

An Intelligent Bird-repellent Device Based on Raspberry Pi

Xinyu Ma^{1, a*} and Chang Chen^{2, b}

North China Electric Power University, Beijing, 102206

^amaxinyu@ncepu.edu.cn, ^bchenchang@ncepu.edu.cn

*The Corresponding author

Keywords: Raspberry pi; Python; Ultrasonic; Picture processing; Frame difference method

Abstract. The paper mainly discusses the invention of an intelligent bird-repellent device based on the Raspberry PI—an open source hardware platform. Based on Raspberry Pi development board, embedded system is used. Python language is used based on Linux system. By controlling of the GPIO pin, detection of ultrasonic distance is realized. In addition, web camera is used to collect images and grayscale images and apply a Gaussian blur. Frame differential method is used to ensure the movement of objects.

Introduction

According To Information Statistics, after lightning stroke and external damage, bird-trouble is the third reason for transmission line malfunction.

Compared with trees, transmission tower's structure is more stable, and it will not be easily shaken by rain and snow storm. So birds prefer to nest on the tower. If it rains, nests built with branches might be blown onto the charged wires or hanging bottles. Branches touch or stay close to the charged wire will cause short circuit grounding accident. Then, the whole lines will have the hidden trouble of tripping operation, which brings massive influence on people's production and normal life. Although some kinds of birds don't nest on transmission tower, they rest on the cross arm of the tower. Their excreta will pollute insulator chains. When the weather is humid or foggy, flashover accident will happen easily.

In order to solve the problem, considering bird habits and other bird-repellent devices, we invent an intelligent bird-repellent device specially used in power system.

Hardware Platform and Related Software

Raspberry Pi. Raspberry Pi is a kind of new microcomputer. It uses ARM11 architecture, and its size only equals with a credit card. It has a powerful system and interface resources. Raspberry Pi can achieve most functions of traditional computers only by being connected to a display and a keyboard, such as word processing, image processing, etc. Furthermore, it is attractive in price and quality. Therefore, it has been widely used in advanced DIY development all over the world.

Python—Raspberry Pi Development Language. Python is the first carrying programming development environment of Raspberries Pi. It is an object-oriented, literal translation type computer programming language. It contains a set of perfect and understandable standard library which allow users to do a lot of common things easily. What's more, it can also realize automatic memory management. So it is often used to deal with advanced tasks such as system administration tasks and network programming.

OpenCV—Raspberry Pi Image Processing. The full name of OpenCV is Open Source Computer Vision, which is a cross-platform library Computer Vision based on Open Source distribution. It can be run on the Linux operating system, Windows and Mac OS. Powerful OpenCV can implement many of the general image processing algorithm through programming languages such as python language.

Intelligent Bird-repellent Device

Ultrasonic Ranging Module

(1) Hardware Setting

The following Hardware is needed:

Ultrasonic Ranging Module, 1k Ω and 2k Ω resistance, bread board.

Hardware connection:

- Connect the Raspberries Pi 6th pin and ground wire to the bread board
- Connect the Raspberries Pi 2th pin and sensor's 5V power supply pin
- The trigger pin can be directly connected to the sensor, actually Raspberry Pi will send a **V signal to the trigger pin, and the signal can be received by the sensor.**
- Link the Raspberries Pi 18th pin with bread board and connect it to the terminal of Raspberry Pi. On the same line, connect one of the 2 k Ω resistance terminal and the other terminal is connected to on the bread board with the same line of Raspberry Pi and sensor ground terminal. The connection diagram is as follows:

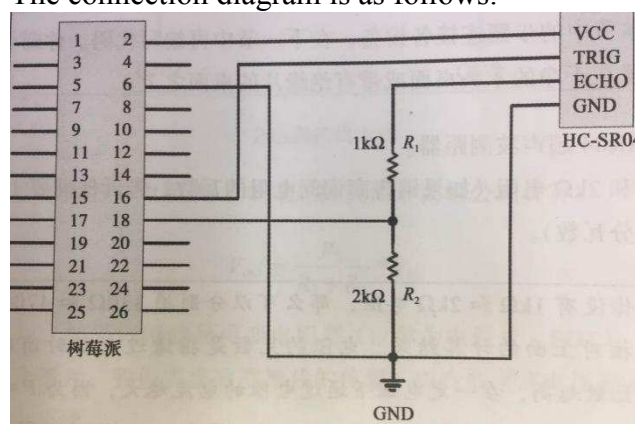


Figure 1. Connection diagram

(2) Software Setting

- First, we need to install or invoke the installed GPIO and time library:

```
import RPi.GPIO as GPIO
import time
GPIO.setmode(GPIO.BCM)
```

