A Novel Octree Partitioning Approach Based on Camera Parameters for Unknown Object Representation

JeongHyun Seo, JaeWoong Kim, Sukhan Lee and Yeonchool Park*
Intelligent System Research Center, Dept. of electrical engineering and computer science,
Sungkyunkwan Univ. 300 Choenchoen-dong, Jangan-gu, Suwon, kyungki-do, Republic of Korea.

ABSTRACT

This paper proposes an improved octree representation method which is necessary for robot manipulation. To manipulate an object in a certain complex environment, robot needs simple, precise, and task oriented environment representations. Octree is a effective and broadly used three dimensional volumetric representation method to do this. But the problem of the conventional octree representation methods is that it generate many useless and incorrect cells because it generates the octree based on the number of 3D points. To solve these problems, we use the new 'Fill-Factor' concept. This paper introduces the concept and shows the experimental results to show the feasibility of the concept.

Keywords: Octree, Fill-factor

1. INTRODUCTION

In these days, robot has been evolved along with rapid development of the fundamental technologies. These developments enlarge the robot's service field from industrial area to human living area. Various behaviors such as navigation and manipulation are necessary for the service robot which lives in home environment. But the environment is different from the industrial area in terms of complexity and change variety.

Therefore real time modeling of the dynamic environment is very important task for service robot. This paper deals with the 1) environment modeling[1]-[5] from 3D point clouds; 2) octree representation method which is necessary for object manipulation.

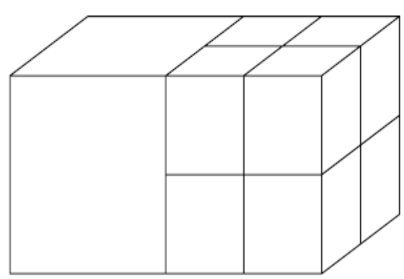


Fig. 1. Octree method

There are several conventional methods for the image based environment modeling such as mesh[6][7]. Among them, we use the octree based approach which is one of the volumetric representation methods. A general octree represents a space and it is decomposed recursively to the eight sub octree-cells until the limited thresholds of the resolution are met[8][9]. The octree concept is broadly used for robot manipulation. because it is intuitive and fast compared with other

*parkpd@skku.edu; phone +82-31-299-6473; fax +82-31-299-6466

ICMIT 2007: Mechatronics, MEMS, and Smart Materials, edited by Minoru Sasaki,
Gisang Choi Sang, Zushu Li, Ryojun Ikeura, Hyungki Kim, Fangzheng Xue, Proc. of SPIE
Vol. 6794, 67942S, (2007) · 0277-786X/07/$18 · doi: 10.1117/12.784520

representation methods. Octree representation has also many decomposition algorithms and approaches. But in this paper,

we propose a novel octree representation method which is optimized to unknown objects by adopting the 'Fill-Factor' concept. This concept decides the octree-cell's decomposition limit by checking the number of the occupied points and size. This concept is described in chapter3 in detail.

Rest of the paper addresses as follows. Section 2 deals with conventional octree representation approaches and our approach briefly. This chapter also expresses some problems of conventional approaches. In Section 3 describes the solutions for the problems addressed in Section 2, which are based on the Fill-Factor. And the experimental results are described in Section 4 to show the feasibility of the approach. Section 5 is the conclusion of this paper

2. CONVENTIONAL APPROACHES FOR OCTREE REPRESENTATION

This chapter deals with the two major concepts of the octree representation methods. The one is to represent a space using a basic cell which is equally divided in advance. The other is to decide the octree size based on Euclidean distance. The detail description of them is as follows.

2.1 Uniform Size Cell Decomposition Method

This method is a way to decompose and represent the space with uniform size octree cell. In this method, the size of the octree cell is pre-decided. At run time, the one octree cell is judged as an occupied cell when the number of points is bigger than threshold value. Otherwise, one cell is judged as a empty cell. The merit of this method is an easiness of its algorithm because this method simply divides the space and counts the points. However, this method has some problems. First, problem is a resolution problem. I.E, there may needs to complete too many times if the octree cell is determined too small so that causes generated many cells to count and check. This problem makes time delay of robot's re-action behavior. On the other hand, if the size of octree cell is determined too big, this method cannot represent the detailed space. This problem obstructs accurate robot behavior.

