A Real-Time 3D Workspace Modeling with Stereo Camera

Sukhan Lee, Daesik Jang, Eunyoung Kim and Suyeon Hong

School of Information and Communications Engineering
Sungkyunkwan University
300 Chunchun dong, Jangan-ku, Suwon, Korea

{lsh, dsjang}@ece.skku.ac.kr

JungHyun Han

Department of Computer Science and Engineering
Korea University
Sungbuk-ku, Seoul, Korea

jhan@korea.ac.kr

Abstract **– This paper presents a novel approach to real-time 3D modeling of workspace for manipulative robotic tasks. First, we establish the three fundamental principles that human uses for modeling and interacting with environment. These principles have led to the development of an integrated approach to real-time 3D modeling, as follows: 1) It starts with a rapid but approximate characterization of the geometric configuration of workspace by identifying global plane features. 2) It quickly recognizes known objects in workspace and replaces them by their models in database based on in-situ registration. 3) It models the geometric details on the fly adaptively to the need of the given task based on a multi-resolution octree representation. SIFT features with their 3D position data, referred to here as stereo-sis SIFT, are used extensively, together with point clouds, for fast extraction of global plane features, for fast recognition of objects, for fast registration of scenes, as well as for overcoming incomplete and noisy nature of point clouds. The experimental results show the feasibility of real-time and behavior-oriented 3D modeling of workspace for robotic manipulative tasks.**

Index Terms – 3D workspace modeling, planar feature, stereo vision, SIFT, 3D object recognition

I. INTRODUCTION

Environment modeling is crucial for autonomous mobile robots, specially for intelligent service robots that perform versatile tasks in everyday human life. Recently, 3D modeling techniques have been investigated to represent the environment for navigation or walk-through. Especially, the approaches based on 3D range sensors have been successful. In contrast, real-time workspace modeling in a cluttered environment is a far more difficult problem, and few research results have been reported.

In this paper, we tackle the problem of 3D workspace modeling for robotic manipulation using a stereo camera in real-time. A new real-time approach to workspace modeling based on recognition of global spatial features and target objects is proposed.

Environment modeling has been actively studied by many research groups [5][6][9][10][12]. Garcia and Solana [5] presented a stereo-vision-based algorithm for determining the 6-DOF trajectory of a mobile robot and simultaneously building a volumetric 3D model of the workspace. Huber *et al.* [9] proposed to use spin-images, which are defined as features describing the local shape around a particular point, for environment modelling. Liu *et al.*[10], Hähnel *et al.* [12] and Nüchter *et al.* [6] suggested environment modeling algorithms based on planar features for approximating/refining the environment and compensating for noise of 3D points. These approaches mainly have been focused on the environment modeling for navigation rather than manipulation. Accurate and rapid modeling of workspace for manipulation has rarely been reported.

To construct a 3D environment model, multiple scans are often *registered*. The most generally used for such registration is Iterative Closest Points (ICP), originally suggested by Besl and Mckay [3] and Chen and Medioni [8]. Ever since, many variants of ICP have been proposed, and a recent survey and estimation can be found in [4]. The main problems of the ICP approach are that insufficient results are generated when the initial relative pose is not known and often there are lots of outliers and noise.

Another approach to registration is the feature-based matching. Surmann *et al.* [11] showed a scan matching algorithm based on edge features, but the matching results are not satisfactory despite the fast computational speed. Spin-images [9] and SIFT (scale-invariant feature transform) [1][5] have often been used for multiple-scan registration as well.

Object recognition has long been studied in the computer vision field. A previous work related to ours is a 3D-feature-based object recognition. For example, Johnson and Herbert [14] proposed a spin-image-based recognition algorithm in cluttered 3D scenes, but the algorithm does not run in real-time.

Extraction of global spatial features plays an important role in the sense that it simplifies the workspace for further processes including motion planning. Nüchter *et al.* [6] extracted planar features using a combination of ICP and RANSAC (Random Sampling Consensus) algorithms. Liu *et al.* [10] proposed to find planar features based on expectation maximization (EM). Hähnel *et al.* [12] proposed to apply the random sampling and region growing techniques to find planar features. Yet, the algorithms do not run in real-time.

