scene reported in Figure A, that shows a color image, a depth map characterized by missing depth values (black point) and a segmented image (also with missing values). Let us denote with $\mathcal{P}_S$ the set of pixels characterized by a segmentation value, and with $\bar{\mathcal{P}}_S$ the set of pixels without a segmentation value. The following procedure allows to propagate the segmentation information to the pixels of $\bar{\mathcal{P}}_S$:

1) Identify all the pixels $p_i \in \bar{\mathcal{P}}_S$ which have at least one of the 4-neighbors $p_i^n, n = 1, ..., 4$ in $\mathcal{P}_S$. The set of such points is denoted with $\mathcal{B}_S$. Each pixel $p_i \in \mathcal{B}_S$ has at least one neighbor and no more than four neighbors for which segmentation information is available. The segments of the neighbors p_n of p_i are denoted as S_i^n.
2) For each $p_i \in \mathcal{B}_S$ compute the cost of assigning it to each neighboring segments S_i^n as

$$\mathcal{C}(p_i, S_i^n) = \Gamma_L(p_i, p_n) + \Gamma_a(p_i, p_n) + \Gamma_b(p_i, p_n)$$

in which

$$\Gamma_L = L(p_i) - median\{L(p_j), p_j \in S_i^n \cap W^n\}$$

$$\Gamma_a = a(p_i) - median\{a(p_j), p_j \in S_i^n \cap W^n\}$$

$$\Gamma_b = b(p_i) - median\{b(p_j), p_j \in S_i^n \cap W^n\}$$

being W^n a segmented window centered around p_n and L, a, b the CIELab channels of the color image.

3) Compute the association $[p^*, S^*]$ which minimizes the assignment cost

$$[p^*, S^*] = \arg \min_{[p_i, S_i^n]} \mathcal{C}(p_i, S_i^n)$$

4) Propagate the segmentation value of S^* to p^*
5) If $\mathcal{B}_S$ is not empty, restart from 1).

This procedure allows to iteratively propagate the segmentation of the pixels in $\mathcal{P}_S$ characterized by the minimum color difference with respect to the local color median of the neighboring segments, for which a segmentation value is available. Figure 20 shows some examples of the results of this method.

ACKNOWLEDGMENT

REFERENCES

[1] P.F. Felzenszwalb and D.P. Huttenlocher, "Efficient graph-based image segmentation," *International Journal of Computer Vision*, 2004.
[2] D. Comaniciu and P. Meer, "Mean shift: a robust approach toward feature space analysis," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2002.
[3] J. Shi and J. Malik, "Normalized cuts and image segmentation," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2000.
[4] Richard Szeliski, *Computer Vision: Algorithms and Applications*, Springer, New York, 2010.
[5] D. Scharstein and R. Szeliski, "A taxonomy and evaluation of dense two-frame stereo correspondence algorithms," *International Journal of Computer Vision*, 2001.
[6] "Mesa Imaging," http://www.mesa-imaging.ch.
[7] "Microsft Kinect," http://www.xbox.com/en-US/kinect.
[8] "Microsoft Photosynth," http://photosynth.net/.
[9] F. Calderero and F. Marques, "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.
[10] M. Harville, G. Gordon, and J. Woodfill, "Foreground segmentation using adaptive mixture models in color and depth," in *IEEE Workshop on Detection and Recognition of Events in Video, 2001. Proceedings.*, 2001.
[11] J. Leens, S. Pirard, O. Barnich, M. Van Droogenbroeck, and J.M. Wagner, "Combining color, depth, and motion for video segmentation," in *Computer Vision Systems*. 2009.
[12] L. Wang, C. Zhang, R. Yang, and C. Zhang, "Tofcut: Towards robust real-time foreground extraction using a time-of-flight camera," in *3DPVT*, 2010.
[13] V. Kolmogorov, A. Criminisi, A. Blake, G. Cross, and C. Rother, "Bi-layer segmentation of binocular stereo video," in *Computer Vision and Pattern Recognition, 2005. CVPR 2005. IEEE Computer Society Conference on*, june 2005, vol. 2, p. 1186 vol. 2.
[14] L. Ladicky, P. Sturgess, C. Russell, S. Sengupta, Y. Bastanlar, W. Clocksin, and P. Torr, "Joint optimisation for object class segmentation and dense stereo reconstruction," in *Proceedings of the British Machine Vision Conference*, 2010.
[15] M. Bleyer, C. Rother, P. Kohli, D. Scharstein, and S. Sinha, "Object stereo- joint stereo matching and object segmentation," in *Computer Vision and Pattern Recognition, 2011. CVPR 2011. IEEE Computer Society Conference on*, June 2011.
[16] V. Mezaris, I. Kompatsiaris, and M.G. Strintzis, "Still image segmentation tools for object-based multimedia applications," *International Journal of Pattern Recognition and Artificial Intelligence*, vol. 18, no. 4, pp. 701–725, June 2004.
[17] A. Bleiweiss and M. Werman, "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.

