

# Team-JSK: MBZIRC Progress Report

Team JSK<sup>†</sup>

JSK Lab, Graduate School of Information Science and Technology, The University of Tokyo  
7-3-1 Hongo, Bunkyo-ku, Tokyo, Japan 113-8656.

## I. INTRODUCTION

This document provides a report of Team JSK's progress in preparing for the Mohamed Bin Zayed International Robotics Challenge (MBZIRC). The team consists of members from the JSK Laboratory at the University of Tokyo. The JSK Lab, founded in early 1980s, has a long history of robotics research with focus on areas including humanoids, drones, robotics manipulation, and perception, and the lab has experience in participating in robotics challenges including the DARPA Robotic Challenge and the Amazon Picking Challenge. JSK lab is also an active contributor to the ROS community.

## II. PROJECT PERSONNEL

Team JSK is made of eleven members: Prof. Masayuki Inaba, Prof. Kei Okada, Dr. Yohei Kakiuchi, Dr. Wesley Chan, Bakui Chou, Xiangyu Chen, Krishneel Chaudhary, Kohei Kimura, Yuki Furuta, and Hiroto Mizohana. The team is roughly divided into three groups corresponding to each task with groups having overlapping personnel.

## III. CHALLENGE 1: LANDING UAV ON A MOVING VEHICLE

### A. Hardware

We developed a hexarotor UAV as shown in Fig.1. As described in Fig.2, this aerial robot consists of onboard sensors such as IMU, barometer, laser sensor, GPS for basic hovering flight control, as well as an original flight controller and a high level processor. The monocular camera is installed for egomotion estimation. The total weight of the UAV is 4.3 kg, while the flight time can reach 20 min with heavy vision processing on the onboard processor. We have achieved outdoor flight with autonomous altitude hold mode using our original sensor fusion algorithm (Fig.3).

### B. Software

The software, including motion planning, visual perception and virtual simulation, are developed on the Robot Operating System (ROS) environment, and we use Gazebo for performing simulations<sup>1</sup>. We use the Gazebo simulator for testing and planning our strategy and for customizing our hardware and software. The visual perception component carries out target (heliport) localization of the moving

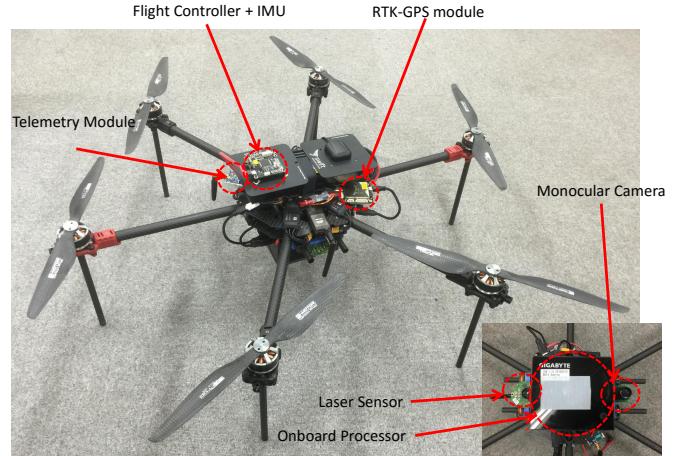


Fig. 1: Image of task 1 UAV (Hawk)

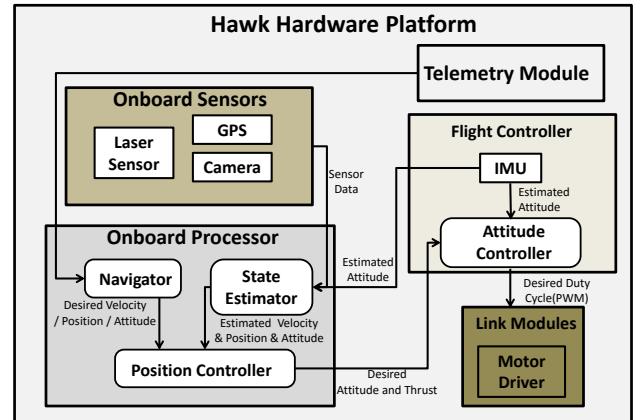


Fig. 2: Hardware platform of Hawk

vehicle, and we plan the efficient approaching and landing strategies based on the motion of both the UAV and the vehicle. Since we use Nvidia TX1 embedded processor for fast computations on GPU, our algorithms for *task 1* and *task 3* are developed in CUDA-C, C/C++ and Python.

