

Lekcija 6:

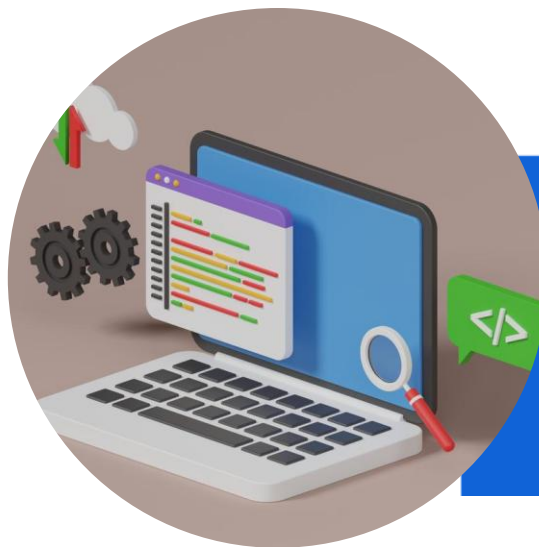
Zbiranje podatkov z droni

Pametno kmetovanje z droni



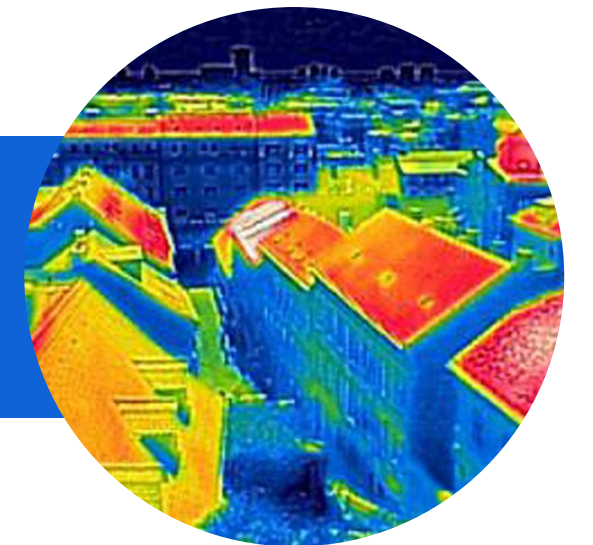
Cilji lekcije

Razumeti, kako droni zbirajo podatke
v kmetijstvu



Naučite se uporabljati Python za fotografiranje

Oglejte si, kako zračno slikanje pomaga kmetom spremljati
njihova polja



TEME

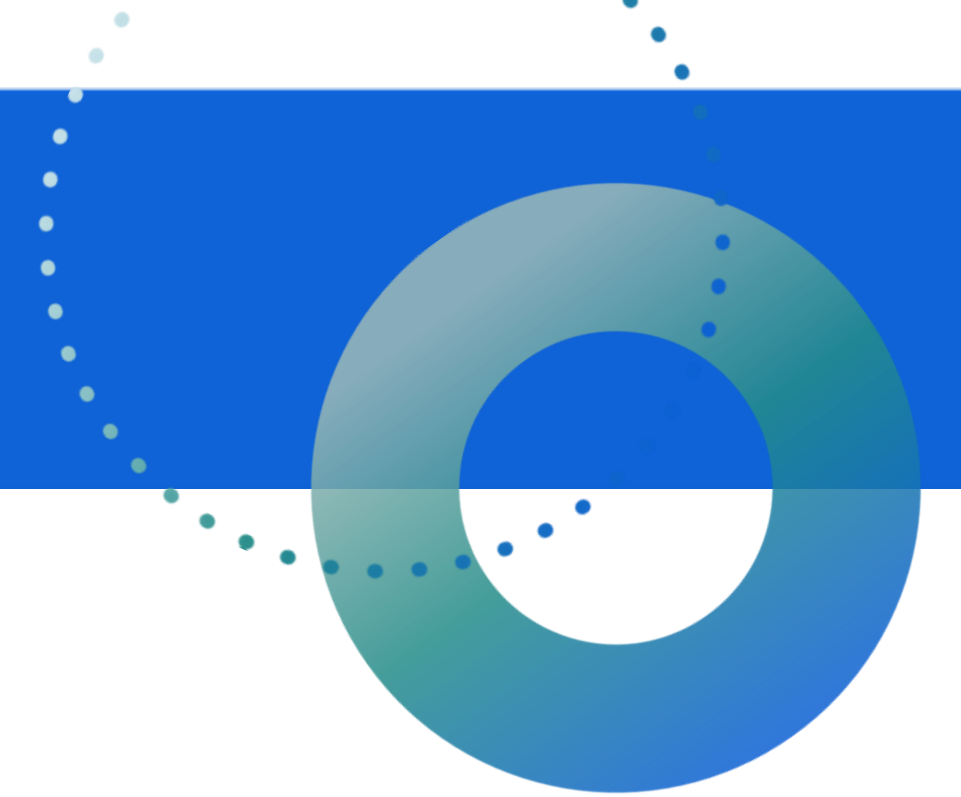
Kaj je zbiranje podatkov v pametnem kmetijstvu

Senzorji, zračne slike in avtomatizacija

Razprava in vprašanja in odgovori

Droni kot zbiralci podatkov

Kaj je zbiranje podatkov v pametnem kmetijstvu



Pametno kmetijstvo uporablja napredne tehnologije, kot so droni in senzorji, za zbiranje podatkov v realnem času polje. Ti podatki pomagajo kmetom pri sprejemanju boljših odločitev o zdravju pridelkov, namakanju, gnojenju in škodljivcih, ter nadzor.

- Droni delujejo kot "leteči izvidniki", ki hitro opazujejo velika polja.
- Kamere in senzorji na dronih zbirajo slike in podatke o okolju.
- Zbiranje podatkov pomaga pri zgodnjem odkrivanju težav – preden se razširijo.

