ARSI Deliverable D26.5

Prototype for Sewer Inspection

In phase I of the ARSI project, a detailed design was proposed for an aerial robot able to perform visual inspections of sewer tunnels [1]. In this document, we describe the changes and developments carried out in Phase II to develop this design into a functional prototype.

1. References

[1] ARSI D2.1 — Operation requirements and system design
[2] ARSI D2.2 — Towards automatic sewer inspection. First results
[3] Brian Gerkey & Kurt Konolige - Planning and Control in Unstructured Terrain

2. Localization

One of the key challenges in this project is that the ARSI drone must be able to estimate its position in real-time, without any direct external source such as GPS. Like most robotic systems, our system achieves this by using information from a range of embedded sensors, and combines it using a probabilistic filter to produce an estimation of the position and velocity of the drone.

Table 1 below lists the various sensors embedded on the ARSI platform. Note that the gyroscope, accelerometer, magnetometer and barometer are all built into the Pixhawk Autopilot hardware unit.

Sensor type	***Model***	***Description***
RGBD camera	Orbbec Astra	Provides color and depth images of the environment. The depth image can be converted directly into a point cloud, with an 8m detection range. This sensor is primarily used for Visual Odometry.
Infrared ranger	TeraRanger One	This infrared Time-of-Flight sensor is mounted underneath the ARSI platform and provides direct range measurements to the ground.
3-axis gyroscope	ST Micro L3GD20H	Provides real-time angular rates for the 3 axis (pitch, roll and yaw)
3D accelerometer	ST Micro LSM303D	Provides linear 3D accelerations.
3D magnetometer	ST Micro LSM303D	Provides 3D magnetic fields
Barometer	Amsys MS5611	High precision pressure sensor

Table 1: Embedded sensors on the ARSI platform

The Pixhawk Autopilot runs two highly configurable state estimator modules:

- **Attitude estimator**: quaternion based filter to estimate the vehicle attitude using information from the accelerometers, magnetometers, and gyroscope.
- **Position Estimator**: filter estimating 3D position and velocity states using the attitude, accelerations, as well as range data and visual odometry.

Both estimators allow adjusting the weight for each secondary source to reflect the level of confidence we have in their measurements. For instance, early tests showed that the magnetometer data often did not match the real trajectory of the ARSI platform in the sewer tunnels (we suspect due to the presence of metallic structures in tunnels, some visible and some embedded into the sewer walls). We therefore gave the magnetometer a relatively low weight in the state estimation configuration, in order to minimize its effect on the overall solution.

Figure 1 presents the general architecture of the localization system in ARSI described in more detail in the following sections. Note that the attitude and pose estimators are part of the Pixhawk firmware, while the rest of the software modules all run on a PC onboard the ARSI platform.

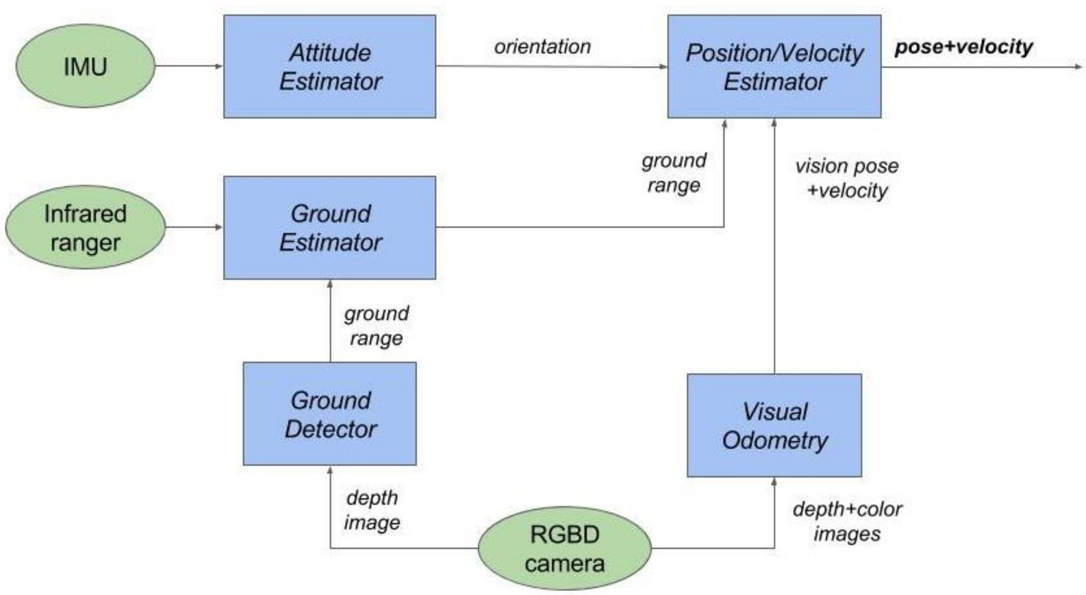

Figure 1: General architecture of the localization system in ARSI

Visual Odometry

Early in phase II of the ARSI project, our team decided to use a RGBD camera as the primary sensor to estimate the velocity and position of the aerial robot during sewer inspections.

RGBD cameras use a combination of optical and infrared sensors to provide both visual and depth information of a scene, and are well adapted to operation in indoor environments. Visual Odometry (VO) algorithms such as RTAB-Map can then combine this information to calculate the movement of the platform between two consecutive data frames, providing an estimation of the instant velocity. Velocity estimates are then integrated over time to extrapolate the position and trajectory of the vehicle.

