Interactive Multi-label Segmentation of RGB-D Images

Julia Diebold, Nikolaus Demmel, Caner Hazırbaş,
Michael Moeller, and Daniel Cremers

Technical University of Munich, Germany
{julia.diebold,c.hazirbas,michael.moeller,cremers}@tum.de,
nikolaus@nikolaus-demmel.de

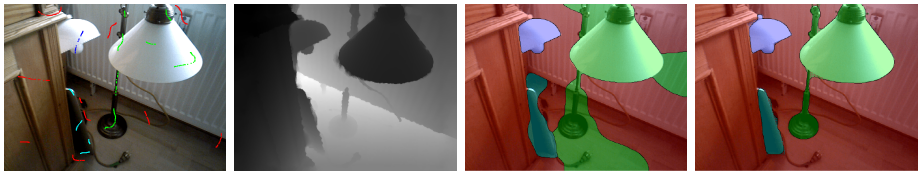
Abstract. We propose a novel interactive multi-label RGB-D image segmentation method by extending spatially varying color distributions [14] to additionally utilize depth information in two different ways. On the one hand, we consider the depth image as an additional data channel. On the other hand, we extend the idea of spatially varying color distributions in a plane to volumetrically varying color distributions in 3D. Furthermore, we improve the data fidelity term by locally adapting the influence of nearby scribbles around each pixel. Our approach is implemented for parallel hardware and evaluated on a novel interactive RGB-D image segmentation benchmark with pixel-accurate ground truth. We show that depth information leads to considerably more precise segmentation results. At the same time significantly less user scribbles are required for obtaining the same segmentation accuracy as without using depth clues.

Keywords: Multi-label Segmentation, RGB-D Images, Interactive Segmentation, Spatially Varying Color Distributions, Total Variation

1 Introduction

A major challenge in computer vision is to compute accurate *image segmentations*, that is, the accurate partitioning of images into meaningful regions. Possible fields of application cover medical imaging, image editing software, object tracking and scene reconstructions. The definition of meaningful regions, however, highly depends on what application the segmentation is needed for. Thus, fully *automatic* image segmentation methods are usually tailored to very specific tasks and try to extract particular objects the methods have learned some prior knowledge about, *e.g.* indoor [5,7,23] or facade [8,26] segmentation.

One way to develop general purpose segmentation tools are *interactive* segmentation methods, where the user indicates the object to be segmented. In this work, we consider user inputs by so called *scribbles*, *i.e.* separate points the user indicated to belong to a certain object. Alternative interactive user input modalities not considered in this work include bounding boxes [12,20,27] or contours [1,3]. Due to their adaptability, interactive segmentation methods have recently attracted a lot of interest. Recent works focus



a) Color image with scribbles | b) Depth image | c) RGB segmentation [14] | d) Proposed RGB-D segmentation

Fig. 1. Depth information significantly improves the segmentation result.

on foreground/background [3,11,12,27,28] as well as on multi-region segmentation [15,21,22], and mostly consider RGB images as input data.

Despite the segmentation constraints given by the user, accurate segmentation remains a challenging task. Extensive studies have led to significant improvements of segmentation quality in recent years [11,28]. Nevertheless, modern approaches often still fail for complex scenes, where objects with similar colors and difficult lightning conditions appear. Moreover, a good segmentation often requires a rather large number of scribbles.

Considering the recent increase and availability of depth-sensing cameras such as the Kinect, we investigate the segmentation of RGB-D images to overcome some of the aforementioned problems. We will mainly focus on the distinction of objects based on color and depth information. While some research has been done on extending interactive segmentation methods to medical imaging data (*e.g.* [4,13]), very little work has been done on the interactive segmentation of RGB-D images. The only other approach we found which explicitly addresses interactive multi-label RGB-D segmentation is the method by Shao *et al.* [22] on the semantic modeling of indoor scenes. Although this method is related to our approach in the sense that it also formulates the segmentation of RGB-D images as a variational approach, it is tailored towards the application of furniture segmentation. Therefore, the algorithm can use learned a-priori information about the objects to be segmented and the user interaction merely serves as a possible correction step for the first automatic segmentation step.