The above mentioned 3D modeling methods are focused more on precise and accurate modeling, instead of evaluating 3D modeling under the framework of robotic tasks. For the latter, the requirement of precision and accuracy becomes relative to task at hand, while the real-time issue becomes more immediate and significant.

(a) a workspace to be modeled (b) a workspace model

Fig. 1 3D workspace modeling by integrating three principles
: global spatial features, object models and target oriented details

Application of 3D modeling to service robots implies that 3D modeling should be integrated with robotic task execution in such a way that the resulting robot behaviors are well acceptable to human. This point is not given proper attention in the past in designing 3D environment or workspace model.

In this paper, we propose a novel method for 3D modeling in real-time. We also integrate different methods to utilize fundamental principles that human uses for understanding the environment.

II. Proposed Approach

The proposed 3D workspace modeling technique attempts to draw and utilize fundamental principles that human uses for understanding the environment. By doing so, we can endow robots with the capability of fast and smooth manipulation and handling of objects in 3D space as human does. The following three fundamental principles are chosen:

1) The approximate yet rapid characterization of spatial features of the workspace by recognizing global spatial features such as planes, walls, and floors and their spatial relations.

2) The quick recognition of known objects in the workspace for the in-situ integration of their solid models from the knowledge or database to the workspace model.

3) The task-oriented modeling of geometric details or resolutions on the fly adaptive to the necessity given by the task at hand.

It should be pointed out that our approach of integrating the above three principles aims at providing a service robot with the capability of not only real-time but also human-like operations, so that the robot offers a comfortable interaction with human.

The approach to the implementation of the above three principles taken in this paper is described briefly in the following:

1) For the approximate yet rapid characterization of spatial features of the workspace, we choose to extract the global *planar features* in the workspace. To ensure a fast and robust processing, we resort to the combined use of the SIFT features with the 3D information (stereo-sis SIFT) and the 3D point clouds from stereo-sis. The extracted planar features are used to reduce the search space, as well as to speed up the process of scene registration. (Sections Ⅲ and VI)

2) For the quick recognition of known objects in the workspace and their replacement by the corresponding solid models from the database based on an in-situ registration, we also use the SIFT features with their 3D information. (Section IV)

3) For an on-the-fly modeling of geometric details adaptively to the necessity for the given task at hand, we use a multi-resolution octree representation of obstacles, where the resolution is determined by the spatial freedom relative to the given task. (Section V)

Note that the SIFT features with 3D position information, stereo-sis SIFT, are used as the main features for plane extraction, object recognition and scene registration to improve the speed of each algorithm and to overcome the defects of incomplete and noisy point cloud. Fig. 1 shows an example of the workspace model, represented using the three principles.

The originality of the proposed approach can be summarized as follows: We propose, for the first time, the three principles on robot 3D modeling as an integral concept for system integration that offers real-time and human-like robotic service behaviors.

In what follows, we present the details of how the principles and approaches described above are implemented.

III. Rapid Characterization of Workspace Configuration

Planes in 3D space (planar features) are used as the global spatial features in this paper. The proposed plane-extraction algorithm works in two stages. The first stage employs the RANSAC (Random Sampling Consensus) algorithm [7], which has been widely used for fitting a model to an experimental data set, and produces a candidate set of planes. The plane set is fed into the second stage for verification, and only the verified candidates are accepted as planes.

In the first stage, the RANSAC algorithm processes only the stereo-sis SIFT features. Three stereo-sis SIFT features (P_1, P_2, $P_3 \in R^3$) are randomly selected, and the plane normal N is estimated. Given a plane (N, P_1) based on the three stereo-sis SIFT features, a point X may be determined to lie on the plane if X satisfies the following equation:

$$|N \cdot (X - P_1)| < \tau \quad (\|N\| = 1) \tag{1}$$

where τ is the user-defined threshold.

