**INDEX**

1. **Abstract .......................................................................................1**
2. **Components and specifications.....................................................1-6**
3. **Operating System .........................................................................7**
4. **Softwares......................................................................................7-9**
5. **Image............................................................................................9**
6. **Future Enhancements ………………………………………………………………...10**
7. **Code .............................................................................................10-11**
8. **Reference…...................................................................................12**

**ABSRTRACT**

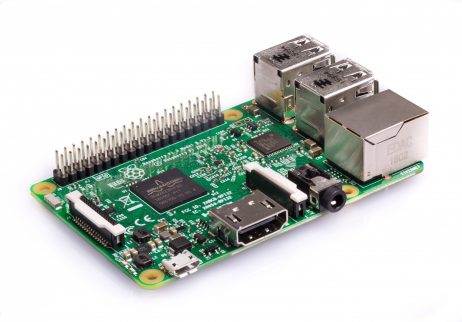
The goal of the project is to construct and programme a camera unit for the use on an agricultural robot. The camera should be built around a chassis that is apt for the agricultural robot and the camera should be able to capture images on command.

The use of Raspberry pi for the completion of the task makes wireless communication possible along with the possibility of future enhancements.

**COMPONENTS AND SPECIFICATIONS**

1. **Raspberry Pi 3**

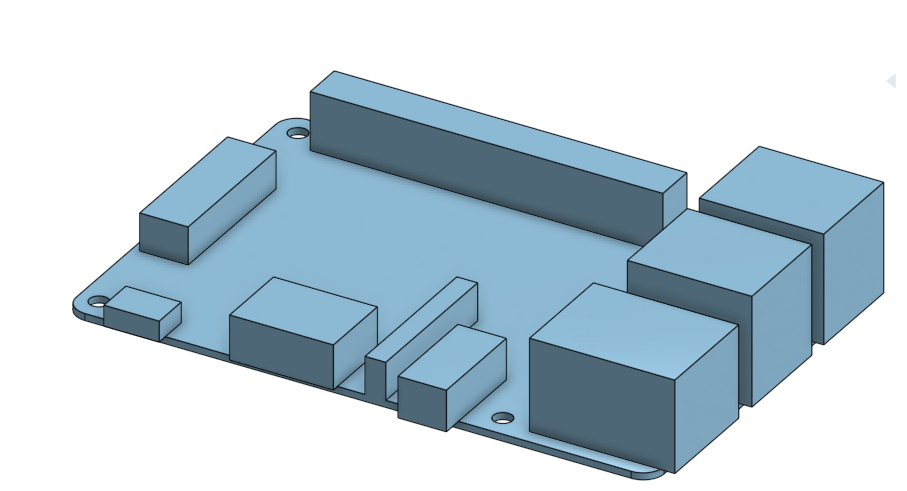
It is a single-board computer with wireless LAN and Bluetooth connectivity. It is run with a micro SD card with the Raspbian Operating System preinstalled.