We investigate the application of interactive RGB-D multi-label segmentation and enhance the recently published work by Nieuwenhuis and Cremers [14] by including depth information. We propose to extend the spatially varying color distributions [14] to RGB-D images in two different ways: a) We consider the depth as an additional color channel. b) We enhance the spatially varying color distributions from varying in a plane to be volumetrically varying. Figure 1 d) shows an example of the improvements that can be obtained by taking the depth into account. In the above example, it is almost impossible to distinguish the radiator from the lamp (Figure 1 c), because both objects have a similar color and are close in the image plane. The proposed volumetrically varying color distributions (Figure 1 d) incorporate the depth information, which yields much more distinct color descriptions and thus better segmentation results.

2 Variational Interactive Segmentation of RGB Images

2.1 Multi-label Segmentation

Let $I : \Omega \to \mathbb{R}^d$ denote the input image, mapping the image domain $\Omega \subset \mathbb{R}^2$ to $\mathbb{R}^d$, with $d = 3$ for an RGB and $d = 4$ for an RGB-D image. Image segmentation denotes the task of partitioning the image plane into a set of n pairwise disjoint regions Ω_i: $\Omega = \bigcup_{i=1}^{n} \Omega_i$. The regions Ω_i can be computed by minimizing the following energy:

$$E(\Omega_1, \ldots, \Omega_n) = \frac{1}{2} \sum_{i=1}^{n} \mathrm{Per}_g(\Omega_i) + \lambda \sum_{i=1}^{n} \int_{\Omega_i} f_i(x)\, dx, \tag{1}$$

where $\mathrm{Per}_g(\Omega_i)$ denotes the perimeter of each set Ω_i, which is minimized in order to favor segments of shorter boundaries. These boundaries are measured with either an Euclidean or an edge-dependent metric defined by the non-negative function $g : \Omega \to \mathbb{R}^+$. For example, $g(x) = \exp\big(-\gamma\,|\nabla I(x)|\big)$, favors the coincidence of object border and image edges. f_i denotes the appearance model and λ is a weighting parameter which regulates the influence of the second term.

2.2 Convex Relaxation

The usual strategy to address the nonconvex energy minimization problem arising from (1) is to use convex relaxation: One represents the disjoint regions Ω_i by indicator functions v_i, with $v_i(x) = 1$ if $x \in \Omega_i$ and $v_i(x) = 0$, else. Since the v_i are indicator functions, we can make use of the fact that the total variation (TV) of an indicator function is nothing but the perimeter of the set described by the functions. Hence, we can reformulate Equation (1) as

$$E(v_1, \ldots, v_n) = \frac{1}{2} \sum_{i=1}^{n} \int_{\Omega} g(x)\, |Dv_i(x)|\, dx + \lambda \sum_{i=1}^{n} \int_{\Omega} v_i(x)\, f_i(x)\, dx, \tag{2}$$

where Dv_i is the distributional derivative of v_i. Determining the optimal segmentation can be stated as solving the minimization problem

$$(\tilde{v}_1, \ldots, \tilde{v}_n) = \arg\min_{v_i} E(v_1, \ldots, v_n) \quad \text{s.t. } v_i(x) \in \{0, 1\},\ \sum_i v_i(x) = 1,\ \forall x. \tag{3}$$

Since the nonconvexity of the above problem comes from the integer constraint $v_i(x) \in \{0, 1\}$, a standard convex relaxation is to replace this constraint by $v_i(x) \in [0, 1]$.

The key to obtain a good segmentation method based on (3) is to determine f_i that lead to a good data fidelity term guiding the segmentation. In the following, we recall the computation of the f_i motivated by maximum a-posteriori probability (MAP) estimates as suggested in [14].

2.3 Likelihood Estimation based on User Scribbles

Let $I : \Omega \rightarrow \mathbb{R}^3$ and $u : \Omega \rightarrow \{1, .., n\}$ be a labeling, such that $\Omega_i = \{x \in \Omega \mid u(x) = i\}$. Motivated by a MAP estimate Nieuwenhuis and Cremers [14] proposed to compute the $f_i(x)$ as the negative log-likelihood of the estimated probability distribution:

$$f_i(x) = -\log \hat{\mathcal{P}}(I(x), x \mid u(x) = i). \tag{4}$$

The expression $\mathcal{P}(I(x), x \mid u(x) = i)$ denotes the joint probability density of observing a color value $I(x)$ at location x given that x is part of region Ω_i. Based on the ideas of kernel based probability estimates (cf. [25] for an overview), it can be estimated from the user scribbles by

$$\hat{\mathcal{P}}(I(x), x \mid u(x) = i) = \frac{1}{m_i} \sum_{j=1}^{m_i} k \begin{pmatrix} x - x_{ij} \\ I(x) - I(x_{ij}) \end{pmatrix}, \tag{5}$$

where $\{x_{ij},\ j = 1, .., m_i\}$ is the set of user scribbles for region i, and k a suitable kernel function. The probability estimate in (5) only has to be computed for pixels $x \notin \{x_{ij},\ j = 1, .., m_i\}$. For $x \in \{x_{ij}\}$ we keep the label given by the user scribble. We discuss the particular choice of k in more detail below.

