Team Cappadocia Design for MAGIC 2010

Alper ERDENER[1], Evrim Onur ARI[1], Yoldaş ATASEVEN[1], Berkhan DENİZ[1], Kutalmış Gökalp İNCE[1], Uğur KAZANCIOĞLU[1], Teksin AÇIKGÖZ KOPANOĞLU[1], Tankut KORAY[1], Kerem Mustafa KOŞANER[1], Ayhan ÖZGÜR[1], Çiğdem ÇAKICI ÖZKÖK[1], Tuna SONCUL[1], Halil Onur ŞİRİN[1], İskender YAKIN[1], Scott BIDDLESTONE[2], Lina FU[2], Arda KURT[2], Ümit ÖZGÜNER[2], Keith REDMILL[2], Örsan AYTEKİN[3], İlkay ULUSOY[3]

(1) ASELSAN Inc., P.K. 1, Yenimahalle, 06172, Ankara, Turkey, Phone: +90 (312) 592 10 00, E-Mail:menguc@aselsan.com.tr ; (2) The Ohio State University, Control & Intelligent Transportation Research Lab, 405 Dreese Labs 2015 Neil Ave., Columbus, OH, 43210, USA, Phone: +1-614-292-2572, E-Mail:umit@ece.osu.edu; (3) The Middle East Technical University (Vision Labs),06531, Ankara, Turkey, Phone: +90 (312) 210 23 01, E-Mail:ilkay@metu.edu.tr.

ABSTRACT

Team Cappadocia is participating in the Multi Autonomous Ground-robotic International Challenge (MAGIC 2010), with a set of fully autonomous ground vehicles that can execute an Intelligence, Surveillance and Reconnaissance (ISR) mission in a dynamic urban environment. The design incorporates dynamic mission planning with automatic task assignment and optimized route generation, automated object of interest detection and tracking, decision making, automatic Unmanned Air Vehicle (UAV) image processing, automated local and global map and information integration, and a novel system architecture with modules compliant with the Joint Architecture for Unmanned Systems (JAUS). The cooperators in the team formed under the leadership of ASELSAN Inc. of Turkey are The Ohio State University of USA (Control & Intelligent Transportation Research Lab), The Middle East Technical University (Robotics & Vision Labs), the Bilkent University (Robotics Lab) and the Bogazici University (AI Lab) of Turkey.

1. Introduction

The Multi Autonomous Ground-robotic International Challenge (MAGIC 2010), co-sponsored by the Australian and US Departments of Defense, invited competitors from worldwide research organizations to develop next-generation fully autonomous ground vehicle systems that can be deployed effectively in military operations and civilian emergency situations.

Report Documentation Page

Form Approved
OMB No. 0704-0188

1. REPORT DATE	2. REPORT TYPE	3. DATES COVERED
25 JAN 2011	**FInal**	**01-01-2010 to 01-01-2011**

4. TITLE AND SUBTITLE
Final Report for Grant AOARD-10-4026 MAGIC 2010 The ASELSAN Team

5a. CONTRACT NUMBER
FA23861014026

5b. GRANT NUMBER

5c. PROGRAM ELEMENT NUMBER

6. AUTHOR(S)
Faruk Menguc

5d. PROJECT NUMBER

5e. TASK NUMBER

5f. WORK UNIT NUMBER

7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES)
ASELSAN A.S. Mehmet Akif Ersoy Mah. 296.Cad. No:16,06172 Macunk?y Yenimahalle ANKARA T?rkiye,06172 Macunk?y Yenimahalle ANKARA T?rkiye,06172 Macunk?y Yenimahalle ANKARA T?rkiye,TR,06172

8. PERFORMING ORGANIZATION REPORT NUMBER
N/A

9. SPONSORING/MONITORING AGENCY NAME(S) AND ADDRESS(ES)
AOARD, UNIT 45002, APO, AP, 96337-5002

10. SPONSOR/MONITOR'S ACRONYM(S)
AOARD

11. SPONSOR/MONITOR'S REPORT NUMBER(S)

12. DISTRIBUTION/AVAILABILITY STATEMENT
Approved for public release; distribution unlimited

13. SUPPLEMENTARY NOTES

14. ABSTRACT
Team Cappadocia is participating in the Multi Autonomous Ground-robotic International Challenge (MAGIC 2010), with a set of fully autonomous ground vehicles that can execute an Intelligence, Surveillance and Reconnaissance (ISR) mission in a dynamic urban environment. The design incorporates dynamic mission planning with automatic task assignment and optimized route generation, automated object of interest detection and tracking, decision making, automatic Unmanned Air Vehicle (UAV) image processing, automated local and global map and information integration, and a novel system architecture with modules compliant with the Joint Architecture for Unmanned Systems (JAUS). The cooperators in the team formed under the leadership of ASELSAN Inc. of Turkey are The Ohio State University of USA (Control & Intelligent Transportation Research Lab), The Middle East Technical University (Robotics & Vision Labs), the Bilkent University (Robotics Lab) and the Bogazici University (AI Lab) of Turkey. During the period of the research, Team Cappadocia has put together and validated an UGV system in order to compete in the MAGIC 2010 challenge. A full technical paper, whose abstract has been reproduced here, details the implementation and many of the significant experimental results of the system; the paper has been published for the Land Warfare Conference

15. SUBJECT TERMS
Cooperative Control, Unmanned Vehicles, Autonomous Agents and Multi-Agent Systems

<table>
<tr><td colspan="3">16. SECURITY CLASSIFICATION OF:</td><td rowspan="2">17. LIMITATION OF ABSTRACT
Same as Report (SAR)</td><td rowspan="2">18. NUMBER OF PAGES
28</td><td rowspan="2">19a. NAME OF RESPONSIBLE PERSON</td></tr>
<tr><td>a. REPORT
unclassified</td><td>b. ABSTRACT
unclassified</td><td>c. THIS PAGE
unclassified</td></tr>
</table>

The Team Cappadocia[1] has been formed under the leadership of ASELSAN of Turkey to enter the competition. Other cooperators are the Ohio State University of USA (Control & Intelligent Transportation Research Lab), the Middle East Technical University (Robotics & Vision Labs), the Bilkent University (Robotics Lab) and the Bogazici University (AI Lab) of Turkey.

1.1. Statement of the Problem

Intelligence, surveillance and reconnaissance (ISR) of an area plays a vital role in the operational theatre before high value protection forces take action. It is the primary role of the ISR mission team to construct enough information before the main acting teams start conducting their mission. However, ISR missions are becoming ever challenging with the increase in threats mainly due asymmetric warfare. Such missions, especially within urban environments are causing high vulnerabilities to the protection forces as the environment is unknown, complex, dynamic and threat motion is unpredictable.

Autonomous robotic systems that can provide the protection forces with necessary knowledge without getting them into the harm's way are in consideration today to save the lives of many, as decreasing vulnerabilities during ISR missions is essential. Team Cappadocia took the opportunity to provide a novel, reliable and robust solution to the ISR missions in dynamic urban environments with the use of autonomous multi-robot teams using high level of autonomy with the expertise of integrating field proven unmanned vehicle systems and components.

1.2. Conceptual Solution Proposed

The solution proposed by Team Cappadocia relies on a central system which plans, schedules and initiates the operations of an autonomous Unmanned Ground Vehicle (UGV) system based on the pre-knowledge of the area, the continuous UAV data feed and the information acquired through the robots. Each robot is given a path by the central system and autonomously explores its environment following this path while sending received information (obstacles, objects of interest, etc.) back to the central system. The central system integrates the ensuing local maps and the information received from the UGVs and upon evaluation, sends new commands as to track and/or neutralize objects of interest or explore further. The robots then autonomously follow these orders by communicating and cooperating with each other.

1.3. Overall Systems Architecture

The system is divided into modules on a functional basis.

All UGVs are equipped with dedicated controllers for the Vehicle and Payload that handle real-time servo actions that do not require decision making. These modules gather sensor information and act accordingly to their predefined actions, such as tracking a path or an object of interest (OOI). The Low Level Controller (LLC) reads sensor information and navigates the vehicle. The Automatic Target Tracking (ATT) module detects and tracks OOIs. Vehicle System Management Module (VSMM) provides JAUS (Joint Architecture for Unmanned Systems) compliant interface for the vehicle and payload specific controllers.

The robots localize themselves both indoors and outdoors, using internal state sensing and scan matching algorithms through a decision making process which processes and identifies the best pose of the robots.

Sensor Fusion Modules (SFM) on each UGV builds a local map from the best pose information and laser scanner readings. The generated local map is shared with the multi robot data fusion (MRDF) module at the Ground Control Station (GCS). Also OOIs detected by ATT are localized by SFM.

The High Level Planning and Control Module (HLC) controls the vehicle in accordance with the high level commands received from the Dynamic Mission Planner (DMP) and the local map provided by SFM. It navigates the UGV with built in obstacle avoidance by commanding the LLC.

MRDF merges the local maps and OOI information received from the SFMs of each UGV and generates an operational map. Generated operational map is stored in the World Model Knowledge Store (WMKS) module.

Dynamic Mission Planner (DMP) module autonomously plans the missions for each robot based on the operational map stored at the WMKS and the status information gathered from each HLC on robots. According to these plans, tasks are dispatched to robots. DMP is “dynamic” because it continuously watches for operational and status information and updates mission plans accordingly.

Finally, the Operator Control Unit (OCU) is the main user interface application that the operators interact with. Operators can monitor each UGV’s actions and statuses in OCU’s status monitoring screen. Also in another screen, OCU displays tactical information using 3D Geographical Information System (GIS) for enhanced situational awareness. OCU is a JAUS compatible system.

