GUJARAT TECHNOLOGICAL UNIVERSITY

L. D. COLLEGE OF ENGINEERING

ELECTRONICS & COMMUNICATION ENGINEERING DEPARTMENT

BIRDS DETECTION AND REPELLING SYSTEM

KACHHADIYA JEMIN(160280111032), MANSURI ADIL(160280111043),PANDYA HET(160280111050)
Guided By. Prof. ABHAY UPADHYAY

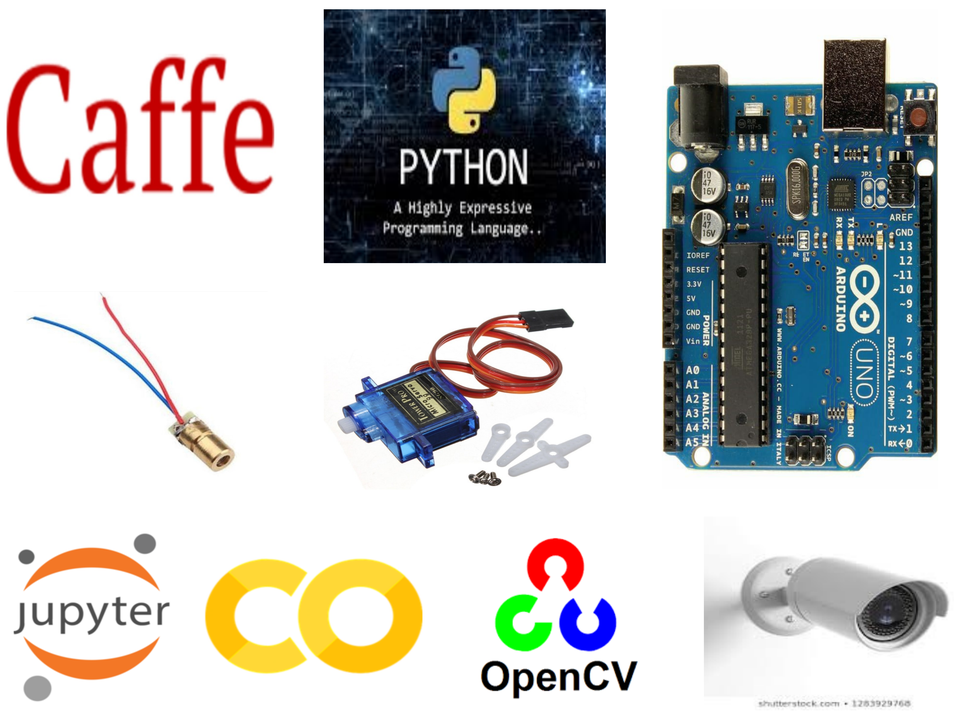
Abstract

The electronic system designed using Fast Image processing and arduino, servo etc which can help to repel the birds away from Engine fan of aircrafts, farms for protect crops from damages caused by birds, commercial wind mills, exhaust fans, etc. For solving problem like airplane accident due to bird hitting, bird hitting at windmill etc. Here we use caffe model for image processing and machine learning part with help of python and arduino, servo, laser as hardware part for repelling birds.

How we are Unique?

There are various precautions are being taken by the respective officials. But we use image processing with machine learning so that we can get exact output as location of bird in frame and then according to that location we will emit laser which help us to repel birds. So with this accuracy result we can say that we are unique and only one who made system like this. System works on caffe model which help us to detect the bird very precisely so it emit laser only when bird will detect with use of two servo we set value of X-Y axis for bird location.

Major software and Hardware Used



Future scope and implementations

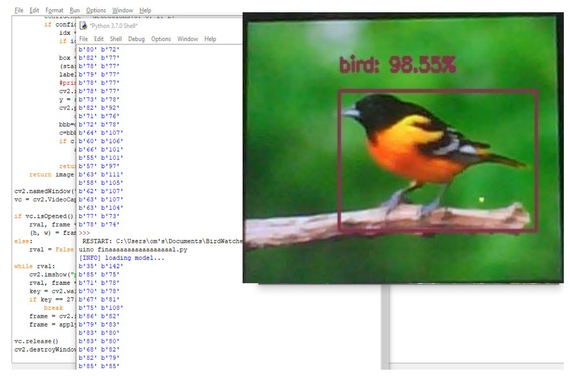
1.We can generate particular frequency by detecting particular bird species.
2. Object of the invention to provide engine guards that can be easily connected to existing aircraft engine housings.
3. This invention is directed to an aircraft engine guard, for protecting an aircraft engine against ingestion of large objects, the guard comprising: a generally cone-shaped body.

