

EE 629 final project report

A Motion Triggered Safety System

Hao Lu

1. Introduction

The project uses PIR motion detect sensor to sense the person passing by. If a person appears in the range of the sensor, the light sensor will decide whether there is sufficient light for a photograph and turn on or turn off the flashlight. Then the camera will take 5 consecutive photos, save the photos on the folder we want and send an email to inform the house owner.

2. Hardware Design

Motion sensor: PIR Motion Sensor (HC-SR501)

Light Sensor: LM393

Camera: DORHEA Raspberry Pi Mini Camera Video Module 5 Megapixels 1080p Sensor OV5647 Webcam

LED (to instead of flash light), botton, alarm, 10k Ω resistors.

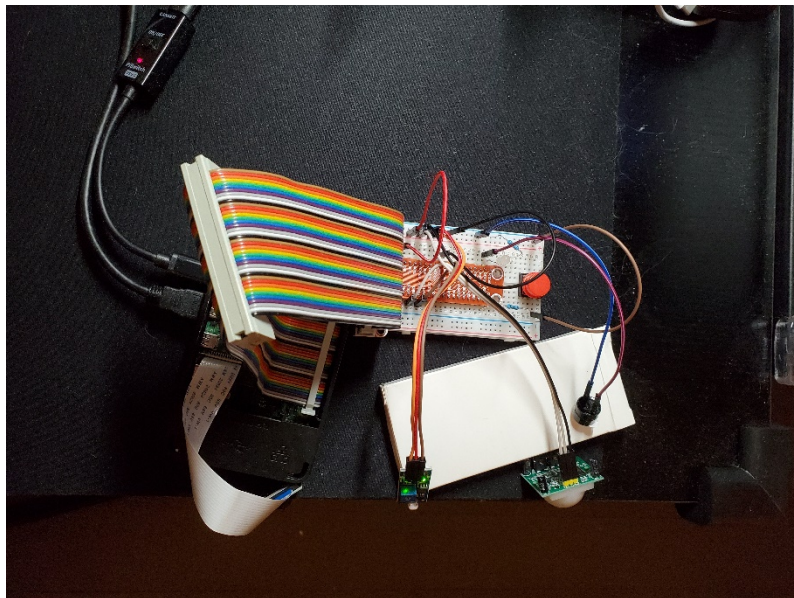


Figure 1. System building 1



Figure 2. System building 2

3. Software Design

The workflow is shown as follow.

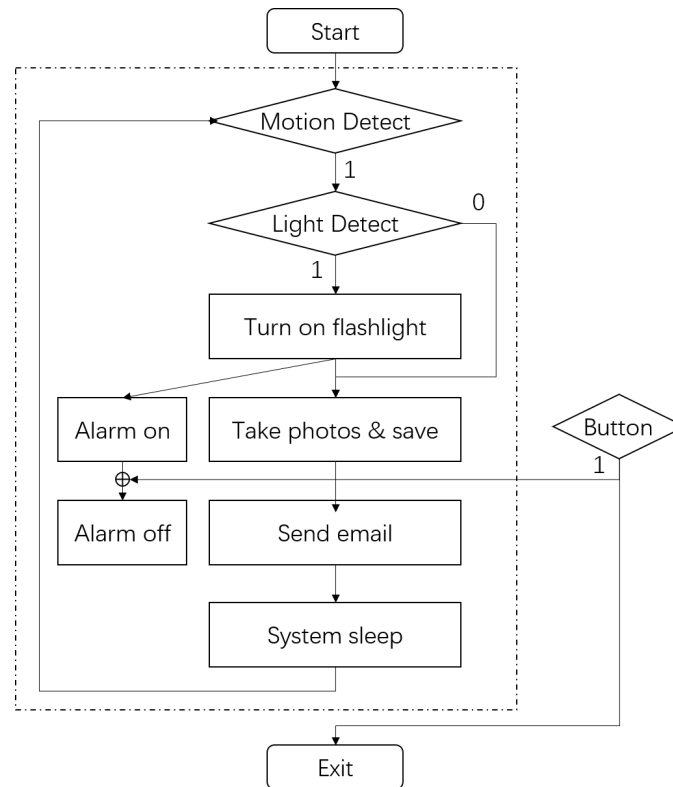


Figure 3. System workflow

Main coding:

Define the function of sending email and attach the taken photo.

```
# define email sending
class Emailer:
    def sendmail(self, recipient, subject, content, image):

        # Create headers
        emailData = MIMEMultipart()
        emailData['Subject'] = subject
        emailData['To'] = recipient
        emailData['From'] = GMAIL_USERNAME

        # Attach text content
        emailData.attach(MIMEText(content))

        # Create image data from the defined image
        imageData = MIMEImage(open(image, 'rb').read(), 'jpg')
        imageData.add_header('Content-Disposition', 'attachment; filename="image.jpg"')
        emailData.attach(imageData)

        # Connect to gmail server
        session = smtplib.SMTP(SMTP_SERVER, SMTP_PORT)
        session.ehlo()
        session.starttls()
        session.ehlo()

        # Login to gmail
        session.login(GMAIL_USERNAME, GMAIL_PASSWORD)

        # Send email & exit
        session.sendmail(GMAIL_USERNAME, recipient, emailData.as_string())
        session.quit

sender = Emailer()
```