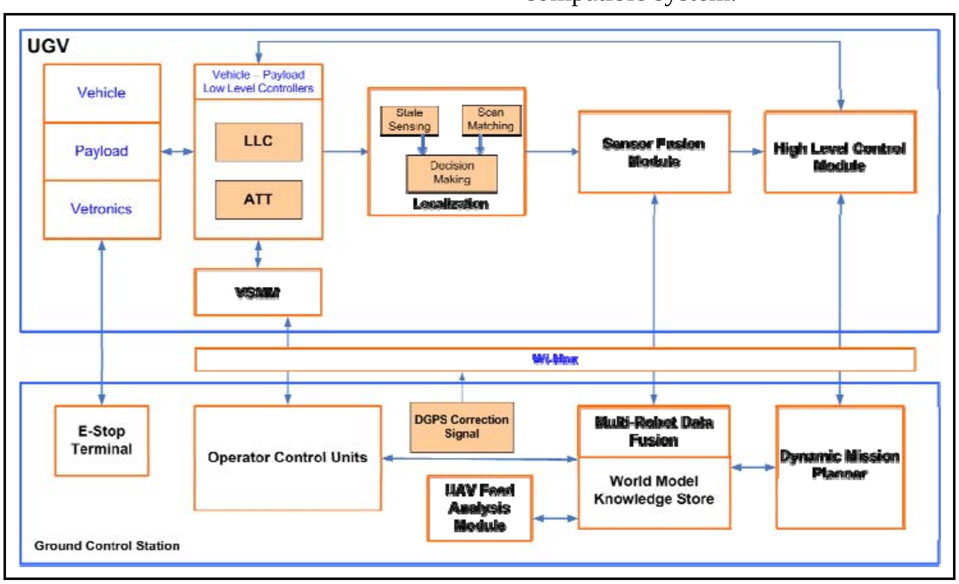


Figure 1: System Architecture

The architecture shown in Figure 1 is platform independent and portable since the interfaces are well defined and based on JAUS. The high level commands are translated through this module. Our system is also scalable and extendible. For instance, by implementing only LLC, VSMM and OCU modules, a basic vehicle with tele-operation capabilities can be built. Target tracking capabilities can be added to such a system by adding an ATT module without changing the rest of the system. Also with this architecture any number of robots and operators can be supported.

2. Ground Vehicle Component & Systems

A team of UGVs composed of Observer and Disrupter robots sharing a common mobile base platform are used. A view of the main UGV components and the subsystems are depicted in Figure 2.

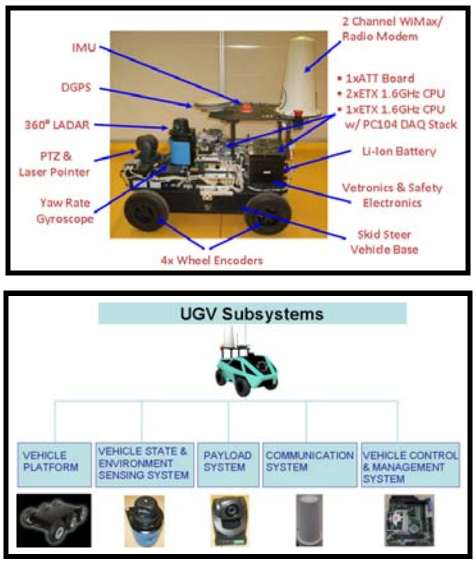


Figure 2: UGV components and subsystems.

Details of the UGV subsystems are explained in the following sections.

2.1. Vehicle Platform

In order to meet the MAGIC 2010 UGV platform and component requirements, a set of COTS vehicle bases have been acquired and outfitted with subsystems readily available and in use within the ASELSAN.

2.1.1. Platform

Vehicle mobility is achieved via a four wheel driven skid steer robot base. Through out the development, modifications were made to chassis, batteries and motor drivers for speed, power and weight considerations.

The UGVs are equipped with four 50W DC motors with 500cpr encoders and are driven by two dual axis digital motor drivers. The motor controllers have embedded over current and over temperature protection and work in the closed loop speed mode.

The vehicle bases are outfitted with custom made electro-mechanical brakes, which are inactive during operation and automatically activated in a failsafe fashion when the power is off.

UGV components are carefully placed to improve the rigidity of sensors during motion, heat dissipation and electromagnetic interference of units. For example, IMU (Inertial Measurement Unit) is placed away from metal surfaces. Also cameras, LADAR (Laser Detection and Ranging), DGPS (Differential GPS) antenna and IMU are placed on the same axis to decrease the effects of parallax/eccentricity.

For aiding the tele-operation mode in an emergency mode a drive camera is also integrated on the UGV and is displayed to the operator through the GCS.

2.1.2. Vetronics

The UGVs are equipped with two independent power distribution subsystems, vehicle and system, respectively. The power sources are isolated from each other to prevent the effects of undesired motor based noise. Circuit breakers and electrical fuses are also placed to protect the power sources from unexpected failures.

The vehicle base has a Ni-MH battery pack used only by the drive system. The main power system is composed of Li-Ion batteries and a DC-DC power converter board. Up to eight Li-Ion batteries can be loaded to the battery slots as shown in Figure 3. The battery compartment design enables fast and hot-swap battery exchange without system shutdown and provides flexibility for the operator in selecting the amount of batteries to load on the system depending on the desired operation time and weight requirements. The complete battery pack can operate the robot electronics for approximately 4 hours.

Figure 3: UGV's Hot-Swap Battery Pack.

The DC-DC Power Converter Board generates necessary regulated and isolated DC voltage levels for the UGV components from the Li-Ion battery pack. The Power Board can generate 5V, 12V and 24V supplies and meets the MIL-STD-1275A-D standard requirements. The Power Board is integrated with the E-stop electronics in order to power down the system during an E-stop request and has a serial communication with the robot controllers so that each individual component's power can de turned off and their status can be monitored.

2.1.3. Low Level Controller (LLC)

A dedicated real-time vehicle control module (Low Level Controller-LLC), based on a COTS CPU equipped with a PC-104 data acquisition stack is used for performing vehicle and payload servo control, sensor data acquisition, system power management, status and health monitoring. The LLC also has an embedded low level reflexive driver that stops the robot before colliding to an obstacle in case other modules fail to avoid obstacles.

<u>Sensor Data Acquisition/Preprocessing</u>: Sensor data acquisition and preprocessing to handle noise, mismatches and scaling are handled.

<u>State Sensing:</u> Runs localization algorithms.

<u>Navigator</u>: Executes waypoint navigation and robotic behaviors (turn a given degrees, proceed a given distance) depending on the HLC commands.

<u>Reflexive Driver</u>: Executes local obstacle avoidance within predefined boundaries.

<u>Primitive Driver:</u> The motion control sub-module converts the steering, speed, and stop/go commands into the low level motor controller commands.

<u>Health Monitor:</u> Continuously checks the UGV components health and provides status to the operator through GCS.

<u>Payload Control:</u> PTZ camera and laser pointer on/off control is handled.

2.2. Vehicle State & Environment Sensing System

A sensing system is used for both internal and external sensing of the vehicle which enables safe and robust localization, navigation and mapping.

2.2.1. Internal State Sensors

<u>Wheel Encoders:</u> 500cpr encoders are integrated on all wheels of the vehicle. They are used in extracting dead-reckoning information (speed, displacement) as well as provide internal feedback for road conditions (move/no move, slip) and zero velocity updates.

<u>IMU:</u> An Inertial Measurement Unit (IMU) is used to measure dynamic attitude (roll, pitch and azimuth). The dynamic roll and pitch is continuously used and the yaw information is only used as a backup emergency feedback.

<u>Yaw Rate Gyroscope:</u> A fiber-optic yaw rate gyroscope is installed to provide high precision yaw rate information for improved heading estimation.

2.2.2. Localization & Mapping Sensors

RTK-DGPS System: A RTK-DGPS (Real-Time Kinematic) system is used in the GCS for RTK-DGPS correction signal broadcasting. The robots are equipped with GPS units capable of receiving these local correction signals.

LADAR: A LADAR with 360 degree field of view and 250 meters range is installed.

2.2.3. Sensor Fusion Module

Sensor fusion module is described in detail in Section 4.2.1.

2.3. Payload System

PTZ Camera: A color Pan-Tilt-Zoom camera is used to detect and track the OOIs and also serves as the main sensor for the operators video feed.

Laser Pointer: A class 1 laser pointer is installed on the disrupter robots.

2.3.1. Automatic Target Tracking (ATT) System

The automatic target detection and tracking from video images is performed on the Video Tracking Unit (VTU) of the ASELSAN. VTU is a dedicated video target tracking board that satisfies the military standards and is in use with several fielded military systems.

The VTU has 2 analog video inputs (PAL/NTSC) and 1 analog (PAL/NTSC) video output where the relevant symbology is added on the video. The board has Ethernet and serial port communications used for sending user commands to the VTU and receiving tracking output (target position) from the VTU. The commands can be sent for setting the operation mode (Detection/Tracking), selecting the OOI and drawing certain symbols and shapes (target bounding boxes and/or centers, etc.) for the user information.

2.4. Communication System

The main communication system is based on the new Wimax technology. Each UGV and the GCS is equipped with Wimax wireless network access units (NAU). These units operate at 5.8 GHz frequency band and are capable of supporting the data rate and coverage area requirements. Wimax units are equipped with MIMO (Multiple Input Multiple Output) antennas and perform well under no-line-of-sight situations (NLOS). These units are also used to transmit the DGPS correction signals.

On each UGV, there is also a low frequency (915 - 928MHz) radio modem. This modem is mainly used for E-stop/Freeze communications. The E-stop Terminal located in the GCS communicates with these radio modems. Also these modems can be used for backup communication.