### C. General Approach

We use the heliport model to train a linear classifier for detection of the landing region. Since the heliport model is known, it is used as an *a priori* for learning. Once the heliport is detected, a visual object tracker running in real time on-

\* <http://www.jsk.t.u-tokyo.ac.jp>

<sup>1</sup>[https://github.com/start-jsk/jsk\\_mbzirc](https://github.com/start-jsk/jsk_mbzirc)



**Fig. 3:** Image of outdoor flight

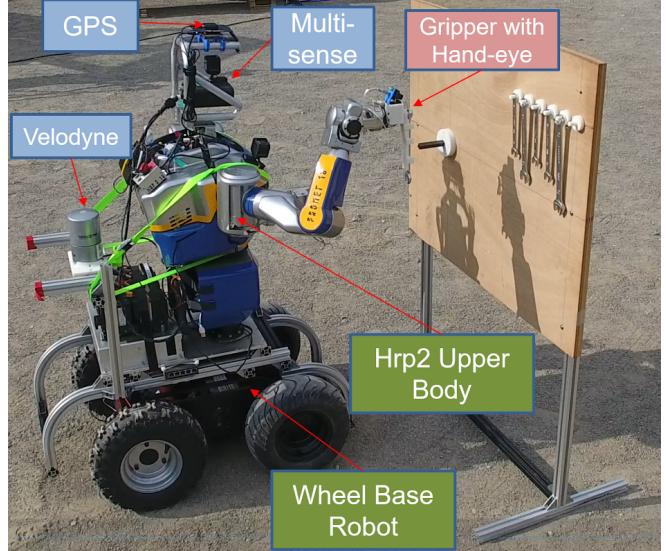
board is autonomously initialized to start tracking the target region. We use a robust tracking algorithm with efficient drift compensation to avoid lost of target when the UAV is in motion. Our visual tracking algorithm is also capable of recovering the target even if it went completely out of view. Once the target is localized, the UAV uses pose information from the visual tracker to navigate towards the target and plan out a landing strategy.

#### D. Results Achieved to Date

In this task, we have completed target detection, tracking and autonomous landing in a Gazebo simulator. Moreover we have also tested our tracking algorithm on an outdoor video sequence of a heliport marker affixed to a vehicle travelling around a track at various speeds. The video was taken from a DJI Phantom 4 and processed offline to test the stability, robustness and speed of tracking.

#### E. Future Plans

The future work on hardware platform for task 1 includes the design of landing gear which will enable the UAV to attach itself onto a moving heliport. A sturdy and light protector for the UAV is also needed to prevent the landing impact with the truck from damaging the UAV. The future work on software for task 1 involves testing the completed software on the customized UAV which is currently under development. This involves fine tuning the current simulator version of our software. Considering the challenges in outdoor environment such as abrupt changes in image space, winds speeds etc. the landing strategy might vary significantly from the simulator version. One very important aspect of autonomous systems which we like to implement is the ability of the UAV to recover from erroneous decisions and false positives that might result from highly cluttered and unstructured scenes. To achieve this, we plan to generate a map of the environment at the beginning for efficiently localizing the vehicle and for trajectory mapping. The idea is that the UAV can use the constructed map to eliminate regions that produce false positives. However, the current



**Fig. 4:** HRP2 robot platform for task 2

limitation is the time required for generating the map which we aim to reduce by using a task oriented approach.

