UAV Ground Detection

Distinguish Tree Species

Christian Ekeigwe, Daehyeon Jeong, Jaeyeong Shim, Jeonghwan Kang, Seoungheong Jeong

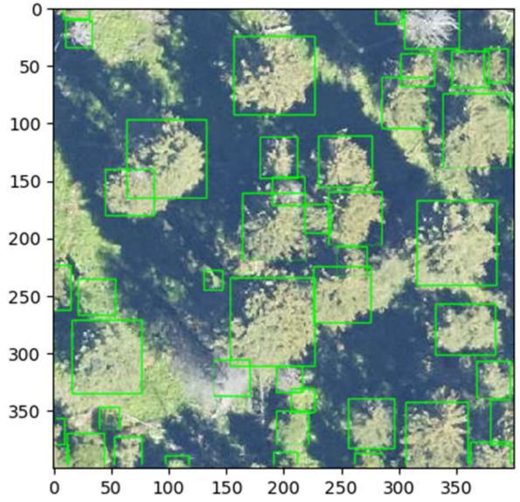
Project 17

PURDUE UNIVERSITY | Polytechnic Institute

Detecting Tree model

Deep Forest

- Training and predicting individual tree airborne RGB image



https://deepforest.readthedocs.io/en/latest/landing.html

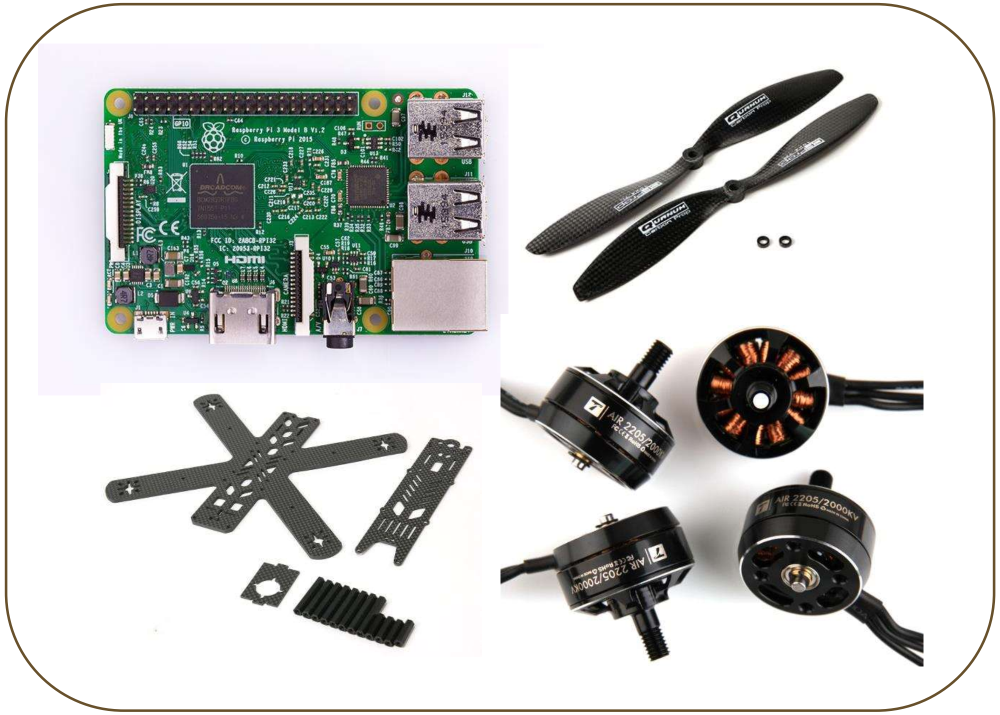
Collecting data using UAV

Collecting data using UAV

- But we think it is hard to use UAV in Korea forest and gathering data.
- We have some several problems
- So, we try to find some new method.

Solution 1

What about UAV?



We have to build drones

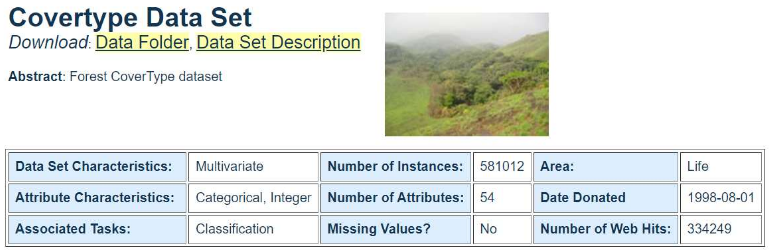
Solution 2

How do we collect map data?

