RoboCupRescue 2015 - Robot League Team <ARC- AUT (IRAN)>

Elham Iravani, Navid Khazaee Korghond, Atefe Moosavinejad, Fatemeh Pahlevan Aghababa, Fatemeh Tavassoli, Majid Jegarian, Omid Safari, Mehran Bahri, Farzam Janati and Saeed Shiry Ghidary

Amirkabir University of Technology (Tehran Polytechnic)
424 Hafez Ave, Tehran, Iran.

e_iravani@aut.ac.ir
http://AUT-ARC.webs.com

Abstract. This paper describes the approaches of the ARC-AUT real rescue team. Most of the team members have good experiences in the national and international RoboCup competition in Real Rescue, @home and humanoid robot leagues in the past years. We have an autonomous robot and also a manual UAV. The key point is that the UGV goes through red and orange arena and generate 3d map and the autonomous robot goes through the yellow arena and generate 2d and 3d map. At the end two robots merge their maps to a unique map with QR code and victim positions. The framework that used is ROS operating system. A novel scenario are designed for victim detection and verification task with some sensor fusion especially vision.

Introduction

Team ARC-AUT (Amirkabir Robotic Center-Amirkabir University of Technology) has been started working on rescue league one year ago. Half of people are experienced people in RoboCup and robotic manufacturing.

We have two autonomous and UAV robot for going through whole arenas. Autonomous robot is four wheel that are derived by four separate motors. The platform is robust to up and down in step field and slopes and also it can change its height. The robot have Kinect and laser scanner to navigate well and generate 2d and 3d maps. We have new method for planning and also victim detection and verification. The robot prioritize victims that seen and give weight to them then the robot navigate through corridor to arrive to the victims in queue. There is a two DOF arm on robot and it moves vertical and one revolute joint for rotate in horizontal plane to move Kinect and victim detection sensor pack. The UAV are like dig-air quad-copter design but it can carry a Kinect and required sensors. It generate a 3d map.

At the end of the mission, both map that generate by the robots send to the operator station system to combine. Then the map has the whole arena with victim and QR code positions.

For software development, we continued using ROS framework which is being used by teams in ARC- AUT since 2013 [1][2]. ROS concepts help the developer to easily manage processes and keep the modules as simple as possible so debugging the whole process of a robot's behavior would not be challenge. Lots of work is done in the open-source community of ROS in form of ROS packages then we could save lots of time and energy and focused on improving algorithms instead of writing everything from scratch. [3]

1. Team Members and Their Contribution

- **Elham Iravani** Team Leader, SLAM, Navigation and Control, ROS framework, Behavior
- **Navid Khazaee Korghond** ROS framework, SLAM, Navigation and Control, Behavior
- **Atefe Moosavinejad** SLAM, Navigation and Control
- **Fatemeh Pahlevan Aghababa** Behavior, ROS framework
- **Shirin Reyhanian** Behavior(especially UAV)
- **Edvin Babaians** ROS framework, SLAM
- **Fatemeh Tavassoli** Vision
- **Majid Jegarian** Team manager, Mechanic member
- **Omid Safari** Autonomous Mechanic Leader
- **Mehran Bahri** UAV Mechanic Leader
- **Farzam Janati** Electronic member
- **Omid Baei** Mechanic Supervisor
- **Shervin Fallahnejad Rostamkolaei** Motion Control, Electronic Supervisor
- **Kiarash Kohansal Nodehi** Mechanic member
- **Vahid Aberoumand** Electronic member
- **Dr. Saeed Shiry Ghidary** Advisor

2. Operator Station Set-up and Break-Down (10 minutes)

The whole system contains two robots (Autonomous Robot and UAV Robot). There are two laptops, one joystick for UAV manual control, and one access point for Wireless connection to robots.

In the station, operator can monitor views and information which are received from both robots. Then, he responds them to reject or accept the situation over wireless and laptops. We need 3 persons to operate autonomous robot, UAV, and station pack. The whole system for operator station will bring in a package and two robots carried to the beginning point in arena. They are responsible to turn on and set the robots and station pack. Then they should test the wireless connection throughout the whole system.

