Indoor robotics summary

Omkar Joglekar

December 2022

1 Altitude Statistics

The "indoor status" message has an "altitude" field, but this field probably contains altitude data from sea level and within an indoor space, it does not have enough granularity to detect small changes. It is a constant 1000.0. To get better altitude measurements indoors, the SONAR sensor is used. The data returned from this sensor is stored in the message "sonar_height". The message is of type "range". So it contains 3 main fields - "min_range", "max_range" and "range", where the "range field contains the actual measurement value. Before using this data, each individual measurement should be tested for 3 things - $min\_range \leq range \leq max\_range$, $range \neq \infty$ and $range \neq -\infty$. We must throw away the points that don't satisfy the first condition. The points violating the third condition indicate object collision. The three conditions were checked during parsing and none of the points violated the second and third conditions. The ones that violated the first condition were excluded from the main data analysis.

The count, mean, standard deviation, 25th/75th percentile, median (50th percentile), min, max, skew, kurtosis are recorded in the CSV file titled "altitude_summary.csv". The box plot is shown in Fig. 1

In all the flights, the average altitude is approximately $1.5[m]$. The average human height in Israel is around $1.6 - 1.7[m]$. The height at which the drone is flying could lead to collision with a human, in an indoor setting. Having information about this can help us design a safer interaction between Tando and people in the workplace. The plots in Fig. 2 summarize the altitude vs time in the airborne state, with the average human height marked on them for reference.

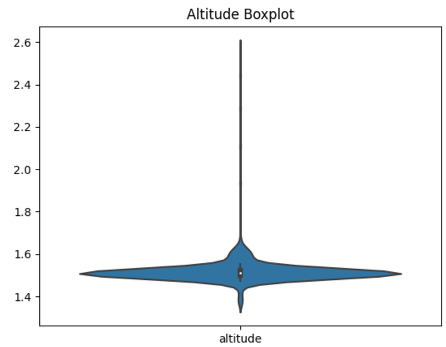
2 IMU Statistics

IMU data consists of linear acceleration along all three axes, angular velocity along all three axes and the orientation of the drone represented as quaternion angles. During parsing, the quaternion angles were converted to Euler angles (in degrees) called "roll(x)", "pitch(y)" and "yaw(z)". The angular velocities were discarded during parsing, because a stable drone will not rotate around it's principal axes, so the data would be relevant only for unstable flights.

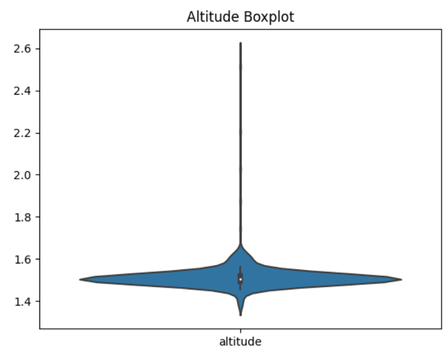
Using the accelerations and orientation angles, we can segment the time axes into different drone orientations. For example, a positive yaw represents the drone facing right and a negative one would indicate the drone facing left. The orientation tells us about the scouting pattern of the drone.

We can use K-Means to cluster the drone orientations across time to gain more information about the nature of the drone trajectory. To optimize the number of clusters, I used the elbow method. The docking and detaching protocols exhibit similar accelerations and orientations as can be seen in the plots in Fig. 3, Fig. 4 and Fig. 5.

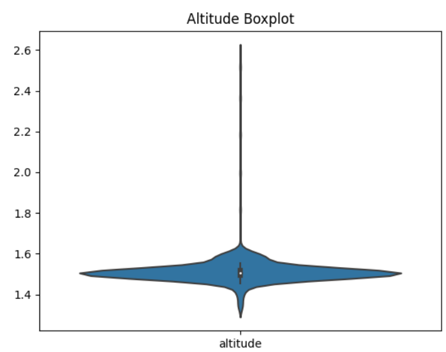
Flight 0 and Flight 2 exhibit a similar trajectory in terms of the yaw data, which potentially points to similar scouting patterns. Flight 1 has a sudden change in the yaw, but this is just because the measurement for $+180deg$ and $-180deg$ represent the same angle in the plane. The plots can be used to reverse-engineer the actual trajectory followed by the drone throughout the flight.