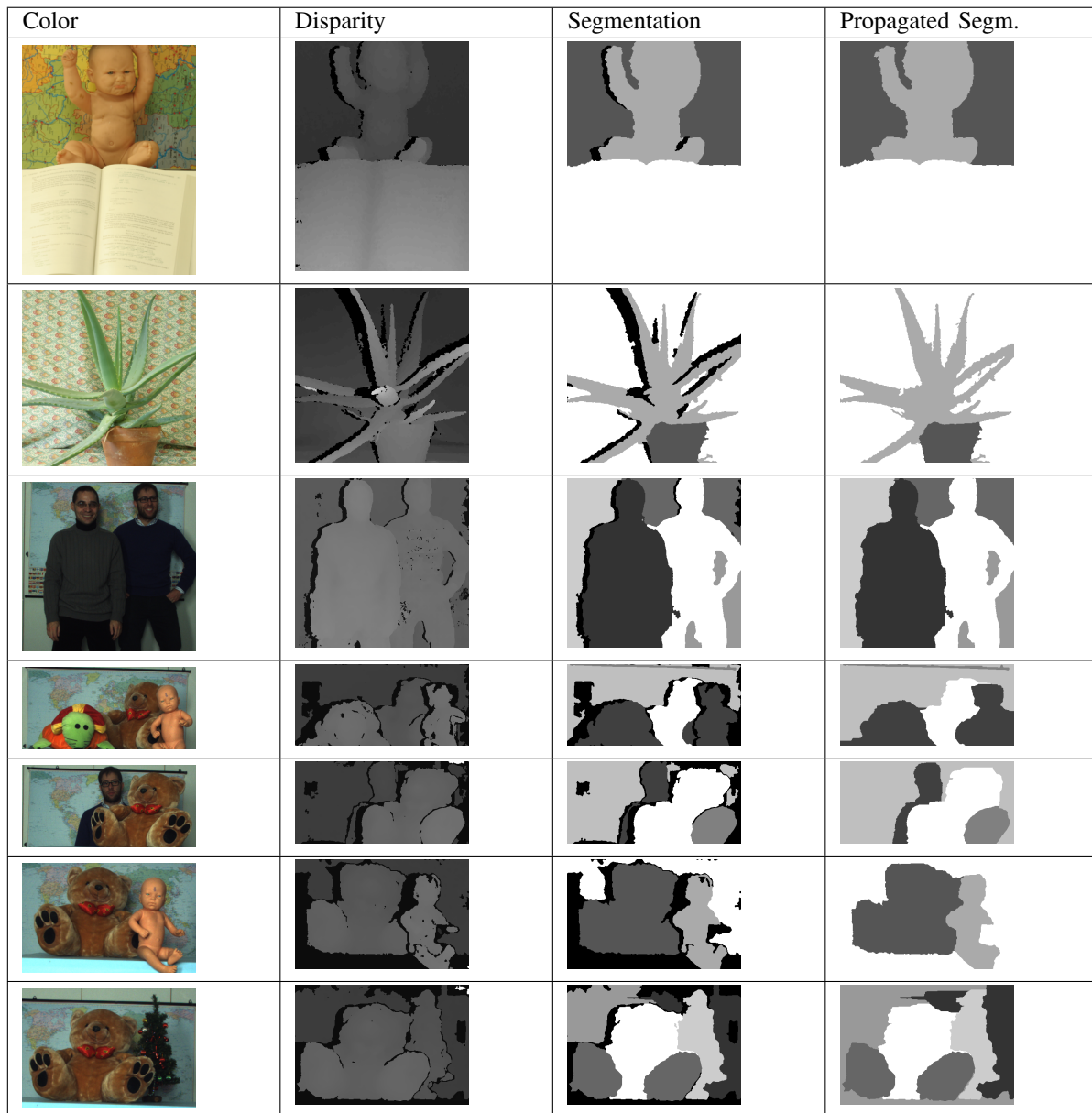| Color	Disparity	Segmentation	Propagated Segm.