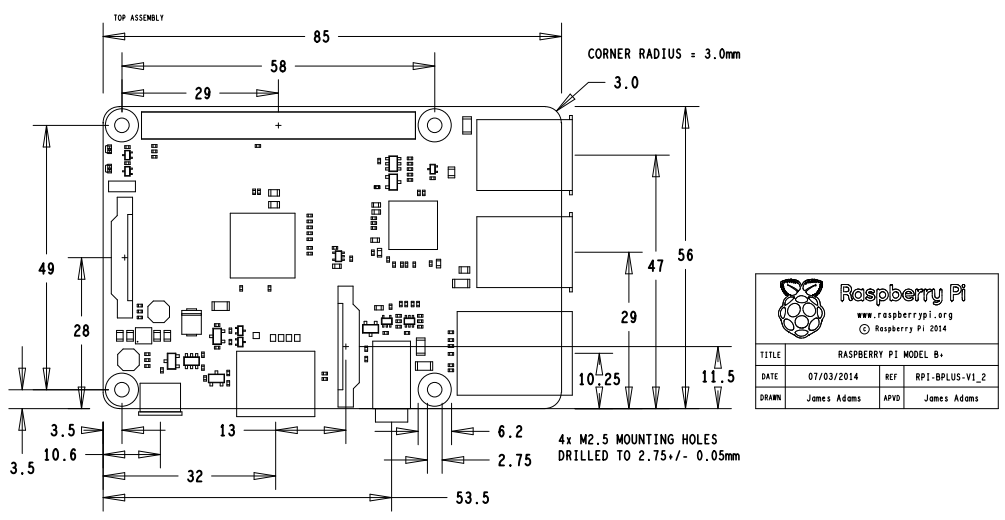


**SPECIFICATIONS**

* Quad Core 1.2GHz Broadcom BCM2837 64bit CPU
* 1GB RAM
* BCM43438 wireless LAN and Bluetooth Low Energy (BLE) on board
* 100 Base Ethernet
* 40-pin extended GPIO
* 4 USB 2 ports
* 4 Pole stereo output and composite video port
* Full size HDMI
* CSI camera port for connecting a Raspberry Pi camera
* DSI display port for connecting a Raspberry Pi touchscreen display
* Micro SD port for loading your operating system and storing data
* Upgraded switched Micro USB power source up to 2.5A

**CAD DESIGN**

****

****

1. **Raspberry Pi Camera**

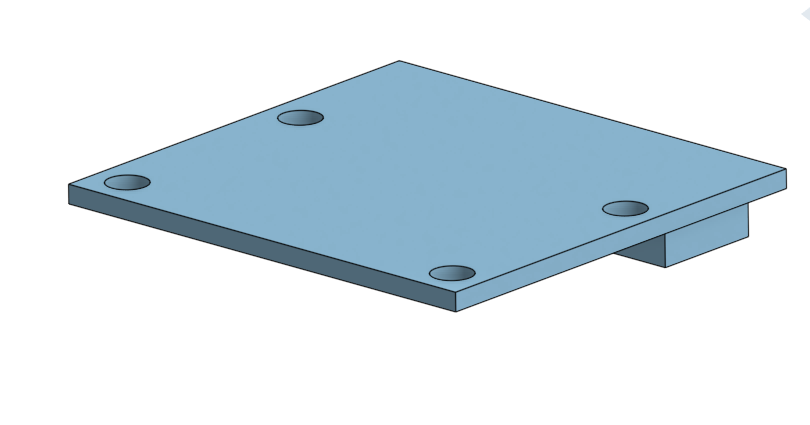
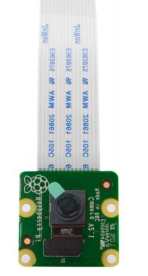
The Camera Module can be used to take high-definition video, as well as stills photographs. It supports 1080p30, 720p60 and VGA90 video modes, as well as still capture. It attaches via a 15cm ribbon cable to the CSI port on the Raspberry Pi.

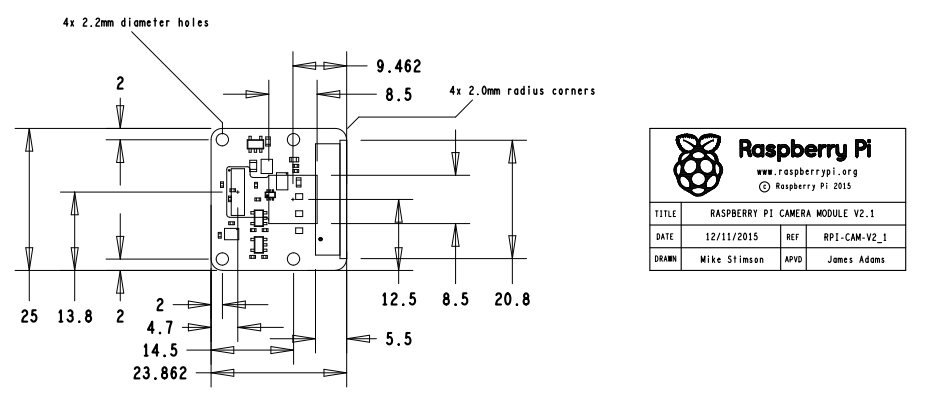
The camera works with all models of Raspberry Pi 1, 2, and 3. It can be accessed through the MMAL and V4L APIs, and there are numerous third-party libraries built for it, including the Picamera Python library.

**SPECIFICATIONS**

* Image Sensor: Sony IMX219.
* Maximum Photo Resolution: 3280 x 2464 pixel.
* Supported Video Resolution: 1080p30, 720p60 and 640x480p90.
* Physical Dimensions: 25mm x 23mm x 9mm.
* Interface: CSI connector (15cm ribbon cable included)
* Supported OS: Raspbian

**CAD DESIGN**



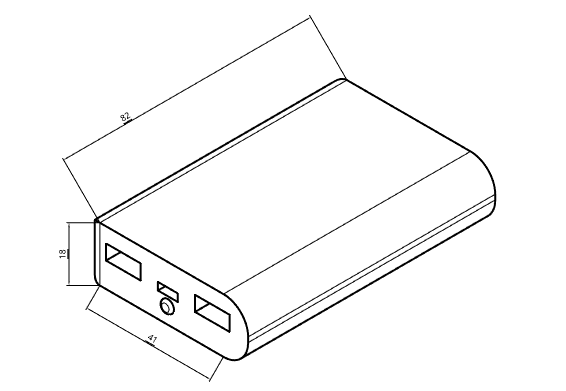
1. **Power bank**

Necessary to power the Raspberry pi over a micro USB cable.