## IV. CHALLENGE 2: OPERATING A VALVE STEM

### A. Hardware

Our robot consists of an upper body humanoid (HRP2) on a high-power mobile base as shown in Fig.4. The robot is equipped with a stereo camera, a long range laser sensor, a global positioning system (GPS), and a custom made gripper. The gripper consists of a magnet embedded link actuated by a servo motor as shown in Fig.5. The wrist is also equipped with a six axis force torque sensor.

The HRP2 is one of our lab's many robots and has been in use for a very long time. It is relatively mature and we have created a lot of software packages for this robot, such as the lisp based programming language euslisp. We acquired a mobile wheel base robot platform to replace the legs of the humanoid robot and designed both the control hardware circuits and the ROS control interface for the moving base.

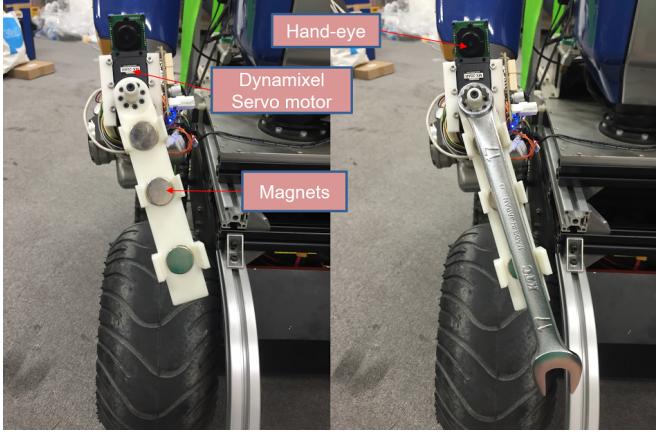
### B. Software

For task 2, the software is also implemented on ROS with utilization of multithreading for fast computation. Euslisp, an integrated robotics programming system developed by the JSK lab was used for kinematics simulation and robot control. OpenCV and in-house developed algorithms<sup>2</sup> are used for recognition and perception.

### C. General Approach

**Navigation:** Our approach is to use the long range laser sensor and GPS positioning for searching and navigating to the panel when the robot is at a far distance from the panel, and the panel is out of range for the stereo sensors. As the panel becomes closer than the minimum range of the laser sensor, the robot will then switch over to use the stereo

<sup>2</sup>[https://github.com/jsk-ros-pkg/jsk\\_recognition](https://github.com/jsk-ros-pkg/jsk_recognition)



**Fig. 5:** Custom made magnetic gripper.

camera. Our high-powered mobile base can reach up to 4 m/s and allows the robot to drive through various outdoor terrains.

**Wrench and valve stem detection:** We experimented and compared infrared camera with stereo camera, and we decided to use stereo camera for close range perception, since infrared cameras tend to fail in outdoor environments subjected to lighting variations, and cannot sense objects that are too close to the robot.

We detect the wrench by using edge detection, Hough transform, and K-means. First, we select a region that includes the 6 wrenches on the camera image as shown in the top image of Fig.6A. We apply Hough line transform to obtain the edge image of the selected region and extract lines which have large slopes. Then, we apply K-means ( $K = 6$ ) to the extracted lines. The size of wrenches can be estimated from the classified lines, but the position of the wrenches estimated from the lines, illustrated in Fig.6B, is sometimes not precise enough for grasping. Therefore, we use Hough circle transform to detect the hanger (circular part of the wrench) on the image (Fig.6C) and obtain its depth from the point cloud data (Fig.6D). Using the described method, we can detect the size and position of the wrenches.

For detecting the valve stem, we also select the region of interest on the camera image (bottom image of Fig.6A). We estimate the plane from the point cloud in the selected region and extract the points in front of the plane (Fig.6E). The centroid of the extracted points is considered as the position of the valve stem (Fig.6F).

**Picking the wrench:** Our robot picks up the wrench by positioning the gripper motor axis in front of the detected wrench hanger, aligning the gripper with the wrench, moving the gripper towards the wrench, and letting the magnets pull the wrench into the gripper. The gripper is designed so that the wrench is grasped firmly, but with some movement possible for passive compliance.

