Motion Planning for Dual-arm Mobile Manipulator — Realization of "Tidying a Room Motion" —

Takayasu Takahama, Keiji Nagatani, Yutaka Tanaka
Graduate School of Natural Science and Technology, Okayama University
Tsushima-naka, Okayama, 700-8530, JAPAN
Email: keiji@ieee.org

Abstract— When a mobile robot executes a given task in human-robot co-existence environment, it is necessary for the robot to perform robust motions and to have abilities to deal with unforeseen situations. Therefore we aim to realize an intelligent and robust motion for a dual-arm mobile manipulator. To carry out a research of intelligent motion, we set a concrete task as "grasping a book and returning it to a bookshelf". In this task, a path of an end-effector that grasps a book should be planned according to the book's position. A motion of returning a book also requires confirming return position and planning manipulator's motion.

In this paper, we introduce our motion planning algorithm for a dual-arm mobile manipulator, and shows an example of integrated motion of "tidying a room" task.

I. Introduction

Recently, target environments for robotics research has been extending from structured environment (e.g. factories environment) to human-robot co-existence environment (e.g. office building and home environment). Therefore it is necessary for robots to have abilities to perform complicate tasks and to deal with uncertainty. To realize such intelligent motion, we have started a research of motion planning for a dual-arm mobile manipulator.

Generally, a mobile manipulator has much wider work space than a fixed manipulator. However, a mobile manipulator has a disadvantage, which an error of base robot's position extremely affects its end-effector's position. To compensate the error, the robot should have an ability to adjust a position of the end-effector. On the other hand, a dual-arm manipulator can perform complicate tasks than a single manipulator. However, dual-arm manipulator requires more complicate motion planning, and a shared work space of dual-arm manipulator becomes narrower than a single manipulator. Therefore, motion planning is very important to perform intelligent motions for a dual-arm mobile manipulator.

To proceed with a research of such motion planning, we have set up a specific task. A theme of the task is "tidying a room", and we have defined a concrete research task as "grasping a book laid on a desk and returning it to bookshelf" by a dual-arm mobile manipulator.

In this paper, we explain a motion planning algorithm for a dual-arm manipulator to realize the proposed task, and introduce a demonstration of integrated motion in real environment.

II. Related Works

The research field of mobile manipulator is vast. Therefore, we introduce researches of "application base" and "motion planning" for mobile manipulator, which deeply relates to our research.

In application base researches, Tomizawa et. al. performed a mobile manipulator's motion to grasp a book from a bookshelf and to browse it via the Internet[1]. Nagatani et. al. performed a door opening motion and passing through a doorway by a mobile manipulator[2]. In above researches, robot's motions are intuitively preplanned by operators, and motion planning is not required.

From the point of view motion planning researches for mobile manipulator, it is popular to consider a manipulability[3] of an end-effector. Bayle et. al. proposed standardization of manipulability for non-holonomic vehicle with n joint manipulator, and checked a problem of kinematic control[4][5]. Desai et. al. proposed a motion planning algorithm for grasping an object and transporting it to another place using multi-mobile manipulators[6]. Nagatani et. al. proposed a motion planning algorithm to draw a line on a wall with keeping manipulability[7]. Khatib et. al. proposed an on-line motion planning of a mobile manipulator using "elastic strips approach"[8]. Yamamoto et. al. proposed a motion planning algorithm to grasp an object using ellipse of work space with wheeled mobile manipulators[9].

Many of motion planning algorithms for mobile manipulator are proposed recently, shown in above. However these algorithms aim to realize a simple task (such as obstacle avoidance), and it is difficult to apply more complicate tasks. Also, many of motion planning researches are only considered in simulation world.

In this research, we aim to realize a complicate motion for a dual-arm mobile manipulator, "grasping a book and returning it to bookshelf" that requires motion planning. Additionally, we perform the robot's motion in real environment to confirm a validity of our algorithm, and to find problems.