**SPECIFICATIONS**

* Battery capacity: 6000mAh 
* Input voltage: 5V 1000mA (micro-USB)
* Output voltage: 5V 1000mA (Port A), 5V 2400mA (Port B)
* Dimension: 100 x 65 x 22 mm

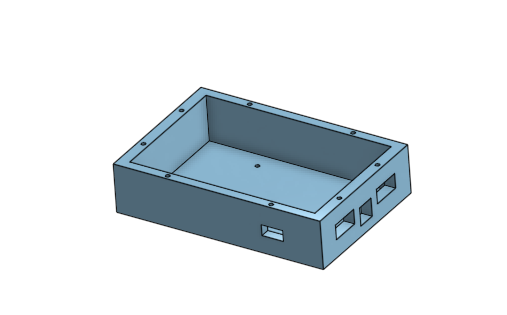
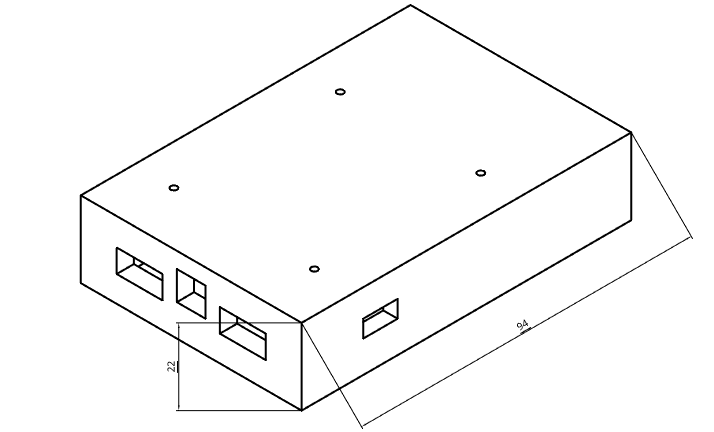
**CAD DESIGN**



1. **Power bank case**

The case is designed to hold the power bank in place near to the Raspberry pi and camera. It is designed with respect to the dimensions of the power bank used for this project and 3D printed.

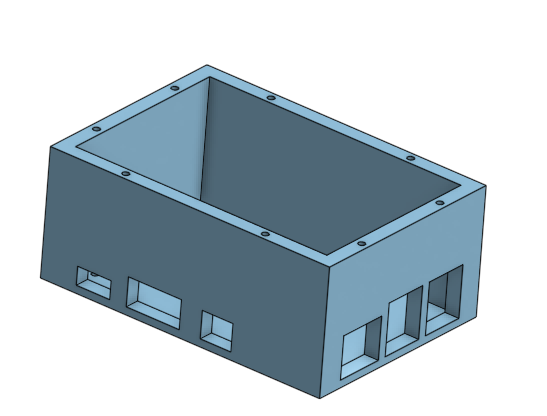
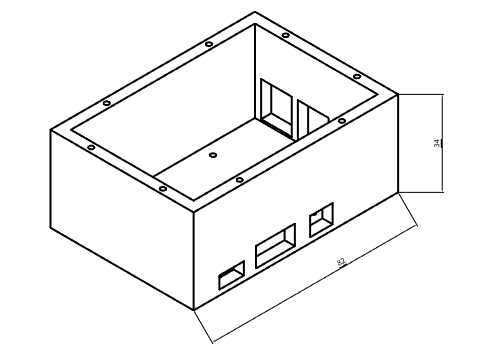
**CAD DESIGN**

1. **Raspberry pi case**

The case is designed to hold the Raspberry pi based on its dimensions and 3D printed. The case is attached with the power bank case as well as the camera holder on either sides.