**Wrench fitting and turning:** To fit the wrench head onto the valve stem, we use force feedback from the wrist to gauge the tool contact state. The robot first moves its gripper above the detected position of the valve stem. Due to error

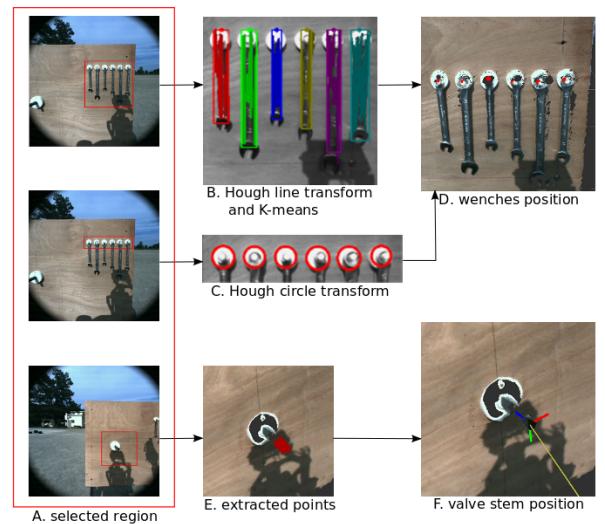
in detection or calibration, the wrench head could be directly above the valve stem or it can be slightly misaligned as shown in Fig.7A and B. The robot moves its gripper down, until a force in the vertical direction is detected, indicating that the wrench has come in contact with the valve stem. Once it detects contact with the valve stem, the wrench can be in one of the contacts states as shown in Fig. 7C - G. The robot then moves its gripper in the horizontal direction in a widening zigzag pattern. Depending on the forces it detects, the robot then begins to turn the wrench, or adjusts its gripper position and retries to fit the wrench (Fig.7).

#### D. Results Achieved to Date

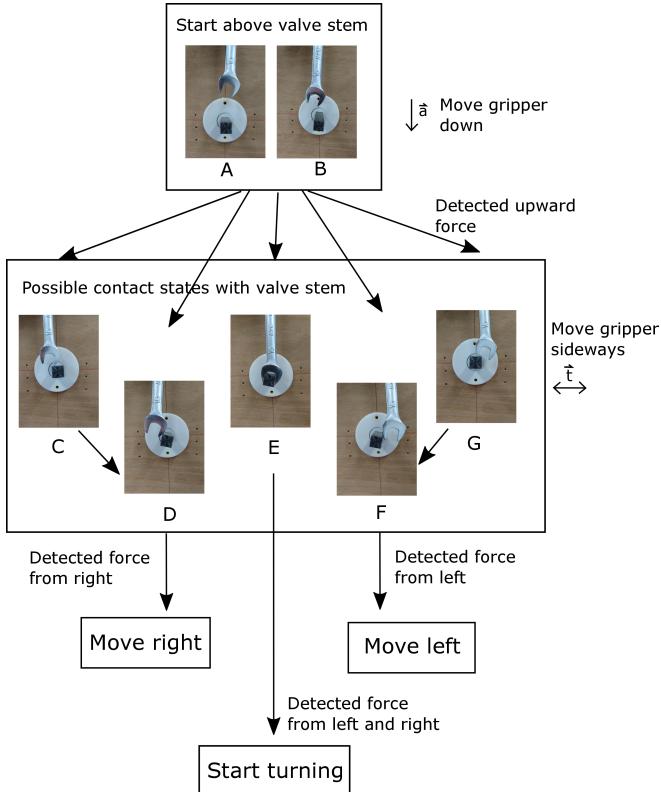
We have completed the prototypes of our mobile base and customized gripper. The entire robot has been assembled and all sensors are functional. The robot can be operated through tele-operation, and we have been able to successfully complete challenge 2 using full tele-operation indoors and outdoors. Recognition of the wrenches and the valve stem has also been implemented. Once we select the region to detect wrenches and valve stem, they will be detected autonomously.

We experimented with wrench fitting and turning with different initial wrench alignments. Among twenty trials, we were able to achieve a 95% success rate with only one failed trial.