Namen poslanstva

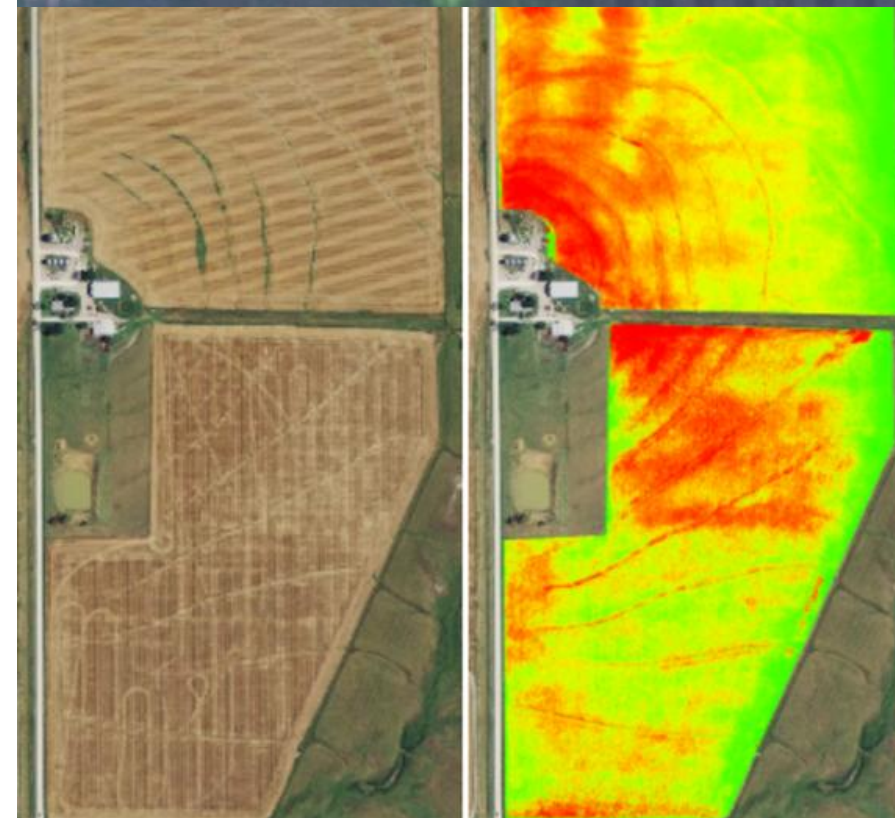
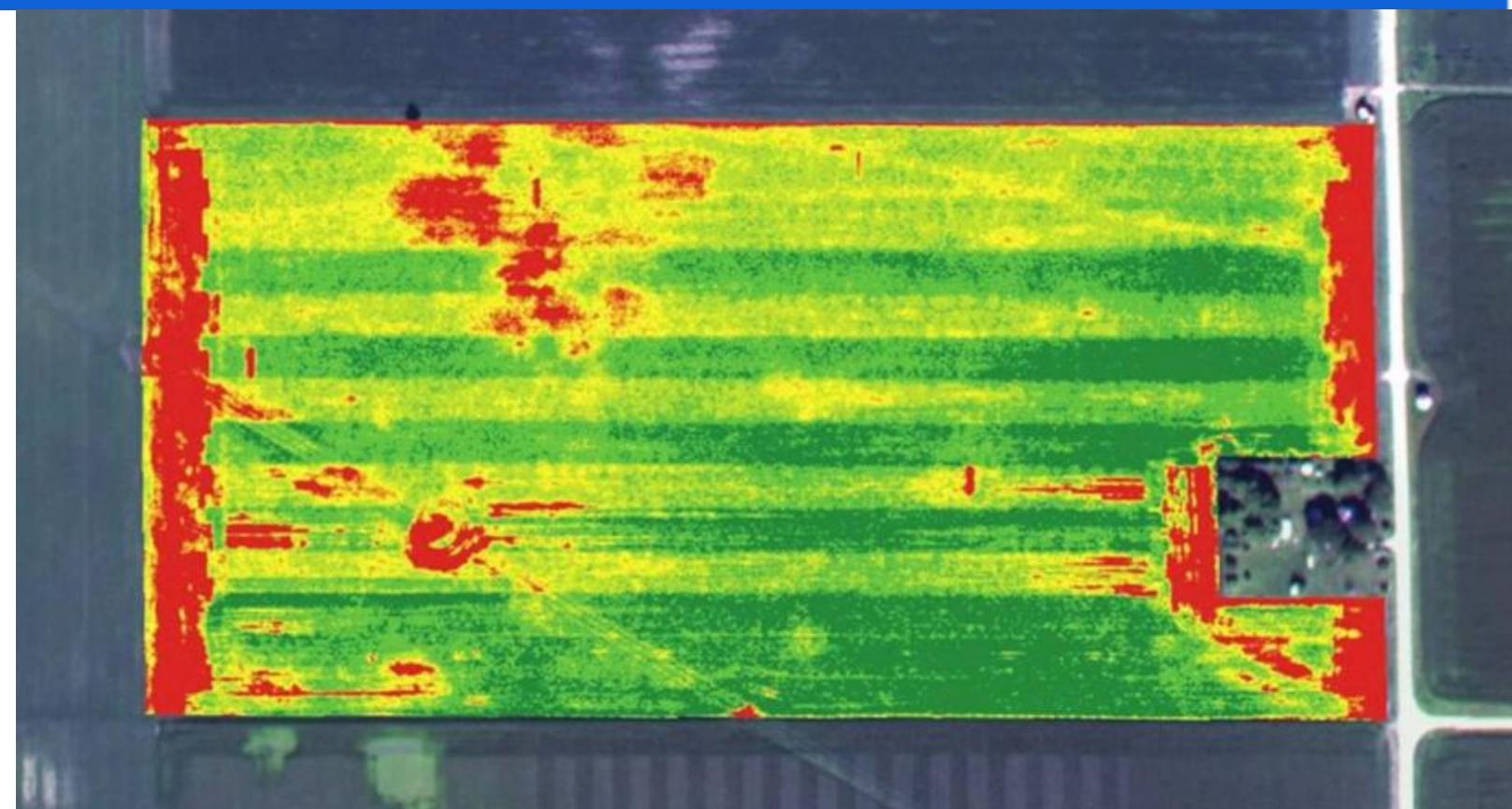
V tej lekciji je naloga našega drona najti določeno problematično območje na terenu in ga posneti fotografija tega.

