Tight Coupling between Manipulation and Perception using SLAM

Benzun Pious Wisely Babu
Robotics Engineering Department
Worcester Polytechnic Institute
Worcester, MA 01609
Email: bpwiselybabu@wpi.edu

Christopher Bove
Robotics Engineering Department
Worcester Polytechnic Institute
Worcester, MA 01609
Email: cpbove@wpi.edu

Michael A. Gennert
Robotics Engineering Department
Worcester Polytechnic Institute
Worcester, MA 01609
Email: michaelg@wpi.edu

***Abstract*—A tight coupling between perception and manipulation is required for dynamic robots to react in a timely and appropriate manner to changes in the world. In conventional robotics, perception transforms visual information into internal models which are used by planning algorithms to generate trajectories for motion. Under this paradigm, it is possible for a plan to become stale if the robot or environment changes configuration before the robot can replan. Perception and actuation are only loosely coupled through planning; there is no rapid feedback or interplay between them. For a statically stable robot in a slowly changing environment, this is an appropriate strategy for manipulating the world. A tightly coupled system, by contrast, connects perception directly to actuation, allowing for rapid feedback. This tight coupling is important for a dynamically unstable robot which engages in active manipulation. In such robots, planning does not fall between perception and manipulation; rather planning creates the connection between perception and manipulation. We show that Simultaneous Localization and Mapping (SLAM) can be used as a tool to perform the tight coupling for a humanoid robot with numerous proprioceptive and exteroceptive sensors. Three different approaches to generate a motion plan for grabbing a piece of debris is evaluated using for Atlas humanoid robot. Results indicate higher success rate and accuracy for motion plans that implement tight coupling between perception and manipulation using SLAM.**

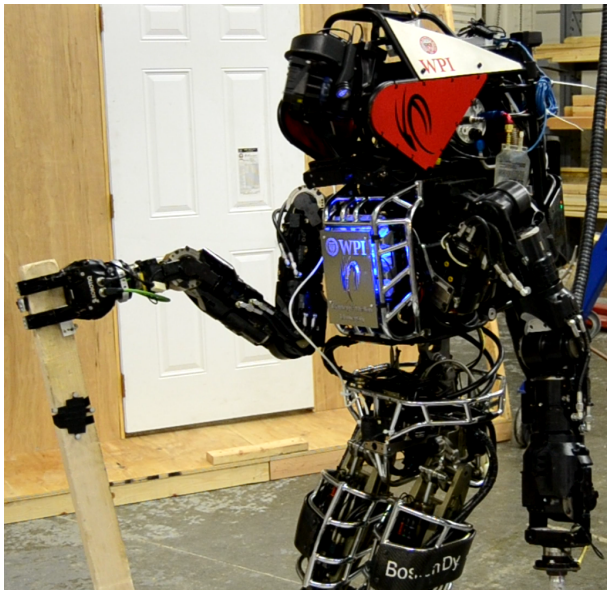

Fig. 1: The Atlas robot performing debris extraction.

I. Introduction

A conventional robotics system relies broadly on three subsystems - perception, planning and actuation. Perception receives raw information about the world from sensors and extracts meaningful internal models. Planning uses the models from perception to generate sets of trajectories for motion. Finally, actuation performs controls to ensure that the robot achieves the desired trajectory. This approach is successful in a statically stable robot with slowly changing environment that can be predicted reliably.

However, in a tightly coupled system, actuation is directly controlled by perception and perception is directly connected to actuation. Planning provides the overall goal for the system and affects both perception and actuation indirectly. In this manner the planner can produce motion trajectories with less accuracy and more robustness.

In order to perform manipulation tasks in an active environment the robot needs to have robust plans that account for changes both in its configuration and the environment. In such a condition, tight interaction between perception and manipulation can be used to create a dynamic planner.

Simultaneous Localization and Mapping (SLAM) [1] is a common technique used in mobile robots to perform estimation of both the robot position and the environment. Proprioceptive information on the robot's internal state, combined with the estimated robot position and kinematics, provide an overall estimate of the robot's total state. The generated map gives extroceptive information about the environment. The proprioceptive information is observed at a higher rate and is used for performing a reactive control of the robot while the extroceptive is slow and provides goals for manipulation tasks. Hence we propose that SLAM is a suitable choice for tightly coupling perception and manipulation for generating motion plans.