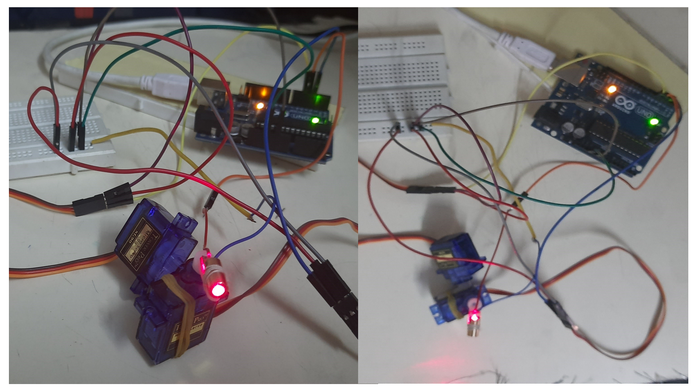
Applications

1. The main application of this project is in the airplane and windmill. Because by analytical study report we found lot of birds are being killed at California.
2. This can also be implemented at artificial flying objects such as air crafts so birds will not hit them and are being saved.
3. This system is also implemented at farms so that the birds which degrades the crops will be saved and this system will only repel the birds by laser so that they would not be harmed'

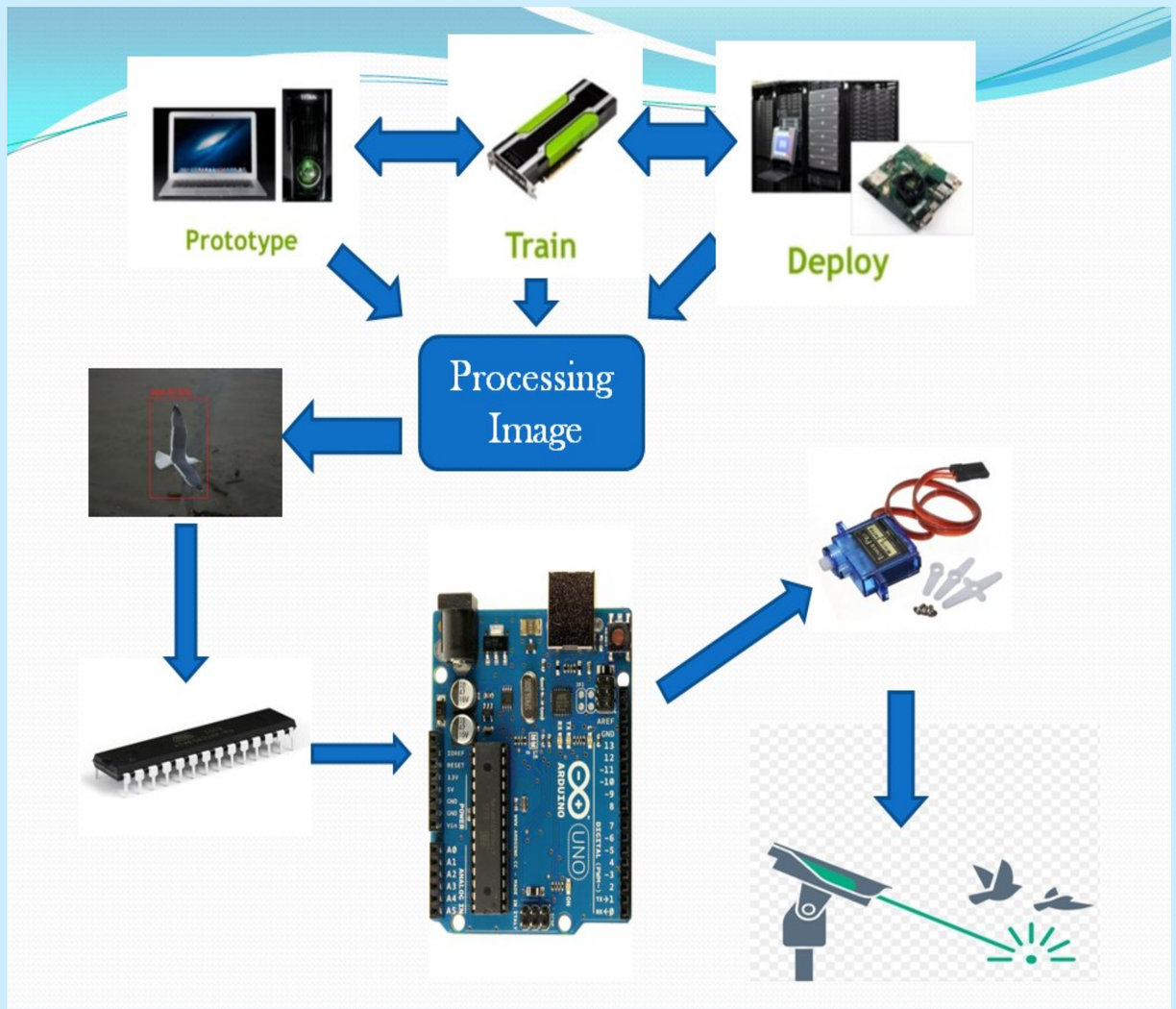
Birds detection and repellent

```
import cv2
from datetime import datetime


confidence_thr = 0.5
arduinodata = serial.Serial('com5',9600)

CLASSES = ["background", "aeroplane", "bicycle", "bird", "boat",
    "bottle", "bus", "car", "cat", "chair", "cow", "diningtable",
    "dog", "horse", "motorbike", "person", "pottedplant", "sheep",
    "sofa", "train", "tvmonitor"]
COLORS = np.random.uniform(0, 255, size=(len(CLASSES), 3))

print("[INFO] loading model...")
ggg=None
shared_dir = '../../MobileNet-SSD/'
net = cv2.dnn.readNetFromCaffe(shared_dir+ 'deploy.prototxt' , shared_dir+ 'mobilenet_iter_73000.caffemodel')

def arduino(startX, startY, endX, endY):
    global ggg
    finalX= (startX+endX)/2
    finalY= ((480-startY)+(480-endY))/2
    Xvalue= str(int(0.1875*finalX)+30)
    Yvalue= str(int(0.25*finalY)+30)
    arduinodata.write(Xvalue.encode())
    print(Xvalue.encode(),Yvalue.encode())
    ggg=datetime.now()

arduino(30,30,30,30)

blob=None
def applySSD(image):
    global blob
    blob = cv2.dnn.blobFromImage(cv2.resize(image, (300, 300)), 0.007843, (300, 300), 127.5)
    net.setInput(blob)
    detections = net.forward()
    for i in np.arange(0, detections.shape[2]):
        confidence = detections[0, 0, i, 2]