- Dron bo letel po vnaprej določenem vzorcu za skeniranje polja.
- Koda QR, postavljena na območje talnega zemljevida z nezdravimi pridelki.
- Ko je QR koda zaznana, se bo dron ustavil in posnel sliko za kasnejšo analizo in vrnitev na zemljo
- Ta postopek simulira, kako lahko kmetje hitro prepoznajo težave in ukrepajo.



Pot leta drona

- Za skeniranje polja dron sledi sistematični poti leta imenovana "kosilnica" ali mrežni vzorec.
- To zagotavlja popolno pokritost območja brez izpuščenih točk. Dron leti v ravnih linijah, naprej in nazaj polje. Nadmorska višina, hitrost in razmik med črtami so vnaprej nastavljeni vnačrt leta.
- Ta vzorec pomaga dronu zbirati dosledne in natančne podatke čez celo polje.

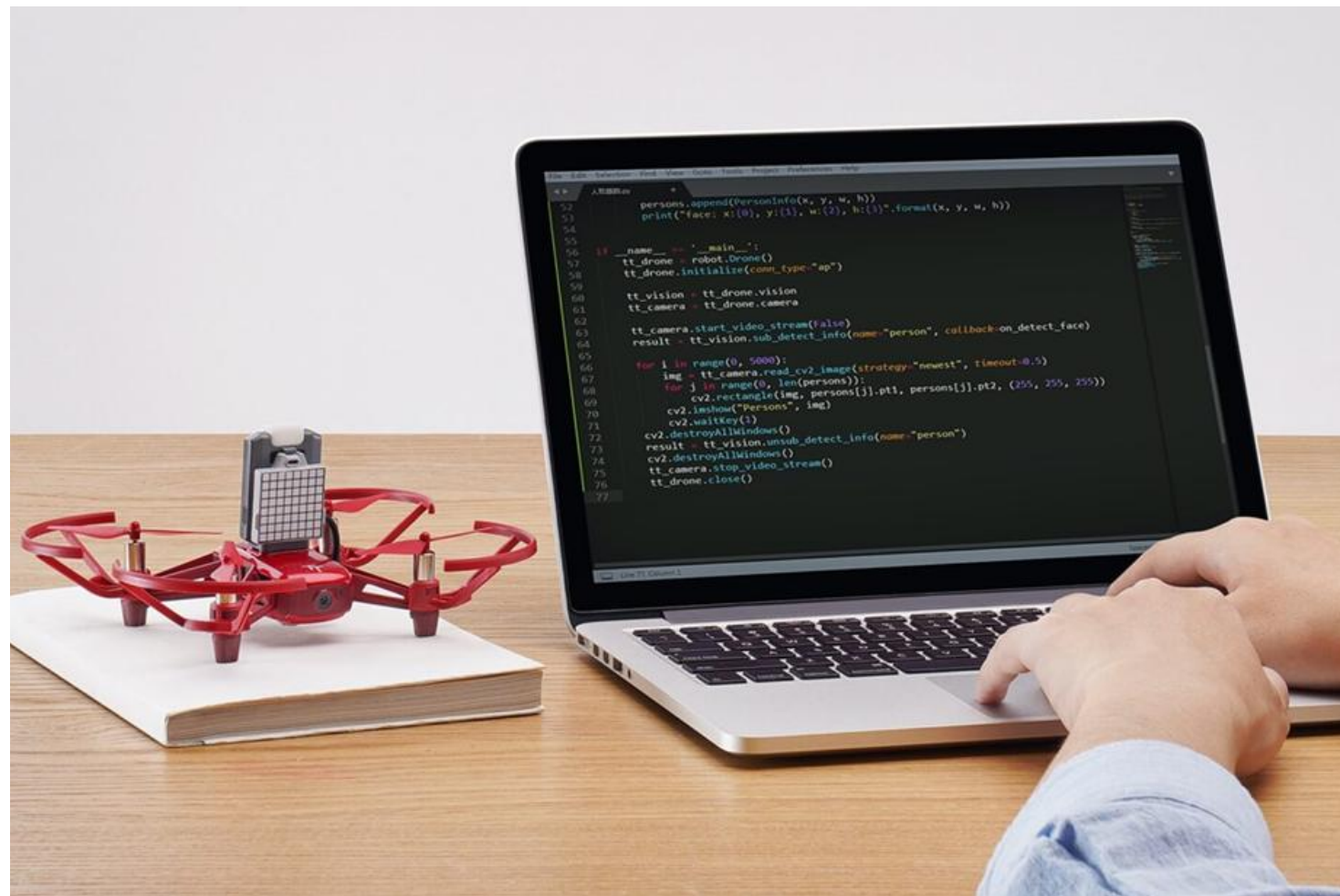


Dejavnost: Programiranje zajemanja slik

- Vzlet.
- Izvedite vzorec iskanja na vnaprej določenem območju
- Poiščite slabe pridelke
- Fotografirajte in pristanite
- Vrnitev domov in na kopno.



Potrebni materiali



STROJNA OPREMA

DJI RobomasterTTDron

Talna podloga Drones4STEM

Osebni računalnik

PROGRAMSKA OPREMA

Platforma MIND+ Scratch GUI (MODUL PYTHON)