3 From RGB to RGB-D Images

3.1 Pre-Processing the Depth Image

Prior to using the depth image, two pre-processing steps have to be conducted. One has to decide how to handle missing depth information and which range of the depth values to use.

Depth inpainting Depth cameras such as the Kinect provide metric depth values in addition to color. However, depth information is usually not available for all pixels. We fill in the missing depths in a preprocessing step with an inpainting technique provided in the toolbox of Silberman *et al.* [24]. The implementation is a slight adaptation of the colorization proposed by Levin *et al.* [10]. For an example see Figure 2 b,c).

a) Color b) Depth c) Filled depth d) Ground truth e) Trimap

Fig. 2. Exemplary RGB-D input, scribbles, ground truth and trimap labeling. a) Color image with scribbles, b,c) (filled) normalized depth image, d) ground truth segmentation, e) trimap used for measuring the pixel labeling accuracy in a band surrounding the object boundaries [9]. The evaluation region is colored gray and was generated by taking a 25 pixel band surrounding the boundaries of the objects.

Normalization For Kinect-like cameras the value range of the depth values $z(x)$ in meters is roughly $[0.5, 6]$. To be independent of physical units, for each image we normalize the actual depth range to $[0, 1]$. Similarly, to be independent of the image resolution, we normalize Ω to $[0, 1]^2$.

3.2 Depth as an Additional Color Channel

Following Nieuwenhuis and Cremers [14], we use Gaussian kernels with different bandwidths to model the joint probability distribution (5). Incorporating the depth image as an additional data channel leads to the following distribution for $\hat{\mathcal{P}}\left(I\left(x\right), D\left(x\right), x \mid u\left(x\right) = i\right)$:

$$\frac{1}{m_i} \sum_{j=1}^{m_i} \underbrace{k_{\rho_i(x)}\left(x - x_{ij}\right)}_{\text{distance kernel}} \underbrace{k_{\sigma}\left(I\left(x\right) - I\left(x_{ij}\right)\right)}_{\text{color kernel}} \underbrace{k_{\tau}\left(D\left(x\right) - D\left(x_{ij}\right)\right)}_{\text{depth kernel}}, \tag{6}$$

with the bandwidths ρ_i, σ and τ. Due to the comparability of their values, the color channels R, G and B are modeled by the same bandwidth σ. A separate fixed bandwidth τ is used for the depth channel. The bandwidth of the spatial kernel ρ_i on the other hand is chosen proportional to the distance to the closest scribble of label i [14]: $\rho_i\left(x\right) = \alpha \min_{j=1,..,m_i} |x - x_{ij}|$. Analogous ideas arise in generalized k-nearest neighbor probability density estimates (cf. [25]), where a similar dependence of the kernel variance on the distance to the nearest samples is considered. Note that although a single multivariate Gaussian could be used for modeling the probability density, this would require an estimation of the covariance matrix, *e.g.* on a training data set.

3.3 Active Scribbles

To overcome the fact that scribble positions are generally not distributed uniformly throughout the image, we furthermore introduce the idea of *active scribbles*. A general problem of (5) and (6) is, that the estimated distribution is heavily influenced by the total number m_i of scribbles in class i. This leads to the undesirable behavior that adding many scribbles in one particular region of the image actually reduces the likelihood of far-away-points belonging to the same class. To avoid this, we determine for each pixel x and each class i all scribbles x_{ij}, $j = 1, .., m_i$ that are within a radius of three times the distance to the closest scribble. We call these scribbles active. The distance is computed in 2D or 3D depending on the availability of depth. If less than 80% of the scribbles are active, we compute the probability density (6) of the active and inactive scribbles separately and combine the two by $0.8 \cdot \hat{\mathcal{P}}_a\left(I\left(x\right), D\left(x\right),\ x \mid u\left(x\right) = i\right) \ + \ 0.2 \cdot \hat{\mathcal{P}}_p\left(I\left(x\right), D\left(x\right),\ x \mid u\left(x\right) = i\right)$, where the subscripts a and p denote the estimates based on the active and passive (inactive) scribbles respectively. Otherwise we use all scribbles to compute (6).