Figure 4. main coding 1

Combine the function of taking photo and calling sending email, the system will send the first photo to the user and tell the user when the system is activated.

```
def take_photo():
    GPIO.output(20, GPIO.HIGH)
    global i
    # If the environment is dark, then turn on the flash LED before taking photos
    if GPIO.input(17)==1:
        GPIO.output(21, GPIO.HIGH)
    else:
        GPIO.output(21, GPIO.LOW)
    # Take 5 photos once then save them to specified address
    for j in range(0, 5):
        i = i + 1
        camera.capture('/home/pi/Desktop/secur/image_%s.jpg' % i)
    GPIO.output(21, GPIO.LOW)
    print('photos taken')
    # Notification when system is activated
    # Send the first photo with the email
    k = i - 4
    image = '/home/pi/Desktop/secur/image_%s.jpg' % k
    sendTo = 'hlu18@stevens.edu'
    emailSubject = "Security system triggered!"
    emailContent = "Intruder found: " + time.ctime() # Tell the triggered time
    sender.sendmail(sendTo, emailSubject, emailContent, image)
    print("Email Sent")
    # Set sleep time
    sleep(30)

# Call stop_camera when the button is pressed
button.when_pressed = stop_camera
# Call take_photo when motion is detected
pir.when_motion = take_photo
```

Figure 5. main coding 2

4. Results

Working condition of light sensor is shown in the figure. The left one is when simulating the dark environment, the right one is with good light condition.

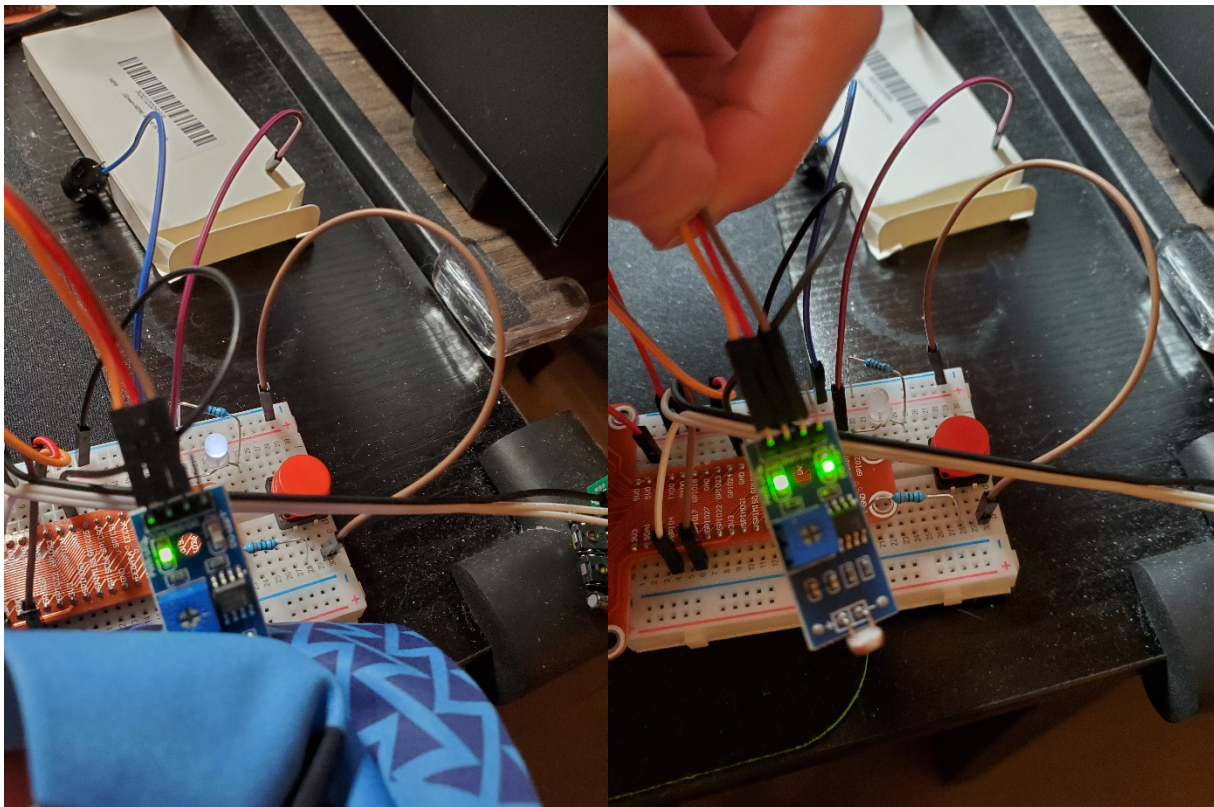


Figure 6. light sensor working condition

Results of the taken photos and informing email. I've copy them to my win10 computer without renaming them to make a better illustration.



Figure 7. Taken photos



Figure 8. Informing email