- Set the GPIO pins to the default mode:

```
GPIO.setup(23,GPIO.OUT)
GPIO.setup(17,GPIO.OUT)
GPIO.setup(24,GPIO.IN)
time.sleep(0.02)
GPIO.output(23, False)
```

- Send and receive pulse:

```
while True:
    GPIO.output(23, True)
    time.sleep(2)
    GPIO.output(23, False)
    while GPIO.input(24)==0:
        start_time=time.time()
```

- Measure and calculate the distance

After receiving the start time and end time of echo pulse, we need to subtract them to obtain time of pulse duration. Therefore is can be used to calculate the distance:

```
time1=end_time-start_time
distance=17150*time1
print ("Measured Distance is:%.1f"%distance, "cms.")
```

Movement Monitoring and Image Processing Module. When ultrasonic tests objects in a certain distance, movement monitoring will start automatically. Raspberry Pi uses USB camera to collect the color images and transformed them into grayscale images. With frame differential method, the target can be separated from the background. Then use Gaussian blur to the grayscale images. By calculating the difference between the first frame and subsequent new frames in video streaming data, the coordinates of a moving object can be set. Therefore, objects movement monitoring is realized.

(1) Invoke the camera

First we need to install necessary software packages in Python compiling environment.

```
import os
import argparse
import datetime
import imutils
import time
import cv2
```

OpenCV has its own order to call the camera data: *cv2.VideoCapture(0)*

Then we get a camera object reference. In the following program, if video path is not provided, we will get a reference to a camera. If a video file path is provided, then we will build a indicator to it.

```
if args.get("video", None) is None:
    camera = cv2.VideoCapture(0)
    time.sleep(0.25)
else:
    camera = cv2.VideoCapture(args["video"])
```

(2) Image processing

- Image gray processing and Gaussian blur

Gray scale means that divide the brightness variation between the brightest and the darkest into a number of parts. Every digital image is made up by many points. these points are called pixels. Generally, each pixel can present many different colors. But they are all made of three sub-pixels red, green and blue (RGB). The background light source of each sub-pixel can show different brightness levels. The more intermediate level means that the more exquisite graphic effect will be shown.

Set a image $I=f(x,y)$, I means the gray value of image. Define convolution operation:

$$(1)$$

The function G is Gaussian function and the function L means the image that is blurred by Gaussian function.

The meaning of the Eq.1 is: at a pixel point (x, y) , put the center of Gaussian function image at it and sum the multiplication of all the pixels around and their corresponding weights. Point (x, y) corresponds to the weight of $G(0, 0, \sigma)$, point $(x+l, y+l)$ corresponds to the weight of $G(l, l, \sigma)$, point $(x+m, y+n)$ corresponds to the weight of $G(m, n, \sigma)$.

We use the following procedure to process frame data and do motion analysis. First, adjust its size to 500 pixels wide——there is no need to deal directly with the large size of original images in video streaming. Then Convert image to grayscale images, because the color data has no effect on our motion detection algorithm. Finally, use the Gaussian blur to smooth our images.

```
frame = imutils.resize(frame, width=500)
gray = cv2.cvtColor(frame, cv2.COLOR_BGR2GRAY)
gray = cv2.GaussianBlur(gray, (21, 21), 0)
frame = imutils.resize(frame, width=500)
# Initialize the first frame of streaming video
firstFrame = None
```

- Motion detection

OpenCV has a lot of methods for motion detection, tracking and analysis. Functions *cv2*.

BackgroundSubtractorMOG and *cv2. BackgroundSubtractorMOG2* are certain forms of foreground and background segmentation based on mixture Gaussian model. All of these methods involve separating the background from the foreground. In the motion detection, we make the following assumptions:

In consecutive video frames, the background of our video stream is static at most of the time, so if we establish a background model, we can monitor whether there is any significant change. If significant change takes place, we can detect it, and the change is related to the movement of our video.

