RGBD Camera Effects

Djalma Lucio, Leandro Cruz, Luiz Velho
Instituto de Matematica Pura e Aplicada - IMPA
VISGRAF Lab, Rio de Janeiro, Brasil
www.impa.br/∼{dlucio, lcruz, lvelho}

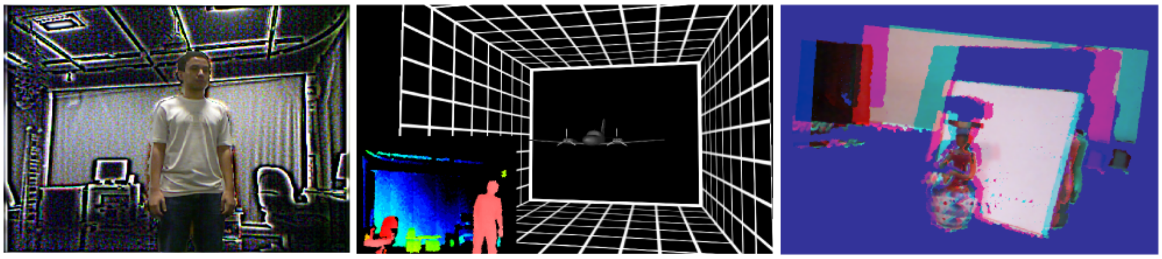

Fig. 1. RGBD camera effects: filtering (left), visualization with head tracking (middle) and stereo image creation (right).

***Abstract*—With the advent of RGBD cameras, like the Kinect sensor, we gain a powerful tool for image processing. The nature of the captured data (i.e. colored image and depth data) could be combined to realize tasks that before were very difficult. In this paper, we will discuss some camera effects done using RGBD images. These effects will be direct, such as image filtering and stereo image creation, or indirect, such as visualization using head tracking.**

***Keywords*-RGBD Image; Camera effects; Filtering; Head Tracking; Stereo Image**

I. Introduction

Most of image processing operates on color channels of the images. Nonetheless, there are others image attributes that can be used in processing. We can cite: depth, normal, luminance, etc. These attributes can carry informations that further allow various procedures with an image.

Nowadays, there are low cost and real time RGBD cameras, i.e. cameras that capture a colored image (RGB) and the respective depth data (D). An example of this kind of device is the Kinect sensor. It captures data with visual and geometric nature. This allows the implementation of some effects which were previously very difficult or impossible. The RGBD images enable to solve old problems with new approaches.

We will show some examples of these effects, which are applied in RGBD data, and we will show some simple approaches to implement them. We process the color attributes using informations obtained with the depth. The RGBD images were captured with Kinect in real time.

In this work, we will show four RGBD camera effects. The first is the RGB image filtering. The second is to create a stereo image from an RGBD image. The third is the implementation of head tracking. And finally, we can do a camera and objects tracking.

A. Related work

Since the Kinect launch, several research were done using RGBD images. Some of these were related with direct and indirect camera effects. Cruz et. al. [1] show a survey highlighting the challenges and applications provided by the Kinect sensor and RGBD images

Bai et. al. [2] introduced a method to select segments along a video. It produces good results on segmentation but this approach could be simplified using the depth data. Silberman et. al. [3] and Prada et. al. [4] show methods to RGBD image segmentation. Izadi et. al. [5] introduced a method to reconstruct a scene (captured with a handheld RGBD camera) that could be used to implement an object extraction.

Lefohn et. al. [6] introduced a method to implement the focus effect on modeled scene. This approach could be adapted to real scene captured with RGBD cameras.

Shotton et. al. [7] introduced a method to calculate a human skeleton using the depth data. With this skeleton it is possible to implement a head tracking to use on visualization. Besides this, Benko et. al. [8] introduced an immersive system with head tracking implemented using the Kinect depth data.

II. Filtering

This effect consists in filtering different image segments. Each segment refers to one or more objects in the scene. In this way, the filtering process could be divided in three steps: the first is the selection of elements that will define the segments; the second is the image segmentation; and finally, the filtering of each segment.

Figure 2 shows the pipeline of filtering. The first is the colored image and the second is the respective depth measured (coded in a colored image). With the depth data, the third image shows the two segments that the image was divided: the blue segment referred to the boy and the black one referred to the background. Finally, on right, is shown the result with only the boy in focus, and the background blurred.

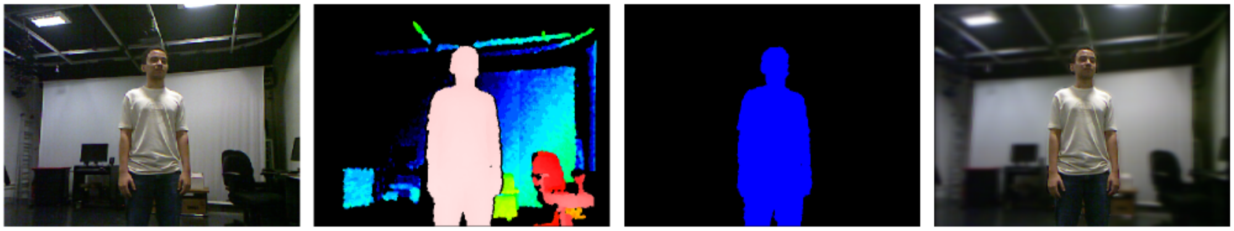

Fig. 2. Filtering Effect.