ali

Aplikacija Drone4Stem

Namestitev MIND+ Pythona

Kontrolni seznam namestitve v Pythonu

- Izberite modul Python

- Izberite RAZŠIRITVE ->

UPORABNIK-RAZŠIRITVE

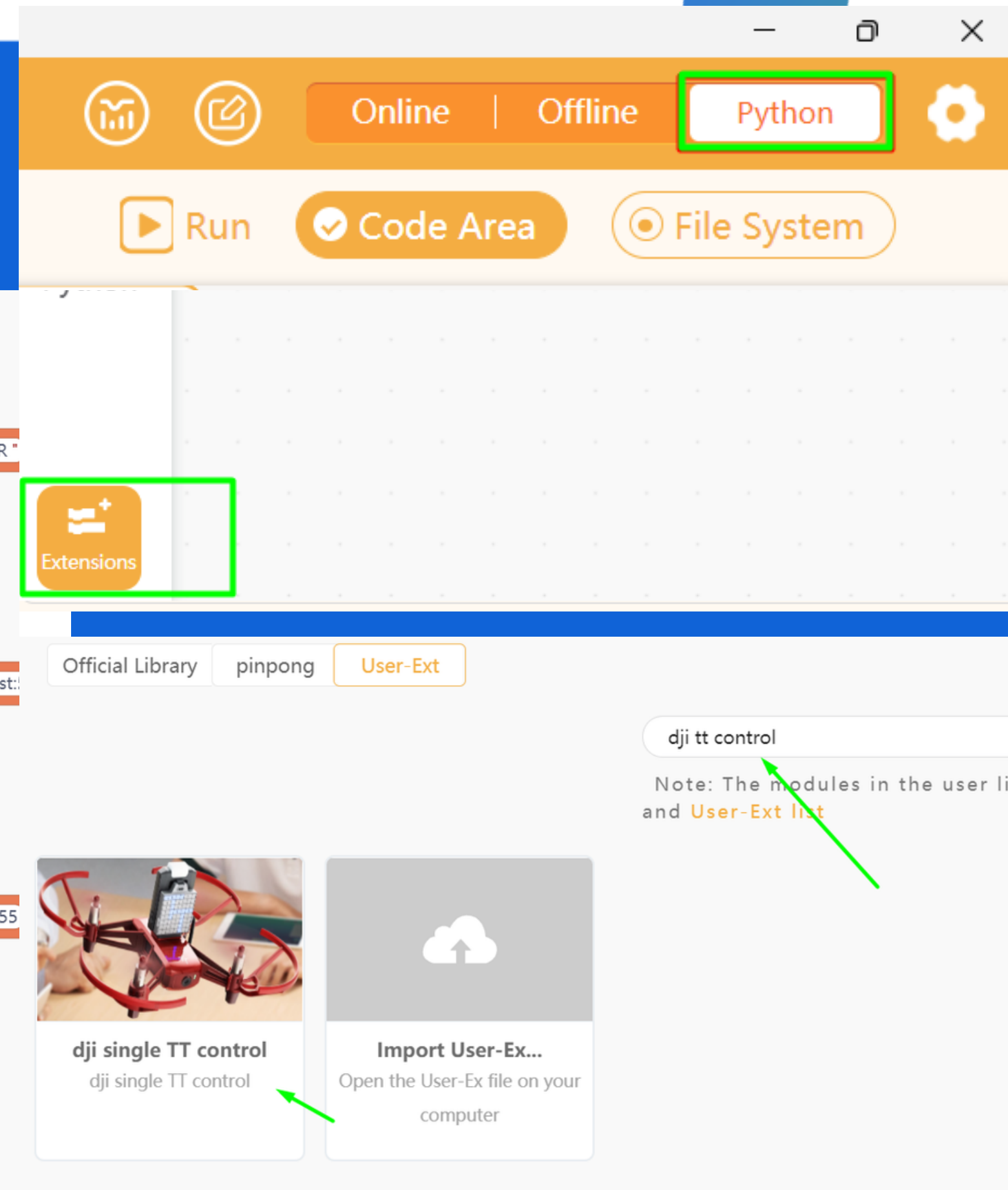
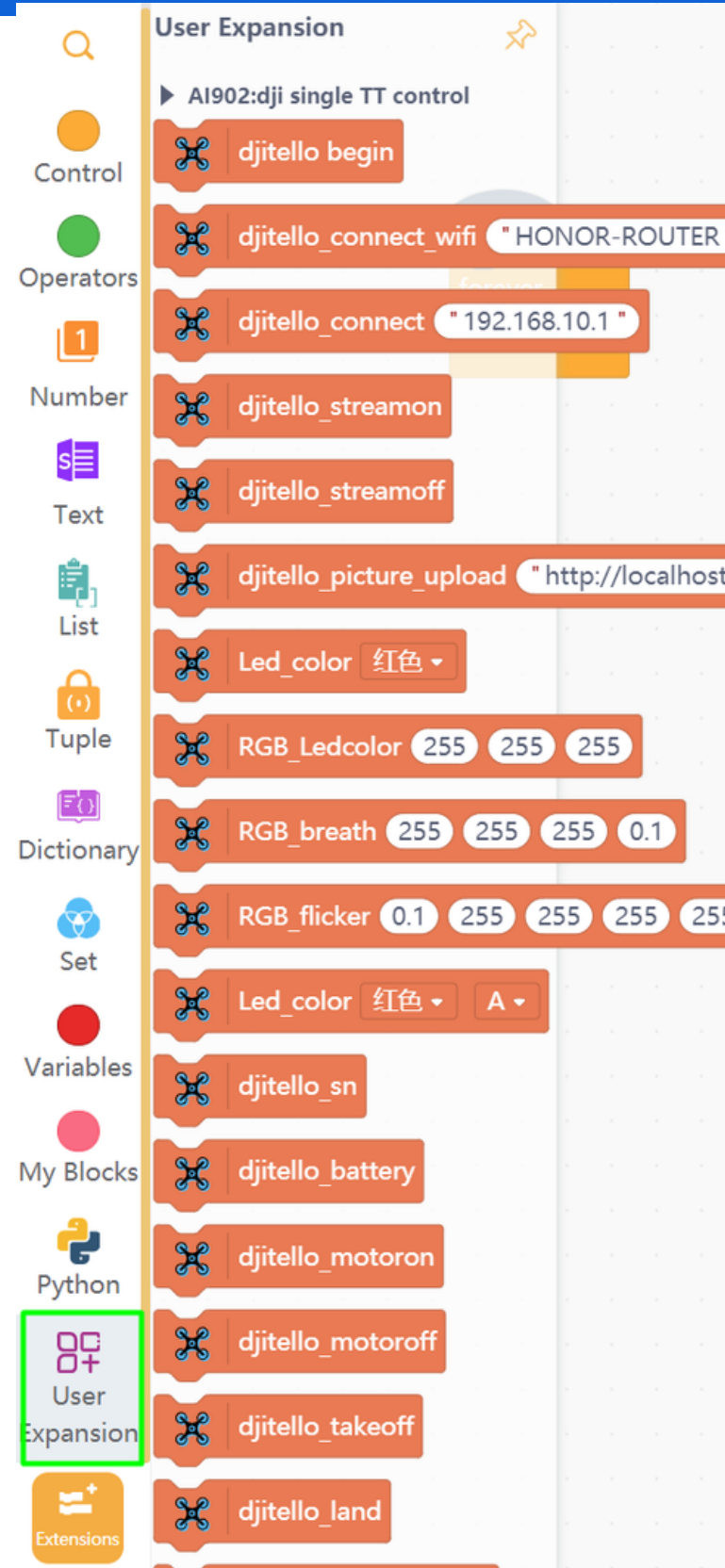
in išči: DJI TT CONTROL in