3. Communications

We use a wireless access point in the station and a USB wireless which is connected to the mini pc on autonomous robot to send/receive data to/from operator station over 5.0 GHz network. Also, we have a 2.4 GHz wireless connection for UAV.

Rescue Robot League		
ARC-AUT (IRAN)		
Frequency	**Channel/Band**	**Power (mW)**
5.0 GHz - 802.11a	8	35
2.4 GHz - 802.11b/g	8	35

4. Control Method and Human-Robot Interface

Since the robot is fully autonomous, no person included in the communication and no command will send from the operator station. The robot explores the environment prior to victim location and it develops the 2D and 3D map. First, it finds the victims. Then, it sends information including temperatures, CO_2 levels, voices, victims' motions, and images containing eye charts and hazard signs to the operator station.

A GUI designed by rqt [4] in ROS that all monitoring data can appear on it. The raw data from sensor sends to the mini pc. Then, it processes them and modifies them to more reliable information. Afterwards, it alerts the operator of the suspicious conditions.

The processor in autonomous robot is Gigabyte Brix GB-BXi7-4500 which is used for all on board processes.

A flight control module based on PX4 autopilot is used to control the Quad-copter. PX4 autopilot is a Low cost open-source autopilot system oriented toward inexpensive autonomous aircraft.[1] The aerial system consists of one UAV that can be controlled manually or autonomously. A remote controller in 2.4GHz Spread Spectrum Technology is used by operator in manual mode to control the robot. In manual mode the operator uses an analog camera mounted in front of the robot, an analog transmitter has is to send the output of the analog camera to the station. The UAV is controlled by an operator. Also, there are victim detection sensors on UAV robot. The operator can monitor the environment that UAV has seen and decide to navigate over the corridors. When the UAV finds a victim, it goes near the victim to detect the signs.

Both UAV and autonomous robots generate 2d and 3d map using Kinect and laser scanner. At the end of the mission the maps will merge and generate a unique map containing QR code and victim position.

The behavior state machine is implemented using ros-smach. Last year one of our laboratory teams (aut-humanoid) started working with this structure and that encourages us to create our behavior structure based on that. [5]

5. Map generation/printing

For SLAM and Navigation, we have used ROS packages from Hector team, hector_slam [6] and hector_navigation [7] with a 2D laser scanner. The SLAM begins at the start and our exploration algorithm plans where to go next, based on weather a victim is detected or more exploration is needed.

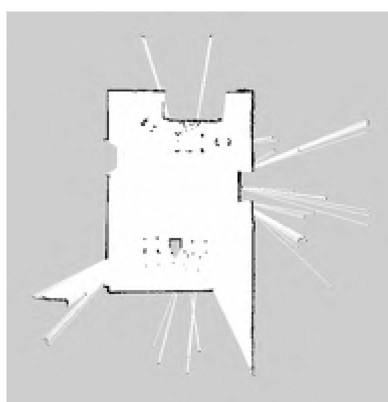

Fig. 1. Output map from laboratory (using hector_slam package [6])

[1] http://en.wikipedia.org/wiki/PX4_autopilot

We use hector navigation to navigate inside the map area whenever is needed. In addition, we developed our own method for planning the robot motion where more exploration is needed to expand the map for unknown areas. The developed method is based on morphological thinning kernels which thins the extracted corridor and converts it to a skeleton so the planner is able to choose the next point along the middle line.