III. Task-oriented Approach

A. Task Definition

Our research task is defined as "grasping a book laid on a desk and returning it to bookshelf" by a dual-arm mobile manipulator. In this task, we assume that the robot has information of a height of the desk, a position of a target bookshelf, a return position of the book, and a size of the book. The bookshelf locates comparatively far from a desk. The book is supposed to be returned at a left side of a specific book in the bookshelf. The robot doesn't know neither an exact position of the book on the desk, nor an exact position of book's return in the bookshelf. Therefore the robot should plan its motion after detecting book's position or return position.

B. Task Division

The proposed task can be realized by the following procedure.

1) The robot moves to a grasping position of the book.
2) The robot detects a book's position by vision sensor.
3) The robot adjusts its position to grasp the book.
4) The manipulator grasps the book.
5) The base robot navigates to the bookshelf.
6) The robot detects a return position of the book by vision sensor.
7) The robot adjusts its position to return the book.
8) The manipulator returns the book.

In our assumption, the robot does not know an exact target position. Therefore it is necessary for the robot to plan a motion. In 2) and 6), the robot needs to detect the book's (or return) position. In 1) and 5), the base robot navigates along pre-planned path.

According to above requirements, there are three important elements, "motion planning", "target detection" and "navigation". In the following subsections, we introduce these elements and motion designs.

C. Motion of Grasping a Book

It is possible for a single manipulator to grasp a book laid on a desk by 1) pushing the book toward an edge of the desk and 2) grasping the showing both side of the book. However, in case that there are obstacles on a desk or there are steps along desk's edges, it is almost impossible to grasp a book by a single manipulator. Moreover, there is a risk to drop the book using above method. Therefore we use a dual-arm manipulator to grasp a book. One manipulator pushes the book and the other manipulator grasps it. We assume that the pushing manipulator moves linearly on the target desk's surface toward a grasping manipulator. This is almost the same motion as sweeping by a broom and a dustpan, shown in Fig.1.

One of the problems is manipulators' work space. If the target book does not exist in manipulators' work space, it is necessary to adjust base robot's position.

D. Motion of Returning a Book

It is possible for a single manipulator to return a book to a bookshelf (shown in Fig.2). In this research, the end-effector moves linearly toward the bookshelf to insert the book. If the return position is not in manipulator's work space, it is necessary to adjust base robot's position.

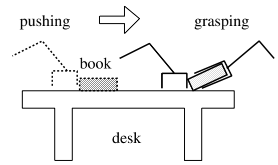


Fig. 1. Grasp motion

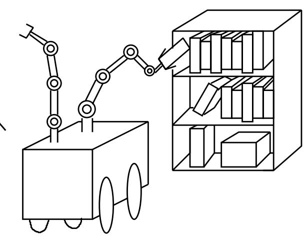


Fig. 2. Return motion

E. Base Robot's Navigation

To transport a target book toward a target bookshelf, it is necessary for the base robot to navigate along a given path. In this research, the path is represented by a series of segments.

F. Target Detection

For grasping a book, the robot is necessary to detect the book's position. For returning the book, the robot is necessary to detect the return position in a target bookshelf. In this research, we assume that the return position is next to a specific book. Therefore both target detections can be performed by detecting books' image using image processing. To realize the detections, we mount vision sensors on an end-effectors.

G. Target Robot

To realize the proposed task in real environment, we use the dual-arm mobile manipulator shown in Fig.3. This robot mounts 5-axis-manipulator and 6-axis-manipulator. Each end-effector has CCD camera, and PSD (position sensitive detector) sensor is mounted on the end-effector of 5-axis-manipulator. The base robot has omni-wheels that enable to move the robot in omni-direction. Fig.4 indicates parameters of links and joints parameters of the dual-arm manipulator. A length of baseline between both manipulators is $0.35[m]$.