izberi knjižnico

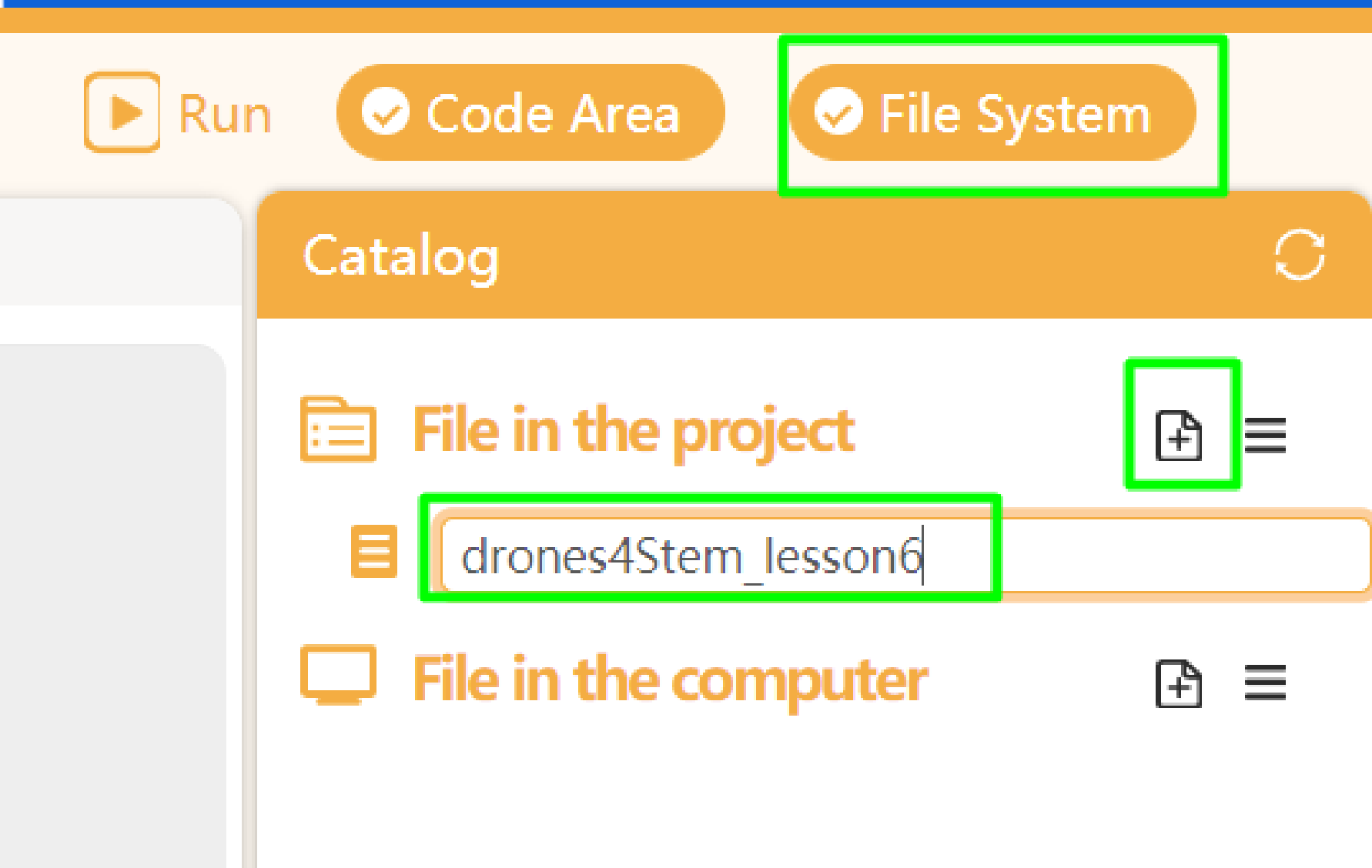
- Blok UPORABNIŠKE RAZŠIRITVE



TELEOTTSpecifične funkcije



Zaženite program v pythonu



- Kliknite Datotečni sistem
- Dodaj datoteko (Datoteka projektu)
- Vnesite me datoteke

Bloki za generiranje kode v Pythonu

MMind+ pretvori bloke v kodo Python

To lahko kopiramo in izboljšamo v načinu KODA.

The screenshot displays the MMind+ KODA interface, which is used for generating Python code from block-based scripts. The interface is divided into several sections:

- Left Panel (Blocks):** Contains a sidebar with various block categories: Control, Operators, Number, Text, List, Tuple, and Dictionary. The main workspace shows a block-based script for a Tello drone.
- Right Panel (Code Area):** Displays the generated Python code. A green box highlights the 'Auto Generate' button, and a green arrow points from the 'djitello_connect' block in the script to the corresponding line in the code.
- Bottom Panel (Catalog):** Shows a file explorer with 'File in the project' and 'File in the computer' sections. The project file 'drones4stem_lesson6.py' is listed.