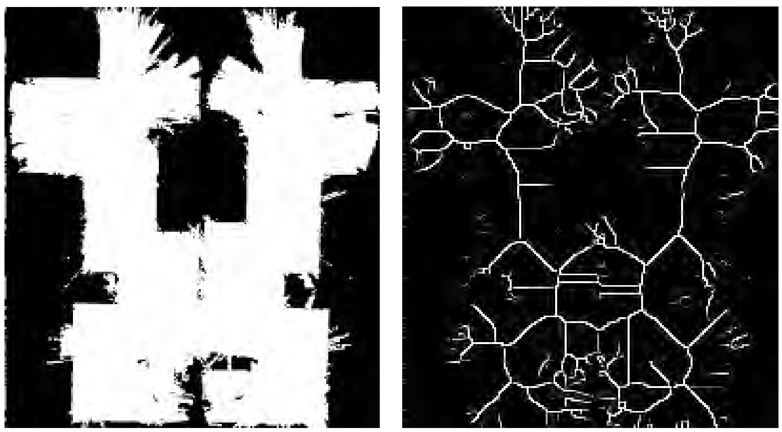

Fig. 2. Output of thinning algorithm for planning

The UAV robot flies over the walls during each mission and generates a 3D map using the mounted Kinect sensor and a modified version of RGBDSLAMv2[8] which makes it possible to try autonomous flight in further steps. This map is represented as a point cloud and victims are marked where the operator has pushed the related button on the controller. We are generating another map in 2D using the bottom camera. When the operator marks the victim, he will take the robot to a higher point above the victim and an image is taken from that view. These images will be stitched into another aerial map which is generated at the beginning of the mission with the same methods [9], by flying all over the mission's area at the beginning. Victim's position in the taken image is preserved by tracking SIFT features of the area in which the operator has marked the victim, hence it will be available in final 2D map. Further processing can be done if the map has to be visually the same as the generated map by the UGV, as the 3D map is available.

6. Sensors for Navigation and Localization

✓ **Laser scanner:**

Hokuyo URG-04LX- F01 Scanning Laser Rangefinder is used in the robot. It is mostly used for 2D mapping but can also otter 3D data.

The Hokuyo UBG-04LX-F01 Scanning Laser Rangefinder is a small, affordable and accurate laser scanner that is perfect for robotic applications. The UBG-04LX-F01 is able to report ranges from 20mm to 5.6m (1mm resolution) in a 240° arc (0.36° angular resolution) but it does so much faster than the Hokuyo URG-04LX Scanning Laser Rangefinder. Its power consumption, 12V 375ma, allows it to be used on battery operated platforms. It has the weight of 260 g, range of 20mm to 5600mm × 240°, accuracy of ±10mm and distance and angle data output with high angular resolution (0.36°). [10] The laser is used by robot for SLAM and Navigation task. The resolution, accuracy and range of view are the important properties that cause using laser.

Fig. 3. Hokuyo UBG-04LX-F01 Laser range finder

- ✓ **Analog camera:**

This camera is mounted in front of the robot and is used to control the robot by the operator, because the video transmitted on Wi-Fi is delayed and not suitable for manual flight.

- ✓ **RGB-D Camera:**

A Kinect sensor is mounted on the UAV to generate 3D maps for each mission. It is also used for our experiments on autonomous flight and victim detection. In manual flight, we have used the depth data from Kinect to help the operator in control of the UAV when it should lower the robot's distance from where the victim is, avoiding collision with the walls. We have also used it as a 2D laser scanner in our experiments, using the ROS package depthimage_to_laserscan [11]. We found it promising in obstacle avoidance in tight indoor areas like the mission's areas so it can be used if the UAV needs to move between the walls.

Fig. 4. Kinect

✓ **Digital Camera:**

A high resolution digital camera is mounted at the bottom of the robot and is used to generate aerial maps and calculating optical flow for navigation purposes. It is also used for tracking visual features of where the victim is marked as described in section 5.

Fig. 5. C920 Logitech

✓ **Inertial Measurement Unit:**

We used a GY-86 IMU on the UAV to estimate robot's pose and control its state. We also fused IMU data with optical flow for navigation purposes in our experiments for autonomous flight. Raw data of IMU is used in the control board of UAV to stabilize hover and movements.

The autonomous robot also used this module to stabilize laser range finder and Kinect.

Fig. 6. IMU gy-80

✓ **Ultrasound Range Finder:**

This sensor is placed under the robot and is used for altitude control. Besides, there are some of them around the autonomous robot for wall detection.