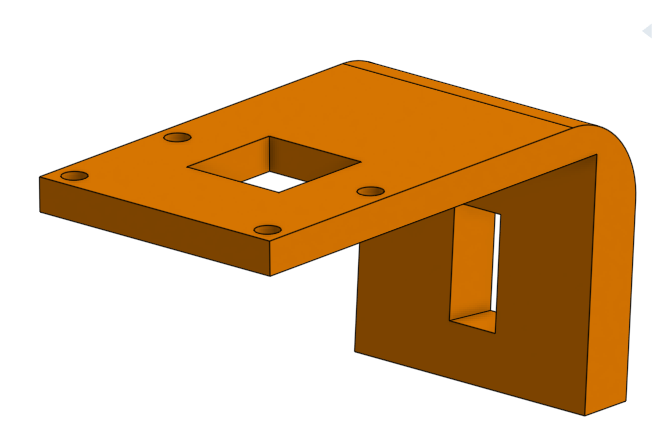
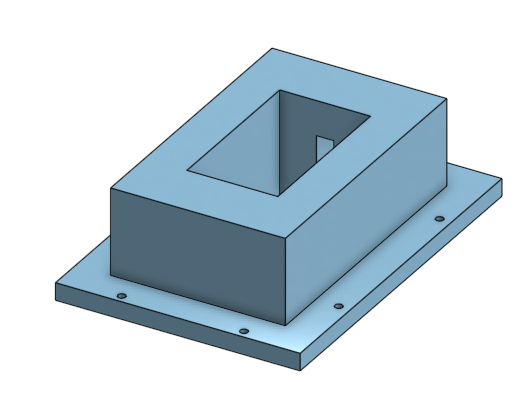
**CAD DESIGN**

1. **Camera holder**

A special holder is designed to keep the camera in place and 3D printed. The camera holder is stable enough to avoid vibrations which could cause blurring of the images taken.

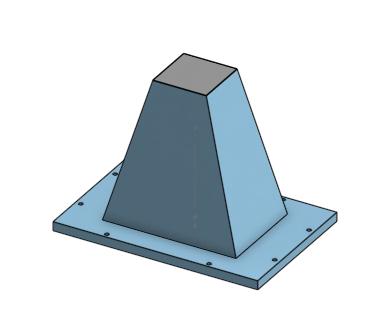
**CAD DESIGN**

1. **Pick up**

This part can be designed in respect to the gripper used by the agricultural robot and can be changed accordingly.

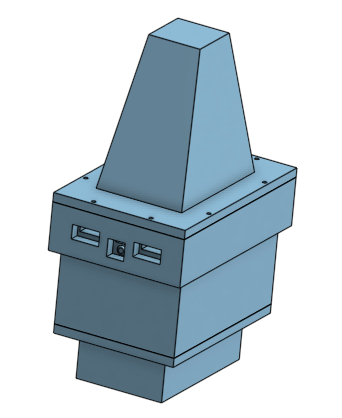
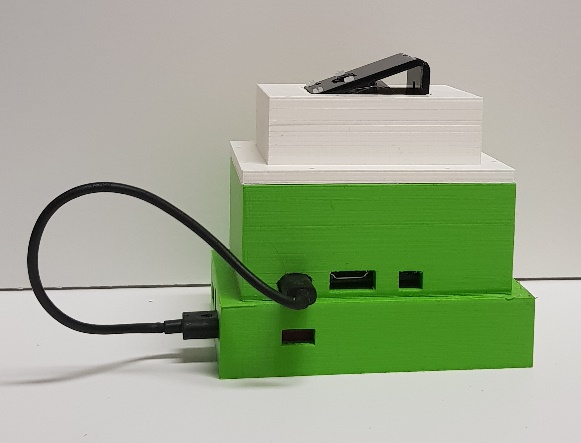
**CAD DESIGN**