We have modeled the background in the *firstFrame* variable. It is used to calculate the differences between the starting frame and video streaming data in the following new frames.

Calculating the difference between the two frames is a simple subtraction, we use both sides of the absolute value of the corresponding pixel intensity difference.

(2)

In OpenCV, order *cv2.absdiff* is used to realize the functions above.

If the first frame is *None*, then initialize it:

```

firstFrame = gray
continue
frameDelta = cv2.absdiff(firstFrame, gray)
thresh = cv2.threshold(frameDelta,100,255,cv2.THRESH_BINARY)
Extended threshold image to fill holes, and then find the outline of the threshold
image:
thresh = cv2.dilate(thresh, None, iterations=2)
(,cnts,)=
cv2.findContours(thresh.copy(),cv2.RETR_EXTERNAL,cv2.CHAIN_APPROX_
SIMPLE)

```

Transverse the outline:

```

for c in cnts:
    if cv2.contourArea(c) < args["min_area"]:
        continue
    (x, y, w, h) = cv2.boundingRect(c)
    cv2.rectangle(frame, (x, y), (x + w, y + h), (0, 255, 0), 2)
    text = "Occupied"

```

- Display the results of image processing

In OpenCV, order *cv2.imshow* shows the current frame:

```

cv2.imshow("Security Feed", frame)
cv2.imshow("Thresh", thresh)
cv2.imshow("Frame Delta", frameDelta)

```

The program above shows the results of image processing, we can see whether there is any movement in the video. By frame difference and the threshold image we can debug our script

Conclusion

Through the Raspberry Pi and the python program, we successfully detected objects within a certain distance and do motion detection. The following pictures are the testing results, including the threshold difference image of obtained by frame differential method (Fig. 3 and Fig. 4) and detected moving object image sticking out by the box (Fig. 2):

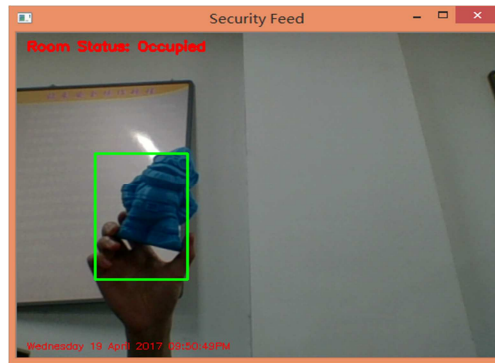


Figure 2.



Figure 3.

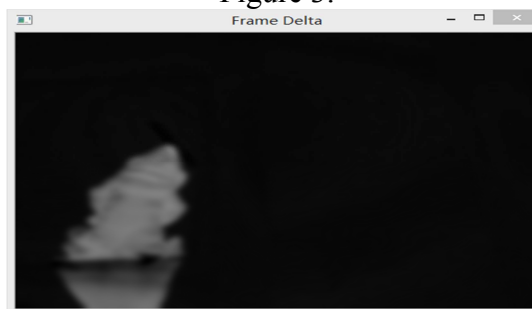


Figure 4.

After the procedures above we can use the order `os.system('omxplayer eagle.wma &')` to play the bird predators' audio files to repellent birds.

References

- [1] Zhao Wei, Lin Tong, Guo Jianquan, et al. Design of intelligent bio-acoustic frightening system against agriculture predators [J]. Transactions of the CSAE, 2010, 26(1):199—204.
- [2] Yang Minxiang. Research on Cause and Grade Division Method of Bird Damage in North China Power Grid[C]. North China Electric Power University. 2011.
- [3] Chen Xiaozhi. Design and realization of warning model and system for transmission line anagianst bird damage[C]. University of Electronic science and Technology of China. 2014.
- [4] Zhao Yuanyuan. Design and application of a bird driving device applied to electric power towers[C]. Dalian Maritime University. 2009.
- [5] P. Bayle. Preventing birds of prey problems at transmission lines in Western Europe. J. Raptor Res. 1999.
- [6] Dirk Uwe Sauer, Jurgen Garcke. Optimum battery design for applications in photovoltaic systems-the oretical considerations. Journal of Power Sources. 2001.