(a) Flight 0

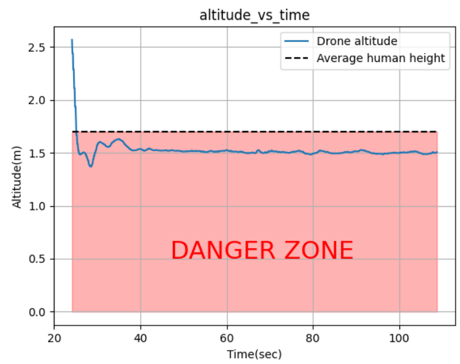


(b) Flight 1

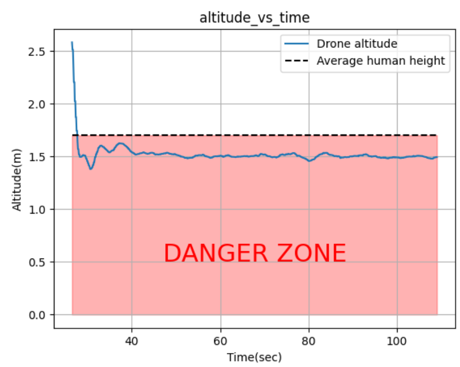


(c) Flight 2

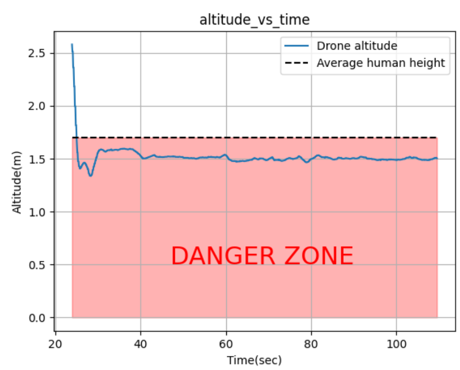
Figure 1: Altitude statistics boxplots



(a) Flight 0

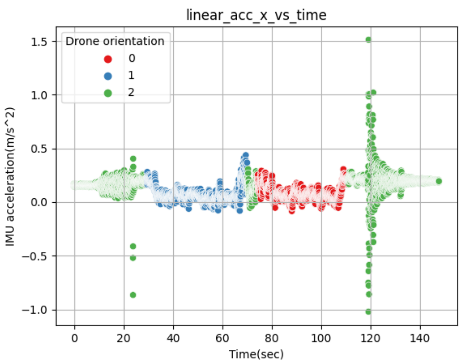


(b) Flight 1

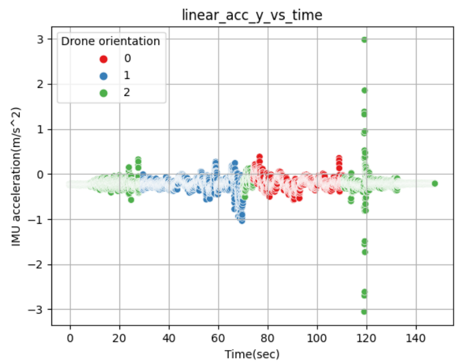


(c) Flight 2

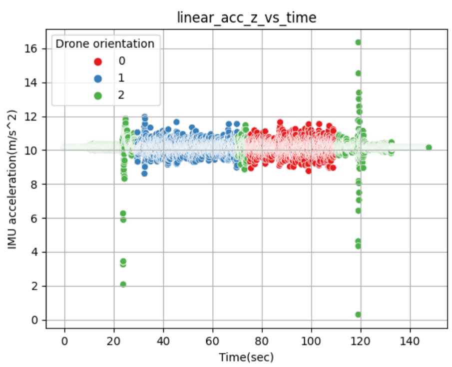
Figure 2: Altitude(m) vs Time(sec)