Google Maps Platform

kaggle

UCI Machine Learning Repository



Covertype Data Set

Download: Data Folder, Data Set Description

Abstract: Forest CoverType dataset

Data Set Characteristics:	Multivariate	**Number of Instances:**	581012	**Area:**	Life
Attribute Characteristics:	Categorical, Integer	**Number of Attributes:**	54	**Date Donated**	1998-08-01
Associated Tasks:	Classification	**Missing Values?**	No	**Number of Web Hits:**	334249

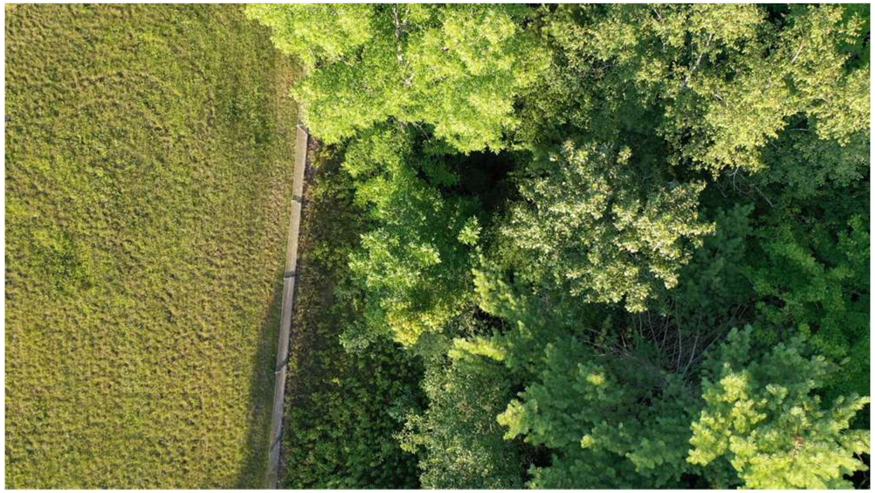
From Google Map API

External Dataset

PURDUE UNIVERSITY | Polytechnic Institute

Data

The video capture shot we get from Purdue



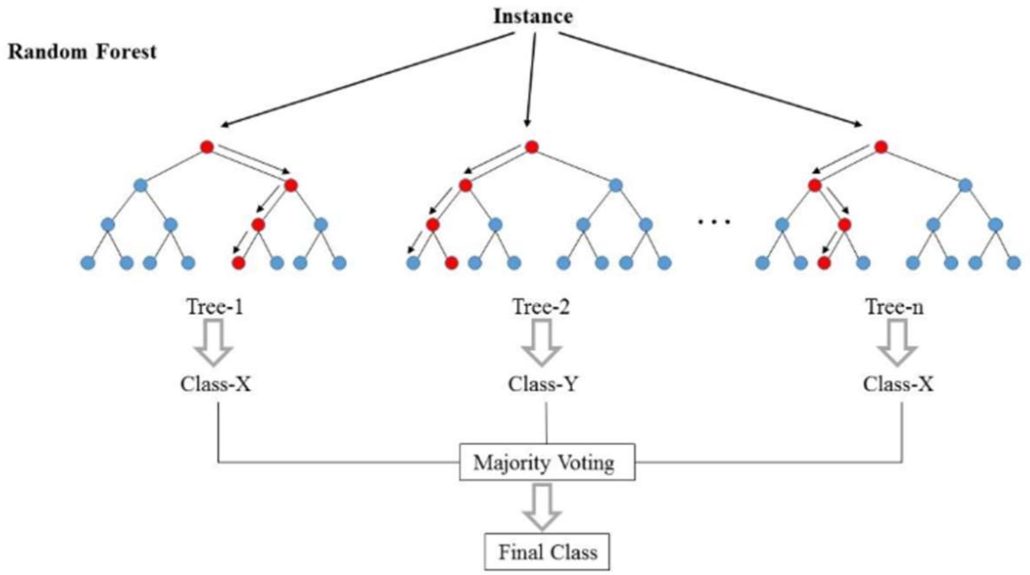
Plan

Step 1	Get data from video	Make modeling system
Step 2	Classify data (Distinguish Tree Species)	Modeling data
Step 3	Real time receive data from UAV	Add/Del data from GEO chart