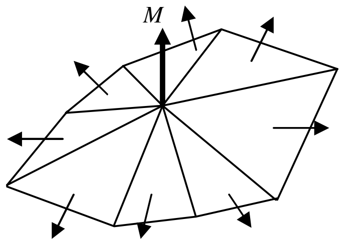


Fig. 2 The normal of a SIFT feature.

In addition to the position of X, our algorithm also considers its orientation. A point X in an image plane normally has 8 neighbor points. The local geometry (triangular mesh) around X is constructed using the 8 neighbor points, as illustrated in Fig. 2. Then, the normal M of X is computed as the normalized mean of the normal of the adjacent triangles. Every stereo-sis SIFT feature satisfying (1) is tested with the following equation again:

$$M \cdot N \; > \; \sigma \qquad (2)$$

where N is the normal of the plane, · denotes the dot product operation, and σ is the user-defined threshold. $M \cdot N$ can have high value when the point X and the plane have similar surface orientation and SIFT features with similar orientations with the plane can be selected with this equation. If the number of SIFT features satisfying both of (1) and (2) exceeds the threshold, the plane (N, P_1) is taken as a candidate for the second stage.

The second stage of our plane-extraction algorithm performs the position test using (1) with the entire 3D points in the scene. If the number of 3D points satisfying (1) exceeds a threshold, the best-fit plane is computed with the 3D points, and is taken as a global spatial feature in the workspace.

When a new scene is captured from a new camera position, the old plane set is updated and incremented with the new point cloud. The new SIFT features extracted from the new point cloud are tested against the current planes and those that satisfy (1) are used to update the planes. In contrast, the new SIFT features which are not covered by any of the current planes may contribute to additional planes through the two-stage plane-extraction process. The complexity of plane extraction algorithm is $O(N_p)$, where N_p is the number of 3D points in a scene.

Note that all computed planes are infinite-sized. For correct modeling of the workspace, however, each plane should have a boundary, because modeling with bounded surfaces is crucial for subsequent processes such as collision detection and motion planning. Assuming a rectangular plane, all the 3D points on a plane are projected to the plane, and then a 2D oriented bounding box (OBB) is computed using a covariance matrix.

Future work for the global spatial feature extraction may be listed as follows. (1) For more autonomous modeling, we may have to adopt a fusion & filtering approach for flexibly fixing the threshold values. (2) The current algorithm will produce a single bounded plane, for example, for the top-planes of two separate tables with the same height. A more elaborate method should be developed to partition such a single plane into a set of connected components. The distribution of the 3D points may have to be investigated.

(a) a 3D point cloud (b) planes extracted

Fig. 3 The scene with planes extracted.

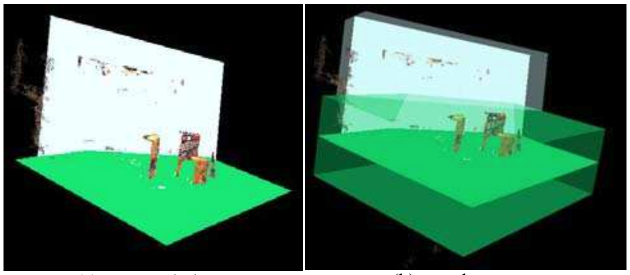
The 3D points comprised in the planar features are simply replaced by the planes in world modeling. Fig. 3 shows the extracted planes and the remaining points that are not lying on the planes. At the next step of the world modeling, some portions of the point cloud will be recognized as objects. The extracted planes not only provide global spatial features, but also reduce the search space for object recognition and therefore enhance the performance.