We have also tested our system on performing the entire challenge 2 with partial autonomy in outdoor experiments. In our experiments, we used tele-operation to drive the robot's mobile base, allowed the robot to detect the wrenches and valve stem with human supervision, and then grasp, fit, and turn the wrench with full autonomy. In our fastest run, we were able to complete challenge 2 in less than ten minutes. This time can be easily shortened as many parts of our code had deliberate pauses for debugging and testing purposes.



**Fig. 6:** Wrench and valve stem detection.



**Fig. 7:** Using force feedback for wrench fitting

### E. Future Plans

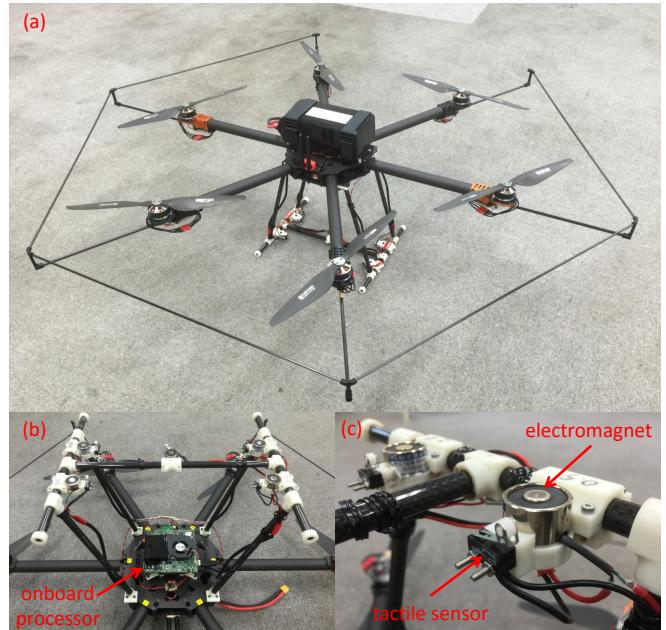
Our future plans include speeding up our task completion time, enabling autonomous navigation of the mobile base, autonomous search of the panel, full autonomous wrench and valve stem detection, and failure detection when grasping, fitting, and turning the wrench. We will also consider and compare alternative wrench detection and wrench fitting methods. Currently, the valve stem we have been operating has very little resistance. While we have successfully turned a valve stem with 5Nm resistance, our gripper prototype broke after turning a quarter turn. We have already strengthened our gripper design, and as future work, we will be testing with valve stems having higher torque resistance. Finally, we are also considering the potential use of another robot platform that is more lightweight and allows us to more easily transport it from Japan to the competition venue, considering airlines regulations.

## V. CHALLENGE 3: SEARCH, PICK AND PLACE

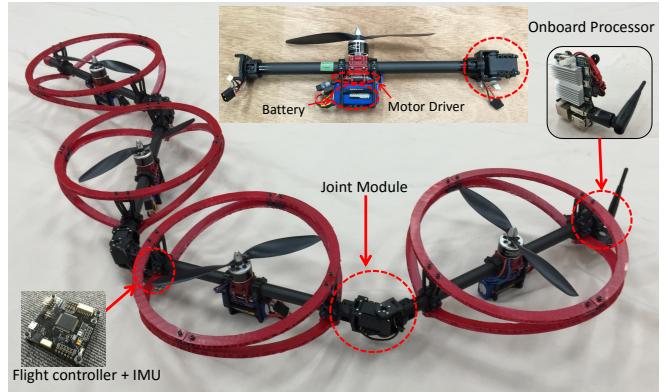
### A. Platforms

For task 3, we used two types of UAVs for the task. The general UAV called "hawk" as shown in Fig.8, which is similar to the one used in task 1, and the transformable multilink aerial robot, which is called "Hydrus" (Fig.9). As described in Fig.10, the hardware platform of "Hydrus" involves the controller for the joints, which enables stable aerial transformation.