Step 2

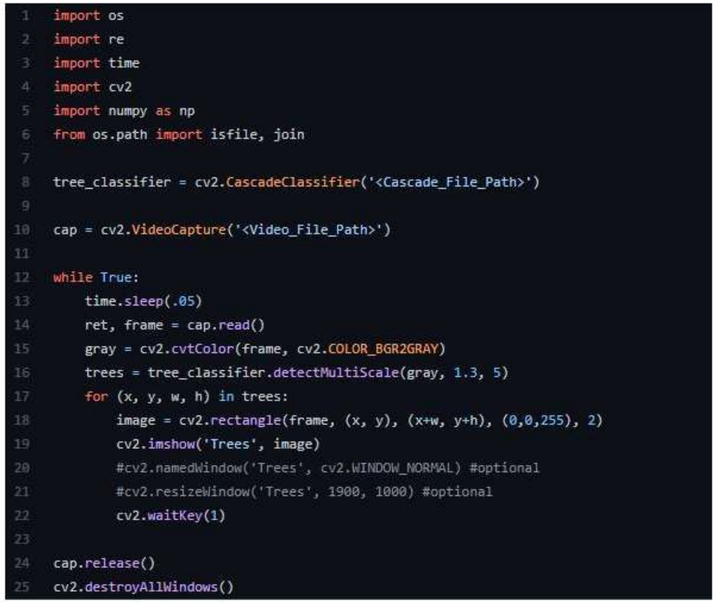
Classify data
(Distinguish Tree Species)

Modeling data



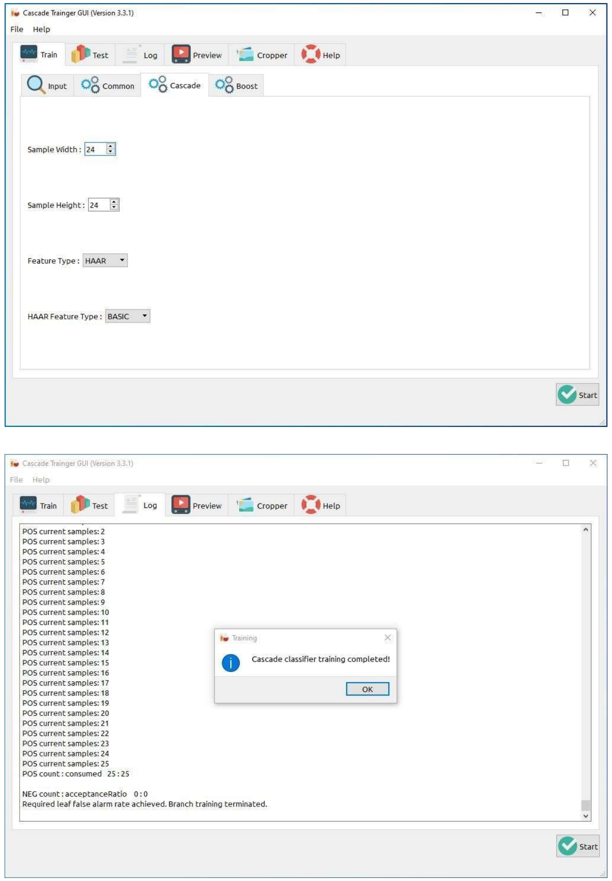
Classification

Classify Data



```python
import os
import re
import time
import cv2
import numpy as np
from os.path import isfile, join

tree_classifier = cv2.CascadeClassifier('<Cascade_File_Path>')

cap = cv2.VideoCapture('<Video_File_Path>')

while True:
    time.sleep(.05)
    ret, frame = cap.read()
    gray = cv2.cvtColor(frame, cv2.COLOR_BGR2GRAY)
    trees = tree_classifier.detectMultiScale(gray, 1.3, 5)
    for (x, y, w, h) in trees:
        image = cv2.rectangle(frame, (x, y), (x+w, y+h), (0,0,255), 2)
        cv2.imshow('Trees', image)
        #cv2.namedWindow('Trees', cv2.WINDOW_NORMAL) #optional
        #cv2.resizeWindow('Trees', 1900, 1000) #optional
        cv2.waitKey(1)

cap.release()
cv2.destroyAllWindows()
```