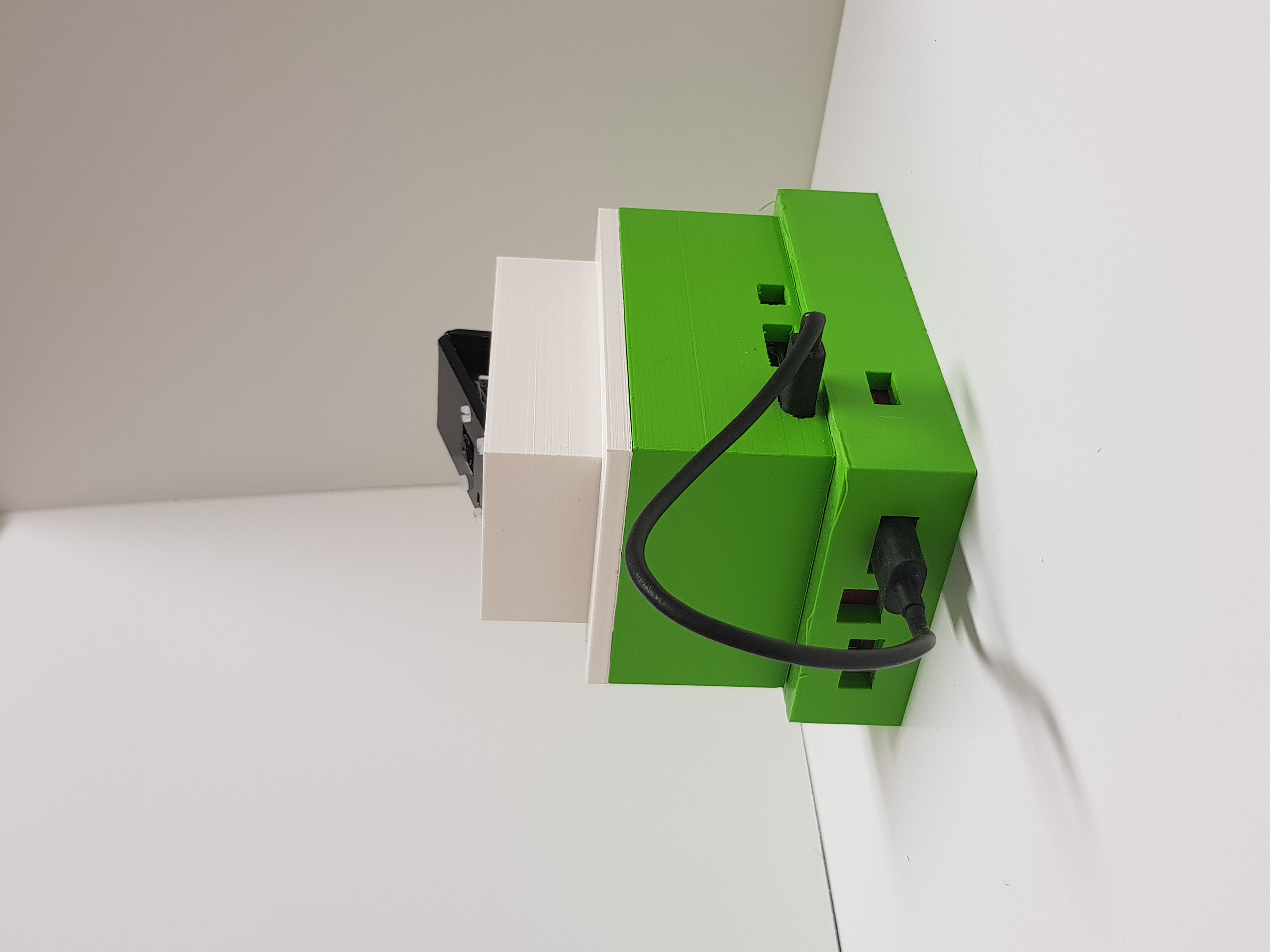


1. **Assembly**

The entire assembly of all the above mentioned components makes up the camera unit used for the agricultural robot.