3.4 Revised Pixel Distance by Depth Values

The main contribution of [14] was to introduce spatially varying color distributions, *i.e.* using a distance kernel in (6). The motivation for this kernel was

that while an object often looks locally similar, its typical color distribution may change with the position that is considered. With the help of the distance kernel, scribbles that are close to the current position gain more influence than those that are far away. A limitation of this approach for RGB images is that the true 3D geometry cannot be represented: Due to the lack of depth information in RGB data, the method considered in [14] is a projection of a volumetrically varying color distribution onto the image plane.

The depth image allows us to compute color distributions that truly depend on the objects' position in space and thus lead to more distinct color descriptions. For illustration purposes Figure 3 a,b) considers a 2d color image. Pixels close in the image are not necessarily close in the 3-dimensional space as we can see in Figure 3 c,d). To better reflect the real object geometry, we therefore improve the computation of the distance kernel $k_{\rho_i(x)}(x - x_{ij})$ by using the depth information.

Back-Projection To perform the distance computation in the 3-dimensional space, the 3-dimensional pixel position X has to be computed from the pixel coordinates x and the normalized depth value $D(x)$. While a physically correct back-projection would be perspective and therefore dependent on the intrinsic parameters of the camera, we found a planar back-projection that simply uses $D(x)$ as the third coordinate to be the better choice for two reasons: It not only compared favorable in our numerical experiments but also is easier to compute as it does not require the knowledge of camera parameters.

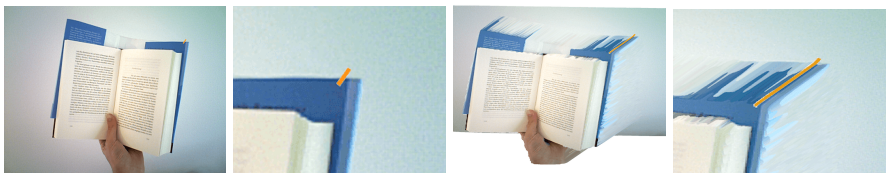
Thus, in Equation (6), instead of evaluating the distance kernel $k_{\rho_i(x)}(x - x_{ij})$ at $x \in [0,1]^2$ we incorporate the depth as a third dimension and evaluate the distance kernel at $X = \big(x, D(x)\big)^\top$:

$$k_{\rho_i(X)}(X - X_{ij}) \quad \text{with} \quad \rho_i(X) = \alpha \min_{j=1,..,m_i} |X - X_{ij}|\,. \tag{7}$$

3.5 The Novel Formulation

Combining the ideas of Sections 3.2 and 3.4 we propose the following appearance model for RGB-D images

$$f_i(x) = -\log \hat{\mathcal{P}}\left(I(x), D(x), x \mid u(x) = i\right), \tag{8}$$



a) Color image (2d dist. in orange) b) Zoom of a) c) Back-Projection (3D dist. in orange) d) Zoom of c)

Fig. 3. Recovering the scene geometry with depth information. Illustration of the distance in the 2-dimensional color image compared to the real distance in the 3-dimensional space. The incorporation of depth information in the computation of the distance kernel allows to capture the real object geometry.

with

$$\hat{\mathcal{P}}\left(I\left(x\right), D\left(x\right), x \mid u\left(x\right) = i\right) = \frac{1}{m_i} \sum_{j=1}^{m_i} \underbrace{k_{\rho_i(X)}\left(X - X_{ij}\right)}_{\text{distance kernel}} \underbrace{k_\sigma\left(I\left(x\right) - I\left(x_{ij}\right)\right)}_{\text{color kernel}} \underbrace{k_\tau\left(D\left(x\right) - D\left(x_{ij}\right)\right)}_{\text{depth kernel}}. \quad (9)$$

Here $\{x_{ij},\ j = 1, .., m_i\}$ denotes the set of user scribbles for region i, X the three-dimensional position $X = \left(x, D\left(x\right)\right)^\top \in [0,1]^3$ and $\rho_i\left(X\right) = \alpha \min_j |X - X_{ij}|$, σ and τ denote the kernel bandwidths. The effect of both ways of incorporating depth information into the segmentation framework will be studied in detail in the experimental results (Section 5).