The block-based script in the workspace includes the following blocks:

- Python main program start
- djitello begin
- djitello_connect (IP: 192.168.10.1)
- djitello_streamon
- djitello_takeoff
- forever loop containing:
 - djitello_challenge
 - djitello_moveforward (Distance: 20)

The generated Python code in the 'Code Area' is as follows:

```
1  # -*- coding: UTF-8 -*-
2
3  # MindPlus
4  # Python
5  import requests
6  from PIL import Image
7  import io
8  from djitellopy import Tello
9  import cv2
10 from djitellopy import TelloSwarm
11
12
13 tello = Tello(host="192.168.10.1")
14 tello.connect()
15 tello.streamon()
16 tello.takeoff()
17
18 while True:
19     tello.enable_mission_pads()
20     tello.set_mission_pad_detection_direction(0)
21     tello.move_forward(20)
22
```

Bloki za generiranje kode v Pythonu

The screenshot displays the Mind+ V1.8.0 RC3.1 interface. The top bar includes the Mind+ logo, 'Project' and 'Learning' dropdowns, and buttons for 'Online', 'Offline', and 'Python'. Below this, a 'Blocks' tab is highlighted with a green box, and a 'Code' tab is also visible. The main editor area shows a Python script for controlling a Tello drone. The right sidebar contains a 'Catalog' panel with 'File in the project' and 'File in the computer' sections. The 'File in the project' section has a file named 'drones4Stem_lesson6.py' highlighted with a green box. The bottom panel is a 'Terminal' window showing the Python 3.8.5 prompt.

Mind+ V1.8.0 RC3.1 new-project-22/3/2025-15:27:50.mp

Project Learning

Online Offline Python

Blocks Code

Save Run Library Management Graphics Area File System

drones4Stem_...

```
1 # -*- coding: UTF-8 -*-
2
3 # MindPlus
4 # Python
5 import requests
6 from PIL import Image
7 import io
8 from djitellopy import Tello
9 import cv2
10 from djitellopy import TelloSwarm
11
12
13 tello = Tello(host="192.168.10.1")
14 tello.connect()
15 tello.streamon()
16 tello.takeoff()
17
18 while True:
19     tello.enable_mission_pads()
20     tello.set_mission_pad_detection_direction(0)
21     tello.move_forward(20)
22
```

Catalog

File in the project

drones4Stem_lesson6.py

File in the computer

Terminal

Python 3.8.5 [MSC v.1924 64 bit AMD64] on win32
Type "help", "copyright", "credits" or "license" for more information.
>>> _

Clear output

Specifikacije talne podloge Drones4Stem



- Vsaka mreža je 40 x 40 cm
 - Gibanje naprej: 100 cm
 - Menjava voznega pasu: 40 cm
- zdrsa
- QR koda 2 je slabo namakanje
 - QR koda 4 označuje slabe pridelke

Initialization and Taking Off

Initialize

```
tello = Tello()
tello.connect()
print(f"Battery: {tello.get_battery()}%")
```

Start video stream

```
tello.streamon()
frame_read = tello.get_frame_read()
```

Enable Mission Pads

```
tello.enable_mission_pads()
tello.set_mission_pad_detection_direction(0) # Downward detection
```

Take off

```
tello.takeoff()
time.sleep(2)
```

Preverite delovanje blazinic za misijo, fotografijo, vrnitev domov

```
# Define helper to check for Pad 4
def check_mission_pad():
    pad = tello.get_mission_pad_id()
    print(f"Detected Pad: {pad}")

    if pad == 4:
        print("Pad 4 detected - taking photo and returning to Pad 1")

        # Take photo
        frame = frame_read.frame
        cv2.imwrite("pad4_photo.jpg", frame)
        print("Photo saved as pad4_photo.jpg")

        # Move to return path
        tello.move_right(80)
        tello.move_back(80)
```

```
# Try to find Pad 1
print("Looking for Pad 1 to land...")
for _ in range(10): # Retry up to 10 times
    pad1 = tello.get_mission_pad_id()
    print(f"Pad ID: {pad1}")
    if pad1 == 1:
        print("Pad 1 found - landing.")
        tello.land()
        tello.streamoff()
        exit()
        time.sleep(1)

print("Pad 1 not found. Landing anyway.")
tello.land()
tello.streamoff()
exit()
```

Vzorec kosilnice

```
# Flight pattern with Pad 4 check
tello.move_forward(120)
check_mission_pad()

tello.move_left(40)
check_mission_pad()

tello.move_back(120)
check_mission_pad()

tello.move_left(40)
check_mission_pad()

tello.move_forward(120)
check_mission_pad()

# End of pattern
print("Pattern complete. Mission Pad 4 not found. Landing.")
tello.land()
tello.streamoff()
```

Varnostna pravila za letenje z droni v zaprtih prostorih

1 Pred poletom:

▪ Preglejte dron

- Preverite, ali so propelerji poškodovani ali zrahljani.
- Prepričajte se, da je baterija popolnoma napolnjena.

▪ Pripravite območje

- Odstranite ovire, kot so stoli, mize in drugi predmeti.
- Zagotovite prostorno in odprto okolje.
- Zaprite okna ali vire zračnega toka

▪ Obveščanje udeležencev

- Vse seznanite z varnostnimi protokoli.
- Ohranjajte varno razdaljo vsaj 2 metra od drona.

Varnostna pravila za letenje z droni v zaprtih prostorih

2 Med letom:

- **Nadzorovano vzletanje in pristajanje**

- Vzletite in pristanite na ravni, stabilni površini.
- Izogibajte se letenju v bližini ljudi ali predmetov.

- **Meje leta**

- Dron imejte v vidnem polju.
- Letite na nizkih višinah, da se izognete udarcu v strop.
- Pristanek, ko je baterija <20 %

- **Protokol za nujne primere**

- Če dron izgubi nadzor, takoj ustavite. (VESOLJE sproži PRISTANAK)
- Po potrebi uporabite gumb za »zaustavitev v sili« v aplikaciji. (»e« ustavi motorje)

Varnostna pravila za letenje z droni v zaprtih prostorih

3 Po letu:

- **Varno izklopi**

- Izklopite dron in odklopite baterijo.