Although the flight control algorithms between "Hawk" and "Hydrus" are fundamentally different, we use the same



**Fig. 8:** Image of task 3 Hawk



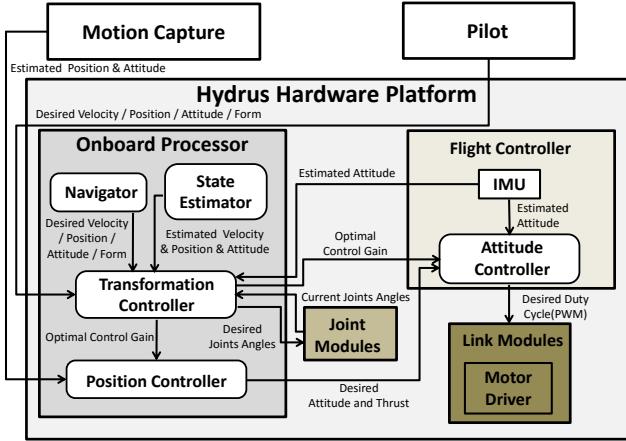
**Fig. 9:** Image of Hydrus

flight controller board which is build by ourselves. For "Hawk", we additionally designed another PCB board for controlling an electromagnet module that can generate attractive forces up to 20[N]. We equipped 5 electromagnets to the UAV and build the attachment with tactile sensors as shown in Fig.8(c). The electromagnet module control board is connected to the flight controller board unit through CAN bus.

For the transformable UAV, we introduce the prototype which contains four links and three servo joints. The modularization of the whole platform is achieved by distributing the power and control system to each link with exception to the flight controller and sensors. Therefore, it becomes easier to the change the speed of the rotors for controlling flight.

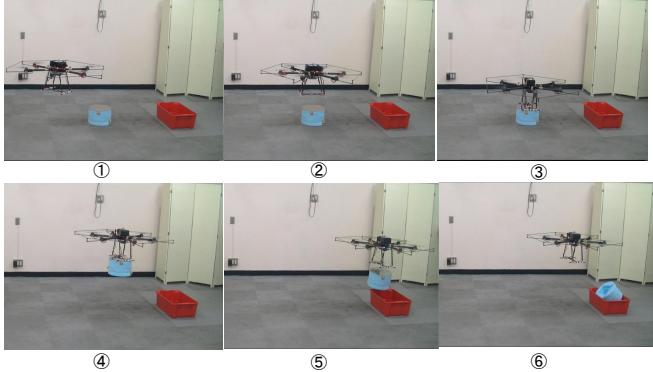
### B. Aerial Manipulation Strategy

For each type of UAV, we develop a different picking method. For "Hawk" type UAV, we apply magnetic force to attract the ferrous object as shown in Fig.11. When contact



**Fig. 10:** Hardware platform of task 3 Hawk

between the bottom of the landing gear and the object is detected by the tactile sensors, the electromagnet module is activated. We successfully picked and carried the object in an indoor environment with the use of a motion capture system, validating the electromagnet based manipulation strategy.



**Fig. 11:** Aerial manipulation method of Hawk

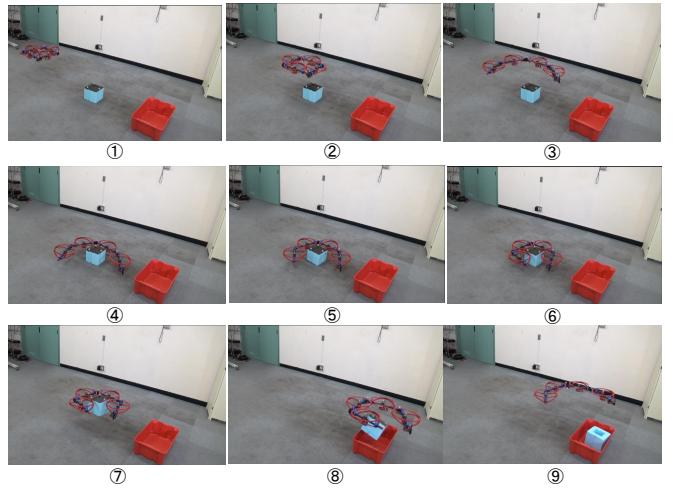
Object transportation based on the whole-body manipulation strategy using "Hydrus" has also been achieved as shown in Fig. 12. Grasp control is achieved based on torque feedback from each joint.