In this paper we present the application of Visual SLAM as a tool for tightly coupling perception and manipulation tasks in a humanoid robot. We demonstrate our approach through an example task of extracting a piece of debris. We examine the success of the three different models of motion

plan generation:

- a conventional static motion planner,
- a dynamic motion planner with loosely coupled visual servoing which accounts for the environment but not the change in the robot configuration, and
- a dynamic motion planner with tightly coupled visual servoing that uses Visual SLAM to account for both the environment and robot configuration.

The following section gives a brief background of existing approaches using reactive planning for manipulation in humanoids. A brief overview of the Atlas robot is presented in Section III. It is followed by an overview of the system architecture in Section IV. The three motion plan created for grabbing the piece of debris is presented in Section VI and results of the application of the different motion plans are presented in Section VII. A summary of results and future work is presented in section VIII.

II. Background

Chitta et al. [2] demonstrated a pick and place robotic system that integrates both perception and manipulation in a reactive manner. Even though sense-plan-act paragdim was implemented, they showed that coupling perception with actuation can produce robust manipulation capability in a cluttered environment. In contrast to our approach, they used a statically stable robot that does not have a dynamic control layer.

A reactive planning strategy for reach task in a cluttered environment is demonstrated by Kanehiro et al. [3]. The reactive planner works by performing both planning and execution in parallel. Similar to Atlas, a full body motion planner [4] was incorporated. In Simulation the robot was able to avoid obstacles and reach a valve.

Stasse et al. [5] discuss the integration of walking with planning in a humanoid HRP-2. Visual SLAM was used as one of the tools for the integration of environment information. The humanoid performed stacks of behaviour in parallel with priority, similar to subsumption architecture [6].

G-SL(AM)^2, introduced by Zhang and Trinkle [7], is motivated towards the problem of grasping while simultaneously estimating the model and position of the object. The algorithm uses a particle filter to perform estimation. The algorithm is not real time and uses a camera external to the robot for making observations. Our approach is real time and relies on a rotating laser scanner mounted on the robot's head.

Numerous approaches have been demonstrated in literature [8] that use vision as a feedback for control. In our work we have avoided the "eye in the hand" approach as it required modifications to the Atlas robot. Also the field of view of the cameras on the Atlas robot is limited to a narrow region infront of the robot, preventing approaches that require tracking a visual fiducial on the robot's hand using cameras.

The Atlas robot can be thought of a camera/laser scanner on a kinematic chain. There have been approaches to perform tracking of kinematic chains such as DART [9]. DART relies on using a dense 3D sensor external to the robot. In our system, the 3D sensor is on the head and can only observe parts of the robot body.

Klingensmith et al. [10] presented an approach for real time visual servoing by tracking the robot arm model. Through optimization they were able to remove kinematic errors and successfully demonstrated manipulation of objects. However, Atlas head does not have a yaw degree of freedom restricting our capability to implement a similar approach.

III. Atlas Robot

Atlas is a hydraulically actuated humanoid robot developed by Boston Dynamics. It has 28 degrees of freedom, and each joint has closed loop force and position control. For the research reported here, a Robotiq's 3 finger adaptive gripper was used as the end effector. A dynamic reactive controller [4] is implemented to maintain the stability of the robot while standing.

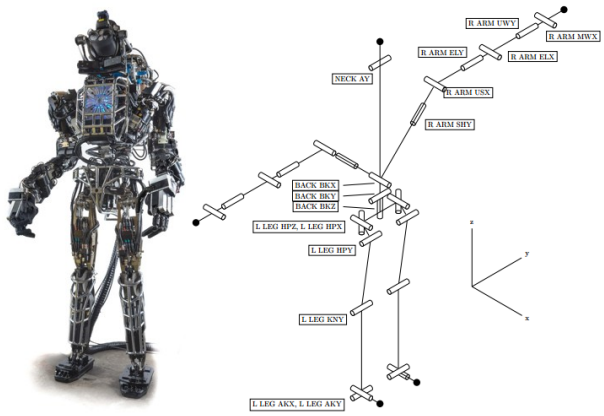

Fig. 2: The Atlas robot with the joint diagram. Observe that only pitch degree of freedom is available at the neck.

