OAK-D DRONE LANDING

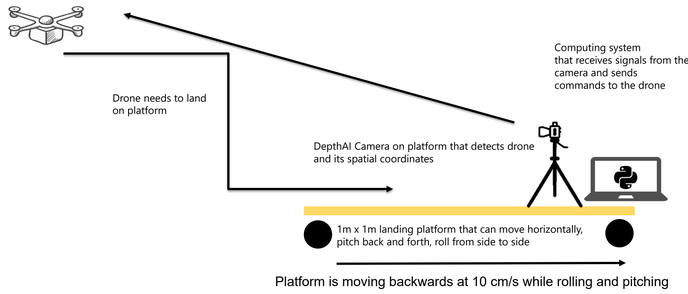
STUDENTS: AADHAR CHAUHAN, ABDULRAHMAN ADAM, ARQUM UDDIN, CHAU HUYNH, AND KIRAN MATHEWS

31

What is the OAK-D Drone Landing Project

Problem Statement: Multirotor Vertical Take-off & Landing (VTOL) vehicles landing on ships are currently controlled by human interaction. Human inputs can be inaccurate and erroneous causing accidents, injuries, and damage to equipment.

Solution: The OAK-D (OpenCV AI Kit with Depth) Drone Landing Project presents a system that uses an AI camera equipped with spatial detection to detect drones flying towards a platform. Once the drone is detected, the system receives coordinates of the drone's location and sends commands to the drone to assist in its landing.



Goals and Requirements

Main Goal: To create a system that guides a drone to land on a platform using a camera programmed with object detection and depth perception.

Hardware Requirements:

CoDrone Pro

[1]

OAK-D Lite

[2]

Requirements:

Criteria for landing:

- The drone must land within 60 seconds.
- The drone must land in a 1m x 1m space.
- The drone must be able to land when the platform is stationary.
- The drone must be able to land when the platform is moving at a speed of 10 cm/s.
- The drone must be able to land when the platform is pitching back and forth and/or rolling from side to side.

Criteria for detection and control:

- The camera must be able to detect the drone in some manner.
- The camera must be able to gain the location of the drone.
- System must be able to use the signals from the camera to control the drone's motion.
- The camera must be stationary relative to the platform.
- The camera's field of vision must be wide enough to compensate for any drone drift.

System Implementation

Implementing Initial Drone Control:

- QR code is implemented using the Pyzbar python library to control the drone outside of camera's detection range (**Fig. 1**).
- Drone uses human face as a fiducial (**Fig. 2**).
- Drone is commanded to fly towards the platform (in the camera's field of view).

Fig. 1. Detection of the QR code to control the drone movement

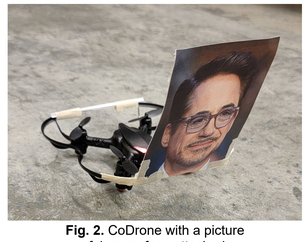


Fig. 2. CoDrone with a picture of human face attached

Detection using OAK-D Camera:

- Our detection model Yolo-V4 yields consistent detection and drone coordinates.
- The camera controls the drone using the drone's coordinates once detected (**Fig. 3**).

Drone Landing Procedure:

- Once the drone is within 1-meter, the drone is commanded to land on the platform.
- The drone pitches forward using a pitch power calculated using translation speed of the platform relative to an anchor.

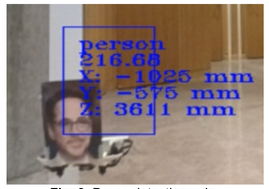


Fig. 3. Drone detection using facial detection.

Fig. 4. Platform translating while the drone flies towards it. The red circle highlights the drone.