#### C. Software

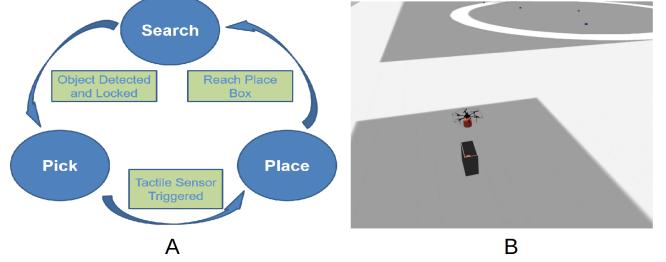
As with other tasks, the software is build using ROS and some functionalities are shared with task 1. Point Cloud and OpenCV libraries are used for visual perception.

#### D. General Approach

1) *Overall Strategy:* We divide the task into three states: Search, Pick, and Place. The UAVs are always in one of these three states and the states automatically transition into the next one when certain conditions are met as illustrated in Fig. 13A. In the "Search" state, the drone will traverse to the center of the arena and randomly generate a search



**Fig. 12:** Aerial manipulation method of Hydrus



**Fig. 13:** Task 3 Demonstration

end-point, the treasure detector will run while the drone is searching. Once the object is detected and locked, a pick motion will be generated. In the "Pick" state, the UAV will turn on the electromagnet and approach the treasure (for *Hawk*) or enclose its body around the treasure (for *Hydrus*). The state transition from the "Pick" state is signalled by the triggering of the tactile sensor. Once the electromagnet of the *Hawk* has caught the treasure or the body of the *Snake* has enclosed the treasure, the UAV transitions into the next state. In the "Place" state, the UAV will fly directly to the placing zone and find the box to place the treasure in. After releasing the treasure into the place box, the UAV re-enters the "Search" state and loops until the task is completed.

2) *Treasure Detection:* As the treasures have distinct color features, we first used a simple detection method to detect the treasures. The inputs are 3D points  $p_i$  from the stereo sensor and the RGB image projection onto the ground by the projection matrix computed using the known camera parameters. HSI color filter is applied to obtain the 3D point candidates of the treasures from the point cloud data. Next, we apply Euclidean clustering to the filtered point cloud  $P_{hs}$ . Euclidean clustering technique can organize points into clusters with respect to distance features in 3D space. For  $\forall p_i, p_j \in P_{hs}$ , clusters  $O_i = \{p_i \in P_i\}$  and  $O_j = \{p_j \in P_j\}$  are obtained by:

$$\min ||p_i - p_j|| \geq d_{threshold} \quad (1)$$

After we obtained all the clusters, we apply a simple tracker to every cluster center, and as we continue to detect the same cluster over time, the weight of the tracker is increased to boost the confidence of tracking. For clusters that are not always detected the confidence are slowly decreased and removed from the treasure candidates vector. The UAV will lock on to the cluster candidate when the weight is large enough and switch into the "Pick" mode to approach the treasure.

#### E. Results Achieved to Date

We first performed full automatic simulation in Gazebo as shown in Fig.13B. To fully simulate the real scene, we add noise and outliers to the detection. In simulation, the UAV takes almost 70seconds to detect, pick and place a single object. With the real robots, we tested tele-operation control, and both the *Hawk* and the *Snake* were able to grasp the treasure and pick and place it into a specified box.

#### F. Future Plans

In our next steps, we will use three UAVs in coordination to complete the task which will not only decrease the time but also can be used to transport larger treasures which a single UAV might not be able to lift. We will also be carrying out more experiments on the real robots to test the detection and motion planning algorithm that have been verified in simulation.