Atlas has a Multisense SL head developed by Carnegie Robotics, which consists of a spinning lidar with a stereo camera pair for sensing the environment. The multisense head generates stereo image pairs at 30 FPS, with a resolution of 1280x760. An on-board FPGA is used to generate disparity maps at 30 FPS. The head also has a rotating lidar. The laser scanner has a 270°field of view with 0.25°resolution. Each lidar scan sweep along a single plane occurs at 40Hz.

The robot neck joint has only pitch degree of freedom. Yaw can only be achieved using full body motion of the robot. Observations from the joint sensors are available through a transform tree that is generated using ROS [11]. A state estimator [12] that uses the inertial measurement unit and the joints encoders is used for full body control of the robot.

IV. System Architecture

Fig. 3 shows the system architecture. Atlas SLAM uses a stereo camera pair to estimate the map of the world and the position of the robot. The map generated by SLAM allows the robot to be rooted onto a fixed coordinate frame. The laser assembler uses the estimates of the robot position to assemble a scan consistent with the fixed frame. Images from

the stereo camera pair are combined with the assembled laser scan for object detection. Object detection is guided by the user by scribbling on the displayed object of interest (Fig. 4a) in the image. Next, the major axis of the the object (Fig. 4c) is used to create candidate samples for the grasp selector (Fig. 4d). The grasp selector uses the robot center of mass and the stability of the trajectory to select suitable grasp from the samples. Finally a complete trajectory is made based on the motion planner. The motion planner will be discussed in more detail in Section VI.

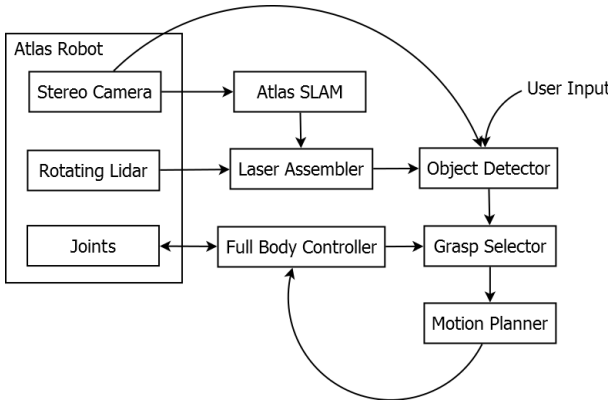

Fig. 3: The system architecture used for grabbing objects.

V. Atlas SLAM

Atlas SLAM is based on the Scalable Visual Simultaneous Localization and Mapping algorithm (ScaViSLAM) [13]. It uses a bundle adjustment based technique to estimate the position of the robot and the environment. The algorithm is defined in two logical blocks: a front end where the features are detected using FAST [14] and a back end where non-linear optimization is carried out to perform mapping. The initial position of the pelvis of the robot is used as the root frame, that is, the origin for the map. The selection of this frame allows us to easily integrate the SLAM information with the kinematic pose estimates of the robot.

ScaViSLAM has near real-time performance, but in order to improve speed and reduce computational load, An image similarity measure allows us to skip processing unchanged frames during periods when the robot is stationary, thereby speeding up processing. The similarity measure is based on the histogram of the images.

Since the robot requires information only about its immediate environment, it is not necessary to have a large map in memory. The double window optimization strategy [13] helps in marginalizing the estimates from information that is not in the immediate environment of the robot.

A. Laser Assembler

The laser assembler rectifies and projects the laser scan onto the map coordinate frame of the robot. Each scan point is transformed based on pose estimates generated by the SLAM. These transforms are synchronized with the camera frame rate and not the laser sweep rate, hence the transforms need to be interpolated before being used for projection. Each scan contains 1081 points. The spindle rotates at 5 RPS.

If $T_1 = e^{x_1} \in SE(3)$ represents the pose of the laser scanner at the start of a scan and $T_2 = e^{x_2} \in SE(3)$ represents the pose at the end of the scan. The transform that is applied to each point on the laser scan is described by Equation 1.

$$T(i) = e^{\frac{i}{1081}x_1 + \frac{1081-i}{1081}x_2} \tag{1}$$

The laser assembler stacks the rectified scans based on the sweep angle after interpolation. A complete sweep is generated when the laser scanner rotates by π radians.