Fig. 7. SRF08 Ultrasonic

7. Sensors for Victim Identification

➢ **Victim Detection and Verification Scenario**

We have designed a multi-sensor fusion platform which detects victims and plans what to do against each one in two phases: Detection and Verification. In the detection phase, robot looks for potential victim candidates and potential places which should be scrutinized more, like a hole in a wall. Possible candidates are detected using three different kind of sensors: The Kinect, two RGB cameras, and heat sensors. Each sensor has a few possible scenarios which may be interpreted as a possible victim:

The Kinect: 3D segmentation of environment is done by detecting planar areas in the depth image, then we can easily look for holes in walls. The RGB image is also used to detect Hazard signs.

RGB cameras: Since the Kinect does not turn around while we are using it for navigation purposes, two RGB cameras are mounted in two sides of the robot to fully cover the environment. We detect the walls by color filters and then run a simple blob detection algorithm on the possible wall areas, which are possible holes in the walls.

Heat sensors: If the robot passes a victim lying on the ground, or left in a box with one or two sides missing, the robot will pass it due to the obstacle avoidance algorithms running in the navigation module, and heat sensors mounted on the will see the heat with a reliable probability. Also, heat in the hole on a wall can be detected by a sensor mounted on the Kinect when the robot is in the proper position against the wall.

These possibilities are calculated in different ROS Services which report their findings to a Verifier service. Since the message includes the type of detection and related information, like "Heat, TEMP" which says detection is done by heat sensor with the value of TEMP, the verifier can plan the proper verification scenario to become sure that there is a real victim, verified by the other sensors. For example, blobs detected on the walls need to be verified with depth image from the Kinect, which obviously implies that the verifier needs to request certain moves from navigation module, e.g. turn left, to verify the detected possible victim. After all, the verifier plans if there is a real victim or more scrutiny is needed, and sends the related data to a planner service. The planner service has a priority list which is filled by candidates' data, like "Hole, X, Y, Z" which has been sent by the verifier. This service plans if the robot has certain victims to mark in the map, or it should navigate to another position to find a real victim, e.g. toward a hole in a wall. During all these phases, new candidates can be added to the list which also has fixed constraints like size, which avoids the robot from continuously switch between possible candidates.

- ✓ **Heat Sensor:**

Both autonomous and UAV robots use heat sensor to measure victim's temperature. There are three of them on autonomous and one of them on the UAV to inform operator form victim's status. TPA81 Infra-Red Thermal Sensor is used. If the robots verify the victim for suitable range of temperature, they will check other victim's signs to ensure about victim existence around.

Fig. 8. TPA81 Infra-Red Thermal Sensor

- ✓ **CO2 Sensor:**

The robots use CO2 sensor for completing information about the victims. After a robot found a victim and measured its temperature, it measures the CO2 level as another vital sign.

The K-30(SE-0018 K30 STA) Sensor Module is a maintenance-free transmitter module intended to be built into different host devices that require CO2 monitoring data. It is an accurate, yet low-cost solution for OEMs who want to integrate CO2 sensing into their products. The compact size, low-power requirements and multiple output options are intended to be easy to implement into analog or microprocessor-based controls and equipment.

Fig. 9. SE-0018 K30 STA CO2 sensor

✓ **QR Code Detection:**

One of the important tasks in the RoboCup rescue competition is the QR code Detection. Robots get scores by detecting QR codes which are placed in different spots of the Rescue environment. This task is done by two eye cameras in the left and right sides of the autonomous robot and a Kinect sensor which is placed in front of the robot. We used the QR code packages which are written on ROS framework using Zbar [12] and OpenCV libraries. The position of the QR codes which are detected autonomously by robots will be indicated on the map.

Fig. 10. QR code detection output

✓ **Voice:**

The voice should send/receive from robots to the operator station and vice versa. Some victims make a sound to draw the attention of rescuers to get help. This voice is transmitted by a tie microphone to the operator station.

✓ **Motion Detection:**

Some victims may have some motions and robots should be able to detect those motions. The robots can do so by frame subtraction methods with its visual sensors.