Detecting images from video



Sample Width : 24

Sample Height : 24

Feature Type : HAAR

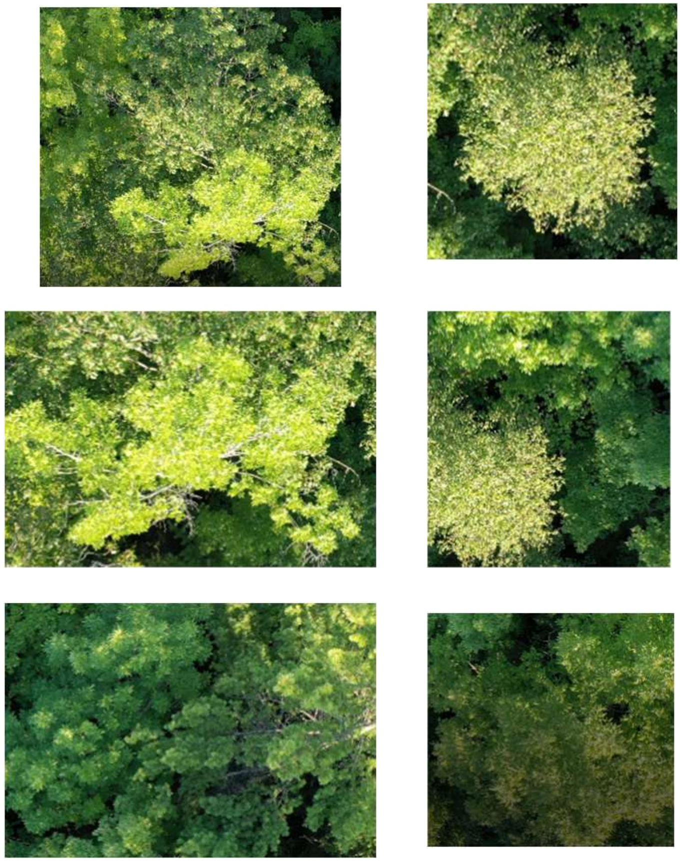
HAAR Feature Type : BASIC

```
POS current samples: 2
POS current samples: 3
POS current samples: 4
POS current samples: 5
POS current samples: 6
POS current samples: 7
POS current samples: 8
POS current samples: 9
POS current samples: 10
POS current samples: 11
POS current samples: 12
POS current samples: 13
POS current samples: 14
POS current samples: 15
POS current samples: 16
POS current samples: 17
POS current samples: 18
POS current samples: 19
POS current samples: 20
POS current samples: 21
POS current samples: 22
POS current samples: 23
POS current samples: 24
POS current samples: 25
POS count : consumed   25 : 25

NEG count : acceptanceRatio    0 : 0
Required leaf false alarm rate achieved. Branch training terminated.
```