**CAD DESIGN**

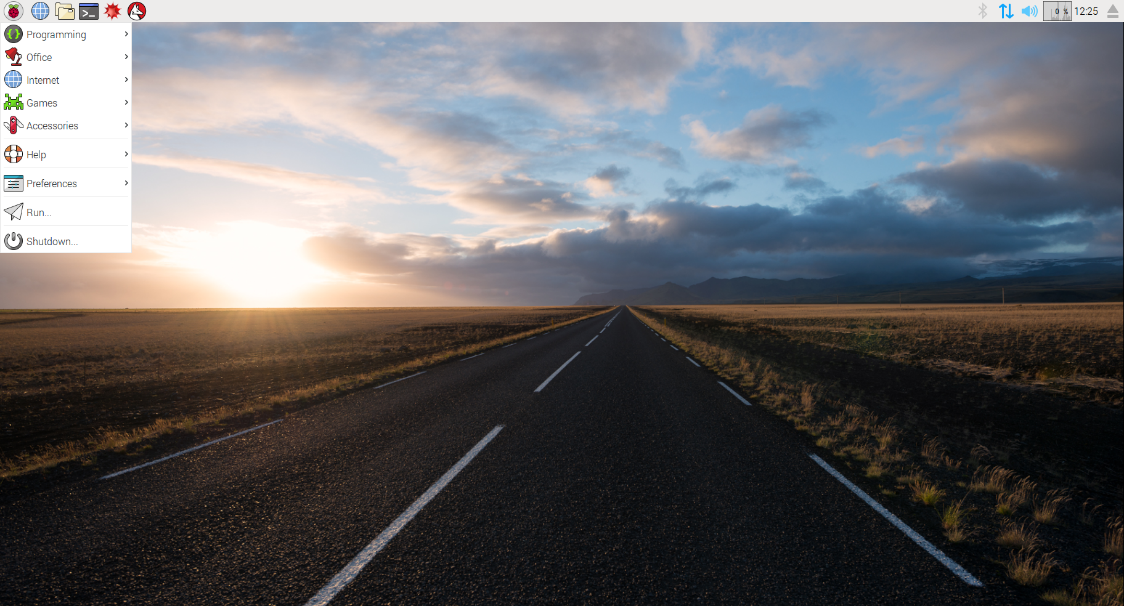
 

****

**OPERATING SYSTEM**

**Raspbian**

The Raspberry pi runs on the Raspbian operating system which is installed through the external SD card attached to the board. Initially the board and the operating system is setup by connecting the board to a monitor across a HDMI cable along with an external keyboard as well as a mouse.

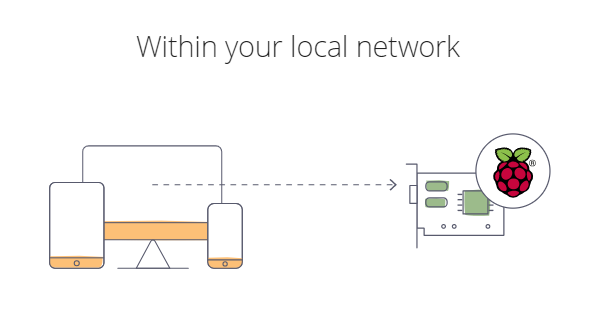


Necessary software are installed on to the system through codes given on the terminal. The code “sudo apt-get update” is used keep the installed software packages up to date.