✓ **Blub Detection:**

In victim detection part, we use a Kinect sensor for frontal view and two webcams for left and right side views. With Kinect depth data, which is acquired by an infrared laser projector and a monochrome CMOS sensor, we can find and mark areas with different depth like holes in the walls that may contain victims for further explorations. Besides, with RGB data which is acquired by a Kinect RGB camera and two webcams, we can find the circular areas. Blob detector algorithm searches over video frames on different scale-space, from fine to coarse scale by using iterative Gaussian blurring, and using Laplacian filter to find the local maxima and consequently blob shape areas. These areas also mark as the interest areas for further exploration to find victims. [13]

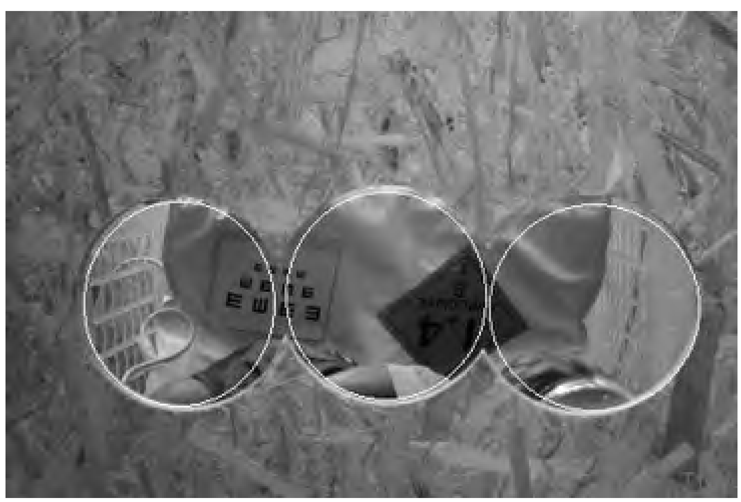

Fig. 11. Blob detector algorithm results on victim location in arena from a captured picture

8. Robot Locomotion

Autonomous Robot:

The mechanical structure of this robot is composed of four wheels and a box in the middle. The battery, the motor control circuit and the ECU are places in the box. The robot is equipped with an inactive suspension system change the angle of the wheels while passing through the obstacles so that the robots can have maximum stability. It also destroys the angle of box with skyline. The suspension system of the robot works with no spring or damper. Each wheel is also designed so that it can be set at angles of 90, 60, 30, 0 degree towards the middle box. We use the offroad RC cars wheels for this robot. The robot motion is provided by four electric motor-gearboxes. The mechanical unit weight is about 5 Kg.

Fig. 12. Autonomous robot design in SolidWork

Fig. 13. Autonomous robot

UAV Robot:

We have a flying robot in the form of quad. The task of the flying robot is mapping the rescue environment and detecting victims. So, it is equipped with a Kinect sensor and a 1 DOF camera in front of it. The Kinect is placed at the top of the board plane and separated by dampers to prevent vibrations of the flight. The body of the robot is made of carbon fiber with square profiles. The Kinect-Robot design is inspired by the robot called “Dog air”.

Fig. 14. Dog air robot [14]

All the flying robots, which are equipped with Kinect, suffer from occlusion of the viewing angle of the propellers of the robot. To solve this problem, there are two approaches: one is to place the Kinect on the top of the robot, and the other one is to place it at the bottom of robot. Both of these methods have some advantages and disadvantages simultaneously.

In the first case, if we put Kinect on the top of the robot, it will have more height and because of its heavy weight this kind of design leads to imbalance robot maneuvers. The Pelican is an example of this kind of design.

In the other case, if we put Kinect at the bottom of the robot, crashing accidents will cause more damages to the Kinect. Also, it will be close to the battery and the heat from the battery will make some problems for it.

Fig. 15. Pelican quad-copter[15]

Fig. 16. Kinect-rigged quad-copter[16]

The ARC Quadcopter solved these problems by presenting a new design for our robot. The weight of the robot is about 2.2kg and its length, width, and height are 30, 30 and 14 inches respectively. We placed the two forward motors below their initial positions. This design gives us the opportunity to achieve a greater viewing angle. The height of rotors and frames provide sufficient distance from the battery and main board of the robot. Our design is illustrated in the following picture:

Fig. 17. UAV robot design in CATIA

In addition, by decreasing the Kinect height about 2 inches with respect to 10 inches propellers, it creates a safe zone for it in the case of crashes. Also, by putting the Kinect in the front side of the robot and putting the batteries in the back, we could keep the Center of Gravity at the Centre of the robot.