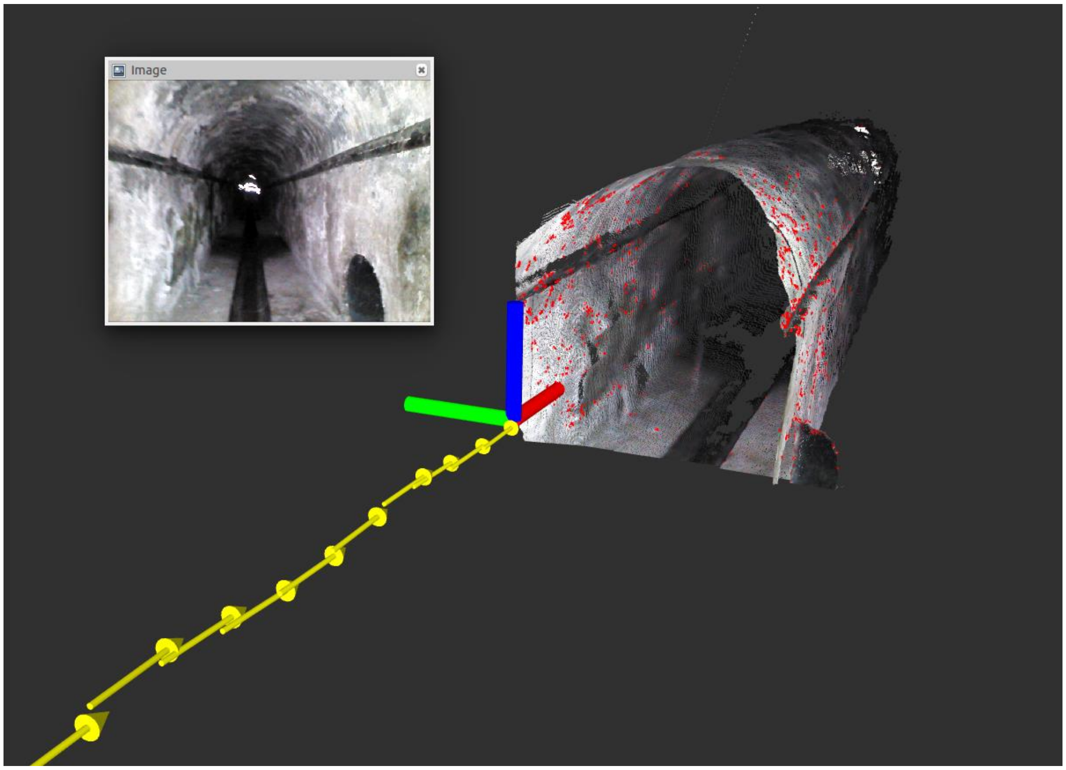

Figure 2: RtabMap running on RGBD data from Mercat del Born. Numerous visual features (in red) are detected on the sewer walls, allowing for the calculation of the vehicle trajectory (in yellow)

Using various RGBD datasets, we confirmed that RTAB-Map is able to detect visual features on the sewer walls, and to estimate the robot velocity with good accuracy (see Figure 2). Velocity estimation is critical to operate an aerial robot in narrow environments like sewer tunnels, being the entry point for the control algorithms running onboard the Pixhawk autopilot module to adjust its attitude and thrust during flight. We validated this approach using RGBD datasets, from our test environment and from the sewers at Mercat del Born in Barcelona (see **¡Error! No se encuentra el origen de la referencia.**).

