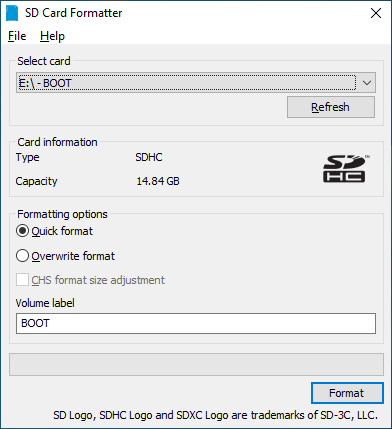
**Beginners Guide to start using Raspberry Pi with OpenCV**

1. Download SD card formatter here, <https://www.sdcard.org/downloads/>
2. Insert the SD card to the computer and format SD card using the formatter

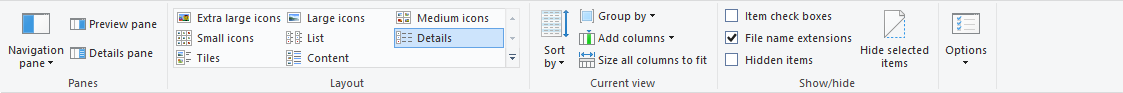


1. Download Raspberry Pi Imager here, <https://www.raspberrypi.com/software/>
2. Open the Imager, choose the options as below and upload Raspberry Pi OS to the SD card



1. When finished writing image to SD card, remove the SD card and reconnect it back.
2. For headless setup

* Configure wireless network
  + Go to Boot folder. For Windows, make sure to tick “File name extensions” in View tab. Then, right click, select New > Text Document.



* + Rename the file to “wpa\_supplicant.conf”. Make sure the file type is “CONF File” instead of “Text Document”.
  + Open the file using Notepad++. Copy text below and paste into the file.

ctrl\_interface=DIR=/var/run/wpa\_supplicant GROUP=netdev

country=<Insert 2 letter ISO 3166-1 country code here>

update\_config=1

network={

ssid="<Name of your wireless LAN>"

psk="<Password for your wireless LAN>"

}

* + Change the text highlighted in yellow to your own preference, eg.:

ctrl\_interface=DIR=/var/run/wpa\_supplicant GROUP=netdev

country=MY

update\_config=1

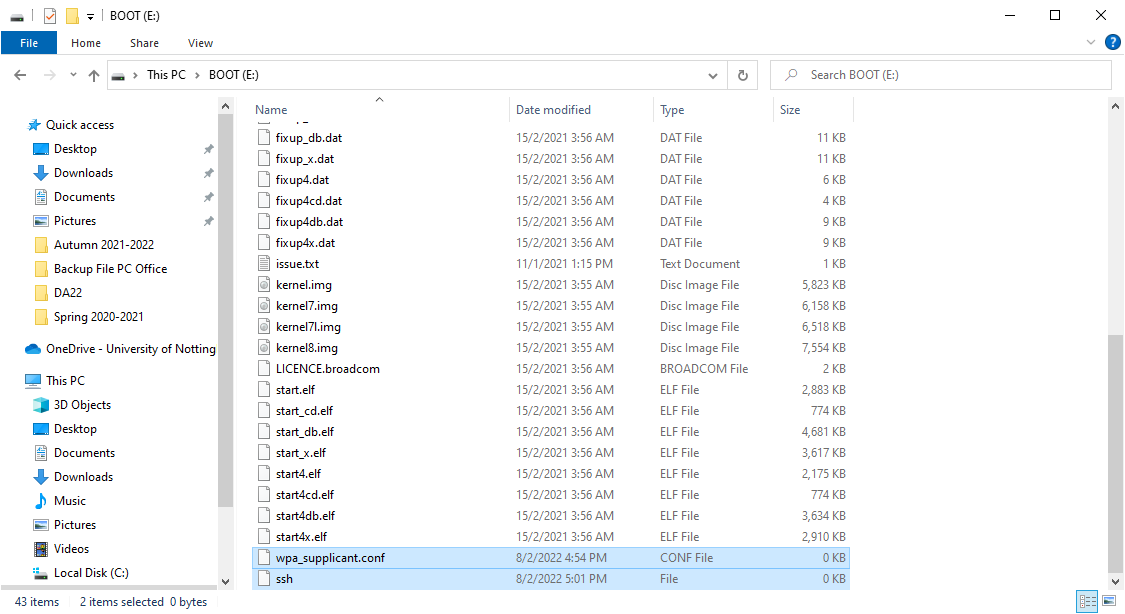
network={pikjsudo

ssid="Rpi\_Hotspot"