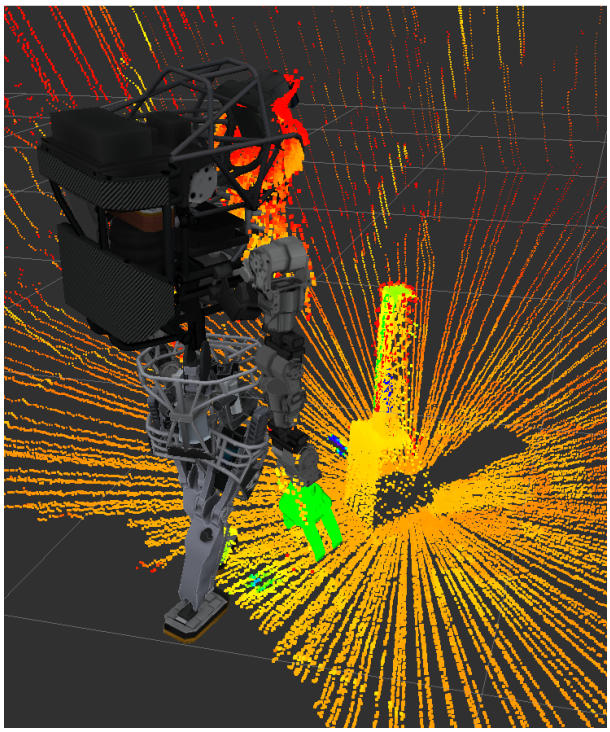

Fig. 5: A complete 360°sweep of the laser scan assembled as one point cloud.

B. Visual Servoing

Kinematic error is observed in end effector frame when the robot moves due to the drift in robots' internal state estimator and the tracking error in the controller. Hence the final approach for the grasp is guided using visual servoing.

Visual servoing is performed to guide the robot hand to the piece of debris using the laser scanner that is attached to the robot head. From the assembled scan, the object of interest is segmented. The segmented points are clustered to remove outliers. The yaw and offset from the mean position of the debris (X_{debris}) and the robot hand (X_{hand}) is used to correct the grasp approach.

$$yaw = atan(\frac{\delta y}{\delta x}) \tag{2}$$

$$offset = \|X_{hand} - X_{debris}\|^2 \tag{3}$$

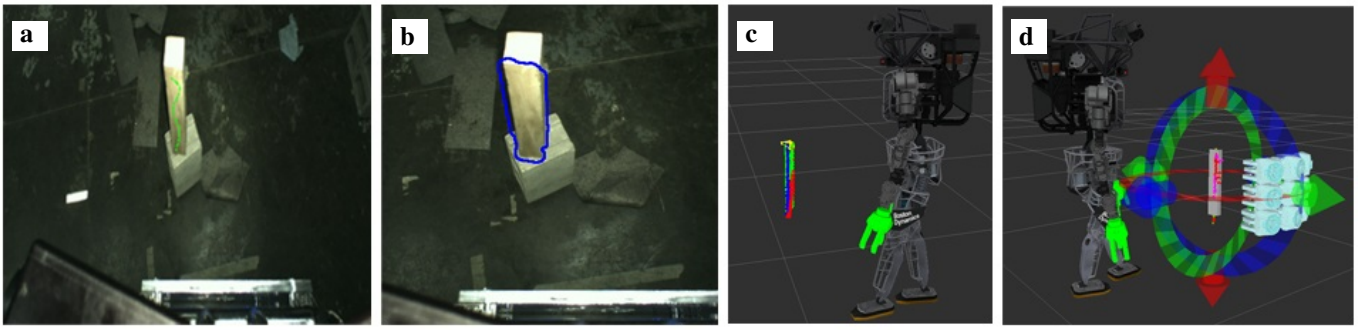

Fig. 4: The different components involved in grabbing the piece of debris: a. The user marks the piece of debris using a green scribble; b. The piece of debris is segmented using the object detector; c. The major axis of the debris is extracted from the laser scan; d. The grasp selector generates sample grasps.