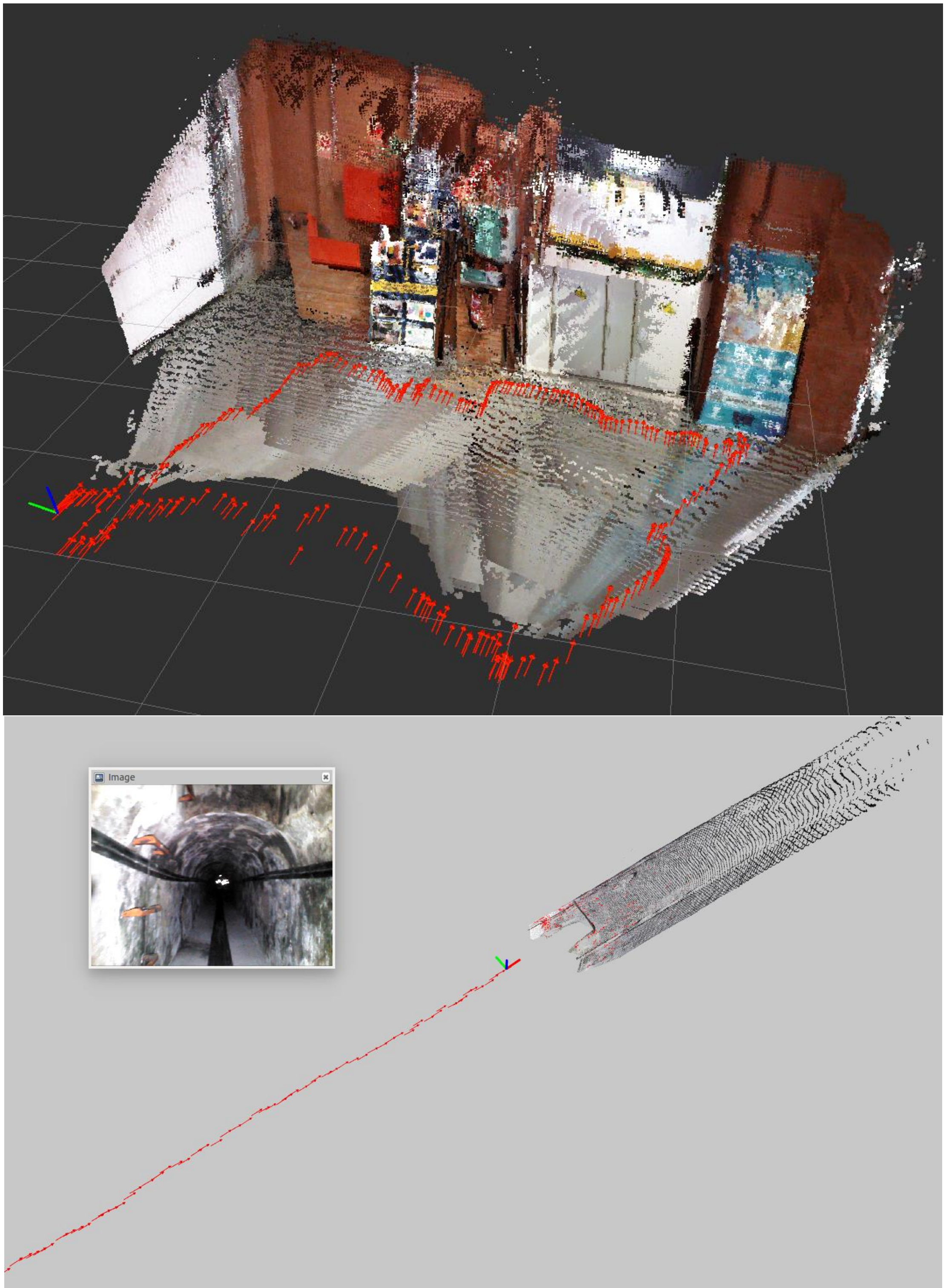

Illustration 1: 3D models and vehicle trajectory (in red) calculated using RtabMap in our flying arena (top) and in a long linear tunnel in Mercat del Born (bottom)

Global localization

It is important to note that position estimation, being the result of an integration process, will inevitably accumulate error ("drift") over time. Since we use a dynamic control approach (see Navigation section), this drift does not directly affect flight along the sewer tunnels. It does however affect high-level commands which are associated with an absolute position, as the accumulated error is likely to cause these commands to be triggered in the wrong location. Landings are particularly problematic for us since they must occur very close to an entry point into the sewer, so that operators can access the platform to replace batteries and deploy the WiFi router (see [1] Concept of Operation).

Navigation drift is a common problem in autonomous robotics, and most systems correct this using external localization systems like GPS, georeferenced landmarks, or by detecting when the platform observes the same area several times ("loop closure"), so the drift can be subtracted from the solution. GPS is not available in the sewers, and loop closure is unlikely since inspection missions will be planned to maximize coverage and flight time. Our chosen approach to bound the drift is to extract known features in the sewers such as entry points (manholes), intersections, ladders, etc. using modern techniques such as deep learning. These features are matched against the GIS data to extract their absolute location (e.g. GPS or UTM coordinates) which will then be passed to the fusion algorithm on the Pixhawk and integrated into the position solution as global "fixes", in the same way that GPS is used to bound navigation drift in outdoor solutions.

However, given the amount of work already required in phase II for the September 2017 evaluation, this feature will not be implemented until phase III. For the September evaluation landings will be triggered remotely by the operator using video feedback over Wi-Fi.

3. Altitude estimation

Altitude control is a challenging problem in a closed environment like sewer tunnels, where standard sensors such as barometers are not accurate enough to allow for the precise control required to avoid colliding with the environment.

Range sensor

For this project, we replaced the barometer by a range sensor mounted underneath the platform and looking directly down. We chose a low-cost, lightweight infrared Time-Of-Flight (TOF) sensor, the TeraRanger One, which was developed by CERN and used in many robotic applications (see Figure 3).

Figure 3: TeraRanger One range sensor

The TeraRanger produces high-frequency measurements (up to 1000Hz) in direct line of sight to the ground, with a precision estimated to ±4cm. However we observed in our tests at Mercat del Born that the measurements were very noisy, due to the presence of water and the proximity of the drone to the ground. This high-frequency noise in turn introduces significant instability in the altitude control of the ARSI platform. Figure 4 illustrates this problem, as well as the result of the low-pass filter we applied to improve the quality of the range measurements.