IV. Integration of Object Models to Workspace

It is assumed that all objects to be manipulated (e.g. a bottle to be picked up) have complete *solid model* representations in the object database. In addition, the database is rich enough to contain the invariant features and local accessibility directions of each object. The workspace will be searched for a *target object*. If a portion of the point cloud is *recognized* as the target object, it will be replaced by the target object's solid model stored in the database. The object recognition algorithm consists of the following steps:

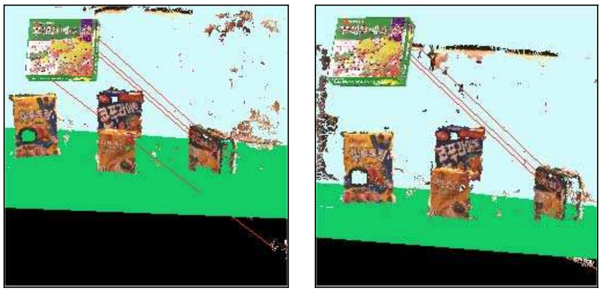
1. Determination of the search space for object recognition.
2. Matching of SIFT features
3. Verification of the matched SIFT features
4. Integration of the solid model of the recognized target object with the workspace model

The first step is optional, but the search space can be significantly reduced by simple heuristics. For example, consider the assumption that all objects lie on the extracted planar surfaces such as table/desk top planes, i.e. no object is placed on top of other objects. In order to utilize the assumption, the distance between a SIFT feature and an extracted plane is calculated. If the distance is larger than the target object's height for all extracted planes, the SIFT feature is discarded. The remaining SIFT features constitute the search space for object recognition.



(a) extracted planes (b) search space

Fig. 4 Reduced search space of a target object



(a) SIFT matching without constraint (b) matching with the 3D triangle constraint

Fig. 5 Object recognition results: The solid model of a cereal box is drawn from the database and virtually placed at the upper-left corner while the other boxes on the plane are real-world objects. In (a), a *mis-matched* SIFT-point pair is found. Note that it is discarded in (b).

Fig. 4 depicts the reduced search space of a target object. (The object recognition process to be discussed immediately benefits from the reduced space. The recognition process in the reduced space takes 234 ms while working with the original space needs 375 ms on average.) Traditionally, various invariant features have been used for object recognition in many domains. Recently, SIFT key points have been used successfully for object recognition in 2D images [2]. The traditional algorithms are based on the invariant feature descriptors in 2D images, i.e. no 3D information is used in the recognition process. In our work, each SIFT feature on the surface of the target object's solid model is matched with the SIFT features in the search space using a traditional SIFT matching algorithm [2]. Unlike the traditional SIFT matching algorithms, however, the SIFT features with 3D position information (stereo-sis SIFT) are used as the main features for object matching.

The SIFT-matching procedure will output a set of matched stereo-sis SIFT-point pairs. The pairs are then verified using a simple 3D geometric constraint, named *3D triangle constraint*. If the number N of the matched SIFT features is greater than 2, ${}_{N}C_3$ triangles can be generated, each in the solid model and in the workspace. The 3D triangle constraint requests that two triangles based on a set of 3 matched point pairs should be congruent within an error bound. Only when the triangles satisfy the constraint, their vertices (three pairs of the matched points) are accepted. Fig. 5 compares the matching results with and without the constraint. The complexity of object recognition is $O(M_s N_s)$, where M_s is the number of SIFT features in an object model and N_s is the number of SIFT features in the reduced search space of a workspace.

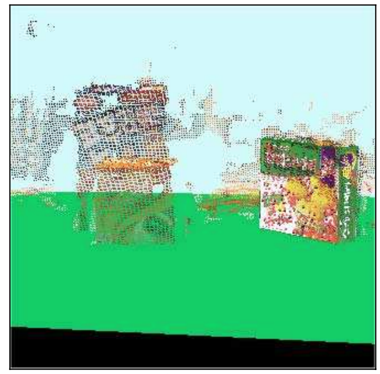


Fig. 6 Integration of the target object's solid model with the workspace

If the total number of the verified SIFT features is less than 3, the object recognition fails. Otherwise, the target object is recognized and the solid model of the target object is plugged into the workspace by the quaternion-based method of Horn [3]. Once the solid model is plugged into the workspace, the 3D point cloud belonging to the target object is discarded, i.e. the point cloud is replaced by the solid model. Fig. 6 shows an example.

It is important to mention the heterogeneous organization of the workspace: The plugged solid models represent the most precise information and the planar features provide simplified global information.

