**Flight test 1.4**

**Why are we testing and what is the purpose of the planned test?**

The point of this flight test is to be able to successfully allow one drone to take off, follow a predetermined route, and be able to successfully land back at home. The software should be able to accept GPS boundary points and the drone will be required to stay within those boundaries. This is the same type of test as Flight Test 1.3, however we modified the code to verify that all points were visited and that the gimbal was pointed down to view the target.

The current configuration of the software is mature enough to be tested at this level because at this point, we believe that the software should be able to handle flying one drone without video and target detection. By completing this test, we will be able to verify what parts of the software are compatible with the drone, and see what parts need to be fixed before the next test. Before the drone was taken off the string, all pre-mission tests were flown with a string attached to the drone. In addition, all code used in the flight test was unit tested, and if possible, the Parrot simulator.

**Requirements Tested:**

## 

|  |  |  |  |
| --- | --- | --- | --- |
| Number | Type | Requirement | Tested in this event |
| 1 | System | The system SHALL allow for a connection between drones and a computer | Yes |
| 2 | Drone | The drone SHALL send GPS coordinates to the computer | Yes |
| 3 | Drone | The drone SHALL send live color video to the computer | No |
| 4 | Drone | The drone SHALL fly to the maximum supplier specified distance | No |
| 5 | Drone | The drone SHALL send metadata | Yes |
| 6 | Drone | After identifying the target, the drone SHALL send GPS coordinates to the computer | No |
| 7 | Drone | After identifying the target, the drones SHALL hover over the target | No |
| 8 | Drone | After identifying the target, the drone SHALL send live video displaying the target | No |
| 9 | Computer | After identifying the target, the computer SHALL display the target | No |
| 10 | Computer | The computer SHALL break up the amount of land into the pieces, with the amount of drones being the number of spaces | Yes |
| 11 | Computer | The computer SHALL allow for GPS coordinate placement | Yes |
| 12 | Computer | The computer SHALL be able to identify a drones location | Yes |
| 13 | Computer | The system SHALL use Wi-Fi to connect to the drones | Yes (SkyController) |
| 14 | Computer | The computer SHALL remember where the previous drones were | No |
| 15 | Drone | The drone SHALL identify any potential flight risks and modify the path | No |
| 16 | Drone | The drone SHALL return to homebase after a critical battery event | No |
| 17 | Drone | The drone SHALL send an SOS alert after going to the ground, if still connected | No |
| 18 | Drone | After a drone send an SOS alert, the remaining drones SHALL receive new GPS locations | No |
| 19 | Computer | After the computer receives a SOS alert, the computer SHALL resize the area | No |
| 20 | System | As an objective, the system SHALL accept more than one drone manufacturer | Yes (Tested by Analysis) |
| 21 | System | The system SHALL be able to be used with one to multiple drone | Partal (One drone) |

**Pre flight checklist:**

* Verify that the battery is charged
* Verify the SkyController is charged
* Verify that the clouds are above 400 feet
  + If cloud coverage is under 400, verify that the software can only go 25 feet under the cloud coverage
* Verify that the wind is under 30 mph
* Verify the laptop is charged
* Verify that the battery is property in the drone
* Verify that the drone’s wings are correctly fitted
* Verify that the location of which we are flying is unrestricted

**Pre-mission tests:**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 1 | Up | SDK | 1. Connect the drone by Wi-Fi or SkyController  2. Input a message in the UI saying that the drone should go up 25 feet  3. Verify that the drone goes up 25 feet from the starting position | Pass |
| 2 | Left | SDK | 1. Connect the drone by Wi-Fi or SkyController  2. Input a message in the UI saying that the drone should go left 25 feet  3. Verify that the drone goes left 25 feet from the starting position | Pass |
| 3 | Right | SDK | 1. Connect the drone by Wi-Fi or SkyController  2. Input a message in the UI saying that the drone should go right 25 feet  3. Verify that the drone goes right 25 feet from the starting position | Pass |
| 4 | Down | SDK | 1. Connect the drone by Wi-Fi or SkyController  2. Manually fly the drone up 30 feet  3. Input a message in the UI saying that the drone should go down 25 feet  4. Verify that the drone goes down 25 feet from the starting position | Pass |
| 5 | Camera | SDK | 1. Turn on the system  2. Connect the drone by Wi-Fi or SkyController  3. Allow the system to accept camera feed  4. Verify that the camera feed is displayed on the computer | Pass |
| 6 | Moving Camera | SDK | 1. Turn on the system  2. Connect the drone by Wi-Fi or SkyController  3. Allow for the system to accept camera feed  4. Verify that the camera feed is displayed on the computer  5. Verify that the user can move | Pass |
| 7 | Take off | SDK | 1. Verify that the drone can take off | Pass |
| 8 | Land | SDK | 1. Verify that the drone is in the air  2. Input a message to allow the drone to land  3. Verify that the drone can land | Pass |
| 9 | Battery | SDK | 1. Verify that the battery level is less than 25%  2. Verify that the drone is coming back home and then landing | Pass |
| 10 | Emergency Landing (Wi-FI) | SDK | 1. Connect the drone by Wi-Fi  2. Input message that allows the drone to fly 20 feet  3. Disconnect the drone from Wi-Fi  4. Verify that the drone can connect to the phone's Parrot App by Wi-Fi | Pass |
| 11 | Emergency Landing (Sky Controller) | SDK | 1. Connect the drone by the SkyController  2. Input message that allows the drone to fly 20 feet  3. Disconnect the drone from the SkyController  4. Verify that the drone can connect to the phone's Parrot App by the SkyController | Pass |