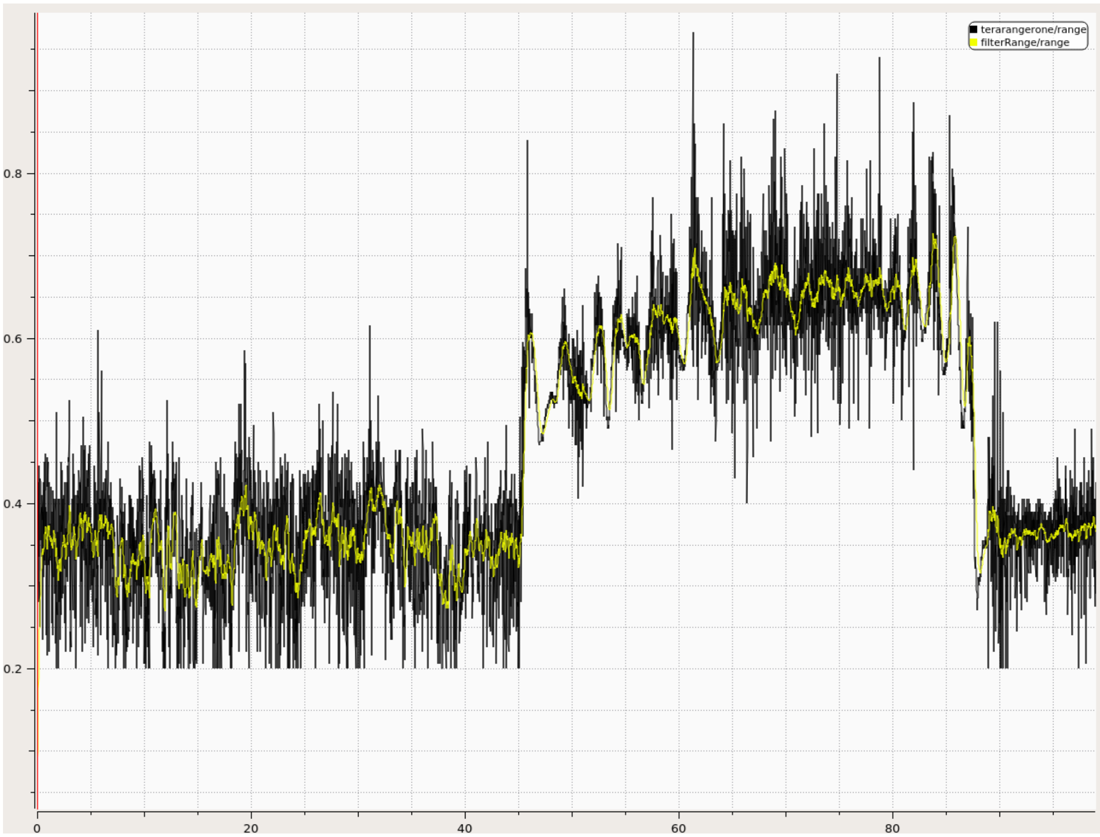

Figure 4: TeraRanger raw data (in black), TeraRanger data Filtered (in yellow)

Ground detection

Most sewer tunnels feature a central canal or drain, where water flows towards extraction points. This means that the ground beneath our platform is almost always uneven, and that range measurements from the TeraRanger sensor will fluctuate during the flight, from the bottom of the drains to the surface area next to them. This affects altitude control, increasing flight instability and risks of collision.

To solve this problem, we processed the 3D point cloud generated from the depth image produced by the RGBD camera, to detect the ground and walls of the sewers during flight using segmentation techniques based on normal estimation (see Figure 5). Because these detections are supported by hundreds of data points, as opposed to a punctual measurement for the TeraRanger, they are more stable and can account for irregularities.

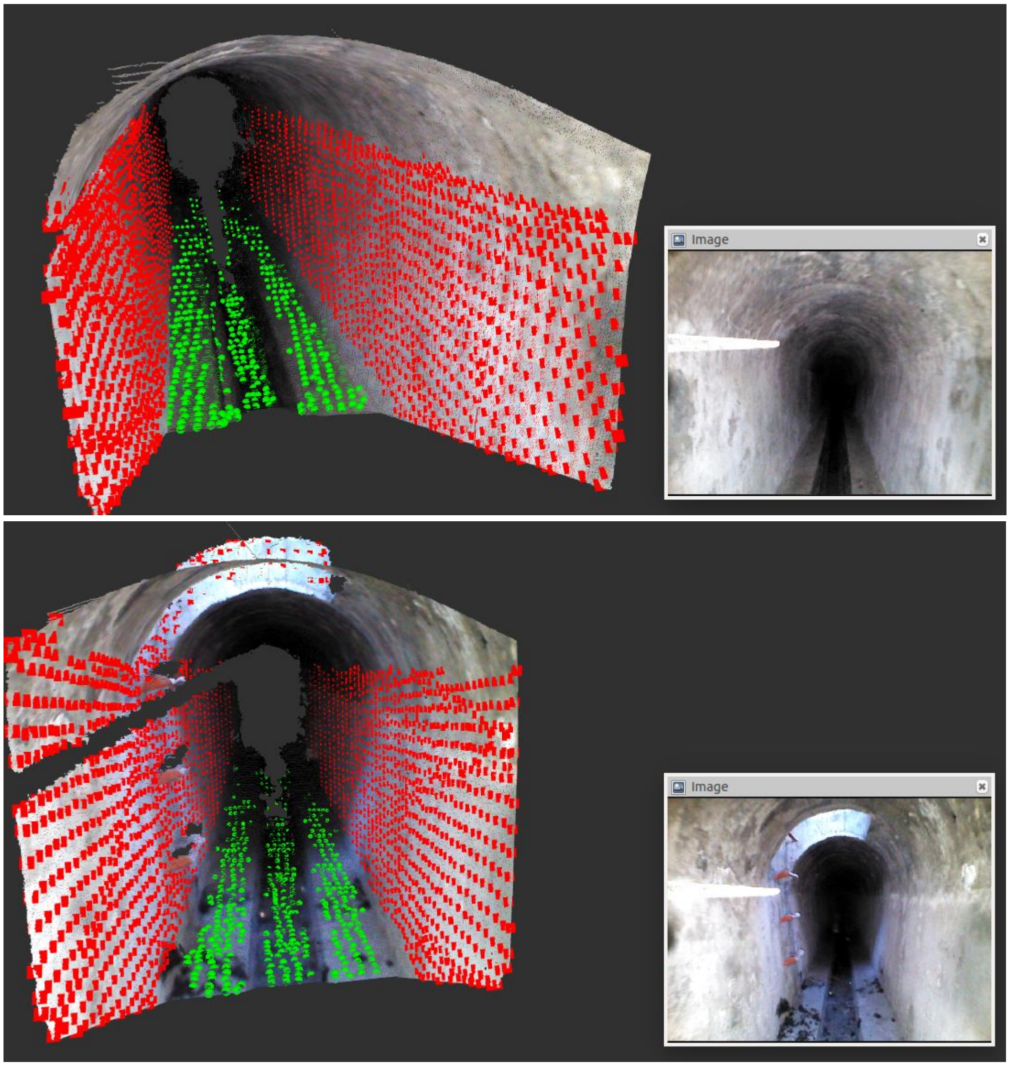

Figure 5: Detection of the sewer walls (in red) and ground (in green)
using RGBD data from Mercat del Born

Since the offsets from the RGBD camera on the ARSI platform are known, we can geometrically derive the range from the drone to the ground detection. We use this range estimation to complement the measurements from the TeraRanger One in order to produce a more robust estimate of the ground range. We developed a software module to receive range measurements both from the TeraRanger and the ground detection module, producing a weighted average that is then passed on to the general sensor fusion algorithm onboard the Pixhawk Autopilot.