V. TASK-ORIENTED REPRESENTATION OF OBSTACLES WITH MULTI-RESOLUTION OCTREE

The point cloud that is not comprised in the extracted planes and also not recognized as target objects is taken as obstacles which should be avoided in motion planning. The obstacles are hierarchically represented in an *octree*, which is a popular modeling technique facilitating accessibility analysis, collision detection, motion planning, etc. In the octree, the entire scene makes the initial cell, and the cell is recursively divided into eight equal-sized cells.

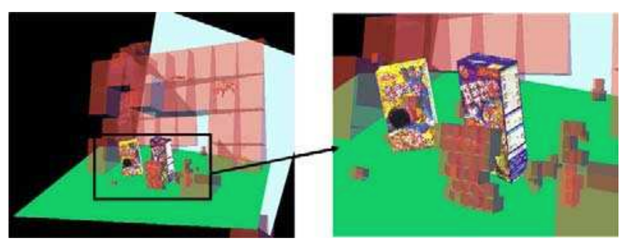


Fig. 7 Multi-resolution octree representation of obstacles

In general, the recursive division continues until the *maximum recursion level* is reached or the cell contains less than the pre-determined number of points. In the proposed approach, the *multi-resolution modeling* is applied to the octree construction. Suppose an octree construction of a scene including the target object to be manipulated exists. Then, a coarser representation would be enough for the obstacles that are far away from the target object. In contrast, a finer representation may be needed for closer obstacles, for example, in order to precisely detect collision between obstacles and the robot arm. In the current implementation, therefore, the cells for farther obstacles are associated with a *smaller* 'maximum recursion level' such that the cells are represented in a coarse resolution. In contrast, the cells for closer obstacles are associated with a *larger* level, leading to a fine resolution.

Fig. 7 shows the multi-resolution octree representation of the obstacles.

(a) scene at a time frame

(b) matched point pairs (connected by red lines) of two consecutive sc-enes

(c) registered scenes, where the matched points are overlapped

Fig. 8 Stereo-sis SIFT and plane based 3D registration

VI. SIFT & Planar Feature Based Registration of Multiple Scenes

This section presents an efficient real-time method to register multi-view 3D data scanned by a stereo camera. A set of 3D points scanned by a stereo camera at a time frame represents a partial view of the workspace. Therefore, the point set needs to be stitched with subsequent scene data to generate a global 3D workspace model. This is the process of registration.

This paper proposes a novel registration method working in a cluttered environment, which satisfies both of the real-time and robustness requirements. The registration process uses only the stereo-sis SIFT features instead of the entire 3D points. Further, the planar features representing the approximate geometric structure of the workspace are used to reduce the size of the SIFT feature set. The proposed registration algorithm can be described as follows:

1. SIFT feature sets S_{t-1} and S_t are calculated at time-frames t-1 and t, respectively.
2. The SIFT feature set S_{t-1} is reduced into S'_{t-1} such that S'_{t-1} contains only the SIFT features lying on any of the extracted planes (within some error bound).
3. The reduced SIFT feature set S'_{t-1} is then matched with S_t, and the matched point-pair set $M_{t-1,t}$ is obtained.
4. The transformation matrix needed for registration is calculated using the matched point-pair set $M_{t-1,t}$ and the quaternion-based method of Horn [3].
5. All 3D points at time-frame t are transformed to the coordinate system of the time-frame t-1.

The complexity of scene registration is $O(N'_{t-1}N_t)$, where N'_{t-1} is the number of SIFT features in S'_{t-1}, and N_t is that in N_t. Fig. 8 shows a result of the stereo-sis SIFT and plane based scene registration.

VII. Experimental Results

It is assumed that a home service robot is requested to manipulate (for example, grasp and move) a predefined object which lies on a planar surface such as the top plane of a table. A stereo camera is mounted on the end effecter of an arm with an eye-on-hand configuration. The range data in the form of 3D point cloud including 2D reference image is acquired from the stereo camera on the fly. (Currently, the arm follows a predefined path, but the path should be planned for future research.) Fig. 9 shows the experimental environment with an eye-on-hand configuration.