- **Preverite morebitne poškodbe**

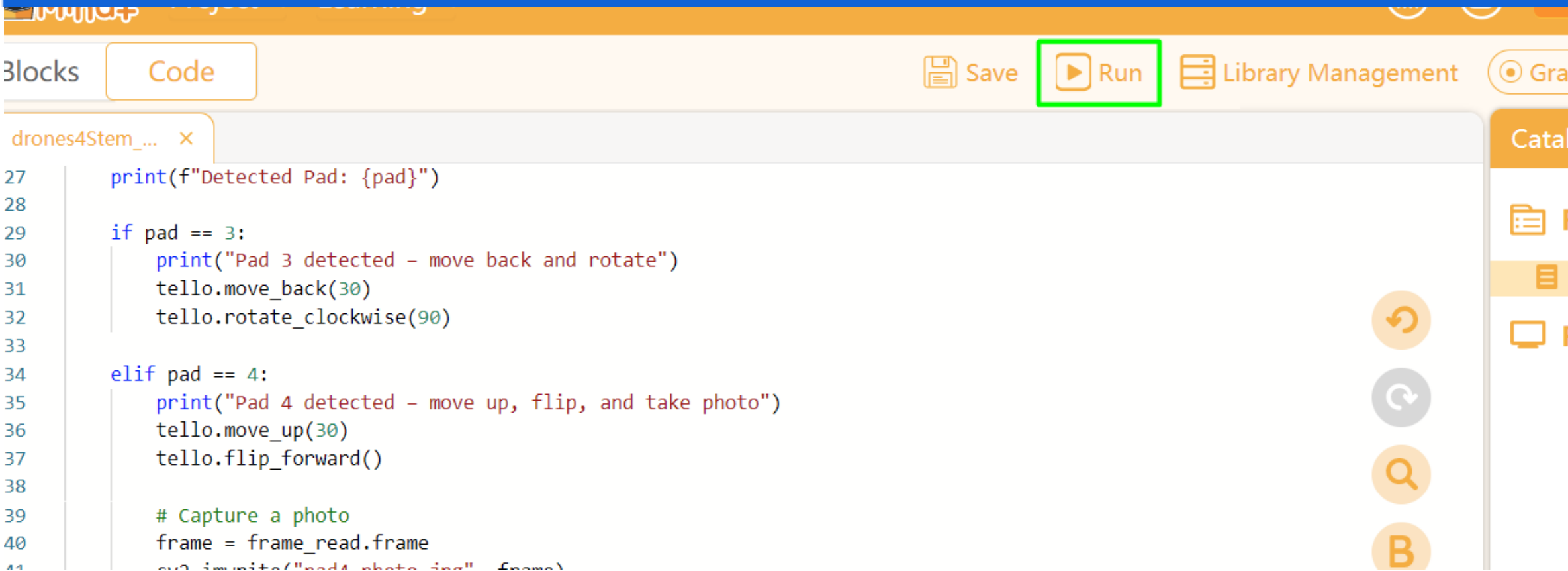
- Po letu preverite morebitne težave.

- **Varno shranjujte**

- Dron hranite na varnem in suhem mestu.

VEDNO OSTANITE MIRNI IN OSREDOTOČENI

Preizkusi program

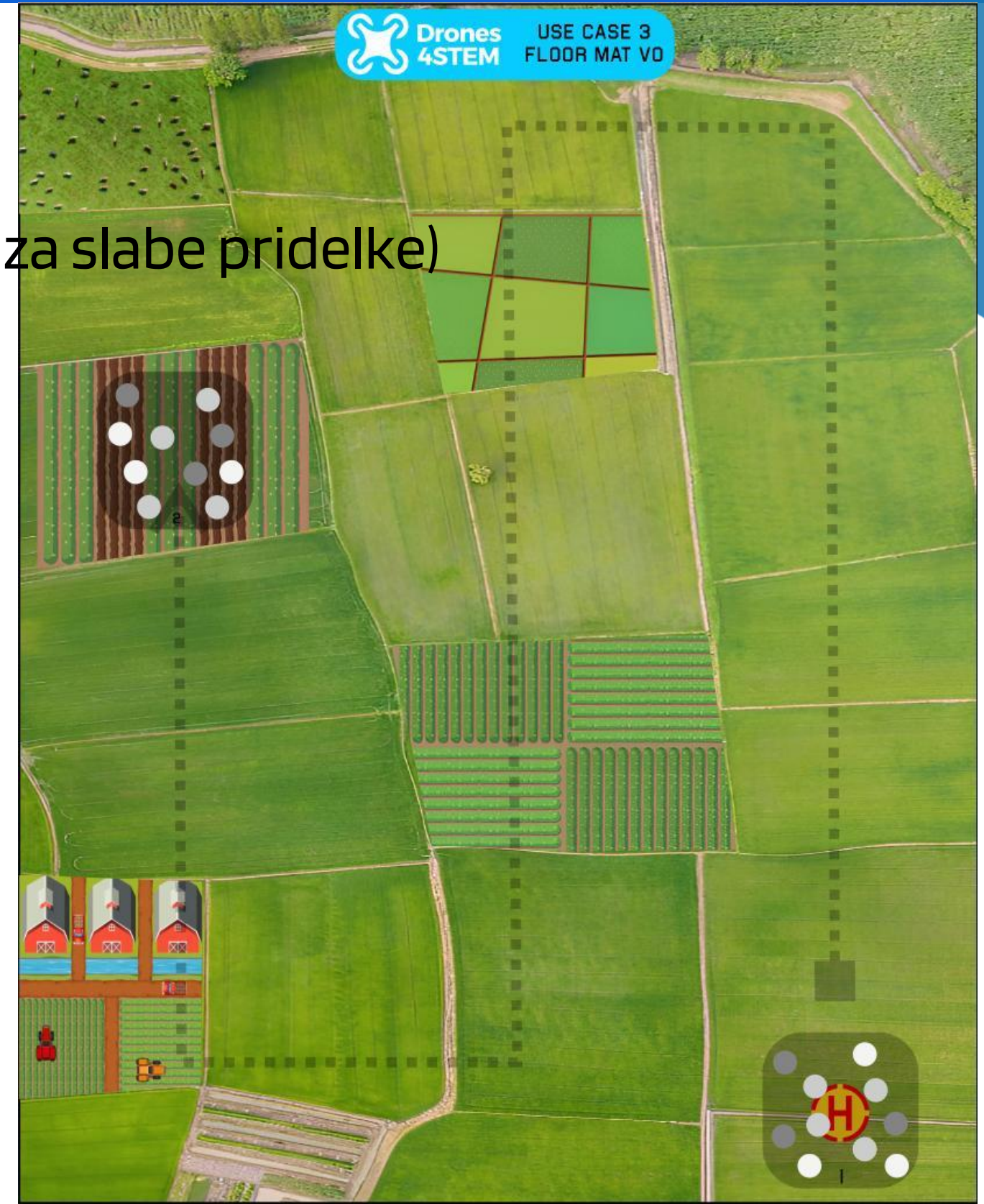


```
27 print(f"Detected Pad: {pad}")
28
29 if pad == 3:
30     print("Pad 3 detected - move back and rotate")
31     tello.move_back(30)
32     tello.rotate_clockwise(90)
33
34 elif pad == 4:
35     print("Pad 4 detected - move up, flip, and take photo")
36     tello.move_up(30)
37     tello.flip_forward()
38
39     # Capture a photo
40     frame = frame_read.frame
41     cv2.imwrite("pad4_photo.jpg", frame)
```