A. Elements Selection

A simple segmentation of the image consists in discriminate background and foreground. It could be done using a threshold on the depth data (the pixels whose the depth is less than the threshold belongs to foreground and the others belongs to background). It could be extended to create a depth range considered like foreground. In this case, we need a minimum and a maximum threshold.

These approaches are very easy. However, the result may not be good. For instance, parts of some objects could belong to background and foreground simultaneously.

Another possibility is to define as background the fixed objects in the scene. It is also a simple method, although it is more discriminative. To do it, we can get an initial frame with only the background elements. Thereafter, for each frame, we define as foreground the pixels which the depth is different than the background one (due to measurement errors, it is recommended to use a tolerance between the depth values).

A more explicit approach is to use object recognition methods. The RGBD data structure improves informations that help this method. This is more precise, although, this is a more complex approach.

We can use specific methods to select an element of a specific object class. In particular, Shotton et. al. [7] introduced a method to track human. This algorithm creates a skeleton of a person that is in front of the device. This method is nowadays available on several tools to handle the Kinect data. It is necessary others methods (specific methods) to track others objects in RGBD videos. Cruz et. al. [1] shows some works to object recognition.

B. Segmentation

Image segmentation is a hard problem of Image Processing and Computer Vision. Moreover, if we use an RGBD image, the depth makes it easy. We can obtain the segment boundary from large gradient variations on the depth data. However, the depth comes from Kinect has noise and holes. Hence, this approach produces coarse results. If it is necessary a more fitted segmentation we must use a another technique.

Prada et. al. [4] introduced an extension of GrabCut method to use the depth. The authors use the boundary of the depth segment (obtained in regions of great depth variations) to define the seeds of the background and the foreground. This method is initialized defining a window around of the object, even as in the traditional GrabCut method. If our object of interest is a person, we can define automatically the windows using the skeleton structure. This method produces satisfactory results.

The main difficult of segmentation using RGBD images from Kinect is the problems on the depth data. If we have a more robust geometric model, we can segment the image using only depth boundaries. Izadi et. al. [5] introduced a method to create a dense surface using a handheld Kinect camera. To get a better surface it is necessary move the camera. The model is represented by a cloud point, but it is possible obtain an implicit model from them. With the implicit model, we can define if the pixel belongs or not to the surface.

C. Filters

After segmentation, we can process each segment with this respective filter. For instance, to simulate a camera focus effect, we can blur the entire image except the element on focus. Figure 2 shows an example of this effect.

We can create other effects only defining different filters. It could be implemented in parallel. For this purpose, we can create a mask to define each segment and implement a shader that applies different filters in different segments. Figure 1 (left) shows other effect generated filtering the background and maintaining the boy, on foreground.

III. Head tracking

An immersive visualization system could be created using the head tracking operation. With the head localization, we can move the scene to exhibit the parts which the user would see if he were facing a real scene. This effect simulates holography. The scene transformations induce the sensation of being really in front of a 3D object (or scene). It could be used to simulate a window, where the scene is the outside landscape.

Different than filtering, the holography simulation uses the RGBD image indirectly. The RGBD images are used to interaction. To use this effect we need the model of the scene (the scene could be modeled or captured).

Using the skeleton, we can easily track a person head. First, we need identify the head. Then, we can create the camera effect defining the virtual camera on the position correspondent to the user head.

A drawback of this method is that this effect only works for one person. If other person look at the rendered scene will see a distorted image.

The Figure 3 shows several scenes rendered according to the head position, obtained with the depth data visualized on the bottom left box.

IV. Stereo Image Creation

Nowadays, 3D movies motivate the creation of stereo images. This process is commonly done using two cameras. Before capturing some adjustment are done to produce some effects such as focus, or improve the depth sensation. Beato [9] shows some technique to create stereo images.

The RGBD images could be used to help these tasks. Beside this, the geometry information could be directly used to produce a stereo image. The Figure 4 shows an example of a stereo image produced with the 3D model created with the depth data.

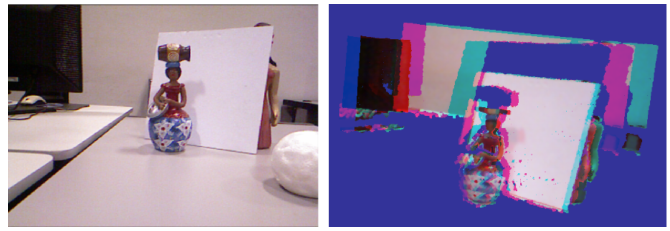

Fig. 4. Stereo Image.