It is also assumed that the workspace is textured enough so that the captured range data contains a plenty of 3D points. In such a well-textured environment, a meaningful bunch of the SIFT features can also be extracted. The range data can be captured with the maximum rate of 30 frames/sec. Fig. 10 shows an example of the captured point cloud and the reference image. For the visualization purpose, the 3D points are displayed with the combined colors of the corresponding pixels in the reference images.

Every target object (requested to be manipulated) is assumed to have a solid model in the database. The solid model is preprocessed to be associated with SIFT features on its surface. Fig. 11 shows some examples of object models used in this paper.

Fig. 9 Experimental environment with an eye-on-hand configuration

(a) the 3D point cloud

(b) the reference image

Fig. 10 Captured range data from a stereo camera

Fig. 11 Object models in the database

TABLE I
MEASUREMENT OF SYSTEM PERFORMANCE

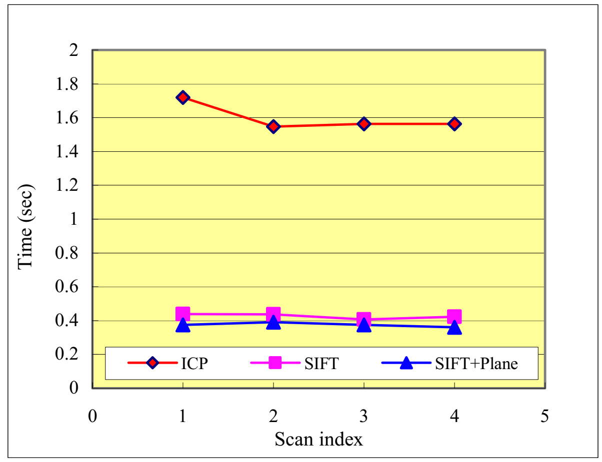
	Average time (ms)
SIFT feature calculation	188
Planar feature extraction	19
Object recognition (3 object models)	350
Multi-resolution octree construction	60
Scene registration	180
Total	797

The experiment is implemented on a modest PC (Pentium 4 2.8 GHz) and it takes on average 3.72 seconds to model a workspace captured with 5 different views. It means that it just takes 1.41 seconds to process a pair of scenes and the modeling process (including planar feature extraction, object recognition, multi-resolution octree construction, and scene registration and transformation) can be done in real-time. It is also found that the proposed method is much faster than the previous modeling approaches [6][9][12]. Table 1 shows the measured results of system performance for each stage of the algorithm.

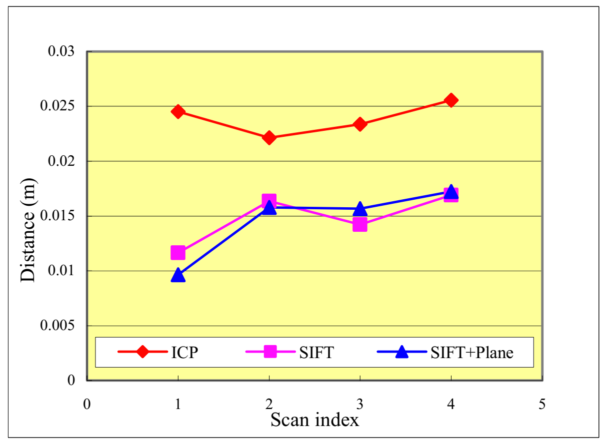
Fig. 12 shows an example of modeling a cluttered workspace. In this case, five consecutive scenes are integrated to build the final workspace model.

We compared the proposed SIFT and plane based scene registration approach with the well-known ICP approach [3]. As demonstrated by the experimental results in Fig. 13, the proposed scene registration is both faster and more accurate than the traditional ICP-based registration methods. Note that, in Fig. 13-(a), the overall computation time includes the time needed for SIFT feature computation, SIFT matching, and transformation. Fig.13-(b) also compares the SIFT based approaches with and without the extracted planes. The distance error after scene registration also shows that proposed algorithm is robust to noisy 3D data. Fig.13-(c) shows examples of registrations using proposed SIFT-based registration and traditional ICP-based method when the distance between two scenes to register is huge. We can see that the proposed method can register scenes more robustly even though the initial positions of two scenes are far way.

