Object Extraction in RGBD Images

Fabian Prada, Leandro Cruz, Luiz Velho
Instituto de Matematica Pura e Aplicada, VISGRAF Lab. Rio de Janeiro, Brasil.
www.impa.br/∼{faprada, lcruz, lvelho}

***Abstract*—In this work we discuss the integration of three depth based methods to the object extraction problem in RGBD images. Each of these three methods provides an insight in the connectedness, proximity, and planarity of the scene. Depth and color are combined in a GraphCut framework.**

I. Introduction

Object extraction can be stated as follows: *Given certain user interaction, identify automatically the set of pixels that belongs to the referenced object.* Classical approaches to object extraction are based only in color data [1]. The aim of our work is to show how the use of structured depth information provided by Kinect lead to reinforce this task.

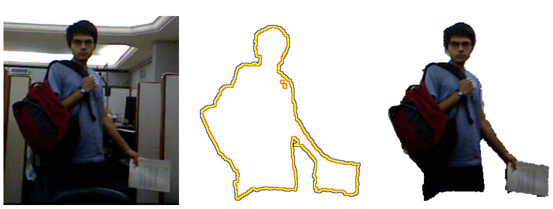
II. Related Work

In most works on RGBD images, the way depth data is used to gain structural information is the result of deterministic and/or heuristic ideas. By deterministic ideas, we mean methods that are able to identify if certain input (e.g., depth map) satisfies some property. Related work in this context is Kahler [2] in recognition of planar patches. On the other hand, heuristic methods are based on reasonable *priors* about the image content. In Silberman [3] the authors classify objects in 13 categories: bed, wall, table, etc., defining location and depth *priors*. Our work proposes three *priors* which are completely different to them. Both methods exemplify the importance of having depth/location assumptions to get robust results.

III. Structural information from depth data

A. Depth Connected Component (DCC)

We say two arbitrary pixels are *depth connected*, if there is a path joining them, where depth difference between consecutive pixels is smaller than a fixed threshold. *Depth connectedness* induces a partition of the image pixels, and we call each of these components a *DCC* (Fig. 1). Whenever the object is not adjacent to any other element of the scene, the information gained throughout *DCC* is almost a perfect object extraction. In this work *DCC* is implemented as *Breadth First Search*.



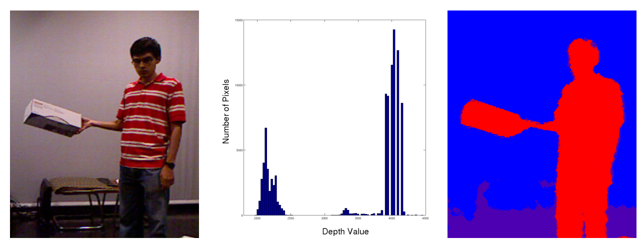
(a) (b) (c)

Fig. 1. a) *Rectangular Selection*. (b) *Contour Graph*. (c) *DCC*.

B. Depth Range Estimation (DRE)

The objective of this technique is identifying an interval in the depth histogram that closely contains the object (Fig. 2). From the set of pixels in the *Selection Center*, we applied *K-means* to identify n clusters of depth (we took $n = 5$). Clusters associated to the object are identified as follows:

1) Sort the n clusters from closest to farthest. Let $d_{(1,2)}, d_{(2,3)}, ..., d_{(n-1,n)}$ be the distance between the center of consecutive clusters.
2) Label the closest cluster as FG.
3) If $d_{(1,2)} < d_t$ the second cluster is labeled FG. Otherwise finish.
4) For $i \in \{3, 4 \ldots n\}$, if $d_{(i-1,i)} < m_t d_{(i-2,i-1)}$, the i-th cluster is labeled FG. Otherwise finish.



(a) (b) (c)

Fig. 2. (a) *Rectangular Selection*. (b) Depth Histogram. (c) *Depth Range Estimation*

C. Background Planar Surfaces (BPS)

DCC and *DRE* have a problem in common: when there is adjacency between object and background we cannot gain any distinction using just raw depth data. To overcome this difficulty we require more structured information about the image. The environment around the object in indoor spaces is frequently conformed of planar surfaces. This is the case whenever the object is over the floor, in front of a wall, etc. Then, *planar pixels* near *Selection Border* are highly probable background pixels. This idea is captured in the next procedure:

1) Take a sample of pixels from the *Selection Border* and store them in a queue. These pixels will be called *component's generators* (Fig. 3).
2) Fix tolerance parameters e_n for normal and e_z for color.
3) Pick the first pixel p of the queue and identify the *depth connected* set of pixels around p satisfying $||n_p - n_x|| < e_n$ and $||z_p - z_x|| < e_z$. Call this set of pixels $PN(p)$, the planar neighborhood of p.
4) In order to confirm $PN(p)$ as a valid plane the following two conditions must hold:

- $PN(p)$ is greater than 5% of *Rectangular Selection*.
- At most 25% of $PN(p)$ belongs to the *Selection Center*.

5) If the previous conditions holds, $PN(p)$ is a *planar component*, and we label its pixels as *planar pixels*.
6) Repeat the process using the next *component's generator* in the queue.

