

# 해야할 일

1. 객체인식, 거리계산 (적외선 카메라)
2. 원하는 위치로 가길
3. 최적의 경로로 쫓아내기 (농작물 피해↓: 각도 조절or강화학습)
4. 방해물 피하기
5. 농가 맵정보

# 1. 객체인식

## (1) 인식해야할것

해로운동물 : 한국에서 거기서 거기일것. 미리 학습시키기

사람 : 안전사고 절대안됨. 추돌피하고 소유자에게 연락 보내기

기타장애물 : 새, 농기계, 나무 등. 피해서가야함

( 농지의 장점 : 장애물이 그렇게 많지않음)

# 적외선 카메라

>

열감지 카메라. 감지가능 온도 범위에 따라 가격차가 좀 있음.  
낮/밤/비바람 관계없이 확실히 객체인식 가능. 거리계산센서있음



# 적외선 카메라



문제 1 : 야생동물을 적외선카메라로 찍은 DATASET 이 있을까?  
찾아보고 없다면 만들어야할수도있음.

문제 2 : 봄/여름/가을/겨울 마다 약간 다르게 나올것같은데 (지열등)  
CYCLE GAN을 이용할수도있다.

CYCLE GAN : 이미지 생성 DNN. 두 영역의 이미지들이 있으면 한 이미지를 다른 영역의 이미지로 만들어낼수있음. 형태분야에있어서는 성능이 아쉬우나 색변화는 뛰어나게 잘해냄 (말<->얼룩말)

# 객체인식 머신러닝

1) Semantic Segmentation

ICNET (cnn이용)

<https://www.youtube.com/watch?v=gWl9idsCuLQ>

2) YOLO

3) SSD



<http://m.edwith.org/deeplearningchoi/lecture/15579>

# 거리계산

적외선카메라를 쓰면 센서에 의해 자동으로 가능. 스테레오 카메라를 쓰면 스테레오 매칭 (비전) 기술을 이용하여 가능.

두경우 모두 정확한 거리를 알수는 없다는 단점



어차피 쫓아내면 되므로 정확한 거리보다는 각도가 중요해보임  
쫓아낼 객체의 모양을 바탕으로 자신이 대략 어느 벡터에 퍼있는지  
추측해서 쫓아낼 방향을 생각해서 자리 잡은 후 소리등으로 쫓아냄

# 남은문제

실시간으로 원하는위치로 가다가 급작스레 화면에 장애물이 잡힐경우 알고리즘 스탱, 피하는 알고리즘 작동  
이런식으로 작동?

실제 자율주행 드론 한번 돌려보기

I

일단 야생동물/ 사람/ 기타 농가에있을만한 장애물 인식하는 객체인식 NN을 짜보고 계속 추가하기

# Red Trail

# RedTrail

I

1) TrailNet

2) 물체감지

3) 저수준의 장애물감지를 위한 깊이 측정, 시각적 주행거리 측정 포함 환경인식을 위한 비전모델

4) 모든 비전프로그램은 Jetson TX1을 통해 MAV에 실시간 실행됨

<https://arxiv.org/abs/1705.02550>

<https://github.com/NVIDIA-Jetson/redtail/wiki>



# NodeCopter

# nodecopter

node-ar-drone 라이브러리 위에 구축된 ARDrone 용 자율 비행 라이브러리. 무인 항공기를 직접 제어하는 대신 Autonomy를 사용하여 무인 항공기가 따라야 하는 경로, 고도 및 방향을 설명하여 임무를 계획하고 실행할 수 있음. 즉, 가고 싶은 곳에 초점을 맞추면, 라이브러리는 드론을 그곳으로 데려감

<https://github.com/eschnou/ardrone-autonomy>

# Yolo Drone

## # YOLO Drone

YOLO Deep Network의 도움으로 현지화하고 사람들을 추적, 흔히 단 한 사람 이상이 무인 항공기 사진에있을 수 있으므로 표준 깊은 학습 인물 / 신체 인식으로는 충분한 결과를 얻을 수 없어서, 셔츠의 색상을 두 번째 기준으로 선택. (뚜렷한 황색 셔츠입은사람 쫓아감)

<https://github.com/durner/yolo-autonomous-drone>

drone.py : 프로젝트의 메인 파일, 수동 인터페이스, YOLO Network와 액추에이터 사이의 자율 인터페이스에 대한 코드를 포함합니다. 모든 멀티 스레딩 및 OpenCV 사전 처리가 처리됩니다.

PID.py : 단순한 PID 컨트롤러 인터페이스로 무인 항공기의 움직임을 쉽게 제어 할 수 있습니다 (무브먼트를 포함).

YOLO.py : Python에서 YOLO Deep 네트워크를 설정합니다. 하위 폴더 utils에는 YOLO net에 필요한 추가 파일이 포함됩니다.

actuators.py : 지역화 된 작업자의 도움으로 액추에이터는 무인 항공기를 조작자 중심으로 이동시켜야하는지 계산합니다. 움직임을 계산하기 위해 PID 컨트롤러를 사용합니다.