Finally, let us mention that the two ways the depth information is utilized in the above model is actually equivalent to using a single Gaussian kernel for the depth information. The single kernel would have a bandwidth that contains a spatially varying part as well as a constant part. Since the latter is rather difficult to interpret, we decided to motivate the proposed approach from two different perspectives. Thus, the depth information appears in our proposed model twice.

4 Implementation

To find the globally optimal solution to this relaxed convex optimization problem, we employ the primal-dual algorithm published in [6,16,17]. It consists of updating a primal and a dual variable in an alternating fashion. The update of each variable decouples for each pixel such that the approach can easily be parallelized and implemented on graphics hardware.

Since we are solving the relaxed problem, there may be pixels x at which $v_i(x)$ take on intermediate values between 0 and 1, *i.e.* we may end up with non-binary solutions. In our numerical experiments, we observed that the computed relaxed solutions $v_i(x) < 0.001$ or $v_i(x) > 0.999$ for 98% of all pixels $x \in \Omega$ and $i = 1, \ldots, n$. In order to obtain a binary solution, we assign each pixel x to the label L with maximum value after optimizing the relaxed problem.

5 Experimental Results

In this section we demonstrate the effectiveness of all proposed RGB-D image adaptions in several numerical experiments. The numerical study is divided into three parts: First, we discuss the data used for the numerical experiments. Second, we compare RGB to RGB-D segmentation and demonstrate that the segmentation accuracy is improved by the additional depth information. Alternatively, less user scribbles are required by the RGB-D segmentation method to obtain the same accuracy as an RGB method. In a third part we demonstrate that not just one but all of our proposed extensions improve the segmentation results in the sense that the addition of each component individually yields an improvement in segmentation quality.

5.1 Experimental Data

As extensively discussed in [21], not every benchmark is suited for testing interactive segmentation. Typical interactive segmentation benchmarks (such as the iCoseg benchmark [2] for foreground/background segmentation or the Icg-Bench dataset [21] for multi-label segmentation) do not provide RGB-D data, and hence could not be used for our experiments. Popular RGB-D benchmarks such as the NYUv2 dataset [24] are not suitable for interactive segmentation since the scenes are typically composed of very many small objects.

Therefore, we chose the Object Segmentation Database (OSD) [18] as the starting point for numerical experiments. We, however, found that the images contained in the OSD were not challenging enough. They all have the same background and same colors. Furthermore, the objects are relatively small compared to the image size and the given depth. Hence, we decided to use 12 images from the OSD along with 16 images we captured ourselves using an RGB-D sensor. The new images were intentionally taken with challenging color and texture similarities between different objects. For all 28 images, we fixed the scribbles and manually created an accurate ground truth labeling.[1] An example is given in Figure 2.

5.2 Depth Information is Crucial

We use the aforementioned image data set to compare our algorithm (using $\lambda = 10$, $\gamma = 5$, $\alpha = 1000$, $\sigma = 0.05$, $\tau = 0.2$ for all experiments) to the results obtained by Santner *et al.* [21] and Nieuwenhuis and Cremers [14]. Due to the similarity of our approach with the one in [14], we used the same parameters (without the additional depth information) for the implementation of [14]. For the framework in [21], we took the parameters that were mentioned to be the best general purpose choice.[2] Using exactly the same scribbles (see Fig. 4 a) for all three interactive segmentation methods, we obtain the results shown in Figure 4 c-e).

We have to mention that our comparison is unfair in the sense that the other methods do not make use of the depth information. However, as we could not find other suitable interactive RGB-D segmentation methods, we chose this comparison to illustrate the importance of depth information for image segmentation tasks.

For images with challenging color and lighting conditions, like *e.g.* in Figure 4 first row, an RGB based method can hardly find the correct segmentation of the scene. The depth channel, however, provides essential information regarding the

