Sunflower Pollination UAV

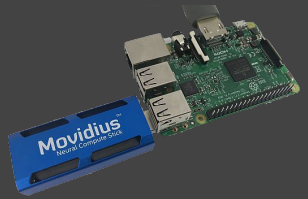
Anna Seib
Giovanni Artavia
Brandon Lammey

Purpose

The Plant Pollination UAV provides a more efficient means of artificially pollinating plants in order to reduce the need for hand pollination in areas without native pollinating species.

The purpose of this project was to create a UAV that is capable of identifying one type of plant using computer vision and machine learning algorithms processed on a Raspberry Pi. Then using the UAV to mimic methods of pollination for that select plant. In this case, we chose to pollinate sunflowers due to their large size and ease of detection.

Components: Hardware



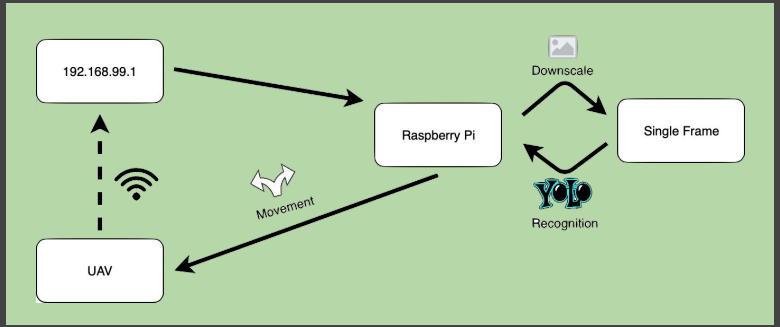
Hardware

- Parrot BeBop II
  - Easily programmable and fairly priced drone capable of carrying weight of components needed for image processing and control
- Raspberry Pi
  - Small lightweight computer used for image processing and control of UAV
- Intel Neural Compute Stick
  - Small deep learning device used for rapid deployment of neural network

Software

- PyParrot
  - Library for basic control of UAV with functions for omnidirectional movement
- OpenCV
  - Computer vision library for image manipulation
- TensorFlow
  - Library used for machine learning model
- FFMPEG
  - Library for handling video from UAV
- DarkFlow
  - Translates YOLO to TensorFlow

Components: Software



Search

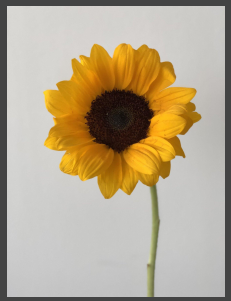
Fly in a grid pattern recording the area in front of the UAV.

Convert Picture

Convert video feed into lower resolution images.

Image Recognition

Use YOLO to determine if the drone is seeing a sunflower.



Pollination

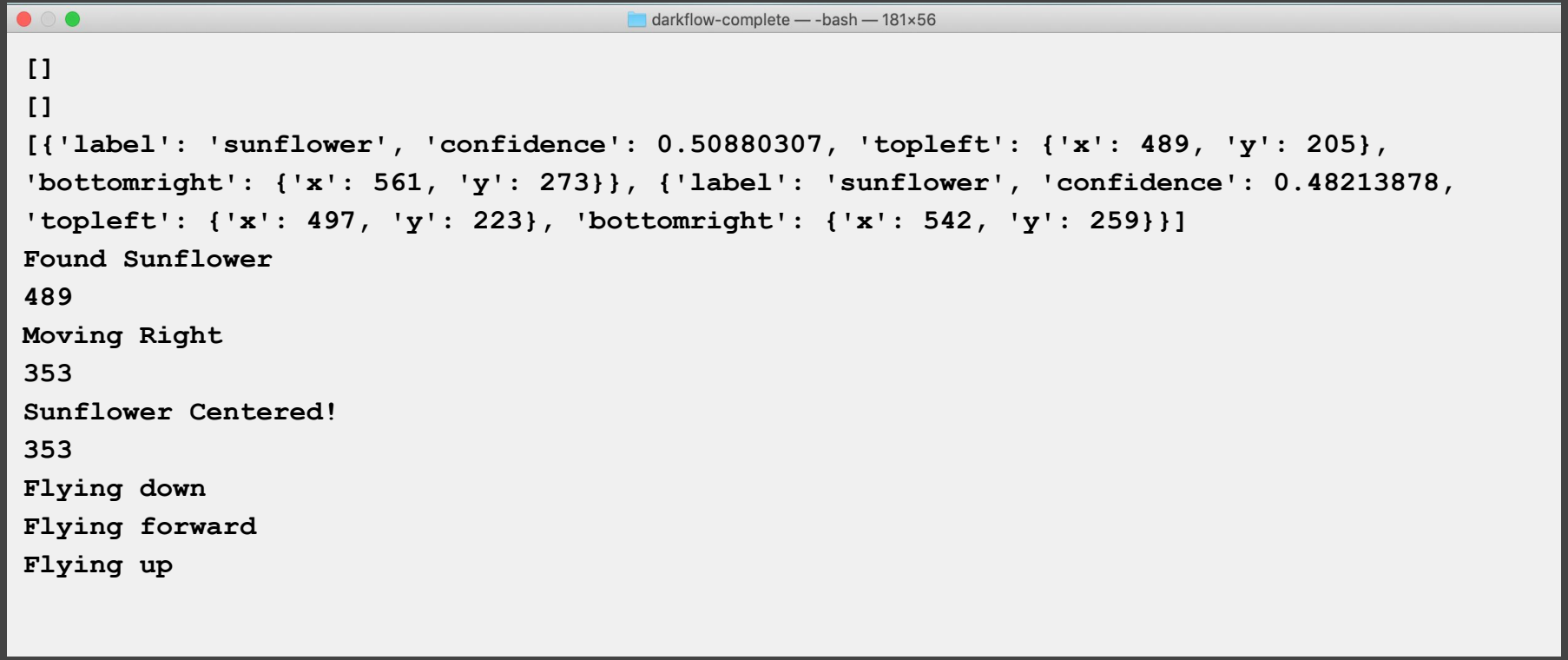
If a flower is found, fly towards it, center the drone and slowly move forward to pollinate