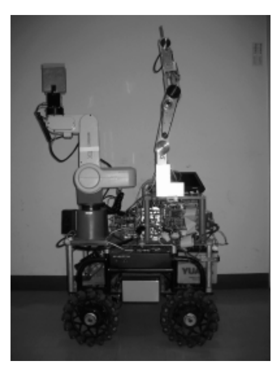


Fig. 3. Target robot

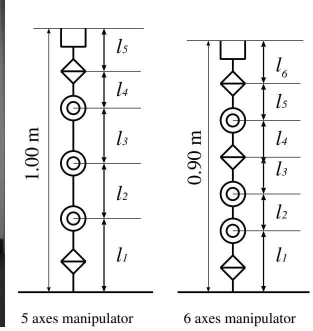


Fig. 4. Dual-arm

IV. Motion planning

"Motion planning" is one of the most important elements to realize the proposed task. Particularly, planning a position of the base robot for grasping or returning book is a critical issue for success of this task. Therefore, after detection of a book's position or return position, we calculate a work space for both end-effectors. If the book's position (or return position) is inside of the work space, the robot carries out a book's grasping (or returning) motion. If the book's position (or return position) is not inside of the work space, the base robot moves to a suitable position for grasping (or returning) the book. To define a work space, we consider a manipulability [3] of manipulators, and collision-free configuration of both manipulators.

In this section, we introduce a definition of manipulability, collision detection, and motion planning algorithm for both grasping and returning a book.

A. Manipulability

When we plan a motion of a manipulator, it is important to consider a manipulability of its end-effector. Manipulability $\omega(\boldsymbol{q})$ is represented by equation (1) using Jacobian matrix $J(\boldsymbol{q})$,

$$\omega(\boldsymbol{q}) = \sqrt{det(J(\boldsymbol{q})J^T(\boldsymbol{q}))} \tag{1}$$

where $\boldsymbol{q}$ is a vector of manipulator's joint angles. When a manipulator makes an arbitrary pose, we can calculate a manipulability as a scalar value by (1). In this research, we assume that both manipulators' $\omega(\boldsymbol{q})$ always keeps more than a fixed value in planned motion.

B. Collision Detection

To avoid interior collision for a dual-arm manipulator, we calculate the shortest distance d between combinations of manipulators' links. For instance, if each manipulator has three links, we calculate 9 combination of d. If one of them is less than a fixed value, we regard that it collides interiorly, and remove this manipulators' configuration from its work space.

C. Motion planning of grasping a book

A motion of grasping a book using a dual-arm manipulator can be realized by "one manipulator pushes the book and the other grasps it". In this motion, the pushing manipulator moves linearly toward the grasping manipulator.

In this research, a work space for dual-arm manipulator is calculated by the following procedure. (We named a grasping manipulator as "M_α", a pushing manipulator as "M_β", the end-effector of M_α as "E_α", the end-effector of M_β as "E_β", and the plane on the desk's surface as "π_α".)

1) Pick up an arbitrary point i in π_α for position of E_α. Then a position of E_α is represented by a vector $\boldsymbol{p_i}$.
2) Once a vector $\boldsymbol{a_i}$ (the direction vector of E_α) is fixed, joint angles $\boldsymbol{q_{\alpha i}}$ for M_α can be calculated uniquely by inverse kinematics. To fix $\boldsymbol{a_i}$, we must specify ϕ_α (the angle between E_α's direction and π_α) and θ_α (the angle between E_α's direction and base robot's direciton) shown in Fig.5 and Fig.6.
   If the following equation is satisfied, go to step 3.
   $$\omega(\boldsymbol{q_{\alpha i}}) \geq \omega_\alpha \;\; (\omega_\alpha \text{ is a fixed value}) \tag{2}$$

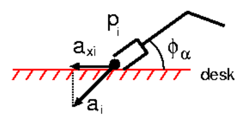


Fig. 5. Vector $\boldsymbol{a}$ and $\boldsymbol{p}$

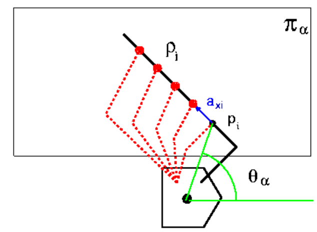


Fig. 6. Motion segment

3) Make a line l by extending the vector $\boldsymbol{a_{xi}}$ that is the horizontal component of $\boldsymbol{a_i}$ (Fig.5). Once the position of the E_β is located on l, and a horizontal component of direction of E_β is equivarent to $\boldsymbol{a_{xi}}$, joint angles $\boldsymbol{q_{\beta i}}$ can be calculated by inverse kimematics at each point on l.
   If the following equation is satisfied, go to step 4.
   $$\exists(\omega(\boldsymbol{q_{\beta i}}) \geq \omega_\beta) \;\; (\omega_\beta \text{ is a fixed value}) \tag{3}$$
4) Now a motion segment ρ_i is defined by a set of points that satisfy the equation of (3) along l, shown in Fig.6. By using collision detection method in subsection (IV-B), we remove the segment ρ_i if it collides interiorly. Moreover, we remove the segment ρ_i whose width is shorter than book's width.
5) Apply from 1) to 4) for all points in π_α. Then work space γ_α is defined by a set of motion segments ρ_i. Each segment is a candidate of E_β's path.