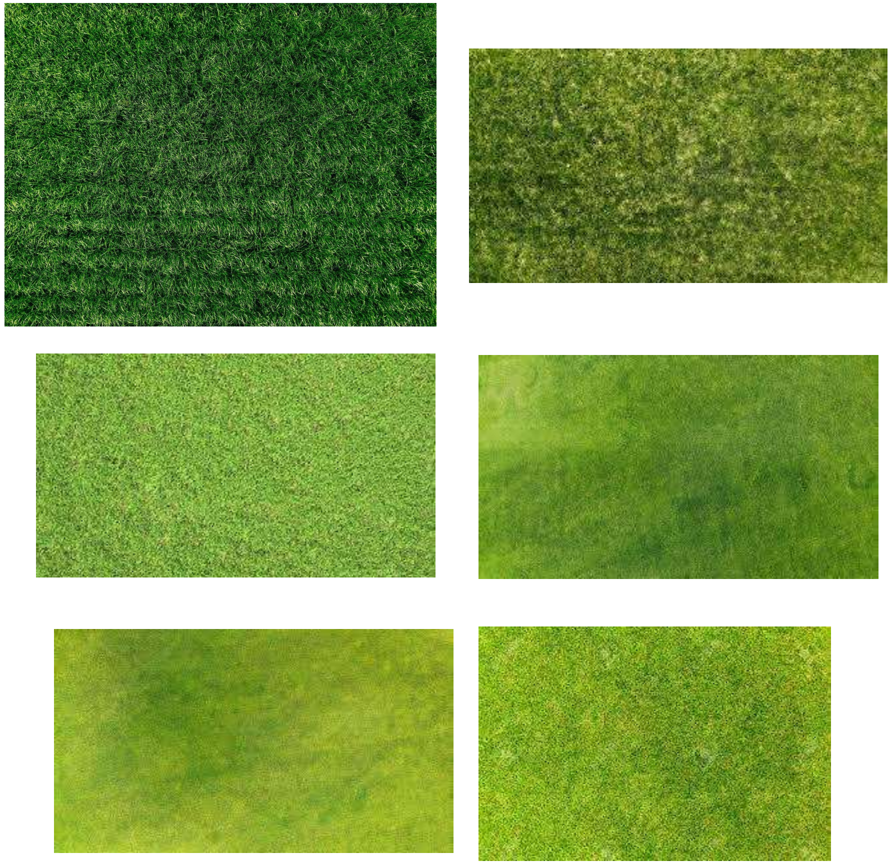
Cascade classifier training completed!