        if confidence > confidence_thr:
            idx = int(detections[0, 0, i, 1])
            if idx!=3:
                continue
            box = detections[0, 0, i, 3:7] * np.array([w, h, w, h])
            (startX, startY, endX, endY) = box.astype("int")
            label = "{}: {:.2f}%".format(CLASSES[idx], confidence * 100)
            #print("[INFO] {}".format(label))
            cv2.rectangle(image, (startX, startY), (endX, endY),COLORS[idx], 2)
            y = startY - 15 if startY - 15 > 15 else startY + 15
            cv2.putText(image, label, (startX, y),
                cv2.FONT_HERSHEY_SIMPLEX, 0.8, COLORS[idx], 2)
            bbb=datetime.now()
            c=bbb-ggg
            if c.total_seconds()>1.5:
                arduino(startX, startY, endX, endY)

            return image
    return image

cv2.namedWindow("preview")
vc = cv2.VideoCapture(0)

if vc.isOpened():
    rval, frame = vc.read()
    (h, w) = frame.shape[0] , frame.shape[1]
else:
    rval = False

while rval:
    cv2.imshow("preview", frame)
    rval, frame = vc.read()
    key = cv2.waitKey(20)
    if key == 27:
        break
    frame = cv2.flip(frame,1)
    frame = applySSD(frame)

vc.release()
cv2.destroyWindow("preview")
```





Architecture Diagram



Reference

1. https://github.com/soeaver/caffe-model
2. https://link.medium.com/K1SMdaAEk5
3. https://youtu.be/rvMVqPsXL10
4. https://www.pyimagesearch.com/2017/09/11/object-detection-with-deep-learning-and-opencv/
5. https://www.coursera.org/specializations/python
6. https://en.wikipedia.org/wiki/Caffe_(software)

Acknowledgment

We express our gratitude to **Prof. Abhay Upadhyay** for his constant encouragement, co-operation and support and also giving the knowledge he have about our project for making it better than we thought.