Based on above procedure, we can calculate a work space for a dual-arm manipulator on the desk. If there is the target book in the work space γ_α, the robot just picks up a segment that overlaps a center of gravity of the book. Then the segment becomes a path of E_β. If there is not the target book in the work space γ_α, the base robot adjusts its position to fit the book toward the work space γ_α, and detects the book's position again.

D. Example of work space to grasp a book

Here is an example of work space γ_α shown in Fig.7. To calculate the work space, we use a model of the dual-arm mobile manipulator shown in Fig.3. In this example, we define a height of the target desk as $0.70[m]$, and the book size as A5. Additionally, we assign the 5 D.O.F. manipulator as M_α, and 6 D.O.F. manipulator as M_β. In this case, θ_α is automatically determined by a position of $\boldsymbol{p}_i$ because of a lack of M_α's D.O.F.. We also set ϕ_α as $-60.0[deg]$ from a horizontal direction, and the angle range (between E_β's direction and π_α) is $\pm 30[deg]$. In this example, we set $\omega_\alpha = 3.1 \times 10^{-8}$ and $\omega_\beta = 3.2 \times 10^{-18}$ heuristically.

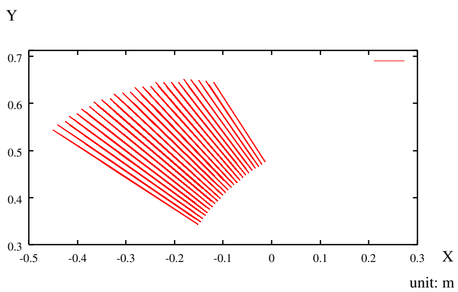


Fig. 7. Work area γ_α

E. Motion planning of returning a book

A motion of returning a book using a single manipulator can be realized by "moving the end-effector toward bookshelf linearly". In this motion, a work space for single manipulator is calculated by the following procedure. (We named the manipulator as "M_δ", the end-effector of M_δ as "E_δ", and a holisontal plane with a fixed height (the same as returning position) as "π_δ".)

1) Pick up an arbitrary point k in π_δ for position of E_δ. Then a position of the E_δ is represented by a vector $\boldsymbol{p_k}$.
2) Once a vector $\boldsymbol{a_k}$ (direction vector of E_δ) is fixed, joint angles $\boldsymbol{q_{\delta k}}$ for M_δ can be calculated by inverse kinematics. To fix $\boldsymbol{a_k}$, we must specify θ_δ (angle between E_δ's direction and base robot's direciton), shown in Fig.8.
If the following equation is satisfied, go to step 3.

$$\omega(\boldsymbol{q_{\delta k}}) \geq \omega_\delta \quad (\omega_\delta \text{ is a fixed value}) \tag{4}$$

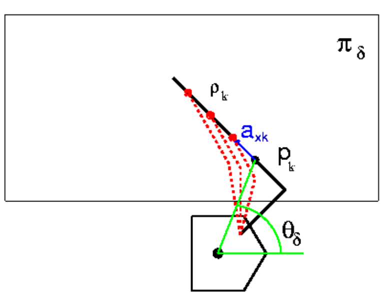


Fig. 8. Returning segment

3) Make a line l by extending a vector of $\boldsymbol{a_{xk}}$ that is a horizontal component of $\boldsymbol{a_k}$. Once the position of the E_δ is located on l, and a horizontal component of direction vector of E_δ is equivarent to $\boldsymbol{a_{xk}}$, joint angles $\boldsymbol{q_{\delta k}}$ can be calculated by inverse kimematics at each point on l.
If the following equation is satisfied, go to step 4.

$$\exists(\omega(\boldsymbol{q_{\delta k}}) \geq \omega_\delta) \quad (\omega_\delta \text{ is a fixed value}) \tag{5}$$

4) A motion segment of ρ_k for M_δ is defined by a set of points satisfying equation of (5) along l (in Fig.8).
5) Apply from 1) to 4) for all points in π_δ. Then work space γ_δ is defined by a set of motion segments ρ_k. Each segment is a candidate of E_δ's path.