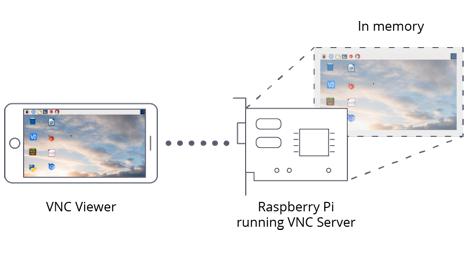
**SOFTWARES**

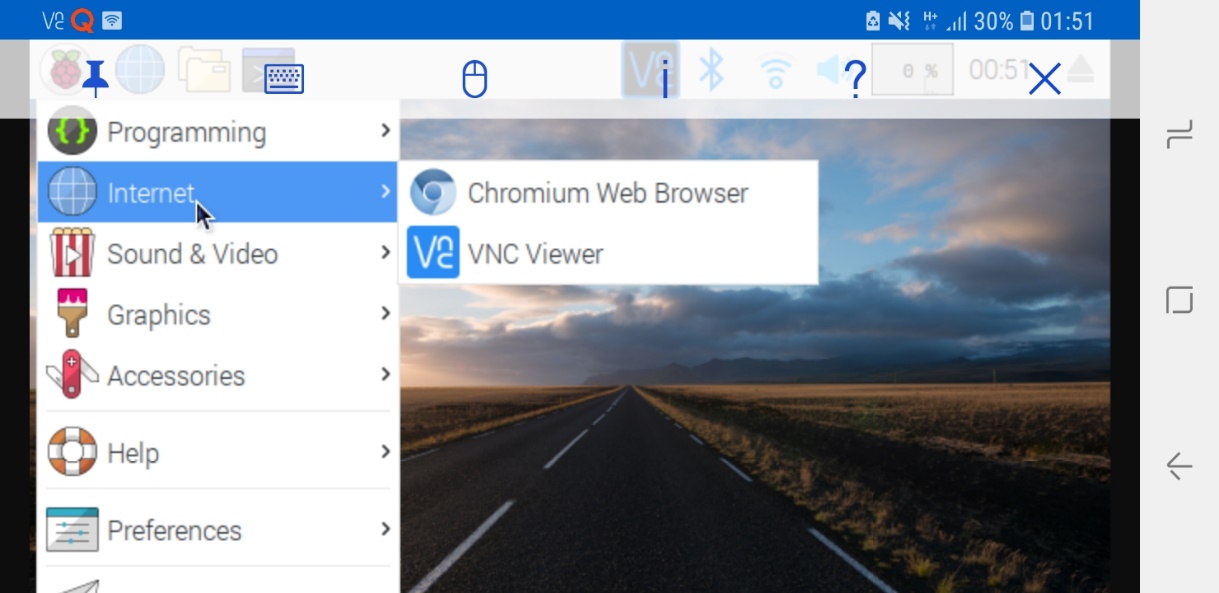
1. **VNC Viewer:**

VNC Connect consists of a VNC Server application that captures the desktop of your Pi in real-time, enabling you to remotely control it, plus supporting programs. Direct connections are quick and simple providing you’re joined to the same private local network as your Raspberry Pi (for example, a wired or Wi-Fi network at home, school or in the office).



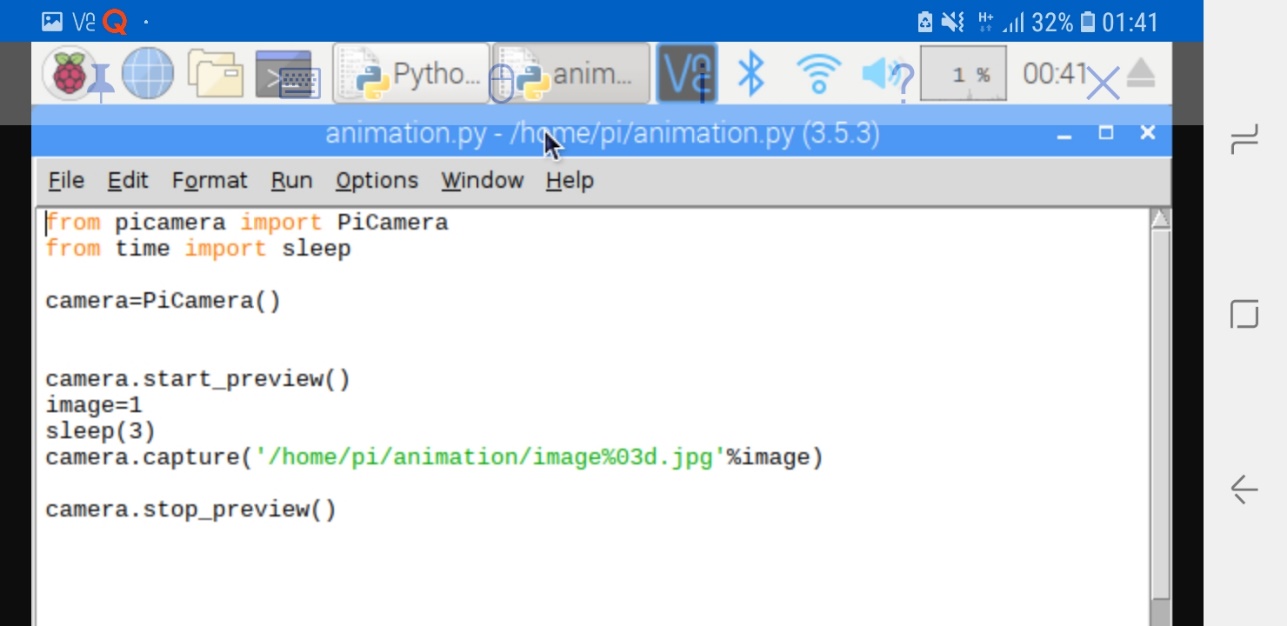
If your Raspberry Pi is headless (that is, not plugged into a monitor) or embedded in a robot, it’s unlikely to be running a graphical desktop.

VNC Server can run in Virtual Mode to create a resource-efficient virtual desktop on demand, giving you graphical remote access even when there is no actual desktop to remote. This virtual desktop exists only in your Raspberry Pi’s memory.****