These tests are a prerequisite to any flight test with the search algorithm. Without these tests, the drone could cause serious damage to itself or others. To verify that all of these tests work, the drone will be tethered to a string to verify that the drone will not fly off and cause damage. In theory, by allowing the drone to fly tethered on this, flying untethered will be more successful by knowing that all basic features work.

**Safety:**

Before flying the drone, we completed all pre-mission tests successfully and the flight was recorded in the flight log. In addition, the flight test was observed by one sUAS pilot and two visual observers in different locations to verify the drone was going in the correct direction. All observations were at point 14. We were able to see the drone the entire time, as complaint with Part 107.

**Location:**

We will be flying at Mendon Ponds Park for this flight test. The image below shows all points taken by the drone as well as the boundary points.



**Results:**

Overall, for the first test conducted under real-life conditions with a SkyController, it was a success.

## 

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Number | Type | Requirement | Tested in this event | Passed/Failed |
| 1 | System | The system SHALL allow for a connection between drones and a computer | Yes | Pass |
| 2 | Drone | The drone SHALL send GPS coordinates to the computer | Yes | Pass |
| 3 | Drone | The drone SHALL send live color video to the computer | No | N/A |
| 4 | Drone | The drone SHALL fly to the maximum supplier specified distance | No | N/A |
| 5 | Drone | The drone SHALL send metadata | Yes | Pass |
| 6 | Drone | After identifying the target, the drone SHALL send GPS coordinates to the computer | No | N/A |
| 7 | Drone | After identifying the target, the drones SHALL hover over the target | No | N/A |
| 8 | Drone | After identifying the target, the drone SHALL send live video displaying the target | No | N/A |
| 9 | Computer | After identifying the target, the computer SHALL display the target | No | N/A |
| 10 | Computer | The computer SHALL break up the amount of land into the pieces, with the amount of drones being the number of spaces | Yes | Pass |
| 11 | Computer | The computer SHALL allow for GPS coordinate placement | Yes | Pass |
| 12 | Computer | The computer SHALL be able to identify a drones location | Yes | Pass |
| 13 | Computer | The system SHALL use Wi-Fi to connect to the drones | Yes (SkyController) | Pass |
| 14 | Computer | The computer SHALL remember where the previous drones were | No | N/A |
| 15 | Drone | The drone SHALL identify any potential flight risks and modify the path | No | N/A |
| 16 | Drone | The drone SHALL return to homebase after a critical battery event | No | N/A |
| 17 | Drone | The drone SHALL send an SOS alert after going to the ground, if still connected | No | N/A |
| 18 | Drone | After a drone send an SOS alert, the remaining drones SHALL receive new GPS locations | No | N/A |
| 19 | Computer | After the computer receives a SOS alert, the computer SHALL resize the area | No | N/A |
| 20 | System | As an objective, the system SHALL accept more than one drone manufacturer | Yes (Tested by Analysis) | N/A |
| 21 | System | The system SHALL be able to be used with one to multiple drone | Partal (One drone) | Yes |

Out of the 8 requirements explicitly tested, 8 passed, leading to a 100% pass rate. Unlike last time, requirement 10 passed because it visited all of the points we told it to visit.

**Lessons Learned:**

Overall, for the second flight test that is fully connected to the computer, it went great. There were no issues with the code, and the drone did as we expected. Our plan for the next test is to connect the computer to 2 drones and see if the drones will visit all of the assigned points. In addition, we plan on running the CV algorithm to verify that a human can be detected.

In conclusion, as a team, this was a great experience and we plan on continuing on the great path we are on. For our next test, we plan on testing this software again to get the correct points visited.