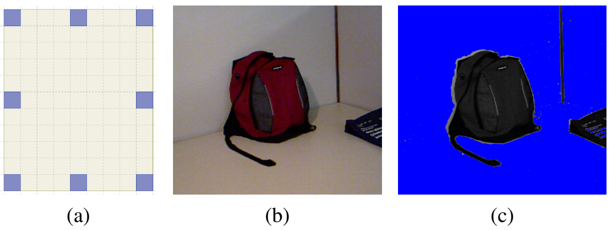


Fig. 3. (a) *Component's Generators*. (b) *Rectangular Selection*. (c) *Planar Components*

IV. Integration with GraphCut

A. Seeds

• *Initial Seeds:* Let m and M be the minimal and maximal depth value in the set of pixels belonging to the *FG* clusters, $\mu = \frac{m+M}{2}$ and $\sigma = M - m$. *FG* seeds are all non *planar pixels* in $I_{FG} = [\mu - \sigma, \mu + \sigma]$. *BG* seeds are all *planar pixels* and pixels in $I_{BG} = (-\infty, \mu - 2\sigma] \cup [\mu + 2\sigma, \infty)$.

• *Seeds Refinement:* Select the largest subset of *FG* seeds that belong to a same *DCC*, and reaffirm these pixels as *FG* seeds. The pixels initially identified as *FG* seeds but not belonging to the chosen *DCC* are then moved to the set of *BG* seeds.

B. Energy Function

The energy function in its general formulation is given by,

$$E(x, z, d) = \alpha_C U_C + \gamma_C V_C + \alpha_D U_D + \gamma_D V_D$$

Here U_C and V_C are the *Color Terms* defined in [1]. U_D is the *Depth Data Term* which is set according to the structural information obtained through the three depth based methods. V_D is the *Depth Smoothness Term* and it is selected to avoid cuts in regions of almost constant depth. $\alpha_C, \gamma_C, \alpha_D$, and γ_D are control parameters.

V. Conclusions

From depth based method we can discover aspects of the scene geometry hardly attainable from color data. Within a GraphCut framework, they allowed design a more robust seeds selection, and define a more flexible Energy Function.

DCC led to satisfactory results in non-adjacency scenes, but this depended on having accurate depth data. *DRE* obtained good results in scenes with large depth ranges, however, in scenes where object had a non-uniform depth range, *DRE* partially identified the clusters belonging the object. *BPS* gave robust results in almost all the experiments done. Very good results were obtained for objects surrounded of planar surfaces. In the non-adjacency case, *BPS* had some limitations but still contributed with valuable information.

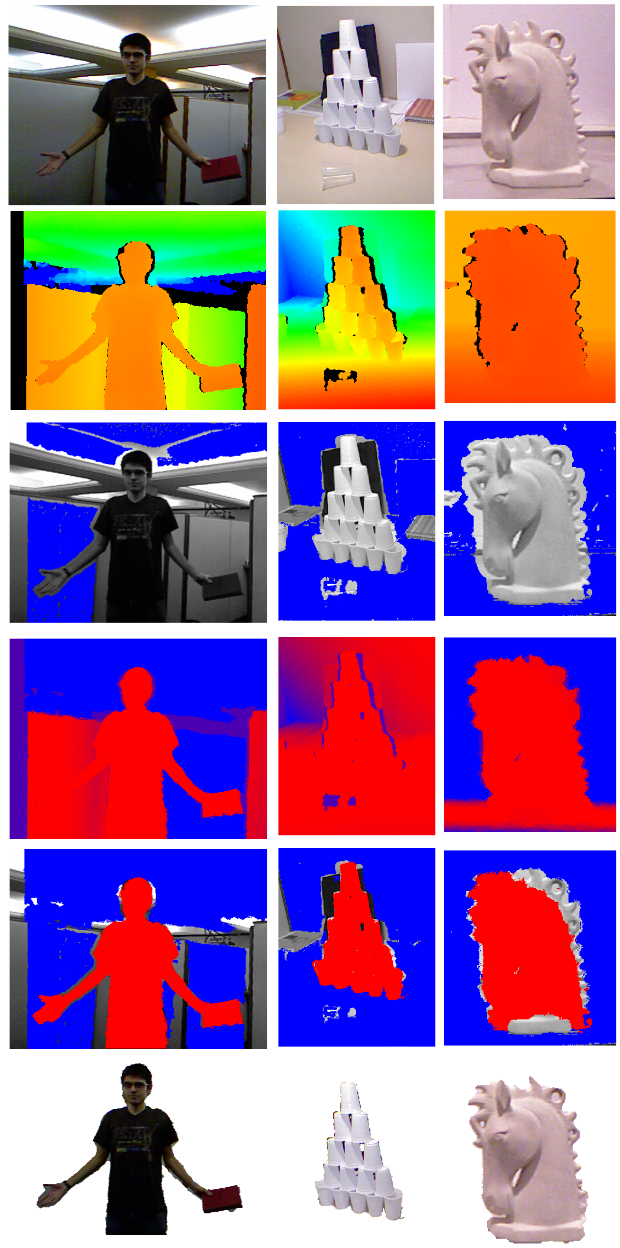


Fig. 4. In Top Down order: Rectangular Selection, Raw Depth, Depth Range Estimation, Background Planar Surfaces, Seeds and Final results.

Seeding process was robust and segmentation results were satisfactory (Fig. 4). Regarding to the Energy Function further work is required on parameter fitting. Some strategies that we consider worth for future research are learning the parameters from an overview of the scene structure, and allowing the parameters to change locally.

References

[1] C. Rother, V. Kolmogorov, and A. Blake, "Grabcut: interactive foreground extraction using iterated graph cuts," in *SIGGRAPH*, 2004.
[2] O. Kahler, E. Rodner, and J. Denzler, "On fusion of range and intensity information using graph-cut for planar patch segmentation," in *Intelligent Systems Technologies and Applications*, 2008.
[3] N. Silberman and R. Fergus, "Indoor scene segmentation using a structured light sensor," in *3DRR Workshop, ICCV*, 2011.