2.2 Decomposition with range considered octree cell

This method is a way that represents the space with a different size of octree cells by according to distance between camera and object. As a result, the object that far from camera is represented with small octree cells, while the object that close to camera is represented with large octree cells. In this case, it can be various representations because this method uses multi- resolution octree cell. Moreover, it makes a robot to concentrate to close environment because robot gets more information from near space. Although this method has merits, this method also has some problems. The Main problem is a noise problem. More far from the camera, the more noises are occurred along the front side direction of camera because of characteristic of the 3D stereo camera. It causes big size octree cell generation which regarded as a meaning full octree cell for robot even if this is just noise. This problem is solved when multi–resolution concept is adopted. But multi-resolution concept in this method cannot solve this problem because this multi-resolutions are only determined by consider the distance from camera. Objects or spaces are represented by same resolution if these are located in same distance from the camera. It means that this method is not free from same problem of Chap. 2.1. if representing target objects are located in same distance.

Above problems are caused from octree generation with absence of complete multi resolution concept. To complete multi resolution octree cell representation, we need clear guide line for limitation which stop timing when decompose the space recursively. In this paper, we introduce 'Fill-Factor' concept to solve these problems. 'Fill-Factor' is more detail described at chapter 3.

3. FILL FACTOR

Existing Octree cell generation methods stop space decomposition when the number of points is larger than threshold. But in this case, useless octree cells are also generated which enlarges octree generation time. This problem makes many difficulties to robot real time environment modeling to object manipulation. In this chapter we use the concept of 'Fill Factor' to decompose the space and generate octree. Since we use points from 3D stereo camera to generate the octree cells, the octree cells are also generated in real 3D space. We focus on this. We can decide decomposition limit timing by whether real points are filled in octree cell or not. In this, judgment base of fill is 'Fill Factor' which mentioned at above.

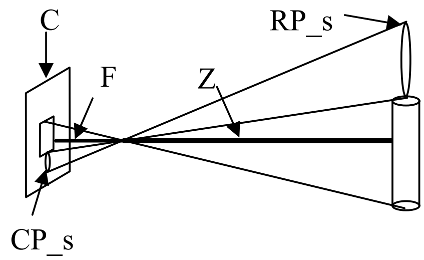


Fig. 2. Points size at 3D stereo camera system

To obtain fill factor, the real size of each points in real space is computed. We can compute the real size of the point in real space using camera parametric information such as focal length and the size of the point on CCD. Fig 2, shows this concept.

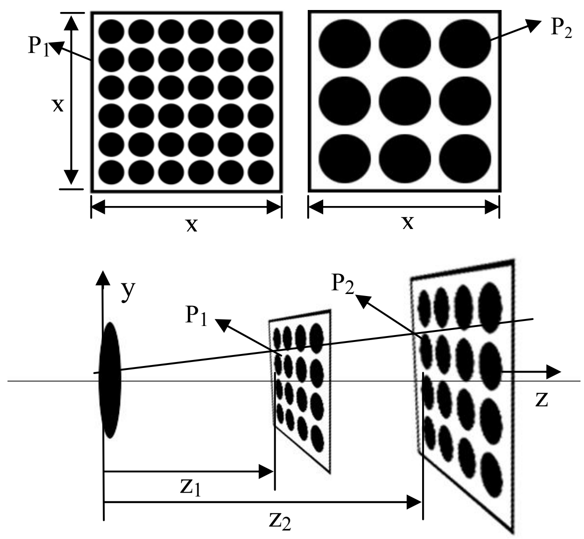


Fig. 3. Points array in 2D area by size x from z distance

In Fig 2. CP_s is the one point size in CCD, F denotes the focal-length, Z is the distance which from camera to the point, RP_s is the point size in the real space. By substitute these camera parameters to (1), we can obtain the real point size in real space.

$$CP\_s : RP\_s = F : Z \tag{1}$$

Using RP_s obtained in (1), we can judge whether octree cell is filled by points or not. In this case, the real point size is differed from distance. Fig 3. shows it.

Therefore, we calculate FillFactor Rate using RP_s in (1).

$$\text{Fill Rate} = \text{Octs} / (\text{sum of RP\_s in each Octree cell}) \quad (2)$$

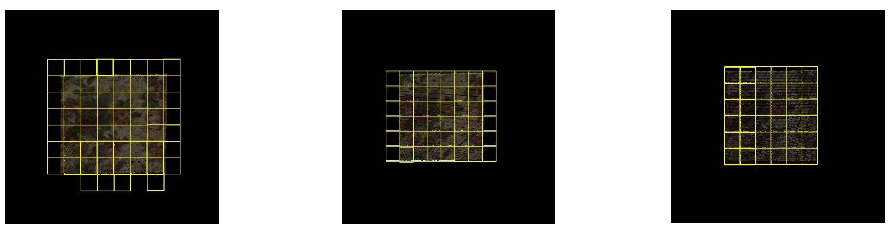
Octs in (2) denotes the size of the octree cell's one face. Because we obtain points that is 2.5D Points.