Figure 6 illustrates how ground range measurements from the TeraRanger and the RGBD detections both contribute to the final altitude estimation produced by the general pose estimation algorithm running on the Pixhawk. Note that the differences in range are due to the physical offsets between sensors, which are taken into account in the pose estimation. Early tests in Mercat del Born have showed that this additional sensor fusion step improved the general altitude control of the platform in the sewers.

Figure 6: TeraRanger Filtered (in black), RGBD detection (in blue) and estimated position (in red).

4. Flight control

Flight control has been an ongoing task throughout phase II. Flight in narrow spaces such as sewer tunnel is very complex due in particular to the turbulences generated by the air flow from the motors of our quadrotor in a closed space. We have done extensive tests in our flight area, experimenting with various control modes in order to find the best configuration for a low-level flight controller, with the goal of obtaining a reactive platform able to stay at safe distance of any obstacles.

This work will be continued in the ongoing flight tests in the sewers at Mercat del Born.

5. Navigation

In the context of a robotic system, navigation is the task of issuing low-level commands to execute a high-level task, such as a sewer inspection by following a predefined route based on GIS data while avoiding any possible collision with the environment. The general approach for navigation in robotics is to combine a **global planner** with a **local planner**.

Global planners calculate the high-level route the robot should follow, typically using existing information such as maps or GIS data. In the ARSI project, this information is provided by the operators who are expected to have access to detailed GIS information of the sewer networks. Using a user interface developed especially, operators are able to visualize GIS data and plan mission routes for the ARSI robot to follow (see figure below, with inspection route in red). These routes, or “global plan”, are then sent directly to the vehicle over Wi-Fi so they can be executed automatically and monitored by the inspection teams.

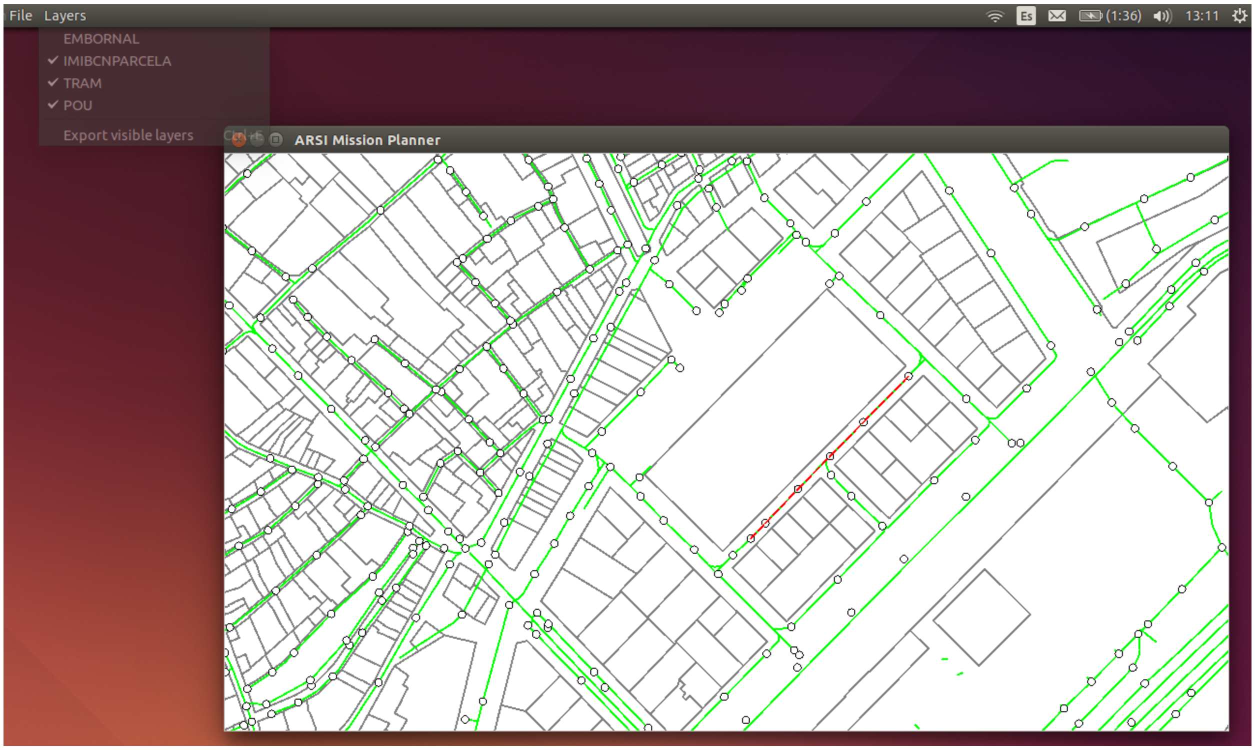

Illustration 2: Prototype for ARSI mission planner using GIS data provided by BCASA

Local planners are designed to generate low-level commands to control a robotic platform along the global path, while reacting in real-time to changes in the environment such as obstacles. In this project we developed a local planner based on the Dynamic Window Approach [3].

Cost map

First, an occupancy grid or “cost map” is generated, to represent the space immediately around the drone. The cost map stores data from various onboard sensors: wall detections from RGBD data, laser data, or range measurements from distance sensors. Obstacles in the cost map are then “inflated”, in order to define the minimum distance to any obstacle that the robot should maintain at all times during flight (see Figure 7).