Based on above procedure, we can calculate a work space for a single manipulator to return a target book. If there is the return position in the work space γ_δ, the robot just picks up a segment that overlaps the return position. If there is not the return position in the work space γ_α, the base robot adjusts its position to fit the return position toward the work space γ_δ, and detects the return position again.

F. Example of work space to return a book

Here is an example of work space γ_δ shown in Fig.9. To calculate the work space, we use a model of the mobile manipulator shown in Fig.3. In this example, we define a height of the bookshelf as $0.70[m]$. Additionally, we assign that the 5 D.O.F. manipulator as M_δ because it grasps the target book. In this example, θ_δ is automatically determined by a position of $\boldsymbol{p_k}$ because of a lack of M_δ's D.O.F.. In this example, we set $\omega_\delta = 3.1 \times 10^{-8}$ heuristically.

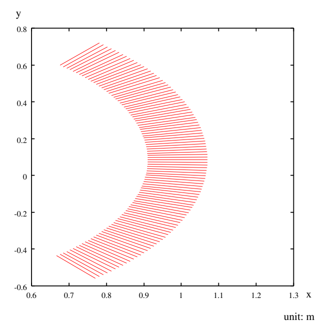


Fig. 9. Work area γ_δ

V. Navigation Method

To move toward a grasping position of a target book, or to transport a target book toward a target bookshelf, it is necessary for a base robot to navigate along a given path. In this research, we have implemented a command system [10] of tracing segments for a base robot, and the path is generated by operators intuitively.

Usually, it is very important to compensate localization error of a base robot. However, we assume that the error is small enough to ignore it.

VI. Target's Position Detection

To detect a position of a target book, or to detect a return position of the book, we use vision sensors that are mounted on end-effectors of manipulators. In this section, we explain our methods to detect target's position.

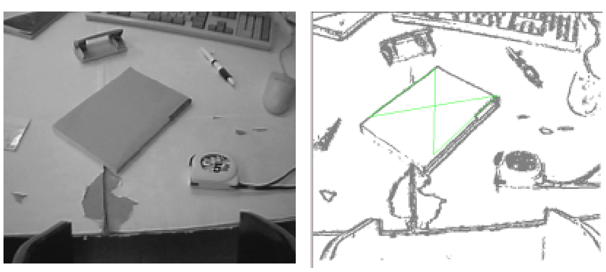
A. Detection of a Target Book's Position

Generally, it is very difficult to find a specific book's position from an image of disorder environment. Therefore, we assume that the target book is laid on a desk (height information of it is known), and the size of the book is also known.

Beforehand, we calibrate camera's parameters by [11] and generates a mapping function between image coordinates and actual coordinates. Then the robot detects a book's position by the following procedure.

1) Capture an original image that includes a target book
2) Generate an edge image using edge filter from the original image
3) Generate a binary image from the edge image
4) Map pixels from the binary image to an actual coordinates using [11]
5) Detect segments by Hough transform method in the actual coordinate
6) Pick up a pair of parallel segments that length is equal to book's width

An example of a book's detection is shown in Fig.10. The left's figure is an original image, and the right's figure includes a detected position of the book. (Diagonal lines are drawn for the detected book.)



Original image Recognition result

Fig. 10. Book's Detection

In case that an image includes a pair of parallel segments that is the same as the width of the target book accidentally (but not the target book's edge), the robot assumes that it is a book's position. In current situation, we do not cope with this problem.

B. Detection of Book's Return Position

We assume that a return position of a target book is next to a specific book in a bookshelf. Therefore detection of the return position is equivalent to detection of a known back image of the specific book. To enable the detection, we use template match method by the following procedure.