We evaluated the robustness of the SIFT-based object recognition algorithm with respect to the variation of the distance between the camera and the workspace. Table II shows the results of this evaluation. In table 2, '# of SIFT for object' is the number of SIFT features for object predefined in database and '# of SIFT matched' means that the number of SIFT features of a sensed scan matched with SIFT features of the object predefined in database.



(a) registration time

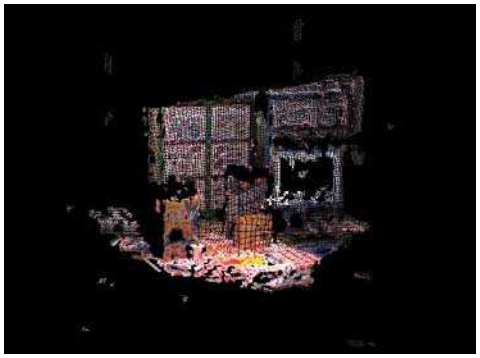


(b) distance error after registration

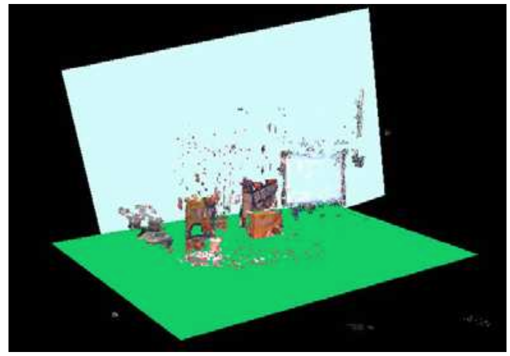
SIFT based registration ICP
(c) Results of SIFT based registration and ICP
Fig. 13 Performance comparison

TABLE II
RESULT OF OBJECT RECOGNITION ACCORDING TO DISTANCE

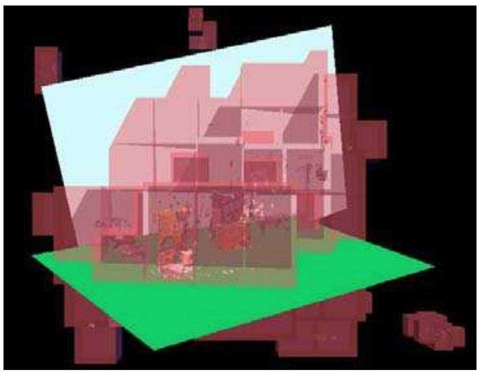
Distance(m)	# of SIFT for object	# of SIFT matched
0.5	373	37
0.8		41
1.1		15
1.4		8
1.7		5
2.0		4
2.3		1
2.6		1



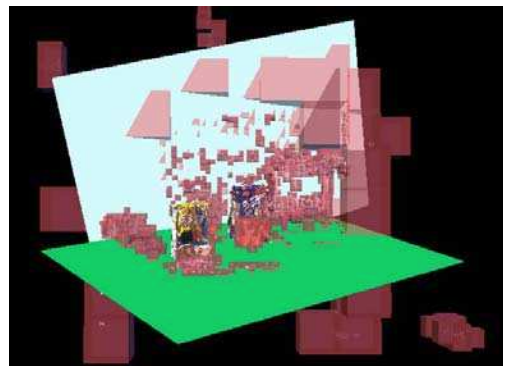
(a) point cloud of the first scene



(b) extracted planes of the first scene



(c) octree representation



(d) a complete workspace model

Fig.12 Example of a workspace model obtained from five scenes

As illustrated in this table, if the camera moves far away from a workspace, the number of SIFT features matched is decreased because the scanned scene has small number of SIFT features and has lots of noises. And also we can infer from this evaluation that the SIFT feature itself is not perfectly invariant to scale variation.