The yaw and offset calculated by equation 2&3 is used to change the desired hand pose for the controller. In order to reduce feedback delay the laser spindle is rotated at the max rate of 5rps. The changes for desired end effector pose in task space is used to update reference joint angles for the robot using a gradient based Inverse Kinematics approach [4].

VI. Approach for grabbing

Three different methods to generate motion plans were carried out. In all the three plans, an extraction trajectory based on a predefined path was executed once the robot grabbed the object of interest. The motion plans are discussed briefly below.

A. Static motion plan

In a static motion plan there is no feedback from perception system once the plan is generated. The plan can be aborted and a re-planning can be performed if the robot fails to execute the plan correctly. Fig. 6 shows the process followed in the plan.

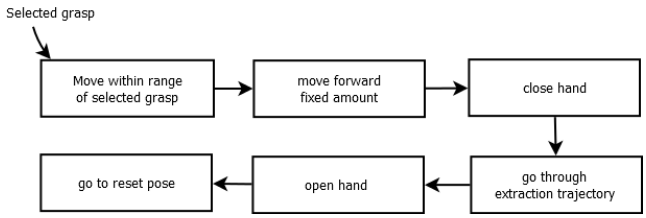

Fig. 6: Static motion plan processing flow.

B. Dynamic motion plan with loosely coupled visual servoing

Dynamic motion plan with loosely coupled visual servoing is performed assuming that only the robot's hand move and the rest of the body is stationary (Fig. 7). Since in this approach there is no SLAM to provide feedback to the laser assembler, the scans are not compensated for any motion that can occur due to movement of the robot's head (Fig. 8). This is similar to the approach followed in a statically stable robot.

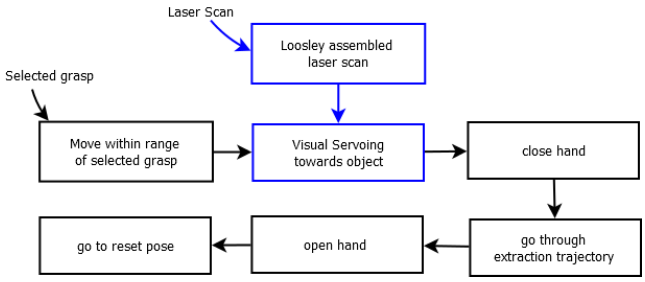

Fig. 7: Loosely coupled motion plan processing flow.

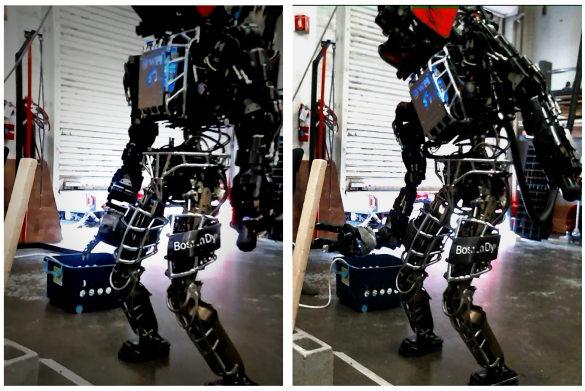

Fig. 8: The left image is the initial configuration of the robot. The right image shows the configuration when the robot is about to grab the piece of debris. Observe the change in the position of the head which affects visual servoing.

C. Dynamic motion plan with tightly coupled visual servoing

The motion plan is made using feedback from Atlas SLAM which estimates the robot configuration. The state estimator of the robot is further improved by the SLAM estimates. This allows the laser assembler to be fixed to a world frame instead of a moving frame on the robot (Fig.8). The scans are compensated for the movement of the entire robot when they are assembled. This assembled scan is used for visual servoing (Fig. 9).

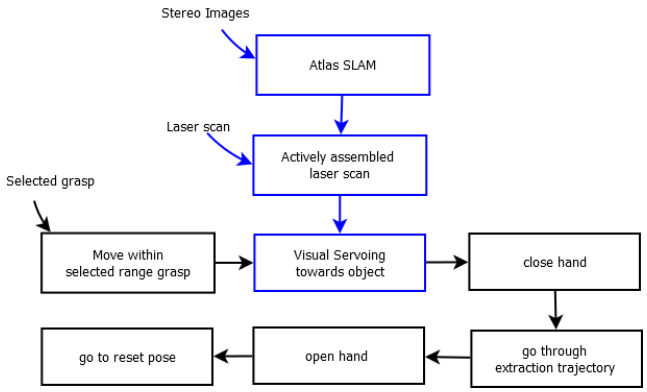

Fig. 9: Tightly coupled motion plan processing flow.