2.5. Vehicle Control Management System

HLC and VSMM modules are included in this subsystem. HLC is described in detail in Section 3.2.

3. UVS Autonomy & Coordination Strategy

The DMP module generates a mission according to the current operational situation. Robots are assigned with planned tasks from the DMP. The primary task for the Observers is exploration. DMP tasks each Observer with an exploration and provides a planned path. The rest is undertaken autonomously by the UGV based on its internal logic, depicted as a hierarchical finite-state machine (FSM).

3.1. General Controller Hierarchy for UGVs

The controller for each robot is designed in a way that emphasizes the difference and interaction between the decision process and the more conventional, regulatory control.

The situation-dependant decision-making controller is in discrete states by nature, while the feedback loop in charge of the longitudinal and lateral controls deals with continuous states such as position, orientation and velocity. This coupling of a discrete-state system (DSS) and a continuous-state system (CSS), as seen in Figure 4, is commonly referred to as a hybrid-state system (HSS) **Error! Reference source not found.**][2].

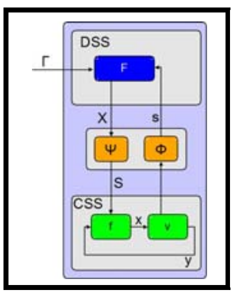


Figure 4: Hybrid system blocks

The DSS portion of the controller is designed as a FSM that makes the necessary high-level decisions for the overall robot operation, and is represented as the High-Level Controller (HLC) in the architecture [3]. In contrast, the CSS portion is part of the LLC; it is implemented as a real-time control system.

The primary interfaces of the HLC are:

- Receive tasks from and report status back to the DMP,
- Send waypoint following or robotic commands to and receive status back from the LLC,
- Receive the locally stored occupancy-grid map and a list of live (local-sensor based) and confirmed (received from central command) OOI from the SFM.
- Send general acquisition, specific tracking and neutralization commands to the ATT for corresponding operations.

3.2. High Level Controller

In the tiered controller structure described, the HLC internally also utilizes another layer of hierarchy. This hierarchical FSM consists of multiple *meta-states*, seen in *Figure 5* for the sensor robot, each of which corresponds to one general mode or situation such as *"Explore"*, *"Neutralize Mobile OOI*" or *"Wait"*.

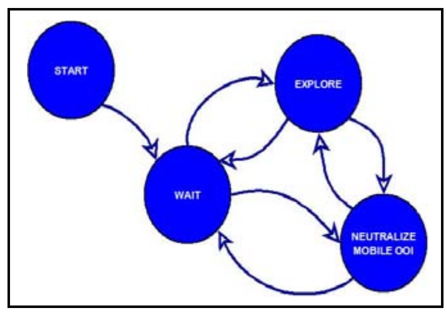


Figure 5: Meta states for sensor robots

The meta-states or "states of states" are connected to each other in the overall FSM, while each of them contains a situation-specific state machine that is responsible for the operation in that particular mode, as seen in Figure 6 for an Observer UGV. So the overall HLC structure can be described as a state machine of state machines, with the meta-states connected through special sub-states of their internal FSMs, marked as "enter" and "exit".

3.3. The Meta States

As seen in the connection of the meta states (Figure 6), the general sensor robot behavior consists of initialization ("Start" meta state), waiting for a task, exploration and interaction with the mobile OOI.

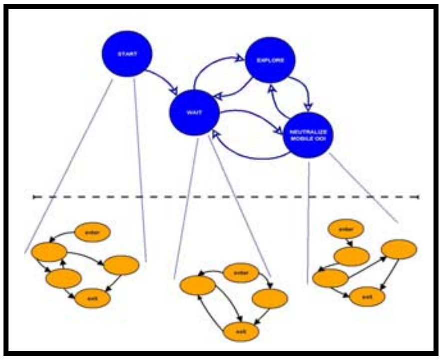


Figure 6: Meta states and corresponding sub-state machines.

When switched to the “Explore” meta state (see Figure 7) the robot utilizes basic obstacle avoidance to reach the given task position, while continuously mapping the environment. If there is an obstacle that cannot be negotiated via simple, local obstacle avoidance, the robot reports back that it is “stuck” and starts waiting for a new task.

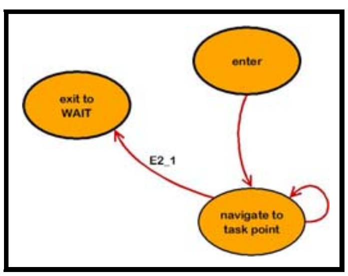


Figure 7: "Explore" meta state for sensor robots.

Once the current task is completed or when the robot receives a “drop task” command, the HLC exits the “Explore” meta state via the designated *exit* sub-state, back to the “Wait” meta state.

This compartmentalized and hierarchical structure of the controller allows the decision-making mechanisms to be developed independently for each situation and additional states can be added to each meta state without affecting the remaining meta state machines.

The mobile OOI operations are handled in the ‘Mobile OOI’ meta state (Figure 8).

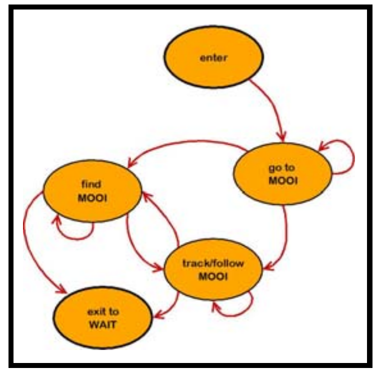


Figure 8: "Mobile OOI" meta state

When tasked by the mission planner to seek and neutralize a mobile OOI, the sensor robots follow the DMP-generated path to get within a certain range of it, while continuously updating the position of the target via the confirmed OOI list that is sent by the MRDF. Once the robot is within a certain range of the OOI, it either starts tracking and distance keeping maneuvers, or if the line of sight to the OOI is lost, seeks clear line of sight using its internal grid map. During tracking and distance seeking maneuvers, the HLC continuously tries to match the confirmed OOI location to the live OOI locations reported by the SFM.

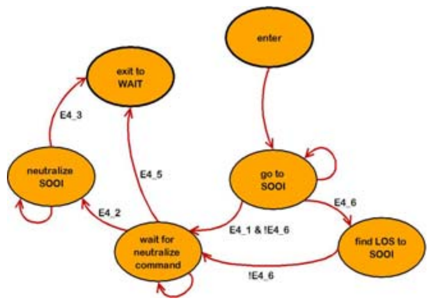


Figure 9: "Static OOI" meta state for disruptors.

Similar to the "Mobile OOI" tasks for the sensor robots, the disruptor robots are tasked to neutralize static OOI (Figure 9). The overall procedure is similar to the mobile OOI case in the sense that the robot approaches the static OOI, seeks line of sight and tracks it by live-to-confirmed matching and commanding the ATT.

Since the static OOI neutralization further requires the marking of the OOI with the laser pointer, this last phase of neutralization is first confirmed from the OCU to avoid any accidental neutralizations.

3.4. Waypoint Navigation and Obstacle Avoidance

The HLC receives the paths generated by the DMP, uses an obstacle avoidance routine on it and provides an obstacle-free path for the LLC to follow (Figure 10). In case of an obstacle-free path that is significantly different from the DMP-intended one or if one of the waypoints that DMP generated is completely unreachable, HLC drops the current task and asks for a new one that takes the newly connected map and OOI info into account.

Given the current grid map in MRDF, DMP generates tasks for HLC (dashed path in the figure). As the robot follows it, new obstacles or OOI are discovered, so the HLC utilizes a width-first graph search from robot grid cell to target grid cell to avoid them.

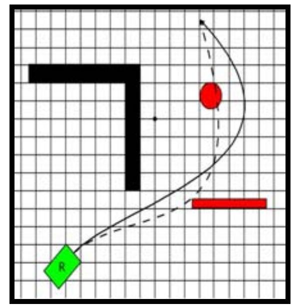


Figure 10: Obstacle avoidance over the local grid map

3.5. UGV Cooperative Activity

Cooperative activity among the UGV’s can be summarized for three types of situations:

1. In the “Explore” state, the GCS could assign one or more UGVs to the same “segment” (See Operational Strategy in Section 8.3).
2. Upon discovering a static OOI, an Observer can call and collaborate with a Disruptor.
3. Upon discovering a hostile mobile OOI, two Observers will be dispatched to track and neutralize the OOI.

4. Sensors, Processing & Mapping for UGVs

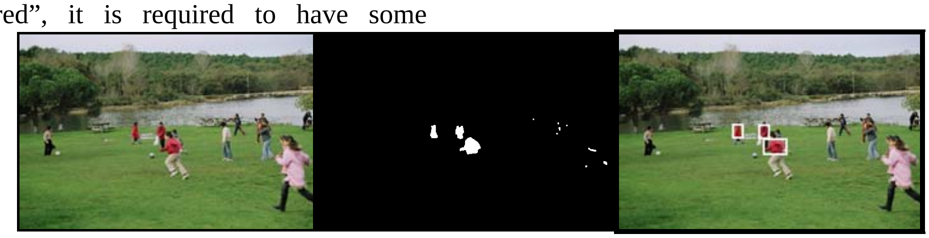
Sensor inputs are preprocessed, i.e. verified, validated and scaled as required prior to processing. This preprocessing include conversion of sensor data into the robotic coordinate frame, removal of incorrect instantaneous data, consistency checking within desired and actual dynamic constraints and conversions to handle dynamic motion like movement in inclination. The fiber optic gyroscope is also calibrated to account for its bias and is done automatically by the system during start up and upon user request when needed.