Future work on hardware for task 3 includes the improvement of the structural strength, as well as the enhancement of the modularization of link system by using CAN communication network. We will also continue to validate the performance of the electromagnet module, and the integrated use of electromagnetic force and whole-body-manipulation will be developed for "Hydrus".

Future work on software for task 3 involves outdoor experiment with actual robots to test the performance of both electromagnet module and whole-body-manipulation. In addition, we will focus on the collaborative motion for picking up the large object using two or three UAV simultaneously, as well as swarm control strategy while searching object.

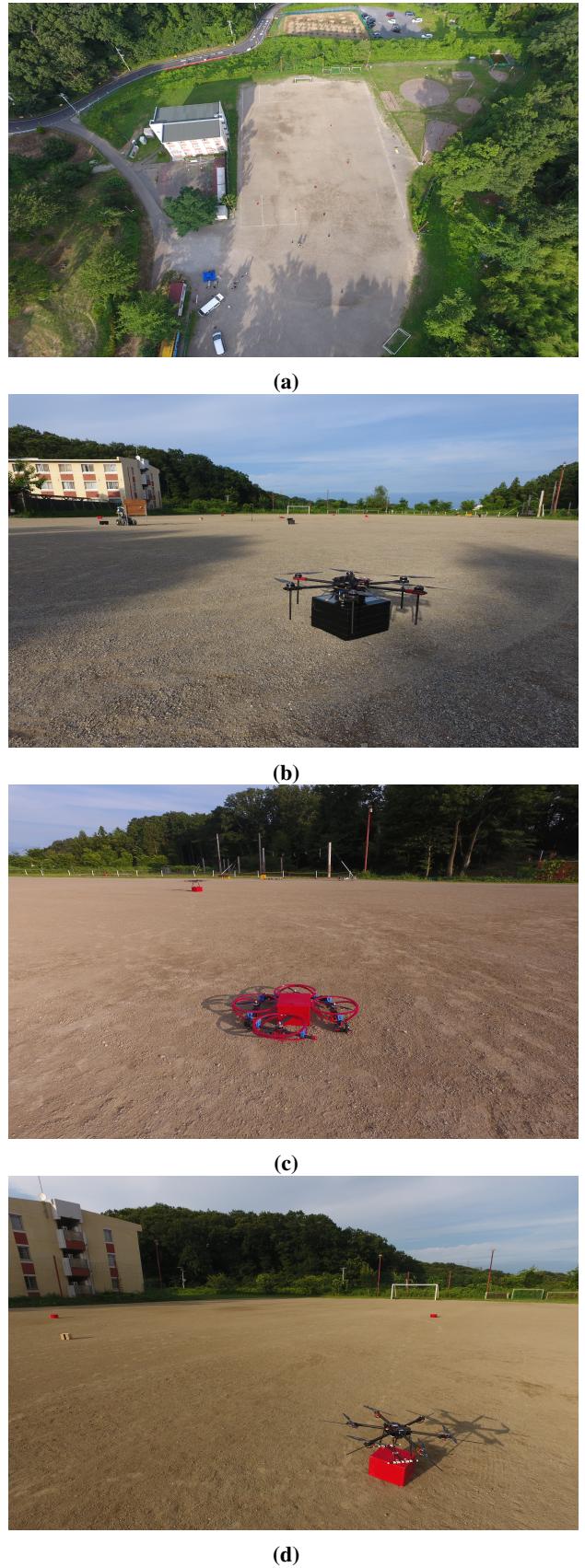
## VI. GRAND CHALLENGE

#### A. Setup of Testbed

We have prepared our testbed where we will perform the outdoor testing in the real world located in Hachioji, Tokyo, Japan, as shown in Fig.14. Thus far, we have developed the robot systems for each task in this testbed individually.

#### B. Future Work

Once we complete each of the three tasks above, for the grand challenge we will combine each of the 3 tasks above, however, we also plan to make some changes such as UAV to UAV and UAV to UGV communications so that all the robots are able to collaborate in completing the tasks.



**Fig. 14:** JSK-Team testbed setup at Hachioji, Tokyo, Japan