1) In advance, a template image of a back of the specific book is stored
2) Capture a image that includes the specific book
3) Perform template match between the template and the captured image
4) Rotate the template image a little bit
5) Repeat 3 and 4 to find the best match

This procedure is performed by functions of image processing board (IP5000: made by Hitachi). In this method, the template may have a size error caused by an error between the camera and the bookshelf. To compensate the error, the base robot adjusts its position using PSD sensor.

VII. Demonstration

We integrated three elements ("Motion planning", "Navigation" and "Target's Position Detection") in our dual-arm mobile manipulator, and perform a demonstration in a real environment. In this section, we introduce the demonstration scene.

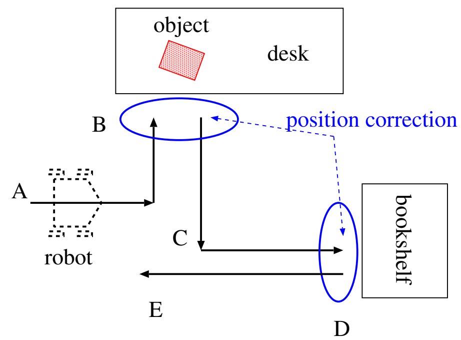


Fig. 11. Given path

First, we set a path for the target robot shown in Fig.11, and performed the robot's motion shown in Fig.12. (In the following explanation, uppercase letters are equivalent to the position in Fig.11, and lower-case letters are equivalent to the images in Fig.12.)

First, the robot navigated from the point A to the point B in (a) and (b). When the robot arrived at the point B, it detected a book's position in (c). In this case, the robot knew that the book was not located in a suitable position, so the base robot adjusted its position. After adjustment of the robot's position, the robot detected the book's position again, and it carried out a grasping motion in (d)–(g). Next, the robot navigated from the point B to the point C in (h)–(j). Using a PSD sensor, the base robot adjusted its position at the point D. Then, the robot detected a return position of the target book by detecting a back of the specified book in (k). Finally, the robot returned the book to the bookshelf in (l)–(n), and navigated to the position E in (o).

VIII. Discussion

A successful ratio of the motion of grasping a book is now about 50 %, and a successful ratio of the motion of returning book is about 60%. Therefore it still be a low successful ratio for whole motion of "tidying a book".

When the robot performed the motion of grasping the book, we have faced the following problems.

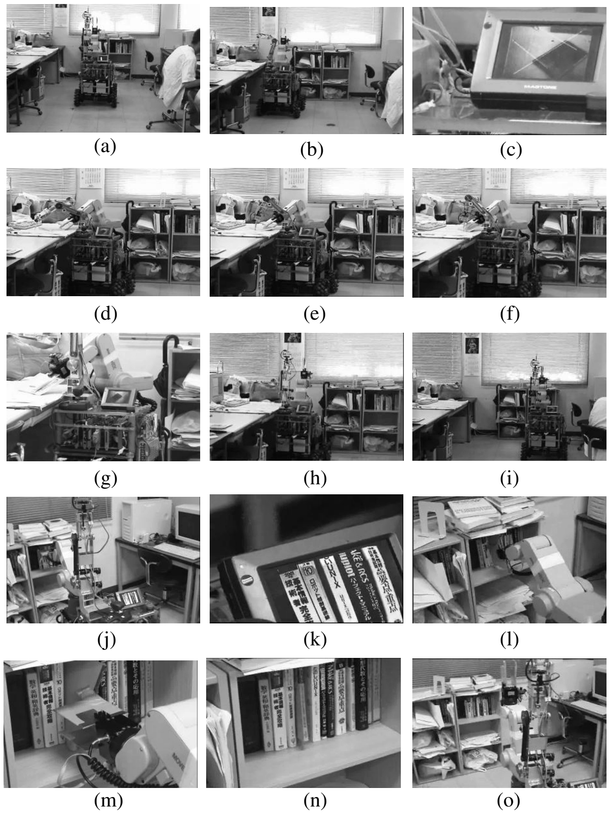


Fig. 12. A motion of tidying a book

1) Sometimes the robot fails to detect the book's position. The reason is that the robot mis-matched to wrong pair of parallel segments.
2) If the book was located at the inner part of the desk, the robot can not reach to the book. Currently the robot can work when the book locates only within $0.30[m]$ from the side of the desk.
3) The robot failed to capture an image that includes the target book because of a localization error of the base robot.
4) A localization error of the base robot at the front of bookshelf caused mis-matching of the specified book.