Most of the information used in the UGV operations is the LADAR as it is the sole source of mapping. It is therefore important that its data is suitably processed and free of major defects. The most obvious correction is to exclude scans reflected off the robot's own body. Further corrections include ignoring long-range scans, as the probability of ground strikes and noisy data increases and determining ground strikes due to the vehicle's roll and pitch.

4.1. Automatic Target Detection and Tracking

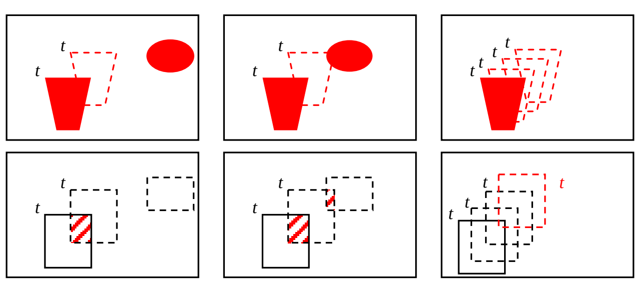
For OOI detection, color information is used. The color red is defined in Hue-Saturation-Value (HSV) color space where the Hue component corresponds to the color information. For an object to be identified as "red", it is required to have some minimum pureness (saturation) and brightness (value). Even for an object that is red, the apparent HSV values for the object vary significantly under different illumination conditions. Therefore, the HSV subspaces for different illumination conditions are provided as presets to the user.

For the detection, the PAL analog video is decoded as YCbCr format. To decrease computational complexity, the YCbCr image is converted into HSV representation by using lookup tables that are generated for red objects in YCbCr color space. At the system start-up, the YCbCr values that correspond to red are marked. These values are placed in lookup tables that are used during the operation. Each lookup table corresponds to a specific illumination condition.



(a) (b) (c)

Figure 11: Red object detection. (a) Original image, (b) result after thresholding and opening, (c) objects that pass the size elimination.



(a) (b) (c)

Figure 12: Illustration for the tracking algorithm. (a) Single overlapping object, (b) multiple overlapping objects, and (c) usage of motion history at time t_3

The detection is made in per-pixel fashion. The resulting binary mask is then cleaned using the morphologic operation, *opening*. By using the connected component analysis (CCA) [5], the bounding boxes of the red objects are determined from the pixels. These are passed to size elimination process to eliminate targets that are too large or too small for the pre-defined OOIs. Here, a simple assumption is made: "The OOIs are on the same plane with the robot". This assumption allows using the line-to-distance relation in a pre-defined image and this relation is used to find the visible size of the objects in the elimination process. The results are presented in Figure 11.

After the targets are found, at most 5 of them are marked by numbers where it is important to keep the labels of the targets consistent among frames. A memory of past targets is kept and in each frame, the overlapping areas of the current targets and the previous targets are calculated. The same IDs are assigned to the targets which happen to have the largest overlapping area with the previous targets.

For tracking operations the bounding box of each object is inspected for its intersection with the bounding box of the tracked object in the previous frame (*model window*). The object with the largest bounding box intersection is selected (**Error! Reference source not found.**.b).

Tracking uses the motion history (trajectory) of the object as follows: The average velocity (pixels/frame) of the object in the latest 50 frames is calculated. The bounding box in the previous frame (*model window*) is shifted by this velocity to obtain the expected location of the target in the current frame and is used for intersections with the candidates in the new frame.

If the OOI is lost due to occlusion or error in the detection phase, the *model window* in the last successful tracking frame is enlarged before the intersection analysis. The enlarging is performed gradually as long as the OOI stays lost. The position update using average velocity information is also used during the OOI-lost frames. The tracking is terminated if the OOI loss state is continued for 75 successive frames, and the VTU returns to the detection mode.

In this system, the video tracking is combined with camera tracking. The PTZ camera is controlled by the LLC through the serial port. The VTU sends the LLC the angle between the center of the image and the tracked OOI's center pixel. The angle is computed from the field of view (FOV) for the current zoom level, and the video image dimensions.

4.2. Sensor Fusion & Mapping

The LADAR data and the pose of the robot are used in generating the map of the area. The raw data obtained from the LADAR is transformed into real-world coordinates using the robot pose. Each robot constructs a local map and these maps are united to form a global map. In generating these maps the "Occupancy Grid Map" method is used. With this method, the errors in the map coming from moving objects and spurious data are also eliminated. The grid cells of the local map are graded according to the traversability of the obstacle ahead of the robot. The grading ranges from "not traversable" to "traversable'. There are also grades such as "not enough information", "likely traversable" or "likely impassable". This methodology has been successfully implemented in the OSU DARPA Grand Challenge vehicles and ASELSAN UGVs [3]

These local maps are transferred to the MRDF and a global map is constructed. The global map is used by DMP for planning, while the local map is used for obstacle avoidance and low level path planning of the UGVs.

4.2.1. Sensor Fusion Module (SFM)

The Sensor Fusion Module has two primary purposes: creating an occupancy grid and matching the ATT tracking information into live OOIs. This data is made available to the HLC and MRDF modules.

Each grid cell in the occupancy grid can take on values from 0 to 255. A ‘0’ value means that the cell is unexplored and no LADAR rays have intersected with it yet. A low value means that it is an unoccupied cell. A high value signifies that the cell is occupied. The first time a cell is explored, the value is changed from 0, unexplored, to 128, explored but uncertain. The mapping of that cell then progresses with each LADAR scan.

When SFM receives pose and LADAR data from the LLC, it starts with positioning the robot into its occupancy grid. SFM then samples each LADAR ray at half the grid cells’ resolution and fixes that point on the grid. After each LADAR scan step, the occupancy grid is updated in accordance with the Bresenham Line Algorithm [6].

Each sampled point is weighted depending on its distance from the robot and the perpendicular distance between the sampled point on the ray and the corresponding cell’s center point, thus avoids wrongful deletion of a wall parallel to the laser ray. Occupied cells are marked on the grid more easily than the unoccupied cells. This provides the DMP with more information for long distance planning, while keeping the far away open areas marked as unexplored.

ATT provides SFM with possible static and mobile OOI positions in terms of an angle relative to the robot's position. SFM will attempt to match this relative angle to the corresponding LADAR ray. It will then look for the closest laser strikes within the bounding box of the given camera angle and translate that distance into UTM coordinates. This constitutes a live OOI, an object likely to be an actual OOI but hasn’t confirmed to be so by an operator. Since the system, by design, doesn’t operate on unconfirmed OOI, SFM attempts to match the live OOI with a list of confirmed OOI that was generated through user action.

SFM sends its grid map and the live OOI that it has generated to MRDF and HLC modules.

4.2.2. MRDF Module

The multi-robot data fusion module receives all of the grid and live OOI data from all robots' SFM modules. All Disruptor robot grid data and live OOI are stored for display purposes, but are otherwise unused for fusion.

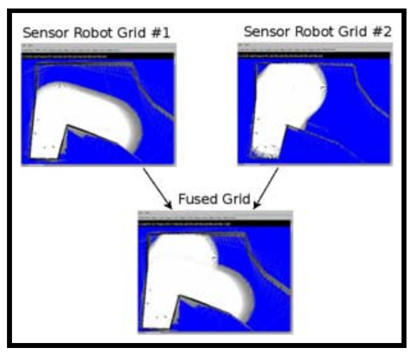


Figure 13: SFM Grids and Fused MRDF Grid

SFM sends two types of grid packets, full and intermediate. The full packet is simply a grid update 50 m^2 around the robot; it is used to initialize the individual maps. The intermediate packet is also the grid 50 m^2 around the robot, but the only nonzero values are cells that were modified during that scan.

During an intermediate update, if the cell is nonzero the number of observations for that robot is incremented for that cell. The number of observations is used as part of a weighted average of all the sensor robots.

Figure 13 shows the grid for two sensor robots and the outcome of the fusion process. After the data is fused it is passed to DMP and OCU.

MRDF is also responsible for broadcasting all confirmed OOIs listed in the WMKS to each of the robots' SFMs for matching with live OOIs. MRDF also receives the live OOIs from SFM modules to add them into WMKS as well, for OCU to display them to the operators.

5. Operations in GPS-denied Environments

One of the challenges present is the multi-robot mapping problem in the absence of GPS, where it is critical to maintain the construction of the map without any drift or becoming distorted. In order to cope with this problem, an approach utilizing a decision making process on two independently running localization algorithms is implemented. The decision making module, checks the consistency with the internal state sensing and the Scan Matching algorithm results and then decides on the best pose of the UGV.

5.1. State Sensing

The state sensing software module running in the LLC is responsible of generating the pose for the UGV from various sensors (DGPS, yaw rate gyro, IMU, wheel odometers) both indoors and outdoors. The state sensing module has two main operational modes: GPS-RTK and unreliable GPS.

At its core, the state sensing module operates Kalman filters with different measurement update models [7], [8]. In the prediction step a differential drive robot motion model is used. The UGVs are restricted to two types of motion, translation and rotation; this provides the chance to use the differential drive motion model on the skid-steer UGVs. As the robots are controlled to turn around their center of mass this assumption holds. After the mean right and left wheel angular velocities are found the motion model used is given as:

$$\begin{bmatrix} x_k \\ y_k \\ \theta_k \end{bmatrix} = \begin{bmatrix} \frac{r \cdot (w_r + w_l)\Delta t}{2} \cdot \cos(\theta_{k-1}) + x_{k-1} \\ \frac{r \cdot (w_r + w_l)\Delta t}{2} \cdot \sin(\theta_{k-1}) + y_{k-1} \\ \lambda . \frac{(w_r - w_l) \cdot r}{L} \cdot \Delta t + \theta_{k-1} \end{bmatrix}$$

Where r is the radius of the robot wheels, w_l and w_r are the angular speed of the left and right wheels, L is the distance between the two wheels (shaft length) and λ is the conversion of the number of wheel rotations of the actual robot to a differential drive robot with the wheel radius of r and the shaft length of L.

