

STIX DataSet README

July 24, 2019

1 Equipment

The STIX datasets were collected with a refurbished iRobot Packbot Explorer, which has been designated "GVRBot". This robot has been augmented with many sensors which are representative of an entry to the DARPA SubT challenge. The sensor modalities chosen represent a superset of typical configurations; this allows a team to experiment with various combinations to evaluate their applicability to the SubT challenge on their specific software.

Images labeled with component callouts can be seen in figures 1 and 1. As shown in these images, the robot is equipped with an Ouster OS1-64 (3D LiDAR) , a FLIR Tau2 thermal IR camera, a Carnegie Robotics Multisense SL stereo camera + illuminators + spinning LiDAR, and a Microstrain GX5-25 IMU. The robot was also equipped with a Point Grey Chameleon which was a spare device and not used in this data collection. Data was saved onto an SSD in the computing payload.

2 Software

The data sets were collected using ROS drivers for sensor components where available. Imagery was collected in compressed or compressedDepth format to reduce file sizes. These can be reconstructed to their raw form through the use of image_transport "republish" ROS nodes, or by using image_transport when subscribing to the topics.

2.1 Ouster OS1-64

Uses https://github.com/ouster-lidar/ouster_example driver. Has been modified to tag output with current system time (`ros::Time::now()`) instead of just using device timestamp, which starts at zero. The timestamp on the device was not set due to lack of supporting hardware on our part, which will be rectified in future collections. To reduce jitter, the offset between system time and device time is continuously estimated, and composed with the device time to get something closer which will only be offset by an unknown delay parameter.

Figure 1: The robot used to collect these data sets, seen from the front

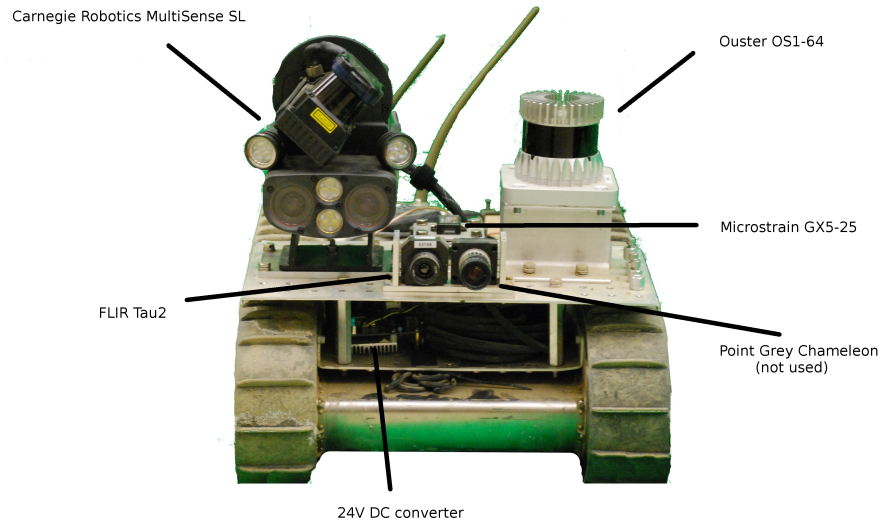
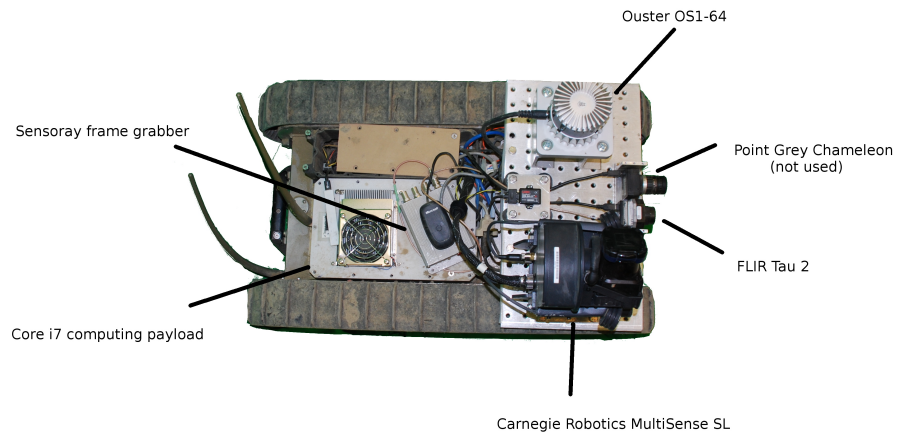


Figure 2: The robot used to collect these data sets, as seen from the top



We have recorded the ouster packets directly to support re-generating the clouds, perhaps with better estimates of time delay, if desired. In addition, the OS1 device generates its own internal IMU data, which would have the correct timestamps for the LiDAR points. We have recorded this but not used it yet. Finally, the point clouds are also recorded at 10 Hz.

2.2 Multisense SL

Uses full driver stack provided by Carnegie Robotics. Device was calibrated at the factory. We attempted to capture all relevant topics and calibration data.

2.3 FLIR Tau2

We have provided the thermal IR data from this sensor, on the topic `cv_camera/image_raw/compressed`.

Intrinsic calibration of this sensor was not performed, so the `camera_info` message should not be used as is. An interested user could attempt to calibrate by looking at common features between the thermal IR image and the Multisense SL, such as lights which show up in both.

2.4 Microstrain IMU

Raw microstrain imu data is recorded. This is also incorporated with the platform's odometry (which is also recorded separately) into a `gvrbot/odom` → `gvrbot/base` (`base_link`?) frame. This can be stripped out if desired through the use of the `tf_hijacker` node which is provided in the bitbucket site at https://bitbucket.org/subtchallenge/stix_ws. This project also contains helpful launch files which can be used to run these bag files.

2.5 Other considerations

We were running our own mapping system while collecting this data, which results in the TF tree containing a `map`→`gvrbot/odom` frame. When evaluating your own mapping system, this frame will need to be stripped through the use of the `tf_hijacker` node, which is provided in the bitbucket site.

The FLIR data cuts out near the end of the long loop bag file.

Extrinsic calibration of sensor positions is quite rough and might be insufficient to generate really accurate maps. Sufficiently motivated parties could use the `tf_hijacker` node to remove inaccurate transforms which could then be re-inserted through the use of a `tf/static_transform_publisher`.

2.6 Run notes

subt_edgar_hires_2019-04-11-13-31-25.bag

- Description: Main loop plus drilling museum. Total length 26 minutes

- Problems:

1. FLIR cuts out at 959 seconds out of 1599 seconds of total run. This means that we didn't see the last Rescue Randy near the ARMY entrance. The robot was in the paved concrete branch off of the ARMY tunnel. FLIR also would have been useful to see at least one more cell phone at the ARMY tunnel
2. Got a good look at the ARMY tunnel entrance gate as well as the initial MIAMI tunnel entrance, but the bag file stops short of re-observing the MIAMI tunnel and getting back in to close the loop. A team was setting up for their run by the time we got back to the MIAMI staging area.

Smoke tests

- *subt_edgar_hires_2019-04-12-15-46-54.bag*
 - Has FLIR, starts just outside the smoke and makes an approach to the "survivor".
- *subt_edgar_hires_2019-04-12-15-52-44.bag*
 - No FLIR data in this run, but still has other sensors. Sees survivor a few more times

Dust tests:

- Three bag files here with good FLIR and all sensors working correctly. These bag files are taken at the steep incline at the back on the MIAMI tunnel. The robot follows closely behind a person who is kicking up a lot of dust into the air.