Fig. 20. Examples of segmentation propagation on different scenes. First column: color images; second column: stereo disparity maps; third column: segmentation produced by the proposed method; fourth column: segmentation propagated by the procedure of Appendix A.

[18] Marcus Wallenberg, Michael Felsberg, Per-Erik Forssén, and Babette Dellen, "Channel coding for joint colour and depth segmentation," in *Proceedings of Pattern Recognition 33rd DAGM Symposium*, Frankfurt/Main, Germany, September 2011, vol. 6835 of *Lecture Notes in Computer Science*, pp. 306–315, SpringerLink.

[19] C. Dal Mutto, P. Zanuttigh, and G.M. Cortelazzo, "Scene segmentation assisted by stereo vision," in *Proceedings of 3DIMPVT 2011*, Hangzhou, China, May 2011.

[20] C. Fowlkes, S. Belongie, F. Chung, and J. Malik, "Spectral grouping using the nyström method," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2004.

[21] G.Sanguinetti, J. Laidler, and N. Lawrence, "A probabilistic approach to spectral clustering: Using kl divergence to find good clusters," in *Pascal Statistics and Optimization of Clustering Workshop*, London, UK, Jul 2005.

[22] H. Zhang, J.E. Fritts, and S.A. Goldman, "Image segmentation evaluation: A survey of unsupervised methods," *CVIU*, 2008.

[23] C. Rosenberger and K. Chehdi, "Genetic fusion: application to multi-components image segmentation," in *ICASSP*, 2000.

[24] "Middlebury stereo vision dataset," http://vision.middlebury.edu/stereo/.

[25] Heiko Hirschmuller, "Stereo processing by semi-global matching and mutual information," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 30, pp. 328–341, February 2008.

[26] C. Dal Mutto, P. Zanuttigh, and G.M. Cortelazzo, "A probabilistic approach to tof and stereo data fusion," in *3DPVT*, Paris, France, May 2010.

[27] "Edison," coewww.rutgers.edu/riul/research/code/EDISON.

[28] "OpenCV," http://opencv.willowgarage.com/wiki/.

[29] "OpenNI," http://www.openni.org/.

Carlo Dal Mutto Carlo Dal Mutto is currently a third-year Ph.D. student at University of Padova under the supervision of Prof. G.M. Cortelazzo. In 2007 he received a B.Sc. ("Laurea Triennale") in Information Engineering in 2009 a M.Sc. ("Laurea Specialistica") in Communication Engineering from the same University (Summa cum Laude). His research focuses on acquisition and processing of color and 3D geometry data. He co-authored several papers on this topic, as well as two book chapters and a book.

Pietro Zanuttigh Pietro Zanuttigh was born in 1978. He graduated in Computer Engineering at the University of Padova (Italy) in 2003 and got the Ph.D. degree from the same university in 2007. He has been visiting researcher at the University of New South Wales (Australia) in 2006 and 2010. In 2007 he became an assistant professor at the University of Padova. Now he works in the multimedia technology and telecommunications group and his research activity focuses on 3D data processing. His main research interests are the transmission and remote visualization of scalably compressed 3D scenes, compression of depth maps and multi-view videos and dynamic 3D scenes acquisition with multiple sensors. He is the co-auhtor of a book and of several publications on international journals and conference proceedings. He also holds a patent on interactive visualization on mobile phones.

Guido M. Cortelazzo Guido M. Cortelazzo holds a "Laurea in Ingegneria Elettronica" degree from the University of Padova (1976), and a Master Degree and a Philosophy Doctor Degree both in Electrical Engineering from the University of Illinois at Urbana-Champaign (1980 and 1984, respectively). From 1983 to 1986 he worked with M/A-COM Linkabit, a satellite communication company of San Diego, California. In 1986 he joined the Department of Information Engineering (DEI) of the University of Padova where he is now full professor. In 1998 he was visiting associate with the Califorrnia Institute of Technology (Pasadena, CA). He was active in the organization of several special sessions and meetings. A founding member of the steering committee of the International Symposium on 3D Data Processing Visualization and Transmission (3DPVT) the proceedings of which he co-edited in in 2002 with Concettina Guerra. 3DPVT has now merged to the 3DIMPVT Conference. The current professional interests of Guido M. Cortelazzo concern the automatic construction of 3D models of still and dynamic scenes and their progressive transmission. He is the author of about seventy journal papers, co-author of two books and co-editor of two special issues. He has been the (co-)recipient of Italian, and international research and industry grants and holds a few industrial patents.