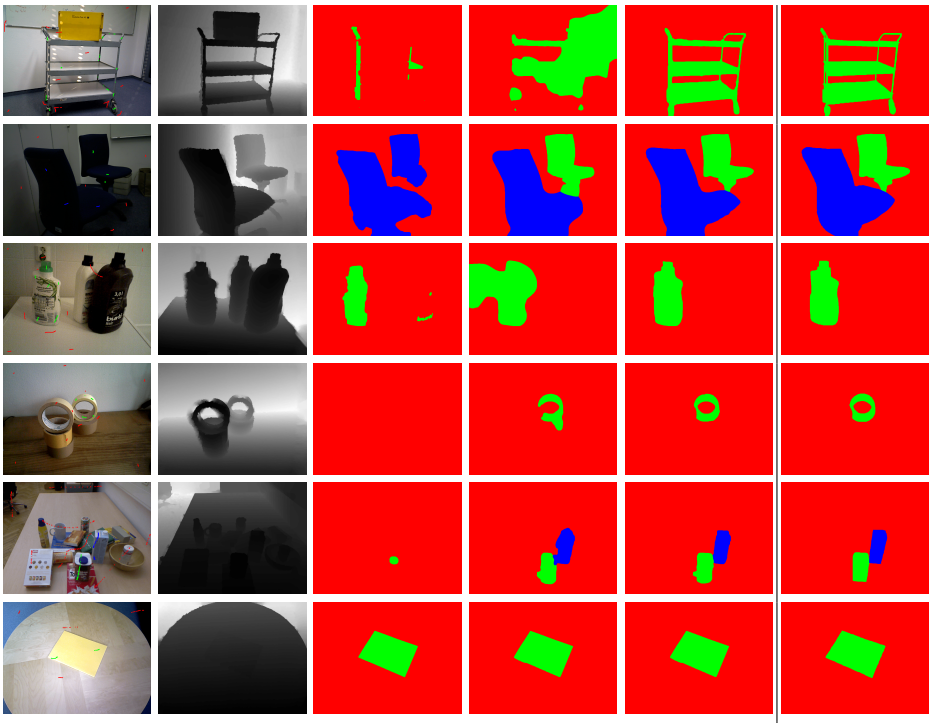
[1] Our framework as well as the RGB-D images, the scribbles and the ground truth labelings are publicly available on our website: `vision.in.tum.de/data/software`

[2] CIELab color space, LBP features with a patch size of 16 and a radius of 3, Random Forests with 200 trees, 750 iterations, $\lambda = 0.2$ and $\alpha = 15$.

spatial relation between the pixels in the image. Thus, the incorporation of the depth image results in significant improvements of the segmentation quality over the RGB based methods. For images in which the depth channel does not provide additional information, such as the image in the bottom row of Figure 4, the proposed method yields the same result as [14], as expected.

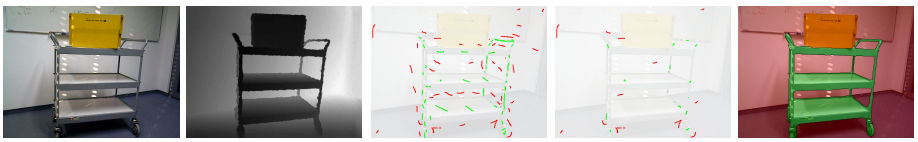
Another benefit which comes from the additional depth information is that less user scribbles are required compared to an RGB based segmentation method. Figure 5 exemplary illustrates this behavior: Running our method with the scribbles shown in Figure 5 d) we obtain the segmentation result in e). We incrementally add scribbles in order to obtain a similar result with [14], see Figure 5 c). Due to the strong color similarity between foreground and background, the RGB based method requires significantly more user scribbles to obtain a similar result.

Finally, let us mention that the runtime of our method is – same as [14,21] – around one second on 640×480 images. The major computational time is needed for the optimization which is independent of our proposed components.



a) Color image with scribbles | b) Depth image | c) Santner *et al.* [21] | d) Nieuwenhuis, Cremers [14] | e) Proposed | f) Ground Truth

Fig. 4. Depth information improves the segmentation. The scribbled RGB-D input data is shown in the columns a,b). Columns c-e) compare the proposed RGB-D segmentation to the RGB segmentations of [21,14].



a) Color image b) Depth image c) Scribbles needed with [14] d) Proposed scribbles e) Segmentation result

Fig. 5. Depth yields less user input. The depth information provides valuable information which reduces the required user input. To retrieve a similar result as in e), the user needs to place more scribbles with [14] c) than with the proposed volumetrically varying color distributions d).