After the prediction step, the pose of the robot is updated according to the yaw rate gyro and the accelerometer readings of the IMU. The measurement model for the yaw rate gyro and the accelerometers together is given as:

$$\begin{bmatrix} x_m \\ y_m \\ \theta_m \end{bmatrix} = \begin{bmatrix} \frac{r \cdot (w_r + w_l)\Delta t}{2} \cdot \cos(\theta_{k-1}) + \frac{a_x (\Delta t)^2}{2} + x_{k-1} \\ \frac{r \cdot (w_r + w_l)\Delta t}{2} \cdot \sin(\theta_{k-1}) + \frac{a_y (\Delta t)^2}{2} + y_{k-1} \\ \omega \Delta t + \theta_{k-1} \end{bmatrix}$$

Where, ω is the rate of turn from the yaw rate gyro and a_x, a_y are the accelerations measured from the IMU. During GPS available environments, a simpler measurement model is applied for the GPS position and the heading is extracted from the GPS update information as they directly correspond to the measured vehicle state.

The preceding equations are implemented using several interlinked Kalman filters. These filters can be switched on and off depending on different conditions. Figure 14 shows the details of this implementation.

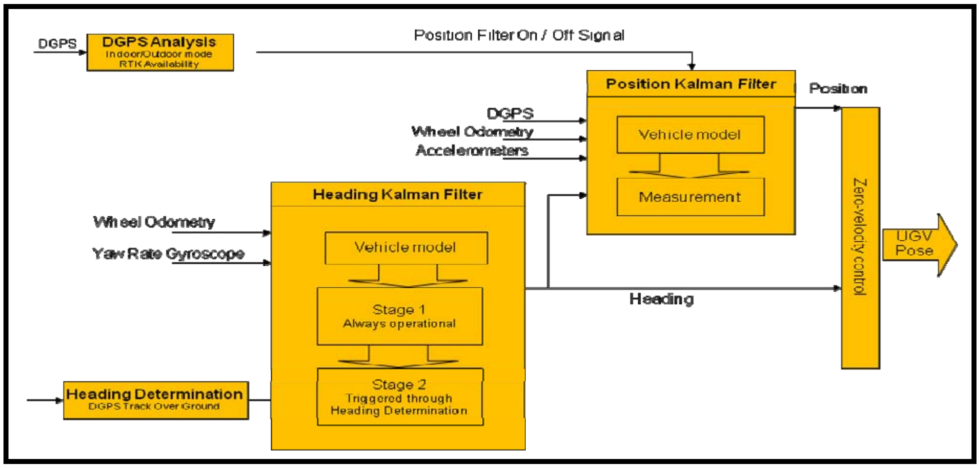


Figure 14: Flow chart for the state-sensing module

The position filter is only operational during GPS-denied operations, since in RTK mode, the GPS unit can provide localization with a precision of 5cm or less. As such, the Kalman filter acts only as a pass-through for the data supplied by the GPS. When RTK precision is lost, however, the filter is brought online. Experimental results have shown that this method gives analogous results to feeding the DGPS information into the filter and having the error matrices updated according to the standard deviation.

The heading block has one Kalman filter that is constantly in operation. This unit receives data from the yaw rate gyro as well. This approach gives very good results for moderately short durations and allows the robot to execute precise turns. However, it has the disadvantage of accumulating errors over time. To solve this particular problem, the state sensing module has a subsystem that extracts heading information from the GPS track of the robot's last straight segment. This value is fed to the second stage of the heading Kalman filter. This filter uses various parameters such as the GPS standard deviation figures and the total error made during the line fitting process to generate the error covariance matrices. On the segments where this heading extraction is not possible, the first-stage Kalman filter operates alone.

Post-processing modules act as oversight modules and ensure that during zero-velocity operations, the state sensing output reflects this and suppress any erroneously calculated change.

Tests have shown that in GPS available environments the GPS-based heading extraction resulted in high-precision maps. During GPS outages or indoor operations, experiments have shown that the state sensing module has a maximum positional error between 1-5% depending on the motion and surface.

5.2. Scan Matching

A major shortfall of the State Sensing method is that it is an incremental result. All calculations are done with respect to the position found at the previous time step; the solution may as a result diverge from the robot position. Incidentally, the UGV is equipped with a laser range scanner that can, in addition to its other duties, provide instantaneous range scans that can be used for localization.

One of the ways in which planar range data can be used for such tasks is scan matching [4], where range readings from the current position of the robot are matched to either an existing map or to data from an earlier scan. Transformations that minimize the discrepancy between the expected and actual range readings hence correspond to most likely alternatives for the current pose of the robot. ASELSAN has developed a novel method for robust real-time ranged scan matching without any odometric information by using a variety of invariant geometric relations derived from line segments fitted to 2D range scan data.

The most novel aspect of this implementation is the exclusion of any assumptions on the presence of odometric information. The second important property of the module is the use of simple features as opposed to point clouds, together with a solution to the correspondence problem through a novel scoring approach guided by pose-invariant geometric features. This reduces the computational complexity of the problem and allows pose information to be generated in real-time on an embedded system.

The algorithm proceeds in three stages. First, raw range data is transformed into line segments, transforming the problem into that of finding corresponding line segments across two different scans. The second stage identifies various different geometric relationships among the line segments in each scan. Finally, a score table based on these geometric relations is constructed where each element corresponds to a list of scores relating to the distinction of a matching geometric relation across two geometric relation sets. The sum of scores in the same list corresponds to the likelihood of correct line pairings across the two scans [9].

The algorithm has a clearly independent nature: it only uses the range scans to provide the robot pose information in real-time. The fact that it doesn't require any a priori information about the world is a definite advantage in reducing the inter-module communication load, as well as giving it the ability to start before any mapping is done. This is a necessary feature as the competition does not allow the UGV collective to have any prior information about the interior of buildings where the Scan Matching algorithm is the most useful.

5.3. Decision Making

The fact that there are two independent modules that both generate pose information requires a method to merge the solutions. The Decision Making module uses the solutions of the two modules to generate what is called the best pose of the UGV. This algorithm has multiple duties: synchronizing information arriving at different sampling rates and deciding the pose given conflicting information.

The first task is necessary due to the fact that the LLC has the fastest sample rate in the system and requires an update at every iteration. The Scan Matching module is unable to provide information at this rate, creating gaps to be filled. In such situations, the decision making module uses the available information, created by the State Sensing module alone, to infer the appropriate pose information between LADAR scans.

The second task is perhaps the more critical of the two: oftentimes the localization algorithms generate conflicting pose information; the decision making module has to decide which of the modules are more trustworthy. The main criterion is the difference between the recently calculated and last known good poses: motion that the UGV is unlikely to make, due to its mobility, is rejected. However, this method will fail to find the correct pose of the UGV in some situations, most of them unintended, like slipping down a slope. Experimental results have shown that the localization algorithms are more prone to erroneously create these solutions than the UGV to fail in its mobility character, justifying this decision.

During trouble-free operation, where both the State Sensing and the Scan Matching algorithms output plausible information, the Decision Making module has to make a choice between the results. The rotational and translational motions, reported by the two localization modules are weighted to create the best pose, where the weights have been determined experimentally. The two modules are not reset to the best pose, as the Decision Making module only considers incremental changes and can cope with the solutions being different from the UGV's best pose.

6. Processing and Fusion of UAV provided Metadata

Automatic segmentation of structures including buildings, roads and vegetation from an UAV image is a primary step in understanding the image for initial path planning. It is required that the aerial picture should be classified into definite classes, such as sealed roads, paths, buildings, trees, grassed areas, sandy ground and so on, for initial planning. There are mainly two types of approaches, which are supervised or unsupervised. As for unsupervised technique, an unsupervised fully-automatic segmentation algorithm that employs structural, spatial and spectral information was used [10], [11]. In this algorithm, segmentation relies on the mean shift algorithm, the bandwidth parameters of which are determined automatically employing DMP without any prespecification. Therefore, this method is fully automatic, unsupervised and data-driven without any user interaction. Basically, the size at which the first maximum is observed, is related to the spatial bandwidth and the mean difference of the structure at that scale from its neighbors is related to the range bandwidth [11]. Figure 15 shows the results of the algorithm.

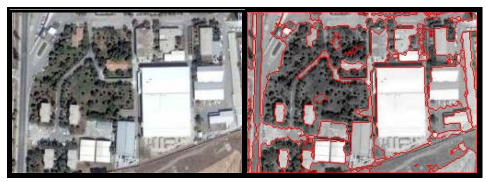


Figure 15: Automatic segmentation

If training images are available then it can be possible to classify the aerial picture into meaningful and consistent modes corresponding to objects of interest such as roads, buildings, grassed areas, sandy ground and so on, for initial phase planning. Two learning methods are utilized: 1) Adaboost

[12], 2) Support vector machine (SVM) [13], [14], [15]. A number of local textural features such as mean and standard deviation of image intensity and gradient, Zernike moments, Haralick features, Circular-Mellin features, Fourier Power Spectrum, Wavelets, Gabor Filters, and a set of statistical features from HSV color space are extracted. Adaboost learning algorithm is employed for both classification and determining the beneficial feature subset, due to its feature selector nature. The extracted features were used as input for SVM. Figure 16 presents the results of SVM and AdaBoost methods, respectively.