VIII. SUMMARY AND CONCLUSIONS

A new real-time approach for workspace modeling based on heterogeneous representation of geometric data is proposed and presented. We model a workspace with three different types of geometric components following the fundamental principles that human uses for modeling the environment: 1) global geometric features such as planar features, 2) known 3D object models in database, and 3) multi-resolution representation of unknown obstacles. The global spatial features represent a contextual description of a workspace and offer an approximated but fast structural information of the workspace. Known 3D object models in the database are integrated into the workspace model as soon as they are recognized, simplifying the modeling process further. Other workspace entities that belong neither to planar features nor to known 3D object models are modeled by the octree representation where its resolution varies according to the task at hand.

The originality and strength of the proposed approach lie in the fact that the integration of these three components leads to real-time and behavior-oriented 3D modeling that is crucial for application to service robots, where a comfortable interaction between human and robot is a must.

The performance and robustness of the proposed approach are evaluated in cluttered indoor environments experimentally. The experimental results demonstrated that the proposed methods are fast and robust enough to model 3D workspace for real-time robotic manipulation.

Future research plan includes the upgrade and expansion of object database, the addition of 3D invariant features, and the integration with motion planners.

ACKNOWLEDGMENT

This research was performed for the Intelligent Robotics Development Program, one of the 21st Century Frontier R&D Programs funded by the Ministry of Commerce, Industry and Energy of Korea. This work was partly supported by the Science and Technology Program of the province of Gyeonggi. Jino Kim and his team at Kwangwoon University, as well as Mr. Jaehoon Jeong and Prof. Hyeonki Choi of Sungkyunkwan University, for their assistance to collect experimental data.

REFERENCES

[1] S. Se, D. Lowe and J. Little, "Vision-based mapping with backward correction," 2002 IEEE/RSJ International Conference on Intelligent Robots and Systems, 2002

[2] D. Lowe. "Object recognition from local scale invariant features," In proceedings of the Seventh International Conference on Computer Vision (ICCV'99), pages 1150–1157, Kerkyra, Greece, September 1999.

[3] P.Besl and N. McKay, “A method for registration of 3D shapes,” IEEE Transactions on Pattern Analysis and Machine Intelligent, vol. 14, pp. 239-256,1992

[4] S. Rusinkiewicz and M. Levoy, “Efficient variants of the ICP,” The 3rd International Conference on 3D Digital Imaging and Modeling ,2001

[5] M. A. Garcia and A. Solana, “3D Simultaneous localization and modeling from stereo vision,” International Conference on Robotics & Automation (ICRA’04), New Orleans, LA, April 2004

[6] A. Nüchter, H. Curmann, and J. Hertzberg, “Automatic model refinement for 3D reconstruction with mobile robots,” 3-D Digital Imaging and Modeling, 2003.

[7] M.A. Fischler and R.C. Bolles, “Random sample consensus: a paradigm for model fitting with application to image analysis and automated cartography,” Commun. Assoc. Comp. Mach., vol. 24, pp. 381 – 395, 1981.

[8] Y.Chen and G. Medioni, “Object modeling by registration of multiple range images,” In proceedings of the IEEE Conference on Robotics and Automation, 1991.

[9] D. Huber, O. Carmichael and M. Hebert, “ 3-D Map reconstruction from range data,” In proceedings of the IEEE Conference on Robotics and Automation, 2000.

[10] 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, Williams College, July 2001.

[11] H. Surmann, A. Nüchter, and J. Hertzberg, “An autonomous mobile robot with a 3D laser range finder for 3D exploration and digitalization of indoor environments,” Journal Robotics and Autonomous Systems, vol. 45, no. 3-4, pp. 181-198, December 2003.

[12] 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.

[13] S. Thrun, W. Burgard and D. Fox, “A Real-Time algorithm for mobile robot mapping with applications to multi-robot and 3D mapping,” IEEE International Conference on Robotics and Automation, San Francisco, April 2000.

[14] A. E. Johnson and M. Hebert, “Using spin images for efficient object recognition in cluttered 3D scenes,” IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 21, May 1999.