PEX 03: Autonomous First Aid Kit Delivery

Due @23:59 on 10 May (T40)

Help Policy

AUTHORIZED RESOURCES: Any, except another cadet's assignment or published solutions to the assigned problem.

NOTE:

- This is a team effort. Never copy another team's work and submit it as your own. Here are a few blatant examples of copying:
  - Making an electronic copy of another team's solution and then modifying it slightly to make it appear as your own work.
  - Reading a printout or other source of another team's work as you implement your solution.
  - Completing your entire solution by following explicit instructions from another team, while they refer to their own solution
- Helping your classmates learn and understand the homework concepts is encouraged, but extensive assistance should generally be provided by DFCS instructors. Only provide assistance up to your depth of understanding, beyond which assistance by more qualified individuals is more appropriate and will result in greater learning. If you have to look at your solution while giving help, you are most likely beyond your depth of understanding.
- You may consult **ChatGPT** to whatever extent desired for this project!
- Help your classmates maintain their integrity by never placing them in a compromising position. Do not give your solution to another cadet or team in any form (hard copy, soft copy, or verbal).
- **DFCS will recommend a grade of F for any cadet who egregiously violates this Help Policy or contributes to a violation by others. Allowing another cadet to see your assignment to help them will result in a zero on this assignment.**

Documentation Policy

- You must document all help received from sources other than your instructor or instructor-provided course materials (including your textbook).
- A **project log** providing details describing main challenges, solutions, and each member's contribution to the overall project. If this detail is missing upon submission, **12% will be subtracted from the final project grade.**
- The documentation statement must explicitly describe WHAT assistance was provided, WHERE on the assignment the assistance was provided, and WHO provided the assistance.
- If no help was received on this assignment, the documentation statement must state "NONE."
- If you checked answers with anyone, you must document with whom on which problems. You must document whether or not you made any changes, and if you did make changes you must document the problems you changed and the reasons why.
- Vague documentation statements must be corrected before the assignment will be graded and will result in a grade deduction equal to 5% (ceiling) of the total possible points.

Introduction

Suppose a downed pilot is lost, stranded in an area where we only have an idea of their location. This individual is injured, must receive medical supplies, and then be extracted ASAP. We need to devise a way, using a drone and AI, to locate this person, deliver much-needed first aid, and then report their location to an extraction. What's involved with locating and identifying a stranded individual and then delivering a first aid package to that person? There are several main points to consider…

While the drone is deployed and fling a predetermined flight path over a defined air space, how might we locate and identify a person from the air? How may we track the objective after it is verified? Perhaps we could create a refined object tracker to track. How might we estimate the stranded individual's geolocation? Finally, how might we deliver a package once the person's location has been determined? What code could we repurpose from PEX02, and what new elements would we need to create?

The Objective

The objective for this project is to lift a care package attached to a cable, send the drone off, using a predefined flight path & pattern, locate and verify the stranded person (dressed, male mannequin) on the ground, safely deliver the first aid packaged within ten feet of that person and return home. Your team will put together all the code necessary to carry out a successful mission. You may use any code we've covered and/or created throughout this course to help. You must determine how to prepare and implement your object recognition model, how you will locate the person in need of help, and how to deliver the care package undamaged. You already have some of the code modules from the previous PEX that you will need for this project: drone_lib.py, RealSense camera routines, etc... Start a new PyCharm project starting with those elements and create a new drone_mission.py to hold your main project code. An initial yolo object recognition model will be provided for you (see section *Problem#2: Object Recognition Model* for details) and this document contains some helpful details as well. The rest will be up to your team to figure out. While some aspects of this project cannot be easily tested via SITL, much of the important pieces can. **It's up to you to discover how SITL can help as you progress through this project.**

Main project elements in this project:

- Use object recognition instead for target identification.
- The use a sophisticated object tracker for positioning over objective.
- Calculate GIS location for package drop rather than relying on non-deterministic algorithm.
- Lowering a package to the ground and detaching from it.

Problem#1: Object Recognition Model

Ideally, we would be able to use a good pretrained object recognition model (CNN) that works out of the box for aerial images. Unfortunately, there are no good aerial object models that we have access to. However, we *do* have access to some imagery data collected from drones called visdrone2019. We should be able to train a good performing object recognition model through some sort of transfer learning process using this visdrone2019 dataset. A promising object recognition model that runs well on small, portable and embedded systems is called yolo4-tiny. We married these two things together to get our solution. We performed some transfer learning on the yolo model, and we will use the resulting weights to recognize a human from the air.

To simplify the process, some of setup/configuration was performed for you; however, the model has some weak points: (1) orientation is important, because the model cannot properly identify people on the ground that aren't more or less upright, and (2) scale is another factor, since the images we trained over were small, due to the altitude the videos were collected from. As you might've guessed, these issues could've been mitigated via data augmentation. Unfortunately, no augmentation technique was applied; we must work with what we have. We are left to consider how we may utilize a weak, but working performer to complete our task. Further details, along with the all the files you'll need will be given in class (lesson 32). We will walk through the process during that time.

Problem#2: Object Tracking

You have the option to employ one of the object trackers you will have experimented with during lab (yet to be announced), or modified one of these trackers to improve its performance. The choice is yours. You will need to calculate where the object is in the images coming from the camera so that you can use your pixels-to-LatLon algorithm to determine the actual location of the object (i.e. the person in distress). OpenCV will draw a bounding box around an object it identifies. You can easily calculate the center of that object and use that to calculate a geospatial location to the object.

Problem#3: From Vision to Lat/Lon

So, how might we calculate where an object actually is by using input we get from our onboard depth sensor? Let's assume that we know our camera's angle and our drone's height from the ground. Combined with the depth sensor reading, how might we go about determining a good lat/long estimate for where the person is standing after we've located and centered on that person?

Problem#4: Deliver the Package

Your python script will determine a safe location 10ft away from the objective (a 10-foot radius from the person). After the calculation is made, the drone will fly to the location, lower the package to the ground, and then release it from the drone. The package will be tethered to a ten-foot cable, and it will contain *extremely delicate materials* (i.e. eggs). The payload **must not be damaged**; therefore, the drone must gently lower the package to the ground before detaching the cable. The drone will be equipped with a mechanism you can control via MavLink (using DroneKit) to detach the cable (details supplied later). Finally, after delivery is made, the drone must return home to consider the mission complete.

Project Evaluation

You will be graded on several fronts.

- First, how effectively your drone was able to identify the objective.
- Second, how close to the 10-foot radius your drone was able to deliver its payload **without injuring the objective**.
- Third, how much damage the payload incurred during the process (ideally, none).
- Forth, how quickly your drone was able to complete its mission.
- Fifth, how smoothly your mission was executed overall.

Each team will be ranked and score weighted accordingly:

- First place receives 100%
- Second place receives 95%
- Third place receives 90%
- Fourth & fifth places will receive 85%
- If your drone fails to reasonably execute larger part of the mission, the *best* your team can receive is 70%.