System Block Diagram

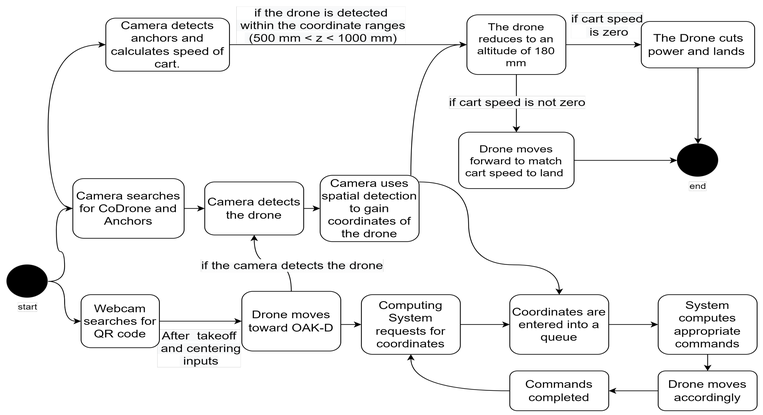


Fig. 5. Block Diagram of Overall System

Results

Detection Range:

- Camera accurately detects within a 1-meter range (**Fig. 6 & 7**)

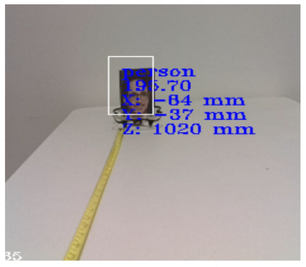


Fig. 6. Camera detecting "person" within the range of about a meter

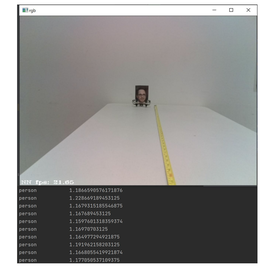


Fig. 7. Camera faintly detects the "person" when the distance increases beyond a meter

Drone Control:

- The drone is successfully controlled via QR code outside of the 1-meter range (**Fig. 1**).
- The drone is then controlled by the camera once within the 1-meter range (**Fig. 2**).

Speed of the Platform relative to the drone:

- The drone is limited to a speed of roughly 23 cm/s at 25% pitch power since the camera is unable to detect the drone at higher speed. This limits the platform speed to 23.24 cm/s as shown in **Table 1** below.

Cart Travelled Distance (cm)	Total time (sec)	Average cart speed (cm/s)	Test description	Test outcome prediction (land or not)	Actual outcome (land or not)	Test passed?
330	14.2	23.24	Optimal speed	Yes	Yes	Yes
105	10	10.50	Required speed	Yes	Yes	Yes

Table 1. Blue: maximum speed the platform can reach for the drone to land successfully. Yellow: required speed.

Tradeoffs, and Future Work

Tradeoffs:

- The drone cannot be detected by itself, hence why facial detection must be used.
- The detection model being used (Yolo-V4) is lightweight and provides accurate detections. However, it sometimes returns multiple detections for one object.
- Cart speed calculations are contingent on both accuracy of anchor coordinates, and consistency of detection, hence, inconsistent anchor readings could lead to false pitch.

Future Work:

- Enhancement of drone detection range and orientation; elimination of QR detection.
- Complete autonomous drone control; minimize human dependence.
- Bigger drone for improved drone detection, drone motion stability and consistency.

References, and Acknowledgments

References:
[1] Robolink . (n.d.). Python with CoDrone Pro/Lite. Robolink Basecamp. Retrieved May 23, 2022, from https://learn.robolink.com/course/codrone-with-python/
[2] Luxonis. (2020). DepthAI's documentation Luxonis. Retrieved May 23, 2022, from https://docs.luxonis.com/en/latest/

Acknowledgement:
•Faculty: Prof. David W. Krout and Daniel King
•Industry: Dr. David B. Laning and Jeff Decker

ELECTRICAL & COMPUTER ENGINEERING
UNIVERSITY of WASHINGTON

ADVISORS: DR. DAVID B. LANING, JEFF DECKER, AND PROF. DAVID W. KROUT

SPONSORS: INSITU A BOEING COMPANY