To solve these problems, "robust book's detection method" and "robust position adjustment method for base robot" are required. Currently, weak points of our "book's detection algorithm" are (1) only use a pair of parallel segments for book detection on a desk, and (2) only use the highest correlation for template matching in bookshelf. Both are insufficient to confirm book's location.

On the other hand, our "localization problem" is our hardware problem, because the target robot uses omni-wheels. It is difficult to control the robot's position precisely, because of omni-wheels' slippage. Our main algorithm does not require omni-directional motion, so the replacement of the base robot will improve the problem.

From the point of view of motion planning, our algorithm worked well. When the robot detected a target's position correctly, the robot succeeded for grasping a target book or returning the book to the bookshelf. However, the problem of "small work space" is impossible to solve without changing manipulator's size. Furthermore, our manipulator does not have force sensor to detect contact force of the hand, and it may become a problem in the future. Currently, backlash of our hand-made manipulator absorbs most of contact forces to a surface of a desk.

IX. Conclusion

In this paper, we introduced a realization of an intelligent motion for dual-arm mobile manipulator along a task oriented approach. Our research task is defined as "grasping a target book and returning it to bookshelf". One of the important elements is a motion planning, so the motion planning algorithm was introduced in detail. We also introduced a demonstration of the proposed task in real environment using actual robot, and reported a several problems from our experiences.

In our future works, we will solve problems shown in the section VIII to perform more robust motion of a dual-arm mobile manipulator. Additionally, we will expand a variety of intelligent motions using mobile manipulators.

References

[1] T. Tomizawa, A. Ohya, and S. Yuta, "Book browsing system using an autonomous mobile robot teleoperated via the internet," in Proc. of IEEE/RSJ International Conference on Intelligent Robots and Systems, 2002.

[2] K. Nagatani and S. Yuta, "Designing strategy and implementation of mobile manipulator control system for opening door," in Proc. of IEEE International Conf. on Robotics and Automation, pp. 2828–2834, 1996.

[3] T. Yoshikawa, "Manipulability of robotic mechanisms," The international Journal of Robotics Research, vol. 4, pp. 3–9, 1985.

[4] B.Bayle, J.Y.Fourquet, and M.Renaud, "Manipulability analysis for mobile manipulators," in Proc. of IEEE International Conference on Robotics and Automation, pp. 1251–1256, 2001.

[5] B.Bayle, J-Y.Fourquet, F.Lamiraux, and M.Renaud, "Kinematic control of wheeled mobile manipulators," in Proc. of the IEEE/RSJ International Conference on Intelligent Robots and Systems, pp. 1572–1577, 2002.

[6] J.P.Desai and V.Kumar, "Nonholonomic motion planning for multiple mobile manipulators," in Proc. of IEEE International Conference on Robotics and Automation, 1997.

[7] K.Nagatani, T.Hirayama, A.Gofuku, and Y.Tanaka, "Motion planning for mobile manipulator with keeping manipulability," in Proc. of the IEEE/RSJ International Conference on Intelligent Robots and Systems, pp. 1663–1668, 2002.

[8] O.Brock and O.Khatib, "Real-time replanning in high-dimensional configuration space using sets of homotopic paths," in Proc. of IIEEE International Conference on Robotics and Automation, pp. 550–555, 2000.

[9] Y.Yamammoto and X.Yun, "Unified analysis on mobility and manipulability of mobile manipulators," in Proc. of IEEE International Conference on Robotics and Automation, pp. 1200–1206, 1999.

[10] S. Iida and S. Yuta, "Vehicle command system and trajectory control for autonomous mobile robots," in Proceedings of International Workshop on Intelligent Robots and Systems, pp. 212–217, 1991.

[11] R.Y.Tsai, "A versatile camera calibration technique for high-accuracy 3d machine vision metrology using off-the-shelf tv cameras and lenses," IEEE Journal of Robotics and Automation, vol. 3, no. 4, pp. 323–344, 1987.