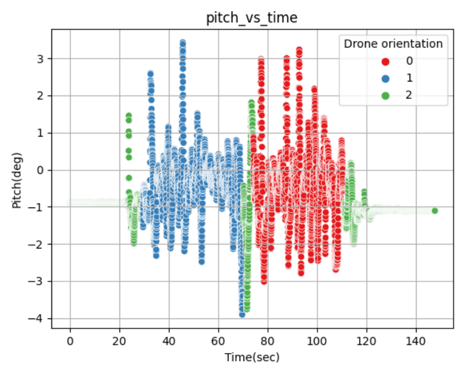
(a) Linear acceleration along x axis



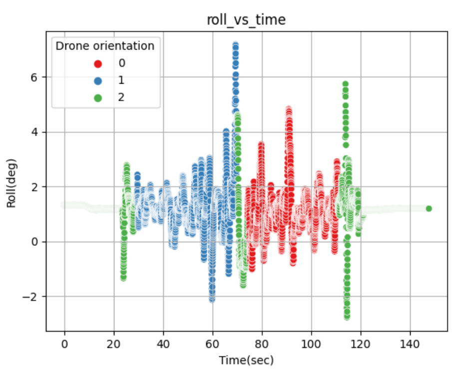
(b) Linear acceleration along y axis



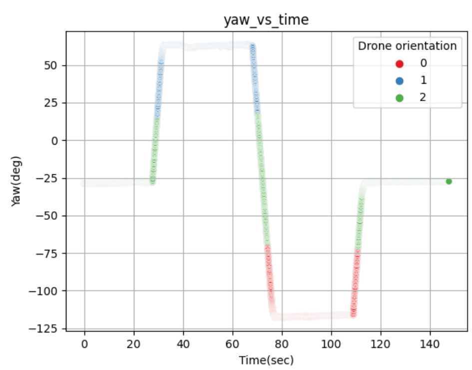
(c) Linear acceleration along z axis



(d) Pitch

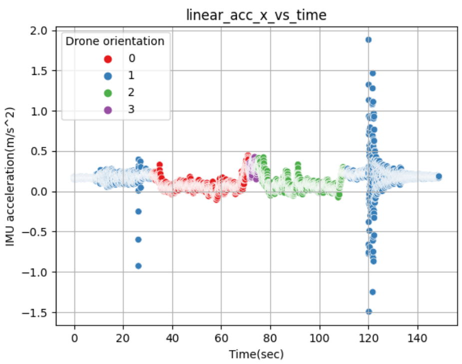


(e) Roll

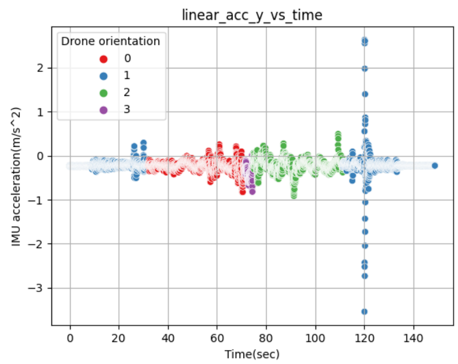


(f) Yaw

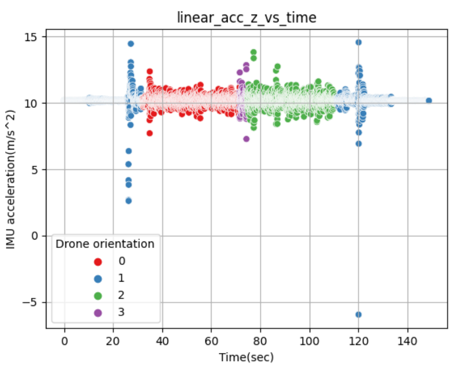
Figure 3: Flight 0 IMU data vs time(sec)