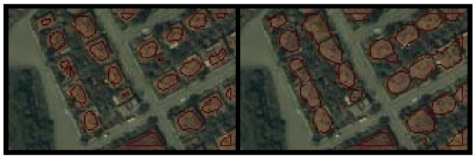


Figure 16: Left: SVM result; right: AdaBoost result

Another approach is to incorporate priori information such as the rough layout of the scene (Figure 17). This information is valuable priori reference data for registration and localization of the objects under investigation. The idea is to find the correspondence points between the reference image and the test image. Consequently, a transformation matrix based on it is formed. Then the atlas image can be transformed to the test image and the image is labeled automatically. A rigid registration with 4 degrees of freedom (2 rotation, 2 translation) is used to find the transformation matrix. Two different methods are proposed for the registration: 1) landmark based registration and 2) registration based on maximizing the mutual information.

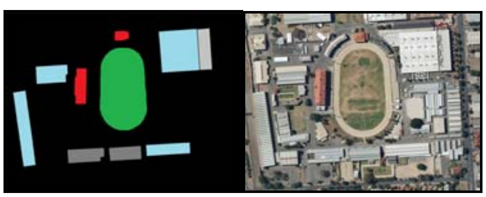


Figure 17: Left: rough representative layout; right: reference image

In the landmark based method an operator is requested to locate at least 4 landmark points on the reference image and also the corresponding points in the test image. The transformation matrix is then calculated based on these points. Landmark based registration produces fast and accurate results and the only disadvantage of this method is the user interaction to mark the landmarks.

The mutual information is a metric that measures the mutual dependence of two variables. For two images, the mutual information can be measured by using intensity vectors [16]. By maximizing the mutual information between the reference image and the test image, the rotational and translational deformations can be estimated. Calculating mutual information for all possible rotation and translation values results in accurate segmentation, but it has a high computational load. This method does not require user interaction and is a fully-automatic method. The results of both methods are shown in **Error! Reference source not found.**.

Figure 18: Registration and classification based on left: landmarks; right: mutual information

7. Situational Awareness Tools and Human Machine Interface (HMI)

A novel Operator Control Unit (OCU) for the supervision of multiple UGVs is developed. Decreasing operators' workload was a significant concern in the design stage. However, unless the HMI is designed to support operators' cognitive capabilities, autonomy can decrease operator's situational awareness. Hence keeping the operator in the loop is an important design criterion. Although the system is acting autonomously, it is important to understand the current situation and the response of the system. The GUI (Graphical User Interface) aims to display the required information in a way that is easy to recognize and comprehend. It is also supported by the audio modality to decrease the operators' load of visual modality and increase situational awareness. The layout of the GUIs and the graphical representations of each element within the GUIs are determined using human engineering methodologies. GUI elements are optimized especially for minimizing keyboard usage and increased human perception using graphical representations rather than using textual presentations.

The Operator Control Unit: Each operator's situational awareness is maintained through an operator control unit set consisting of one computer, two monitors, one wheel and a joystick (Figure 19). Both operators can supervise any of the UGVs as they require. There is also another set of computer and monitor for dynamic mission planning, which can be considered as an external mission planner node such as a C4I or observer in the operational theatre.

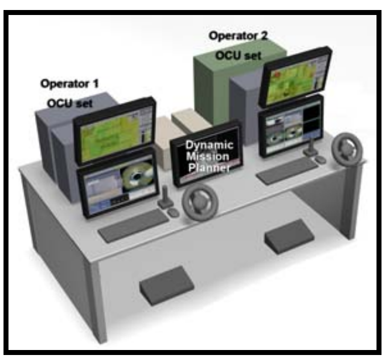


Figure 19: The Operator Control Units.

The OCU is developed as a JAUS component that can control and monitor other JAUS components which can either be an unmanned vehicle or its payload. The OCU can work with any JAUS compliant system and the number of units that the OCU can simultaneously work with is unlimited. The OCU also supports plug & play COTS HMI units.

One of the OCU monitors is used to display information for situational awareness on the 3D GIS tactical map and the other monitor is used for UGV status monitoring, command and control.

Tactical Graphical User Interface: The tactical GUI completes the situational awareness picture by displaying the UGVs and OOIs on the GIS map with enhanced 3D capabilities. Beyond the bird's-eye view, 3D monitoring gives the operator a more realistic and comprehensive image. The color scheme of the tactical graphical user interface is chosen to make a distinction from the UAV image and display in an easily recognizable contrast. The tactical GUI enables the operator to follow the robots' real time positions, the tracks they have passed before and their planned paths on the generated map. OOIs and their lethality zones are also displayed. Hence the operator is able to comprehend the situation and make projections for future plans; when necessary the operator can interrupt an operation and supervise the robots, e.g. via marking high interest zones or no-go zones. The neutralization processes can be easily followed and the types of OOIs can also be determined directly from the map. The map can be focused on any entity the operator selects. Entities in the operational area are organized into map layers which can be hidden or displayed. The area entities can be modified by point and click map tools provided, to draw, edit or delete map areas like buildings, no-go areas, high-interest and high-risk areas. A context menu for each entity contains related actions.

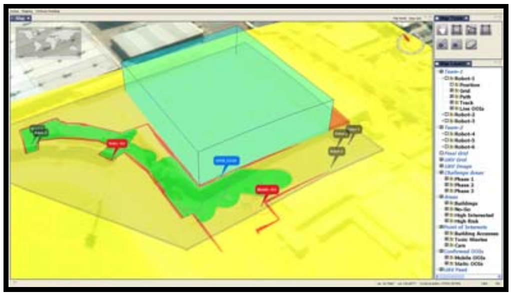


Figure 20: A view from the tactical GUI.

Status Monitoring: This screen mainly shows status information about one or more robots with their live video feed. Also the user interface for the tele-operation of the UGVs is presented. Specifically, there are two perspectives for displaying the information of the UGVs. One perspective displays detailed information with a large video feed only for a single robot in case detailed information is required. Still, in the single robot perspective the vehicle information bars of the other two robots are visible as not to lose track of them. The other perspective is for maintaining situational awareness of the three robots at the same screen showing mission-critical information of the UGVs (Figure 21).

The challenge information view is designed to give information about the challenge. The time spent in each phase, the number of static OOIs neutralized and active in each phase, the number of confirmed OOIs neutralized and active in each phase, and the total time spent in the challenge are displayed on the screen.

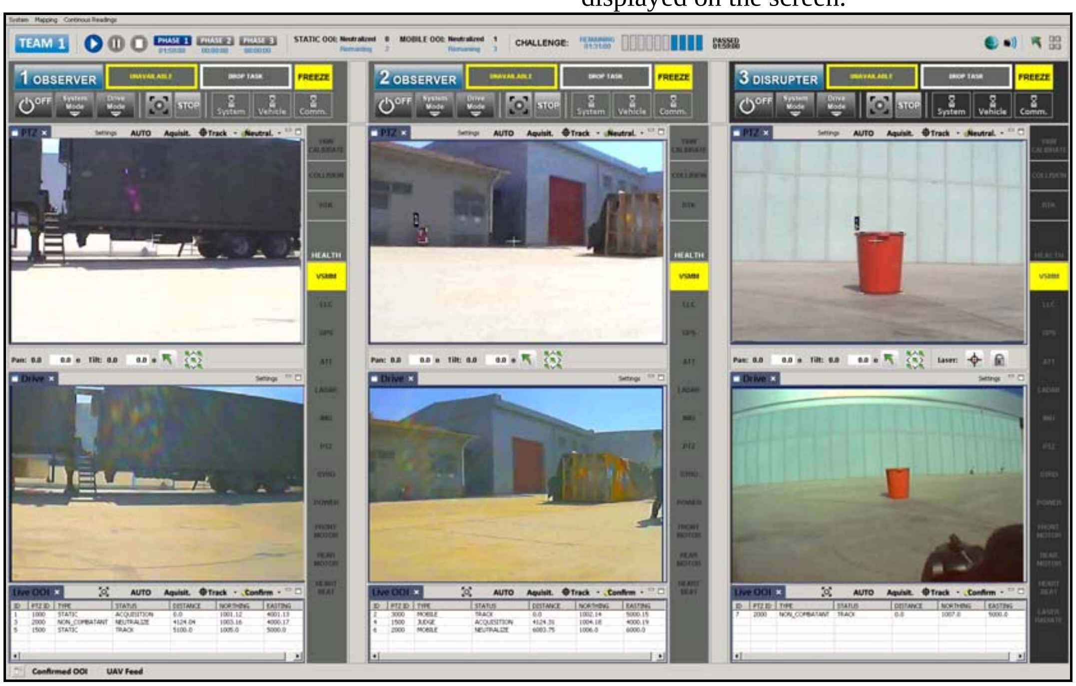


Figure 21: Perspective for three robots

Vehicle Information Bar: The vehicle information view is designed to provide online information about the UGVs in a user friendly way. In order to support users' situational awareness the information shown is coded in colors, which enables the operator to recognize and comprehend easily. The information provided are the power status, the system and drive modes, the vehicle and system battery levels, the communication level and the system status. The commands and states received from the dynamic mission planner are also shown in this view with the graphical indicators. The user is also capable of focusing on a robot on the map and stopping the UGV.

Figure 22: Vehicle Information Bar

Health Monitoring: An intelligent health monitoring capability is embedded within the UGVs to increase the situational awareness for the UGVs. The health monitoring module performs a continuous built-in-test function for the UGVs, and warns the user if any of the UGV components has a problem. There is also a tele-operation mode that can be activated for each robot through the OCU screen. Each operator can select and control one robot at a time using the HMI equipment.

Layout of the GUI elements can also be customized according to the operator needs. For example, operators can have their own customized layouts for different operation modes or their preferences during operation. Customized layouts can be saved, restored and switched in runtime.

The human-UGV interaction within MAGIC is limited to phase starting/ending, Static and Mobile OOI confirmation and neutralization (operator permission for neutralizing) and emergency situations (i.e., UGV mobility problems).