9. Other Mechanisms

The robot has an arm with two degrees of freedom including a rotational and axial movement. A Kinect is mounted on top of this arm and the arm can move it up to 60 cm. This arm works with two servo motors. The mechanism of this action is similar to the car antenna.

10. Team Training for Operation (Human Factors)

We have two robots for rescue, autonomous unmanned ground vehicle (A-UGV) and unmanned air vehicle (UAV). A-UGV is fully autonomous robot that can move in the arena and gather information from environment such as heat, voice, image, and etc. Then, it sends them for an operator station. Also, UAV robot can fly in the arena and make 2D or 3D online map. Then, it sends them for the operator station.

Finally, the system in the station will merge the maps of two robots. So, the human rescuers can see the location of victims as well as find the best path to them.

11. Possibility for Practical Application to Real Disaster Site

We use UAV robot which can fly in the real disaster site to make a practical map of the environment and find victims very quickly. Then, it sends this information to human rescuers to help them in the upcoming rescue operations.

Besides, the robust mobility platform of A-UGV robot enables it to move in harsh environments in order to locate victims and check their vital signs.

12. System Cost

Autonomous Robot:

No	Object	Unit Price	Total Price
1	Motion Motor	40$	200$
2	Servo Motor(mx 106)	493$	1972$
3	Servo Motor(mx 28)	225$	450$
4	Laser Cutting	80$	80$
5	Turning	30$	30$
6	Bolts	10$	10$
7	Sensors	2550$	2550$
8	Camera	90$	180$
9	Driver	40$	80$

10	Main board	200$	200$
11	Mini PC	530$	530$
	Total Price		8817$

UAV Robot:

No	Object	#	Unit price	Total price
1	IMU	1	16$	16$
2	flight control	1	100$	100$
3	video transmitter	1	22$	22$
4	main board	1	200$	200$
5	Sonar	1	40$	40$
6	Camera	2	100$	200$
7	Motor	4	55$	220$
8	speed control	4	40$	160$
9	Battery	1	100$	100$
10	Kinect	1	150$	150$
11	Prop	4	30$	120$
12	Radio	1	445$	445$
	Total Price			1593$

13. Lessons Learned

Our goal of participating in this competition is far more than just winning awards. First of all, this is our first experience in RoboCup competitions and we have new robots to test them in a real challenge. So, this contest could be a very good benchmark for identifying our strengths and weaknesses. We can use this precious experience to improve our team and robots for the future competitions.

References

[1]http://fei.edu.br/rcs/2014/TeamDescriptionPapers/RoboCup@Home/AUT@HomeTDP_RoboCup2014.pdf

[2]http://fei.edu.br/rcs/2014/TeamDescriptionPapers/Humanoid/TeenSize/RoboCup2014-AUTUofM%20Humanoid%20TeenSize%20Team.pdf.

[3] http://wiki.ros.org

[4] wiki.ros.org/rqt

[5] Bohren, J., and Steve Cousins. "The SMACH high-level executive [ROS News]." Robotics & Automation Magazine, IEEE 17.4 (2010): 18-20.

[6] http://wiki.ros.org/hector_slam

[7] http://wiki.ros.org/hector_navigation

[8] F. Endres, J. Hess, J. Sturm, D. Cremers, W. urgard , "3D Mapping with an RGB-D Camera", , IEEE Transactions on Robotics, 2014.

[9] http://wiki.ros.org/mapstitch

[10] http://www.robotshop.com/en/hokuyo-urg-04lx-f01-scanning-laser-rangefinder.html

[11] http://wiki.ros.org/depthimage_to_laserscan

[12] http://zbar.sourceforge.net/

[13] T., Lindenberg, "Detecting Salient Blob-Like I mage Structures and Their Scales with a Scale-Space Primal Sketch: A Method for Focus-of-Attention," International Journal of Computer Vision, 1993.

[14] https://www.airdog.com

[15] http://www.rcheliresource.com/tag/quad-copter/

[16] http://mikeshouts.com/kinect-rigged-quadcopter-flies-autonomously-and-avoids-obstacle/