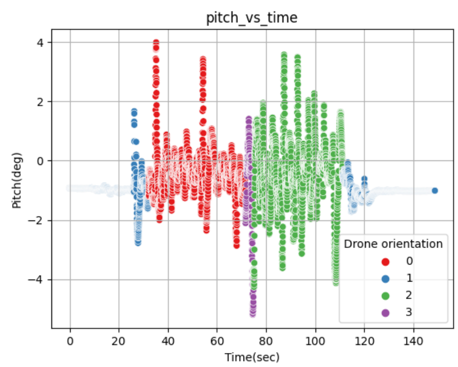
(a) Linear acceleration along x axis



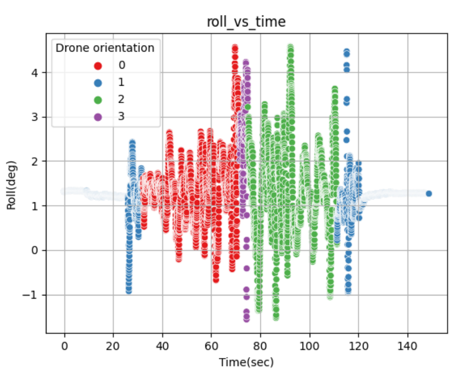
(b) Linear acceleration along y axis



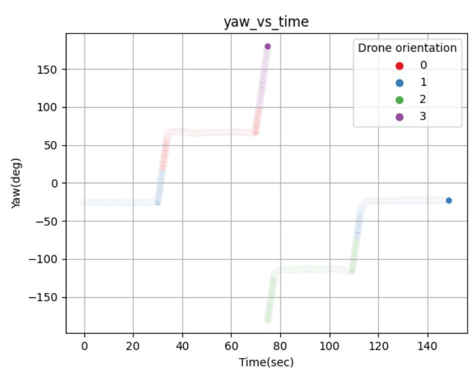
(c) Linear acceleration along z axis



(d) Pitch

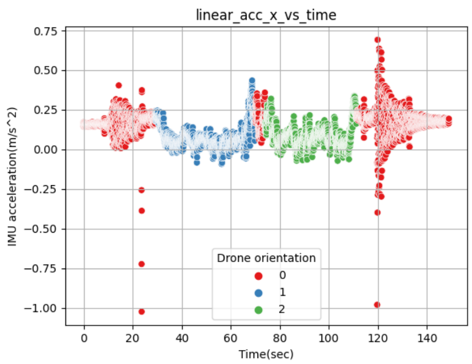


(e) Roll

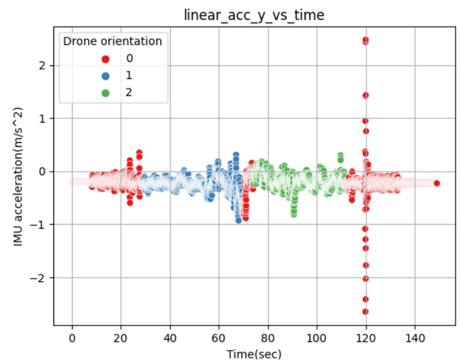


(f) Yaw

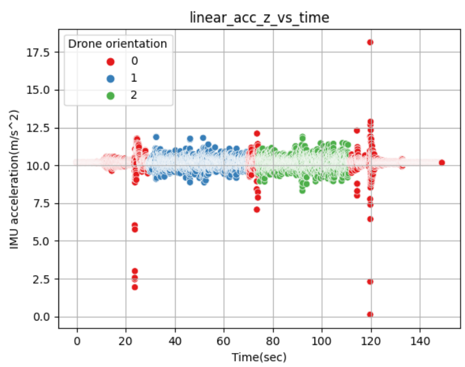
Figure 4: Flight 1 IMU data vs time(sec)



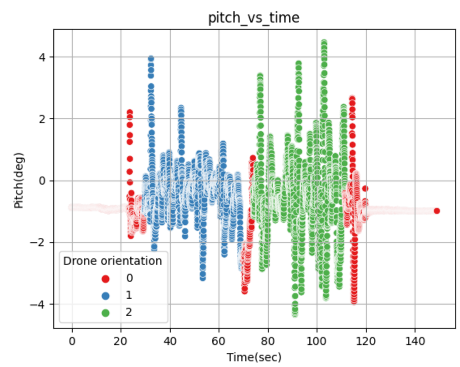
(a) Linear acceleration along x axis



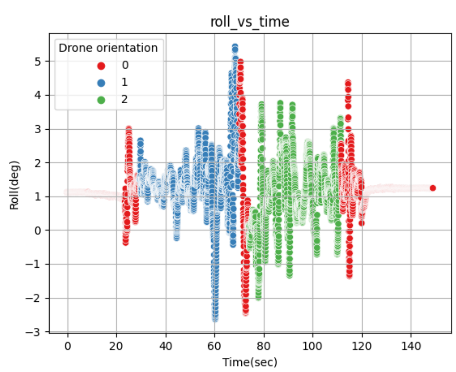
(b) Linear acceleration along y axis



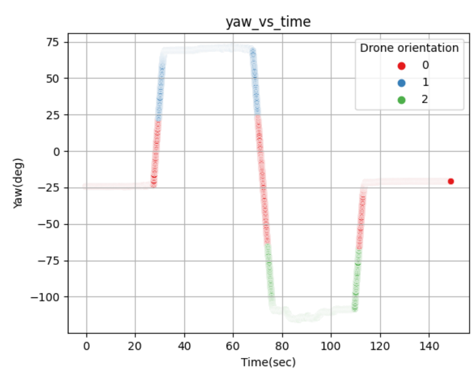
(c) Linear acceleration along z axis



(d) Pitch



(e) Roll



(f) Yaw

Figure 5: Flight 2 IMU data vs time(sec)