CascadeTrainer

Image Labeling



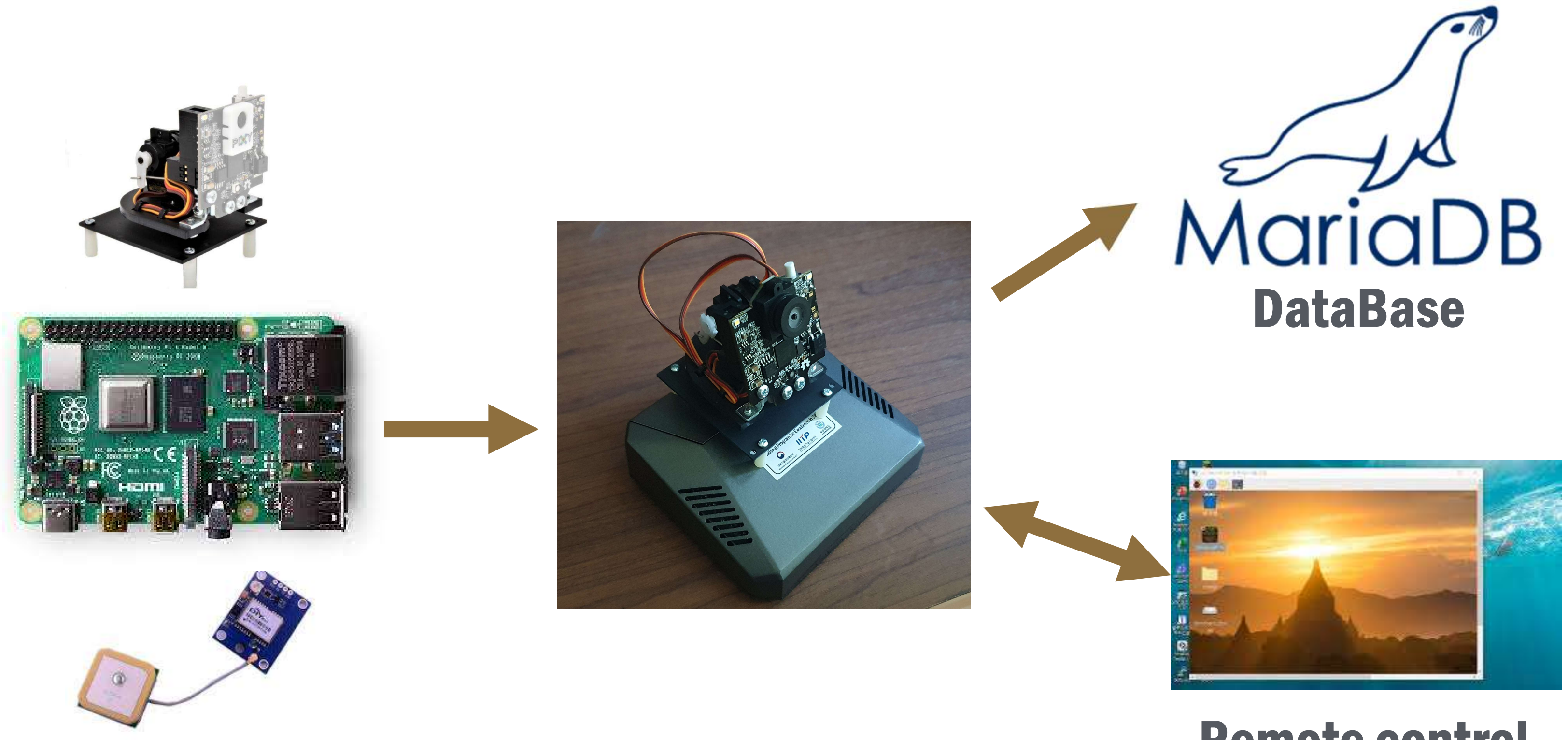
Positive



Negative

NEXT

Real time communication

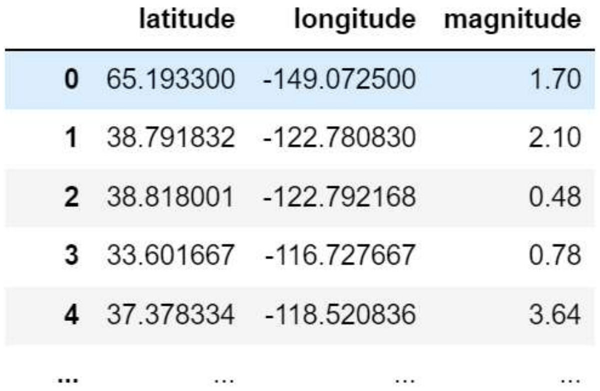


DataBase

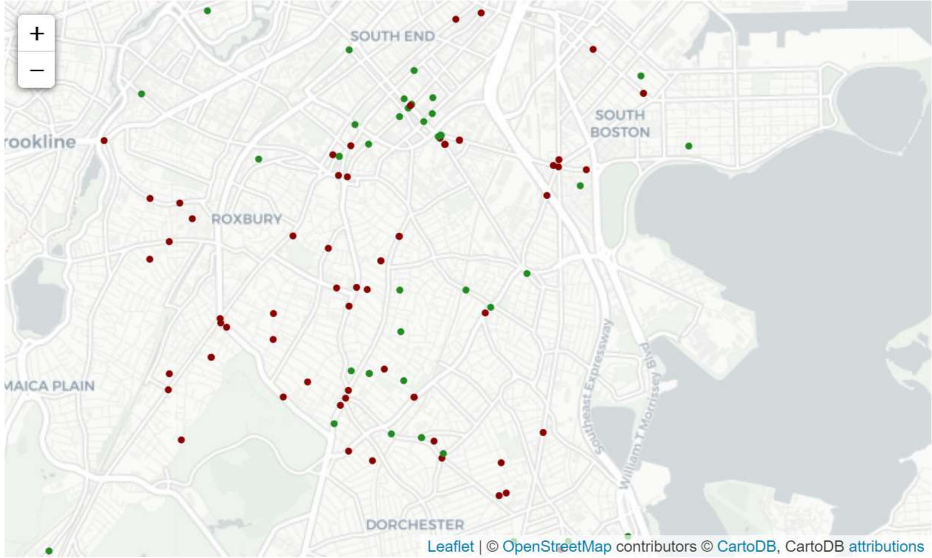
Remote control

Step 3

Data Visualization



	latitude	longitude	magnitude
0	65.193300	-149.072500	1.70
1	38.791832	-122.780830	2.10
2	38.818001	-122.792168	0.48
3	33.601667	-116.727667	0.78
4	37.378334	-118.520836	3.64
...	...	...	...



Leaflet | © OpenStreetMap contributors © CartoDB, CartoDB attributions

Marker on Map

Thank you

Questions?

polytechnic.purdue.edu

/ TechPurdue

PURDUE UNIVERSITY | Polytechnic Institute