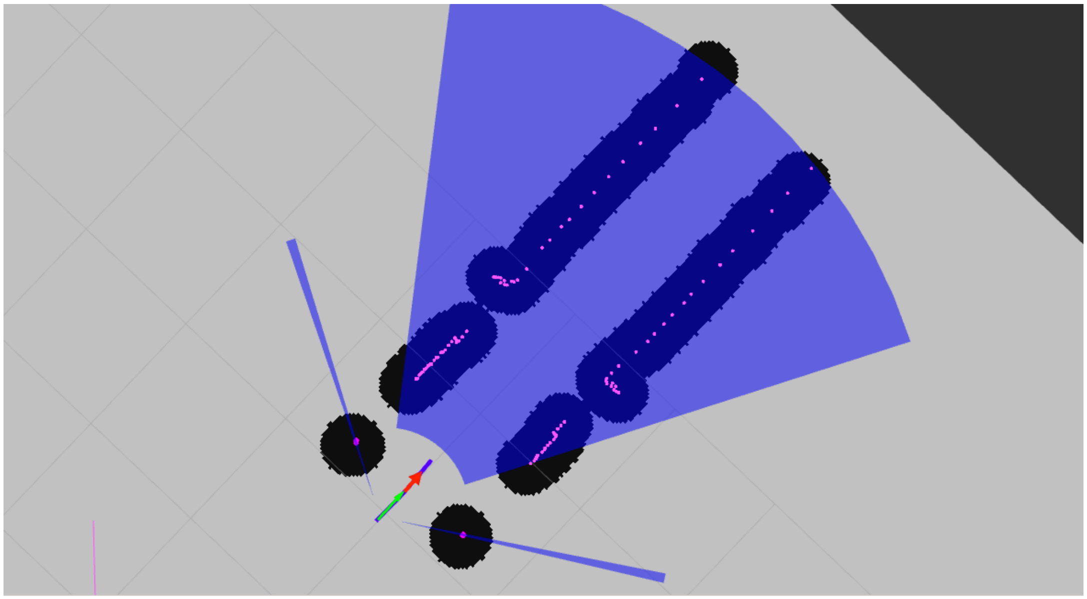

Figure 7: Cost map (in black) generated by inflating point cloud data (in purple). Here we simulate data from a front facing RGBD camera and two lateral distance sensors

During our tests in Mercat del Born, we found that the air flow generated by the drone lifts a lot of dust (as well as spider webs) from the sewer walls and ground. This dust becomes visible by the onboard sensors, in particular the laser data, and introduces numerous “false obstacles” in the cost map (see Figure 8).

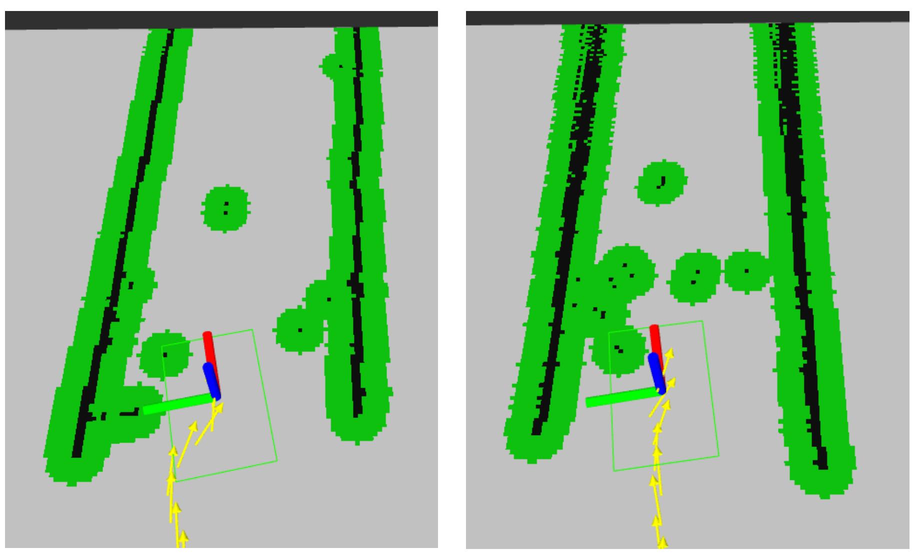

Figure 8: False positives caused by dust in the sewers in Mercat del Born.

These “fake obstacles” dramatically affect the vehicle control as the local planner tries to navigate between them, constantly changing direction are the dust particles appear and disappear in the sensor data. After these tests we introduced a filter for the sensor data in the cost map to remove data points that are inconsistent with their surroundings, based on how many neighbors they have within a certain radius. Using this filter we were able to remove most of the false positives, as shown in.

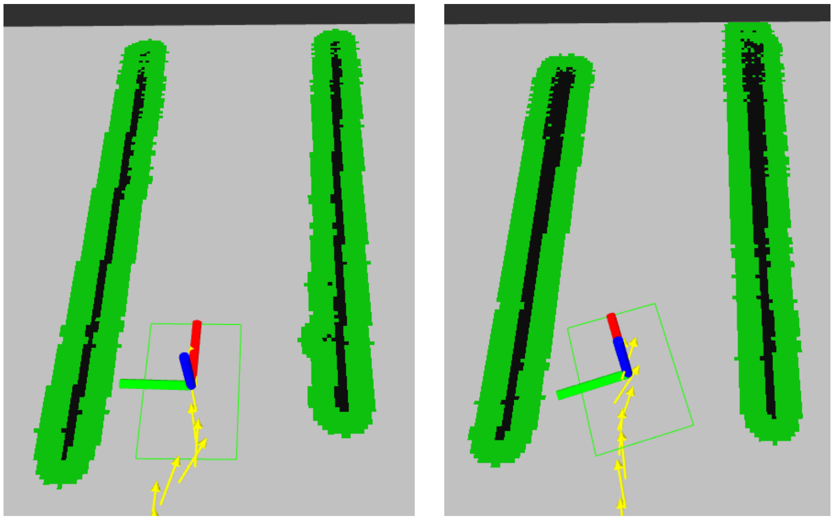

Figure 9: Same cost map as Figure 8, after filtering of dust particles

Local planner

Using the cost map and the known dimensions ("footprint") of the vehicle, the ARSI local planner evaluates and ranks all possible low-level movements within pre-defined degrees of freedom (typically yaw and velocity). A weighted multivariate score is calculated for each possible movement, based on parameters such as distance to the global goal, orthogonal distance to the global path, and shortest distance to any obstacle in the cost map.