8. Mission Operations Strategy

The operational system has multiple stages as shown in **Error! Reference source not found.**. The DMP accomplishes task decomposition, scheduling and path planning. UGV operations are then initiated. When mission-interrupting events occur, the system returns to the Planning Stage. The DMP monitors robots' status and periodically updates the map from the WMKS. New plans are generated whenever necessary.

8.1. Dynamic Mission Planning: Segmentation-based exploration

We utilize a segmentation-based exploration scheme [17] towards exploration, target search and neutralization. The segmentation-based approach takes advantage of prior and newly acquired knowledge of the structured environment and sends the robots to different parts of the area to improve efficiency, thus reducing interference between individual robots.

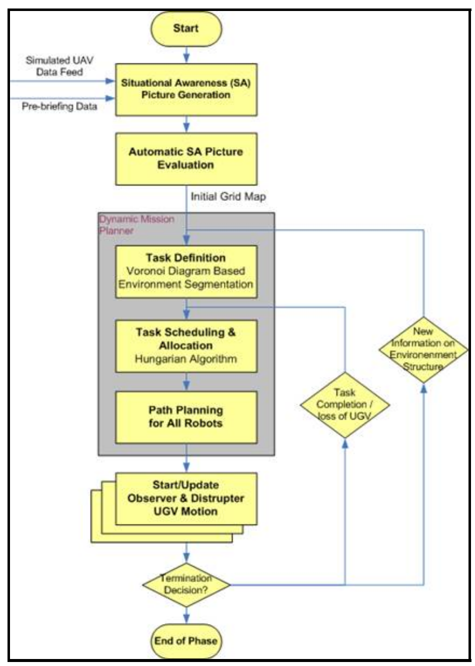


Figure 23: Dynamic Mission Planning Architecture

8.2. Segmentation of area

A segmentation of area is to discompose the area of interest into non-overlapping segments based on the structure of the area.

Given an occupancy grid map, a Voronoi diagram is constructed to interpret the structure of the area in terms of occupied and free space. Specifically, the points on the Voronoi diagram with minimum local clearance are identified as critical points. They typically represent doorways and other relatively narrow passages. The overall area is divided at these points into obstacle-free, non-overlapping zones. Figure 24 illustrates the segmentation scheme for an example area.

8.3. Task Assignment and Path Planning

After task decomposition, a cost matrix is constructed based on the predicted traveling time of the robots to individual segments. The cost calculation also takes into account the exploration priority of designated regions. The Hungarian Method is employed to assign the segments to appropriate robots for a minimum overall cost. The path planning algorithm generates an obstacle-free path for each robot to navigate to its designated segment with minimum cost.

8.4. Exploration within individual segments

The robot coordination and planning within a single segment relies on a frontier-based approach [18]. A frontier grid is defined as a grid on the line separating explored and unexplored parts of the segment. Robots are sent to selected frontier grids based on traveling distance and the utility of the frontier which is calculated based on the size of the unknown area that will be covered when the robot arrives at the grid.

8.5. Dynamic Re-planning

During the exploration, the mission planner communicates with all robots to monitor their status. It also updates its map from the MRDF and the information regarding OOI's from WMKS periodically. When a robot completes its task or cannot follow the given path due to newly discovered obstacles, the mission planning module re-calculates segmentation and task assignments, and sends a new path to the robot. In the event of loss of a robot, the module puts the unfinished task back into the task list, and generates new mission plans for the rest of the robots.

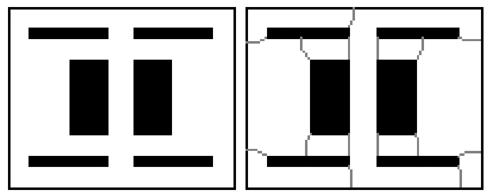


Figure 24: An example map segmentation; Initial map (left), Voronoi segmentation (right)

8.6. OOI Neutralization

When the operators confirm the presence of an OOI, DMP acts accordingly to neutralize it. For static OOI, DMP assigns a neutralization task to a disruptor and plans a path so that the robot can have a line of sight. In the event where that disruptor is unable to complete its task, the other disruptor robot is also sent to the area. For mobile OOI, DMP chooses two observers and tasks them to follow that OOI.

9. Risk Reduction Strategy

All the subsystems, sensors and payloads of the UVS were selected after extensive and detailed investigation. Special focus was given to check the applicability of COTS items to satisfy the challenge environmental conditions. Also, for subsystems especially at the board level, new designs incorporated MIL-STD adaptations. This approach was useful both on the integration and the testing stages where known design rules are proven to work and reduced any possible risks that could be faced upon.

Additionally, in order to reduce the possible risks associated with the development stage, the identification of the risks was done and respective solution strategies were created. This approach was applied to all UVS subsystems.

9.1. EMI/RFI & Electrical

The precautions in order to minimize Electromagnetic Interference (EMI) are generally taken in three levels; Board Level, Line Replaceable Unit (LRU) Level and System Level. These precautions deal with susceptibility to EMI and the emissions of EMI from the equipment or the components.

At the board level, filters are used on board to eliminate conducted interference on cables and wires. The filtering elements are selected based on the operating frequency range and the characteristics of the components to be filtered.

At the system level, twisted and shielded cables are used on the vehicles when necessary, and system cables are grouped according to their functions such as power, analog and digital. General shielding and gasketting techniques are also applied.

9.2. Vibration & Physical

The selection of the vehicle base as well as the design of the structure of the UGV was made to meet all the vibration and physical requirements for the challenge environment.

9.3. Modeling & Simulation

The strategy of Team Cappadocia in realizing the solution was to accelerate algorithm development and verification by using the advanced modeling and simulation infrastructure that was already available. The modeling of the environment, the UGVs and the GCS with the associated interactions was accomplished through the available in-house capabilities of the team. Especially, the simulator and several test bed robots were used until the MAGIC 2010 test areas and MAGIC vehicles were deployed in the field.

The core of the simulator is a software architecture called the “Cooperative Algorithmic Test Bed Software Environment”, or CATSen for short, which is a modular package, developed by OSU, that allows testing of different algorithms, full or partial hardware-in-the-loop simulations, and finally can be used for run-time operations by exactly replicating hardware and software module interfaces in the simulation environment. The core of the simulation is based on a modified version of the open source Gazebo software package [19].

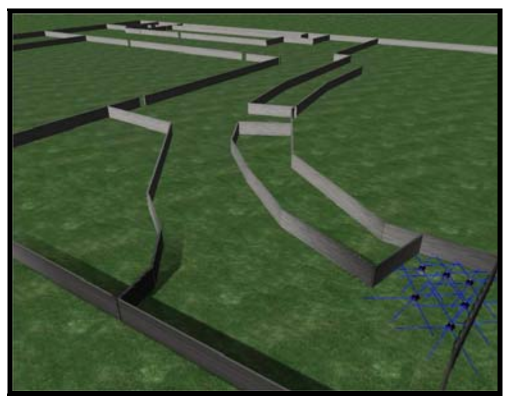


Figure 25: MAGIC Phase 1 Simulation Example

This open-source world physics simulator was further developed and adapted for the specific requirements of the challenge. The main purpose was to have a dedicated testing environment for the higher-level control modules and mission control strategies. The simulator has also enabled initial qualification tests of some modules, such as sensor fusion, in a no-noise environment.

The simulator environment consists of an abstraction layer, where the physical world, robot sensors and payloads, and the low level control modules reside. The simulated physics have been designed so that the vehicle/payload model conforms to the real UGV. The static and dynamic OOIs within the challenge area were modeled and implemented in the simulator as well including movement behavior for mobile objects.

The communication between the LLC and the interface software is established through the network using UDP packets. This allows the simulator to be used in validating the actual LLC hardware. The software modules that physically reside in the LLC computer can thus be tested and optimized without the need to use the actual robot hardware.

The simulator takes advantage of the Ethernet-based architecture: messages in both the real and the simulated system are exactly the same. The abstraction layer therefore operates as a true hardware in the loop system, where the outer modules operate in the same manner whether they are in the simulation or in the real world.

This setup has had the impact of reducing the total development time by a large margin. The simulator system can easily run the operational software of multiple robots at once. This allows changes to be rapidly tested before wide-scale deployment into the robot hardware.

9.4. Safety, E-Stop, Freeze & Lost Link

A dedicated E-Stop Terminal and a separate E-Stop hardware within the UGV vetronics are developed to satisfy the E-Stop, Freeze and safety requirements. The E-stop, Freeze and Run commands are sent to the UGVs through the E-stop Terminal and each individual robots status is also fed back to the operator on the associated LED displays as shown in **Error! Reference source not found.**.

The terminal is designed as a standalone, portable and rugged unit capable of controlling up to 6 vehicles. The front panel is used to execute Run/Freeze and E-stop commands. LEDs associated with the switch positions display the actual status received from the UGVs. The communication status between the terminal and the UGVs is also observable through the blinking Power LED display.

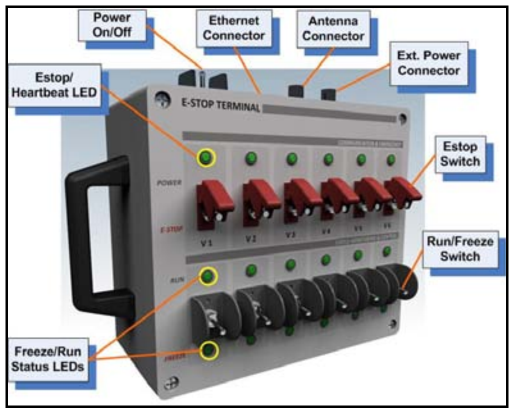


Figure 26: E-Stop Terminal

Communication between the E-stop Terminal and the UGVs are handled with radio modems operating at 900MHz ISM Band. The E-stop Terminal and the UGVs build a communication network after power on. This board supports advanced features such as encrypted messaging and fail-safe relays. The terminal periodically sends commands to the UGVs and receives replies from them. This also acts as continuous heartbeat control. In case of a 10 second communications blackout, or when the terminal user activates an e-stop switch, the vehicle electronics shuts down the power and applies the brakes immediately.