VII. Results

The experiments were carried out using the Atlas humanoid robot. The task of grabbing a piece of debris was used to evaluate the approach. An object detection module was used to segment the piece of debris. A grasp based on the major axis of the piece of debris was automatically selected by the robot. The plan for grabbing the piece of debris was generated using three different motion plans. Multiple attempts under the three different motion plans were carried out. In each attempt the robot position, object location and orientation were chosen randomly. Here we discuss the results from the different motion plans that were tested.

TABLE I: Results for grabbing a piece of debris

Motion Plan	# Attempts	# Success	% Success	Avg Completion Time(sec)
Static plan	6	0	0%	-
Dynamic plan with loose coupling	20	11	55%	57.266
Dynamic plan with tight coupling	20	16	80%	48.075

We evaluate the approach based on success rate, accuracy and completion time. Accuracy and average completion time is defined only for successful attempts. Table I shows the results of the experiments. During grasping the robot must not perturb the piece of debris until it closes the gripper as it can push the piece of debris over. Hence we define accuracy as the root mean squared error in the pose of the object from its initial stationary pose before the object is extracted (Fig. 10). The completion time accounts for the complete process of debris removal.

A. Static motion plan

We observed that the robot completely failed to extract the piece of debris. In most of the attempts the error in accuracy from the object detection algorithm affected the robot motion. In situations where the object detection algorithm was accurate, the error in kinematics and motion of the robot led to failure in accomplishing the task. Since we could not get the robot to grab the piece of debris, the accuracy and completion time was not evaluated.

B. Dynamic motion plan with loose coupling

The inclusion of visual servoing greatly improved the ability of the robot in extracting the piece of debris. The errors in object detection and robot kinematics were compensated by the visual servoing. In situations where the robot moved from its initial configuration for balance, the visual servoing failed because the effect of change in the robot head was neither being observed nor accounted for. This causes the robot to perturb the piece of debris before grabbing. This is visible in Fig. 10 shows that compared to the tightly coupled visual servoing, loose coupling has more position and orientation error and is thus less accurate. The additional time that is required for correcting these perturbations make this approach slightly slower (Table I).

C. Dynamic motion plan with tight coupling

There was better alignment and accuracy in the robots extraction attempts. This was because the motion of the robot's body was observed by visual SLAM and was compensated during visual servoing. There were still failures due to delay in updates from the SLAM. There are cases where the position of the debris infront of the robot make the inverse kinematics solutions to be unstable for the robot.

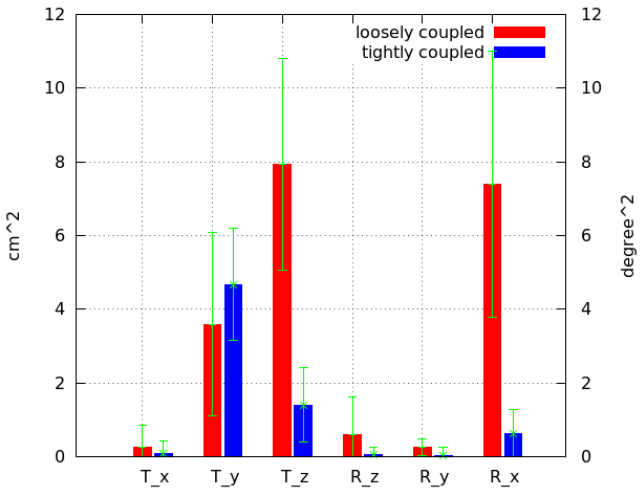

Fig. 10: The accuracy is measured in terms of RMS error: red - The RMS error in loosely coupled visual servoing. blue - The RMS in tightly coupled visual servoing. T_x, T_y, T_z are transnational errors measured in cm^2 and R_x, R_y, R_z are orientation errors measured in $degree^2$.