Mark Flower

Mark current flower as pollinated and fly back up


sunflower

Output


darkflow-complete — -bash — 181×56
[]
[]
[{'label': 'sunflower', 'confidence': 0.50880307, 'topleft': {'x': 489, 'y': 205},
'bottomright': {'x': 561, 'y': 273}}, {'label': 'sunflower', 'confidence': 0.48213878,
'topleft': {'x': 497, 'y': 223}, 'bottomright': {'x': 542, 'y': 259}}]
Found Sunflower
489
Moving Right
353
Sunflower Centered!
353
Flying down
Flying forward
Flying up

Design Issues

Software

- Detection only works for sunflowers due to the ease of datasets and training and does not differentiate between parts of the sunflower
- No collision avoidance is used for flight to avoid obstacles or human/ animal interference
- Landing procedure does not check for clear area to land but does use a reducing speed land procedure

Hardware

- Added weight makes drone susceptible to loss of balance in moderate to high winds
- Pi + neural compute stick has limited memory/ processing power which can result in program crash
    - Components out of date
- Short flight time of approximately 20 minutes (drone attempts to land at 15% battery as a safety feature)
- Pi + Neural compute stick processing is also battery intensive
- Battery technology limits flight time

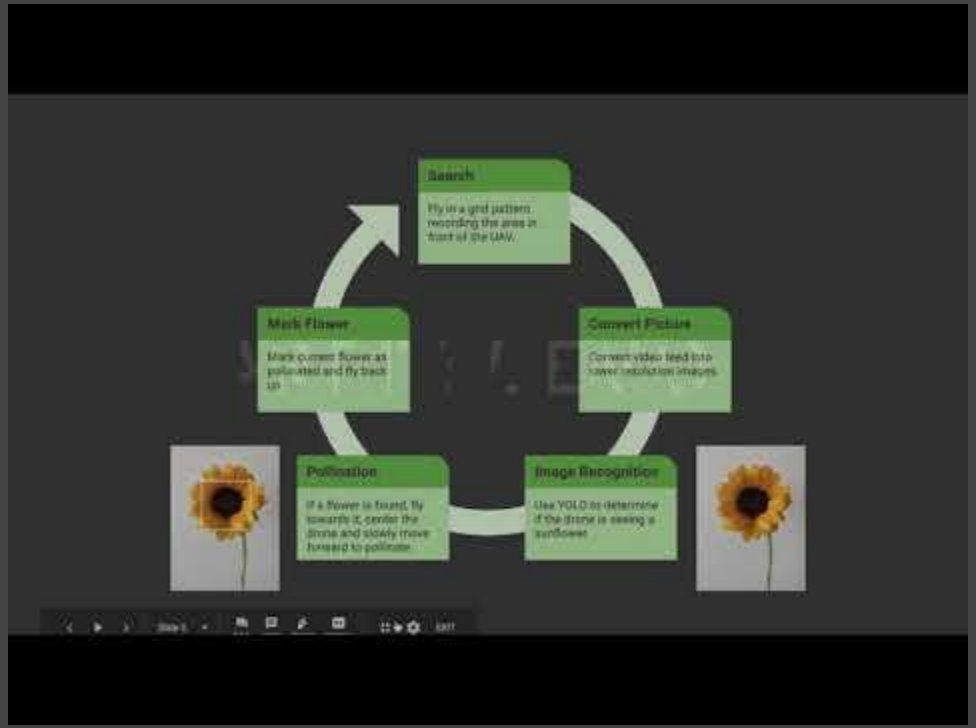
Alternative Designs

- Cloud Computing
  - Use for processing image to improve battery life
  - Must resolve connectivity issues
- Newer onboard computer
  - Nvidia Jetson Nano (released later into project)
  - Intel NCS 2 (released later into project)
- Reduce cost of drone
  - Build or purchase a drone for less than the Bebop 2 at $300
  - Presents issues with flight commands as it must be compatible with PyParrot Library
- Use of UI
  - select grid/ GPS area to pollinate
  - kill switch
  - Select type of plant to pollinate (include plant library)

Ethical Considerations

- UAV has limited flight time and is not tested or recommended for use in moderate to severe weather
  - including rain and high wind)
- No current object collision avoidance
  - Could run into people/ animals/ plants/ objects causing lacerations
  - Impact detection is implemented but takes a brief amount of time for propellers to shut off
  - Not recommended for use in populated areas, near roads, near airports to remain FCC compliant and to avoid accidents
- No area check for safe landing in landing sequence
  - Landing function only causes a slow land feature with blades shutting off when bottom altitude sensor detects object with close proximity (assumes landing on solid ground)
- Connection is initiated over unsecure Wi-Fi
  - video and commands are sent over unsecure channel

Video


Search
Fly in a grid pattern recording the area in front of the UAV.
Convert Picture
Convert video feed into lower resolution images.
Image Recognition
Use YOLO to determine if the drone is seeing a sunflower
Pollination
If a flower is found, fly towards it, center the drone and slowly move forward to pollinate
Mark Flower
Mark current flower as pollinated and fly back up