The UGVs can also be frozen using the terminal.

The vehicle is equipped with active-on wheel brakes in order to keep it immobile in conditions including ramps up to 15 degrees. The vetronics control the brakes and apply them only when necessary to save battery.

9.5. Communication Failure Recovery

Each robot continuously checks its communication with the GCS. If the robot drops out of the main communications network, HLC tries to recover by sending the robot to the last known good communications point where it was able to communicate with the GCS.

Low frequency communication between robots and GCS can be used as a backup communication system where WiMax communication fails.

9.6. Spectrum Plan & Usage

The spectrum plan for communications and their usage area are given in Table 1.

Table 1: Spectrum allocation plan

Unit	Spectrum alloc.	Usage
Wimax NAU	5.8 GHz	UGV-GCS high-bandwidth comm. DGPS correction signal broadcasting
Radio Modem	915-928MHz	E-stop/Freeze and emergency backup comm.

9.7. Test Plan

The initial testing of the hardware components was done using an incremental method. Building on a COTS robot platform, each new sensor, electronic equipment or payload component was added incrementally and thoroughly tested. The main advantages of this approach were to ensure that the devices were compatible with each other and quickly identify any mismatched components. The initial testing also included the integration of software components. When real world sensor data, with its inherent noise and occasional unreliability (sometimes triggered artificially to test system robustness,) destabilized software that otherwise worked properly, using robots that were partway complete allowed the development teams to focus only on the relevant parts instead of dispersing their attention. This incremental approach to building and testing both the hardware and software components in accordance with our system architecture allowed the corresponding interfaces to be quickly validated and precautions with required modifications to be handled immediately before proceeding to the next stage.

During the initial testing stage, multiple-robot tests were also conducted, concerning, in particular, the communications architecture. As the communications system doesn't absolutely require the robots to be fully operational, tests were conducted where one or two mobile robots would carry their communications nodes while the rest of the nodes would be tested on tabletops with artificial network loads (such as streaming video) attached.

Once the robots were built in their final configuration, individual robot tests were handled. A separate team conducted individual performance tests with one of the robots, to achieve a reference performance measure of what can be achieved and the rest of the robots were tested accordingly. These benchmarks were constantly re-evaluated as system-wide problems were resolved; all robots were thus periodically run through standardized tests.

Problems encountered during the single-unit tests were also categorized to build a knowledge base later, including the identification of problems that would cause the robot to cease doing any meaningful contribution to the challenge effort. These major problems are mostly handled by having the system watch for them and stop operating when detected; in some cases where it can't be done automatically; operators were trained to look for the symptoms.

The final, multi-robot tests were separated in two tiers. The first tier involved system tests, evaluating multi-robot mapping accuracy, robot cooperation and mission planning issues. These tests were conducted without regard to the actual competition rules and aimed to improve specific aspects of the mission planning process rather than the overall system. The second tier was aimed to test the operational team's capability and the HMI interfaces. A dedicated test team prepared different test scenarios involving large areas, including multiple OOI. During the tests, the operators were not aware of the number or the placement of the OOI, and were only given the challenge area boundaries at the last minute. The tests were therefore representative of the operations team coming into contact with an area previously unknown to them as well as the robots. These tests have shown that over several runs spanning different layouts and configurations, most static OOI were detected (either autonomously or through human observation) well before activation zones. Of the detected objects, all of them were located within a 1m radius of their pre-surveyed location (with about 75% of them located autonomously and 25% corrected manually.).

10. Summary

ISR operations present challenges to the dismounted protection forces, where MAGIC 2010 aims to improve the mission effectiveness using a cooperative multi-UVS team with high autonomous skills.

Team Cappadocia, as a group of highly qualified members with experience in military products development and unmanned system applications, has created the solution to the MAGIC 2010 challenge provided in this document. The solution takes advantage of both the field proven, reliable and robust expertise and unites it with novel method and technologies.

The key points of Team Cappadocia's solution are;

- Standardized UGV components with JAUS compliant modules,
- Fully automated OOI detection and tracking,
- Intelligent Localization utilizing decision making,
- A novel technique which enables automatic UAV image processing,
- Advanced dynamic mission planning with automatic and optimized route planning,
- Reliable communications architecture,
- Highly automated mission implementation with reduced human interaction,
- Run-time configurable smart HMI displays.

11. Acknowledgements

The authors would like to express their heartfelt thanks to everyone who has made the realization of this project possible. The entire ASELSAN family was instrumental in this process, though special recognition must go to the people who put endless hours working on the project: Aykut Başaran, Bülent Bilgin, Bahar Cavcar, Emin Çetinbaş, Soner Çınar, Kurtuluş Ersoy, İlker Gürel, Hasan Karakaş, Hasan Ozan Kaya, Atila Kaymak, Kenan Kürekli, Neslihan Özlü, Diler Şimşek, Soner Yiğit, Tolga Yurdakul, everyone at the Unmanned Systems Department and our project manager and team leader, Mr. Faruk Mengüç, who sponsored the project from its inception to the completion.

The group also thanks Prof. Dr. H. Levent Akın, Prof. Dr., Oğuz Tosun, Can Kavaklıoğlu, Tekin Meriçli, Serhan Daniş, Barış Gökçe, Ergin Özkucur of the AI Lab of the Boğaziçi University, Asst. Prof. Dr. Afşar Saranlı of the METU Robotics Laboratory and Dr. Uluç Saranlı of the Bilkent University.

12. References

[1] K Passino,. Ü. Özgüner, *Modeling and Analysis of Hybrid Systems: Examples*, IEEE International Symposium on Intelligent Control, pp. 251-256, 1991.

[2] A. Kurt, Ü. Özgüner, *Hybrid State System Development for Autonomous Vehicle Control in Urban Scenarios*, 17th World Congress The International Federation of Automatic Control, pp. 9540-9545, Seoul, Korea, 2008.

[3] K. A. Redmill, U. Ozguner, S. Biddlestone, A. Hsieh, and J. Martin, *Ohio State University Experiences at the DARPA Challenges*, SAE International Journal of Commercial Vehicles, vol. 1 no. 1, pp. 527-533, 2009.

[4] J.-S. Gutmann, C. Schlegel. AMOS: *Comparison of Scan Matching Approaches for Self-localization in Indoor Environments*, Euromicro Workshop Advanced Mobile Robots, pages 61–67, 1996.

[5] M. B. Dillencourt, H. Samet, M. Tamminen, *A General Approach to Connected-component Labeling for Arbitrary Image Representations*, J. ACM, 1992.

[6] J.E. Bresenham, *Algorithm for Computer Control of a Digital Plotter*, IBM Systems Journal, Vol. 4, No. 1, pp 25-30, 1965

[7] C.K. Chui, G. Chen, *Kalman Filtering with Real-Time Applications*, Springer, 4th Ed., 2009.

[8] G. Welch, G. Bishop, *An Introduction to the Kalman Filter*, Technical Report, University of North Carolina, 2001.

[9] İ. Yakın, *Line Segment Based Range Scan Matching without Pose Information for Indoor Environments*, Msc Thesis, Bilkent Uni., 2008.

[10] Ö. Aytekin, İ. Ulusoy, *Object Oriented Adaptive Mean Shift Segmentation of VHR Images Based on Mathematical Morphology*, IEEE Trans. Geosci. Remote Sensing., 2010.

[11] Ö. Aytekin, İ. Ulusoy, U. Halıcı, Segmentation of High Resolution Satellite Imagery Based on Mean Shift Algorithm and Morphological Operations, SPIE Europe Remote Sensing, 2009.

[12], Y. Freund, R. E. Schapire, A Short Introduction to Boosting, 16th International Joint Conference on Artificial Intelligence, 1401–1406, 1999.

[13] M. Çetin, U. Halıcı, Ö. Aytekin, *Building Detection in Satellite Images by Textural Features and Adaboost*, IAPR ICPR, 2010.

[14] U. Halıcı, U. Zongur, Ö. Aytekin, *I. Ulusoy*, *Airport Runway Detection in Satellite Images by Adaboost Learning*, SPIE Europe Remote Sensing, 2009.

[15] C. M. Bishop, *Pattern Recognition and Machine Learning*, 2006.

[16], F. Maes, , A. Collignon, D., Marchal, G. Vandermeulen, P. Suetens, , *Multimodality Image Registration by Maximization of Mutual Information*, IEEE Transactions on Medical Imaging, 16 (2), pp. 187-198, 1997.

[17] K. M. Wurm, C. Stachniss, W. Burgard, *Coordinated Multi-robot Exploration Using a Segmentation of the Environment*, IEEE/RSJ International Conference on Intelligent Robots and Systems, pp. 1160-1165, 2008.

[18] W. Burgard, D. Fox, M. Moors, R. Simmons, S. Thrun, *Collaborative Multi-Robot Exploration*, IEEE International Conference on Robotics and Automation, vol.1, pp. 476–481, 2000.

[19] *3D Multiple Robot Simulator with Dynamics*, acquired at 20.07.2010, on http://playerstage.sourceforge.net/index.php?src=gazebo

1 Cappadocia, a world natural heritage located in the heart of Anatolia meaning “world of beautiful horses” resembles the multi-robot teams as Troy horses discovering and neutralizing threats encountered in the operational area.