5.3 Impact of the Proposed Components

To quantify the results on our benchmark dataset, we compute the dice-scores suggested in [14,21] on the regular ground truth as well as on a trimap surrounding the object boundaries: Let S denote the labeling obtained for an image, GT the respective ground truth labeling. Then the dice-score is computed as

$$dice\,(S) = \frac{1}{n}\sum_{i=1}^{n}\frac{2\,|GT_i \cap S_i|}{|GT_i| + |S_i|}, \tag{10}$$

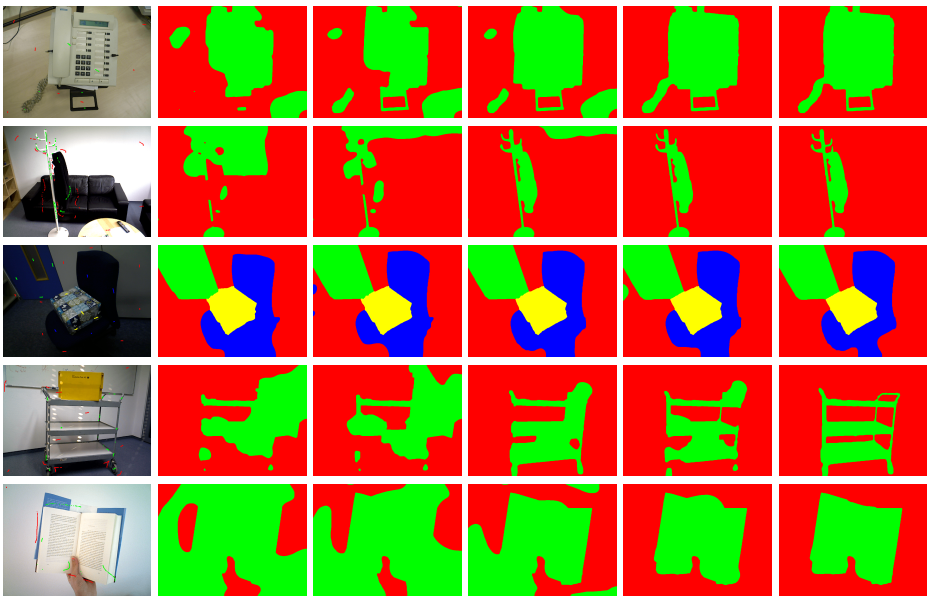
where the index i denotes the label i and $|\cdot|$ the area of a segment.

Table 1 shows the dice scores averaged over all images obtained by [21], [14], and a step by step addition of the proposed algorithm components. The scores not only give us the possibility of quantitatively evaluating the results obtained by the different methods, but also allow to study the effect of each of the proposed extensions of [14], namely using active scribbles, using depth as an additional data channel and using depth for the 3D distance.

It is interesting to see that the usage of active scribbles – which does not require any depth information – already improves the score on the regular ground truth by 0.7% and on the trimap by 2.2%. Additionally including the depth for either the 3D distance or as an additional color channel again improves the score. The best results are obtained when combining all three components as we can see in the last row of Table 1. To visualize the results from Table 1, Figure 6 shows a qualitative comparison of the different components. As we can see, in each column, from left to right the result improves.

Input	Segmentation method	Reg. GT	Trimap
RGB	Santner *et al.* [21]	72.56	67.69
RGB	Nieuwenhuis and Cremers [14] (Figure 6 b)	87.09	86.17
RGB	[14] with proposed AS (2D) (Figure 6 c)	87.79	88.40
RGB-D	[14] + AS (3D) + Depth for 3D distance (Figure 6 d)	91.51	93.63
RGB-D	[14] + AS (3D) + Depth as color channel (Figure 6 e)	92.93	93.07
RGB-D	Combination of all proposed components (Figure 6 f)	**93.70**	**94.84**

Table 1. The proposed method outperforms the previous ones. The dice scores are compared by means of the regular ground truth segmentations as well as the trimap width of 25 (compare Figure 2). The usage of active scribbles is abbreviated by 'AS'.



a) Input with scribbles | b) Result of [14] | c) [14] + AS | d) [14] + AS + 3D dist. | e) [14] + AS + depth as col. ch. | f) Full model

Fig. 6. Each of the proposed components improves the segmentation. We compare the segmentations obtained with different components of the proposed method. The usage of active scribbles is abbreviated by 'AS'. f) The combination of all components: Active scribbles, depth for 3D distance and depth as an additional color channel leads to the best result.

6 Conclusion

We proposed a powerful extension of the spatially varying color distributions [14]. Our contributions include the idea of active scribbles to overcome the problem of non-uniformly distributed user scribbles. Furthermore, we improve the estimation of the data fidelity term by incorporating the depth as an additional color channel as well as using it to construct volumetrically varying color distributions in 3D. We have demonstrated that each of the proposed components contributes separately and improves the segmentation results. Due to the additional depth information, reliable segmentations are obtained with significantly less user input. For future work, one could also use a regularization that takes into account the geometry of the 3D surface as suggested in [19].