The denominator in (2) is obtained by camera parameters and (1). This means a sum of each point's sizes which included in the octree cell. The numerate in (2) is octree size. So, the value of Fill Rate equals the size of the one octree cell's one face if the octree cell filled by points.

4. EXPIRIMENT

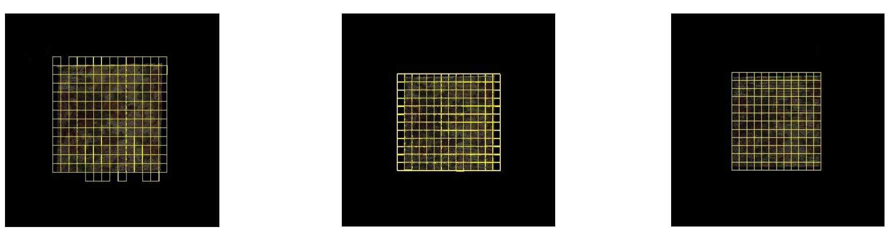
We use Pentium - 4 pc and structured light. In Fig 3. (a) is original image of object (20 cm * 20 cm) , (b) is result using same resolution(3.22 cm) octree decomposition, (b) is result using same resolution(1.66 cm) octree decomposition, (d) is result using Fill-Factor.

(a) Original Image



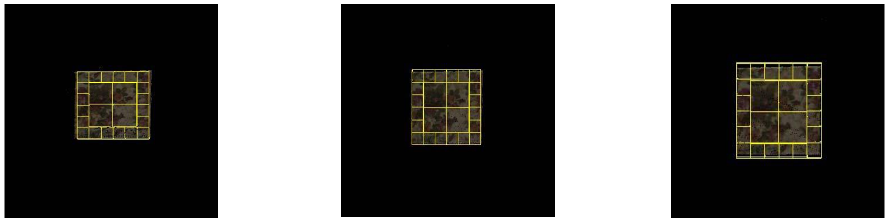
80cm 110cm 140cm

(b) Same resolution octree decomposition (octree size 3.32cm)



80cm 110cm 140cm

(c) Same resolution octree decomposition (octree size 1.66 cm)



80cm 110cm 140cm

(d) multi resolution octree decomposition using Fill-Factor

Fig. 4. Compare previous Octree decomposition with Octree decomposition using Fill-Factor

Octree cell which generated by previous approach is different along the distance as shown as Fig 4 (b), (c). But as shown as Fig 4 (d), octree is always same if we generate octree cell using Fill - Factor.

(a) Original Image

(b) Octree decomposition

Fig. 5. Octree decomposition using Fill-Factor

Fig 5. is result that is represented by octree using Fill-Factor. In Fig 5, we represent obstacle that except recognition object

REFERENCES

1. M. A. Garcia and A. Solana, "3D Simultaneous localization and modeling from stereo vision," *International Conference on Robotics & Automation (ICRA'04)*, Vol.1, Page(s):847 – 853, New Orleans, LA, 2004
2. Nüchter, H. Curmann, and J. Hertzberg, "Automatic model refinement for 3D reconstruction with mobile robots," *3-D Digital Imaging and Modeling*, Page(s):394 – 401, 2003.
3. D. Huber, O. Carmichael and M. Hebert, "3D Map reconstruction from range data," *In proceedings of the IEEE Conference on Robotics and Automation*, Vol1, Page(s):891 – 897, 2000

4. Y. Liu, R. Emery, D. Chakrabarti, W. Burgard, and S. Thrun, "Using EM to learn 3D models of indoor environments with mobile robots," *In proceedings of the 18th Conference on Machine Learning*, Vol.1, Page(s):502 – 507, 2002.
5. D. Hahnel, W.Burgard and S.Thrun, "Learning compact 3D models of indoors and outdoor environments with a mobile robot," *Journal Robotics and Autonomous systems*, Vol. 44, 2003
6. DeHaemer, m. J., Jr. and Zyda, M. J., "Simplification of Objects Rendered by Polygonal Approximations," *Computers & Graphics*, Vol. 15, Page(s):175-184, 1992.
7. Alan D, Kalvin and Russell H. Taylor. "Superfaces : polygonal mesh simplification with bounded error." *IEEE Computer Graphics and Appl.*, 1996.
8. R. Shekhr; E. Fayyad; R. yagel; F. Corn-hill. "Octree-based decimation of marching cubes surfaces." *IEEE In visualization'96*, Page(s):335-342, 499, 1996.
9. Sukhan Lee; Daesik Jang; Eunyoung Kim; Suyeon Hong; JungHyun Han;. "A real-time 3D workspace modeling with stereo camera,*" Intelligent Robots and Systems, 2005. (IROS 2005), IEEE/RSJ International Conference* on Page(s):2140 – 2147, 2-6, 2005