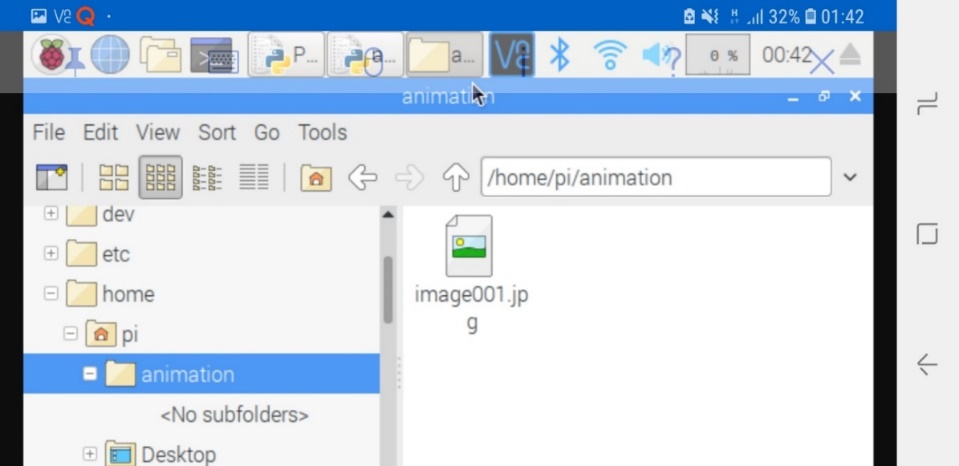
1. **Python**

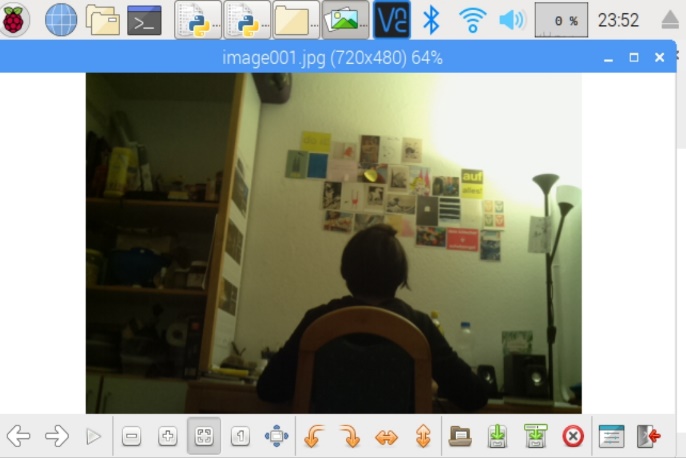
Instead of running the code of the Pi camera on the terminal, the code can be written in Python and run. Python is a programming language that lets you work quickly and integrate systems more effectively. Python features a dynamic type system and automatic memory management. It supports multiple programming paradigms, including object-oriented, imperative, functional and procedural, and has a large and comprehensive standard library.

****

**IMAGE**

The images that are captured are saved in the specified folder and send as attachments via email simultaneously. The images are captured only when the specific python code is run. A small delay of 3 seconds is provided after the capture of every image to obtain as clear images as possible. Python provides *smtplib*module, which defines an SMTP client session object, which is used to send mail to any Internet machine.





**FUTURE ENHANCEMENTS**

Advancements and changes to the code can be made without dismantling the camera unit making room for a vast variety of improvements. The python code can be easily adjusted to capture videos or to create an animation of the pictures taken over a period of time. This can be useful to observe the periodic changes in agricultural field.

The gripper designed for the camera unit can be also redesigned to match the agricultural robot as well.

**CODE**

import smtplib,ssl

from picamera import PiCamera

from time import sleep

from email.mime.multipart import MIMEMultipart

from email.mime.base import MIMEBase

from email.mime.text import MIMEText

from email.utils import formatdate

from email import encoders

camera=PiCamera()

camera.start\_preview()

image=1

sleep(3)

camera.capture('/home/pi/animation/image%03d.jpg'%image)

camera.stop\_preview()

def send\_an\_email():

toaddr = 'naveenmathewgeorge@gmail.com'

me = 'raspiprojekt94@gmail.com'

subject = "Forschungsprojekt"

path='/home/pi/animation/image001.jpg'

msg = MIMEMultipart()

msg['Subject'] = subject

msg['From'] = me

msg['To'] = toaddr

msg.preamble = "test "

part = MIMEBase('application', "octet-stream")

part.set\_payload(open(path, "rb").read())

encoders.encode\_base64(part)

part.add\_header('Content-Disposition', 'attachment; filename=image')

msg.attach(part)

s = smtplib.SMTP('smtp.gmail.com', 587)

s.ehlo()

s.starttls()

s.ehlo()

s.login(user = 'raspiprojekt94@gmail.com', password = 'forraspberry')

s.sendmail(me, toaddr, msg.as\_string())

s.quit()

send\_an\_email()

**Reference**

<https://www.raspberrypi.org/>

<https://www.raspberrypi.org/documentation/hardware/raspberrypi/README.md>

<https://www.realpower.de/details.php?artnr=206615#.XDaTjVxKg2w>

<https://www.onshape.com/>

<https://www.realvnc.com/en/connect/docs/raspberry-pi.html#raspberry-pi-connect-direct>

<https://www.python.org/>