VIII. Conclusion and Future Work

For a static plan to work in a dynamic robotic system there is a need for very accurate observation of the environment and accurate prediction of the motion plan. It is possible to overcome these drawbacks by using a dynamic planning

model that couples perception and manipulation. Even though a loosely coupled motion plan performs better than a static motion plan, there is a need for tight coupling in dynamic robots whose configuration changes. We show that SLAM can be used as a tool for tight coupling between perception and manipulation in such situations.

This paper reports on results where the robot is dynamic, but the environment is static. More work needs to be done on selection of the robot foot placement before grabbing the debris. Future work will involve more rigorous testing on dynamic environments to validate the approach. This approach will be applied to other tasks, including manipulating tools, walking, opening doors, turning valves, and driving.

ACKNOWLEDGMENT

The authors would like to thank the other members of team WPI-CMU for their assistance conducting experiments. This work is sponsored by Defense Advanced Research Project Agency, DARPA Robotics Challenge Program under Contract No. HR0011-14-C-0011. We also acknowledge our corporate sponsors NVIDIA and Axis Communications for providing equipment support.

REFERENCES

[1] R. C. Smith and P. Cheeseman, "On the representation and estimation of spatial uncertainly," *Int. J. Rob. Res.*, vol. 5, no. 4, pp. 56–68, Dec. 1986. [Online]. Available: http://dx.doi.org/10.1177/027836498600500404

[2] S. Chitta, E. G. Jones, M. Ciocarlie, and K. Hsiao, "Perception, planning, and execution for mobile manipulation in unstructured environments," *IEEE Robotics and Automation Magazine, Special Issue on Mobile Manipulation*, vol. 19, 2012.

[3] F. Kanehiro, E. Yoshida, and K. Yokoi, "Efficient reaching motion planning and execution for exploration by humanoid robots," in *Intelligent Robots and Systems (IROS), 2012 IEEE/RSJ International Conference on*, Oct 2012, pp. 1911–1916.

[4] S. Feng, E. Whitmanc, X. Xinjilefu, and C. G. Atkeson, "Optimization based full body control for the atlas robot," in *Humanoids*, 2014, p. submission.

[5] O. Stasse, B. Verrelst, A. Davison, N. Mansard, B. Vanderborght, C. Esteves, F. Saidi, and K. Yokoi, "Integrating walking and vision to increase humanoid robot autonomy," in *Robotics and Automation, 2007 IEEE International Conference on*, April 2007, pp. 2772–2773.

[6] R. Brooks, "A robust layered control system for a mobile robot," *Robotics and Automation, IEEE Journal of*, vol. 2, no. 1, pp. 14–23, Mar 1986.

[7] L. Zhang and J. Trinkle, "The application of particle filtering to grasping acquisition with visual occlusion and tactile sensing," in *Robotics and Automation (ICRA), 2012 IEEE International Conference on*, May 2012, pp. 3805–3812.

[8] P. I. Corke, "Visual control of robot manipulators – a review," in *Visual Servoing*. World Scientific, 1994, pp. 1–31.

[9] T. Schmidt, R. Newcombe, and D. Fox, "Dart: Dense articulated real-time tracking," in *Proceedings of Robotics: Science and Systems*, Berkeley, USA, July 2014.

[10] M. Klingensmith, T. Galluzzo, C. Dellin, M. Kazemi, J. A. D. Bagnell, and N. Pollard, "Closed-loop servoing using real-time markerless arm tracking," in *International Conference on Robotics And Automation (Humanoids Workshop)*, May 2013.

[11] T. Foote, "tf: The transform library," in *Technologies for Practical Robot Applications (TePRA), 2013 IEEE International Conference on*, ser. Open-Source Software workshop, April 2013, pp. 1–6.

[12] X. Xinjilefu, S. Feng, W. Huang, and C. Atkeson, "Decoupled state estimation for humanoid using full-body dynamics," in *International Conference on Robotics and Automation*. IEEE, 2014.

[13] H. Strasdat, A. J. Davison, J. M. M. Montiel, and K. Konolige, "Double window optimisation for constant time visual slam." in *ICCV*. IEEE, 2011, pp. 2352–2359.

[14] E. Rosten and T. Drummond, "Fusing points and lines for high performance tracking." in *IEEE International Conference on Computer Vision*, vol. 2, October 2005, pp. 1508–1511.