References

1. P. Arbelaez, M. Maire, C. C. Fowlkes, and J. Malik. From contours to regions: An empirical evaluation. In *CVPR*, 2009.
2. D. Batra, A. Kowdle, D. Parikh, J. Luo, and T. Chen. iCoseg: Interactive co-segmentation with intelligent scribble guidance. In *CVPR*, 2010.

3. A. Blake, C. Rother, M. Brown, P. Perez, and P. Torr. Interactive image segmentation using an adaptive GMMRF model. In *ECCV*, 2004.
4. Y. Y. Boykov and M.-P. Jolly. Interactive graph cuts for optimal boundary & region segmentation of objects in ND images. In *ICCV*, 2001.
5. C. Couprie, C. Farabet, L. Najman, and Y. LeCun. Indoor semantic segmentation using depth information. In *ICLR*, 2013.
6. E. Esser, X. Zhang, and T. F. Chan. A general framework for a class of first order primal-dual algorithms for convex optimization in imaging science. *SIIMS*, 2010.
7. A. Hermans, G. Floros, and B. Leibe. Dense 3D Semantic Mapping of Indoor Scenes from RGB-D Images. In *ICRA*, 2014.
8. J. Hernandez and B. Marcotegui. Morphological segmentation of building facade images. In *ICIP*, 2009.
9. P. Kohli, L. Ladicky, and P. H. S. Torr. Robust higher order potentials for enforcing label consistency. *IJCV*, 2009.
10. A. Levin, D. Lischinski, and Y. Weiss. Colorization using optimization. In *TOG*, 2004.
11. Y. Li, J. Sun, C.-K. Tang, and H.-Y. Shum. Lazy snapping. *TOG*, 2004.
12. D. Liu, K. Pulli, L. G. Shapiro, and Y. Xiong. Fast interactive image segmentation by discriminative clustering. In *MCMC*, 2010.
13. H. Lombaert, Y. Sun, L. Grady, and C. Xu. A multilevel banded graph cuts method for fast image segmentation. In *ICCV*, 2005.
14. C. Nieuwenhuis and D. Cremers. Spatially varying color distributions for interactive multilabel segmentation. *PAMI*, 2013.
15. C. Nieuwenhuis, S. Hawe, M. Kleinsteuber, and D. Cremers. Co-Sparse Textural Similarity for Interactive Segmentation. In *ECCV*, 2014.
16. T. Pock and A. Chambolle. Diagonal preconditioning for first order primal-dual algorithms in convex optimization. In *ICCV*, 2011.
17. T. Pock, D. Cremers, H. Bischof, and A. Chambolle. An Algorithm for Minimizing the Mumford-Shah Functional. In *ICCV*, 2009.
18. A. Richtsfeld, T. Morwald, J. Prankl, M. Zillich, and M. Vincze. Segmentation of unknown objects in indoor environments. In *IROS*, 2012.
19. G. Rosman, A. M. Bronstein, M. M. Bronstein, X.-C. Tai, and R. Kimmel. Group-valued regularization for analysis of articulated motion. In *ECCV*, 2012.
20. C. Rother, V. Kolmogorov, and A. Blake. Grabcut: Interactive foreground extraction using iterated graph cuts. In *TOG*, 2004.
21. J. Santner, T. Pock, and H. Bischof. Interactive multi-label segmentation. In *ACCV*, 2011.
22. T. Shao, W. Xu, K. Zhou, J. Wang, D. Li, and B. Guo. An Interactive Approach to Semantic Modeling of Indoor Scenes with an RGBD Camera. *TOG*, 2012.
23. N. Silberman and R. Fergus. Indoor scene segmentation using a structured light sensor. In *ICCV*, 2011.
24. N. Silberman, D. Hoiem, P. Kohli, and R. Fergus. Indoor segmentation and support inference from RGBD images. In *ECCV*, 2012.
25. B. Silverman. *Density estimation for statistics and data analysis.* Chapman and Hall Ltd, 1986.
26. O. Teboul, L. Simon, P. Koutsourakis, and N. Paragios. Segmentation of building facades using procedural shape priors. In *CVPR*, 2010.
27. S. Vicente, V. Kolmogorov, and C. Rother. Joint optimization of segmentation and appearance models. In *ICCV*, 2009.
28. J. Wang. Discriminative gaussian mixtures for interactive image segmentation. In *ICASSP*, 2007.