By minimizing this score, we can select the optimal velocity command enabling the system to execute the global path while taking into account all information provided by the embedded sensors. This process is repeated at a high frequency (20-30 Hz), allowing the ARSI platform to navigate safely in the sewer tunnels, following missions planned by operators.

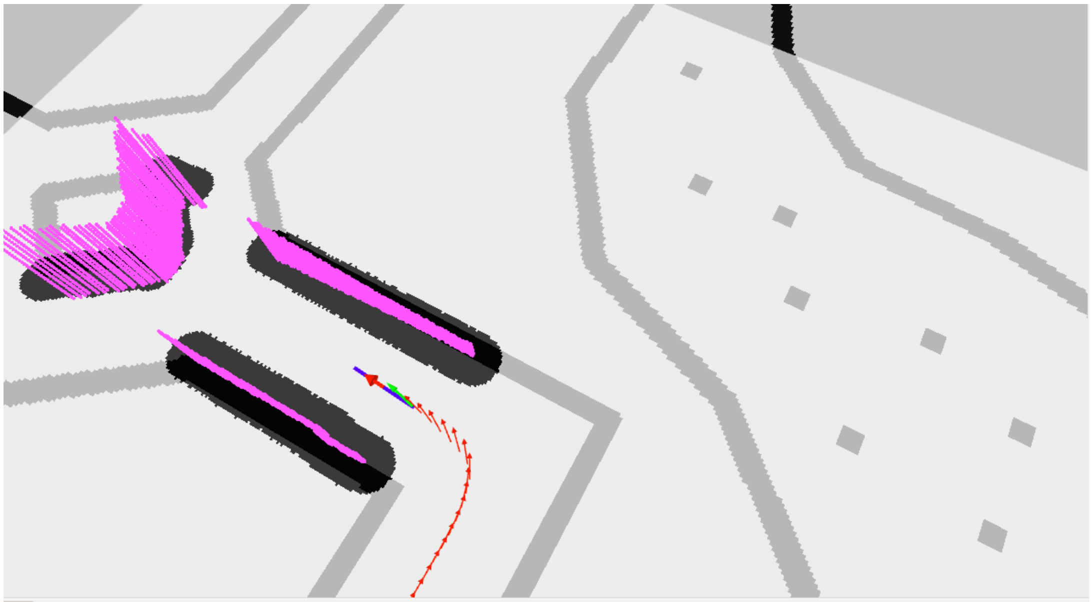

Figure 10: ARSI local planner in a simulated sewer environment

This approach was first developed and tested in simulation (see Figure 10); then various configurations were created using cardboard boxes in our indoor test environment at Eurecat (see Figure 11):

- Straight lines in tunnels with widths varying from 1m20 down to 80cm;
- Section changes;
- Turns and intersections;
- Uneven ground for altitude control;
- Flight underneath obstacles with 60cm clearance (as in Mercat del Born).

A video summarizing these tests can be downloaded here.