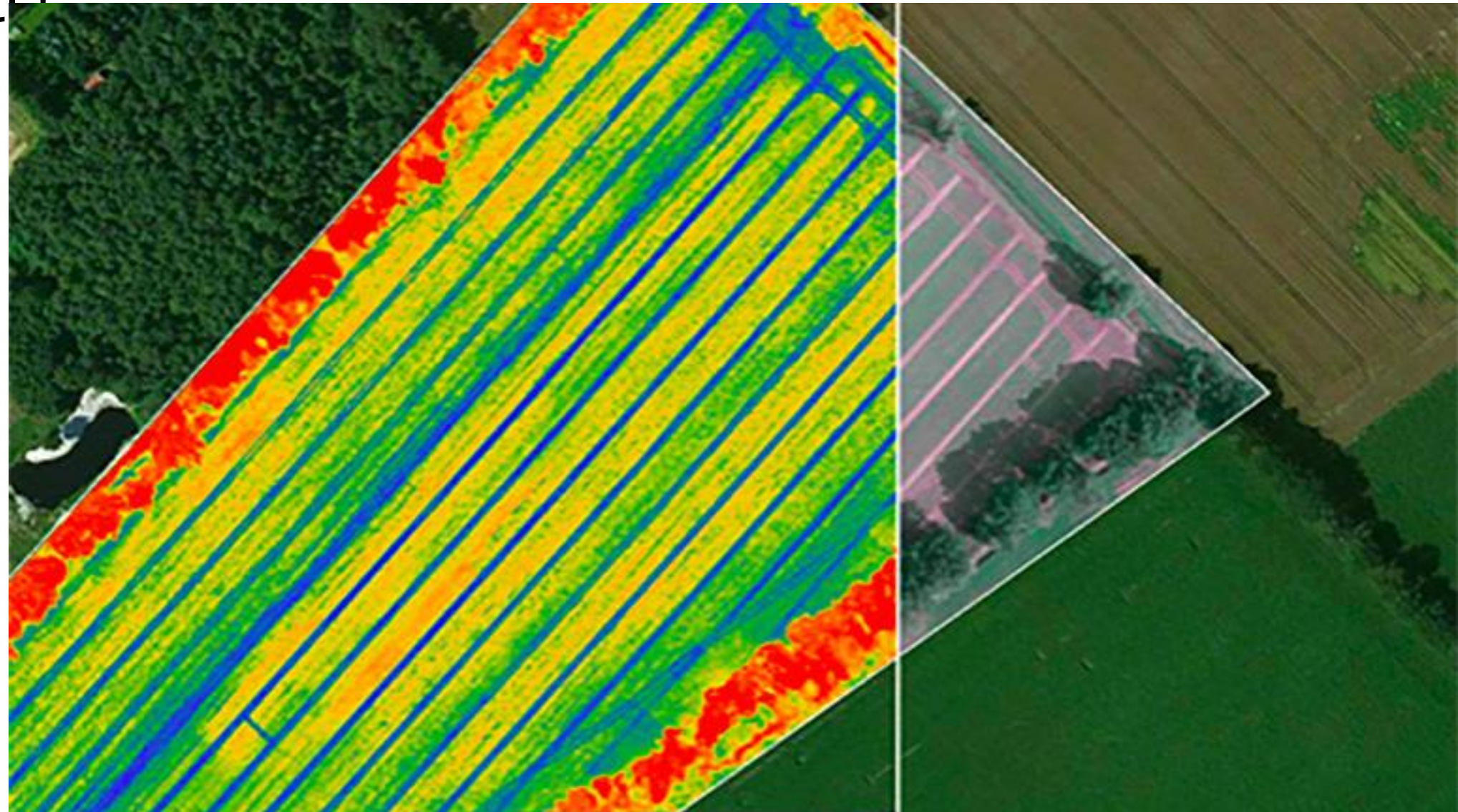
Izboljšave in omejitve

- Kaj se bo zgodilo, če Mission Pad ni prepoznan?
- Ali lahko dodate kakšen vzorec z LED diodami (npr. rdečo za slabe pridelke)
- Kaj se zgodi, če se baterija izprazni?
- Zakaj dron poveča višino pred fotografiranjem?



Osnovna analiza slik

- Zdravi pridelki: Svetlo zelene barve.
- Problematična področja: rumene ali rjave lise, venenje listov.
- Težave s tlemi: Suha ali razpokana tla



POVZETEK

- **Kaj smo se naučili:**

- Zakaj je dron pomemben v pametnem kmetijstvu.
- Kako namestiti modul Python.
- Kateri so osnovni ukazi za inicializacijo DJITello s pythonom.
- Kako narediti funkcijo za branje misijskih blazinic.
- Kako narediti pot za kosilnico.

- **Varnost na prvem mestu:**

- Pomen čiščenja območja za letenje.
- Ohranjanje varnega okolja za vse.



Dodatni viri

Nadaljnje branje

- <https://djitellopy.readthedocs.io/en/latest/tello/>

Videos

[How to install python module](#)



Hvala za vašo pozornost!



Co-funded by
the European Union

Funded by the European Union. Views and opinions expressed are however those of the author(s) only and do not necessarily reflect those of the European Union or the European Education and Culture Executive Agency (EACEA). Neither the European Union nor EACEA can be held responsible for them.