V. Camera and Object tracking

The camera tracking operation was not completely exploited in this work. Moreover, Izadi et. al. [5] show a method to track a camera with the depth data. The authors use a coarse-to-fine Iterative Closest Point to calculate the 6 degrees of freedom of the camera to tracking it.

Effects like focus were done using the camera and the object position, in each frame. We can use the depth data to implement this effect. Using the camera tracking, we can also track fixed objects and to keep it in focus (or out focus), even if the camera is moving. Others filtering effects are analogous.

To filter a moving object we can track it too. In the SubSection II-A we cite some object extraction methods. Some of these use (or could be extended to use) temporal information to track an object over time.

VI. Results and Discussion

In this work we show some possibilities of camera effects that could be done using RGBD images. They could be direct, like filtering, or indirect, like visualization with head tracking.

On one hand, a Kinect drawback is the noise and holes on the depth data. Beside this, as RGB image as the depth data are low resolution (640x480), although the RGB image could be captured on 1280x1024 resolution. On the other hand, it is a low cost device that obtains both colored image and depth data in real time.

An approach to compensate the low resolution and noise on the colored images is to use another colored camera to capture better quality colored images or videos. To merge the colored image captured with another camera and the Kinect depth data it is necessary to calibrate both cameras positions.

The depth data also is noisy and low resolution. For this reason, if we need good fitted object segmentation we cannot use directly it. If we use a more robust scene reconstruction process then the segmentation result will be better. We can attain a more robust scene model using the several depth data, comes from Kinect, or using a better device.

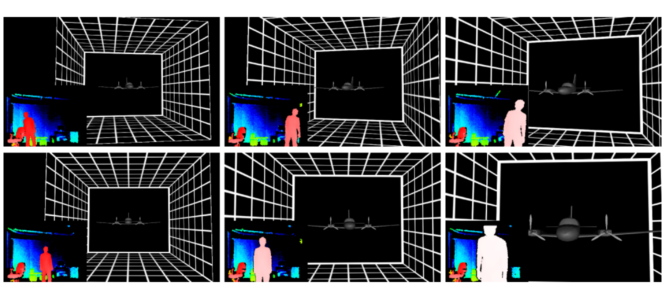

Fig. 3. Visualization using Head Tracking.

Furthermore, the skeleton detection is also unstable because the depth problems. Because of this, the head tracking has some noise. Hence, the scene visualized using head tracking could shake, even the person is stopped. A naive solution is to filter the head position on time. It reduces the shaking, but can create a delay on the scene move relative to the head.

These difficulties come from the adaptive use of the Kinect. This device was created to control the Xbox 360 console using gestures. Nevertheless, this technology created new possibilities that are being used to other purpose. Wherefore, together with these possibilities come new challenges inherent to its non-characteristic use.

We believe that soon will be launched other low cost and real-time RGBD cameras, but with data resolution and quality better than the actual Kinect release. Using this device, the proposed effects will be done more natural and easily.

Acknowledgment

We would like to thank the CNPq, for funding the second author of this article, and the VISGRAF Lab, at IMPA, by providing an environment for fruitful research.

References

[1] L. Cruz, D. Lucio, and L. Velho, "Kinect and rgbd images: Challenges and applications." SIBGRAPI Tutorial, 2012.

[2] X. Bai, J. Wang, D. Simons, and G. Sapiro, "Video snapcut: robust video object cutout using localized classifiers," in *ACM SIGGRAPH*, 2009.

[3] N. Silberman and R. Fergus, "Indoor scene segmentation using a structured light sensor." International Conference on Computer Vision - Workshop on 3D Representation and Recognition, 2011.

[4] F. Prada and L. Velho, "Grabcut+d." VISGRAF PROJECT, 2011, http://www.impa.br/~faprada/courses/procImagenes/.

[5] S. Izadi, R. Newcombe, D. Kim, O. Hilliges, D. Molyneaux, S. Hodges, P. Kohli, A. Davison, and A. Fitzgibbon, "Kinectfusion: Real-time dynamic 3d surface reconstruction and interaction." ACM SIGGRAPH, 2011.

[6] A. Lefohn, U. C. Davis, J. Owens, and U. C. Davis, "Interactive depth of field using simulated diffusion," Tech. Rep., 2006.

[7] J. Shotton, A. Fitzgibbon, M. Cook, T. Sharp, M. Finocchio, R. Moore, A. Kipman, and A. Blake, "Real-time human pose recognition in parts from a single depth image." IEEE CVPR, 2011.

[8] H. Benko, R. Jota, and A. Wilson, "Miragetable: freehand interaction on a projected augmented reality tabletop." Conference on Human Factors in Computing Systems, 2012.

[9] A. Beato, "Understanding comfortable stereography." Techical Report, 2011, http://64.17.134.112/Affonso_Beato/Understanding_Comfortable_Stereography.html.