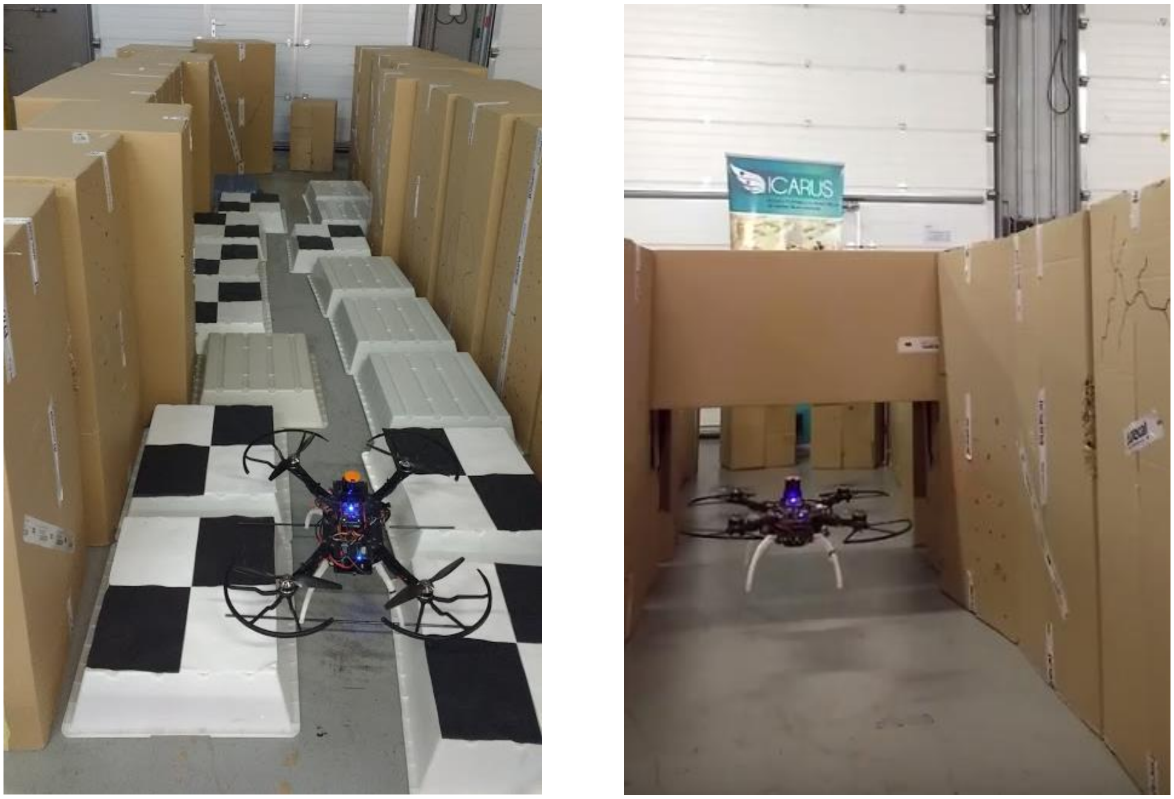

Figure 11: Example setups for lab tests:
(left) altitude control tests with simulated uneven ground and walls
(right) flight underneath obstacle with 60cm clearance like in Mercat del Born

In all these tests, the drone was flying autonomously; missions were planned in advance and executed over Wi-Fi from our control interface (see Figure 12). The control interface allows visualizing RGB images from the drone during flights, as well as complete information about the onboard navigation (odometry, cost map, trajectory, setpoints, velocity requests, etc.) in a 3D environment. The interface also contains a summary panel with critical information such as battery charge, velocity, distance travelled or alarms.

Finally, the interface contains controls to remotely arm the drone, start and stop an autonomous mission. In the future it will also allow pausing to observe a defect, or reversing the direction of travel. An additional panel is also being developed for online capture of points of interest during an inspection, which are then stored in the mission logs onboard the vehicle for further analysis.

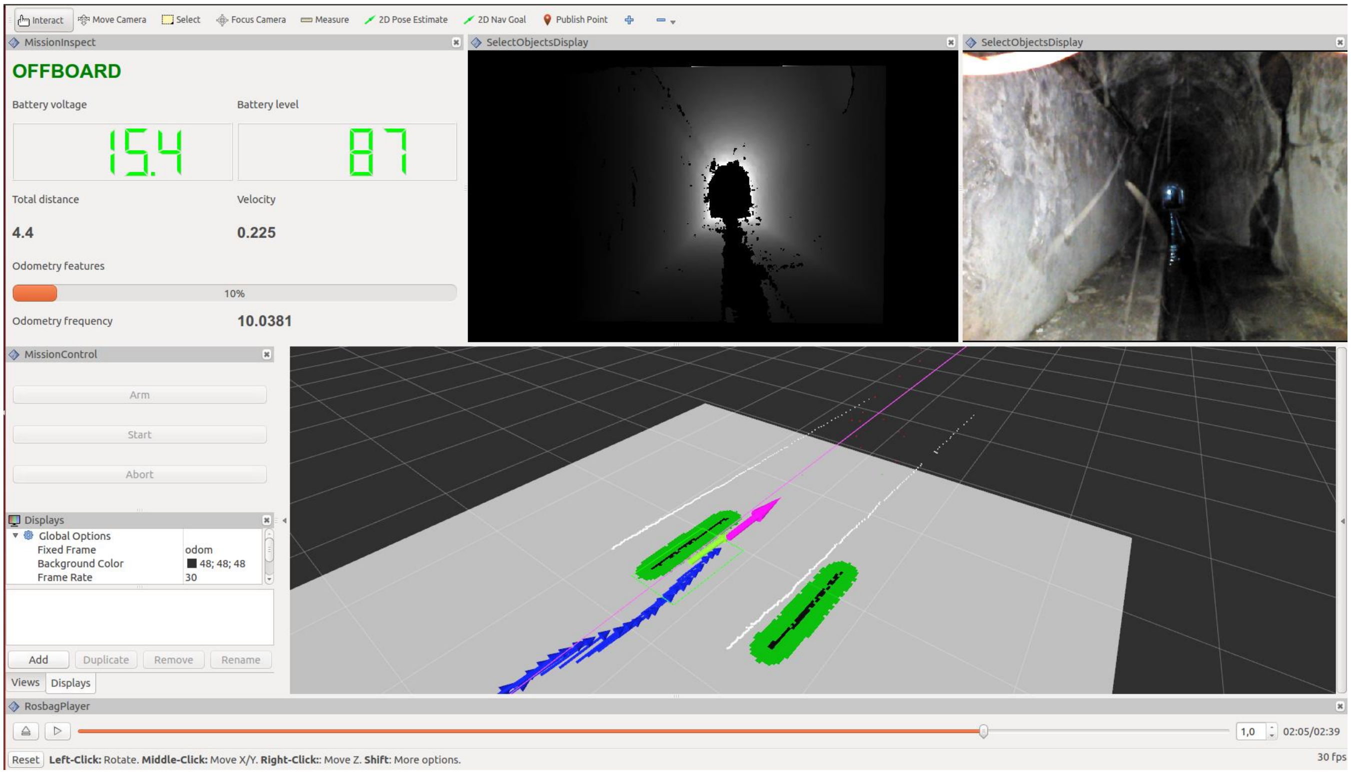

Figure 12: ARSI control interface used in Mercat del Born

6. Communications

Communications tests took place in July in Mercat del Born, using signal boosters and antennas to measure how Wi-Fi signal propagates in the various tunnels, and how much signal is lost at the intersections.

The objective of these tests was to verify the propagation model and mission planning approach presented in phase I. The test results will be described in deliverable D26.4 along with a detailed inspection plan for Mercat del Born using the ARSI system.