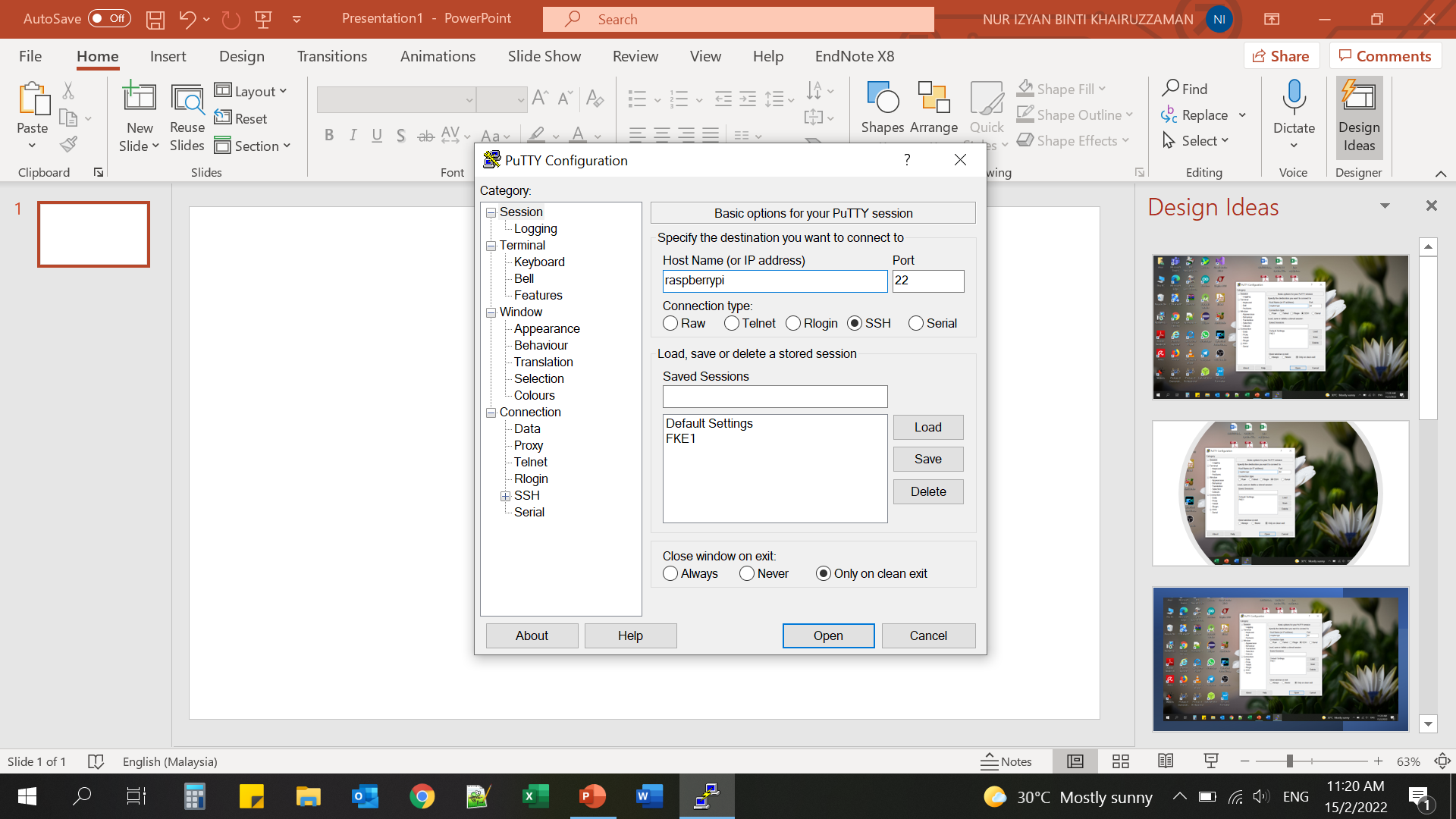
psk="abcd1234"

}

* Enabling SSH
  + Create SSH file named “ssh” without any extension into the Boot folder and leave it blank.
  + Finally, you should have 2 new files in the Boot folder.



* + Remove SD card from computer
* Connect to Raspberry Pi using SSH (using command line)
  + Download and install PuTTY in your remote desktop, <https://www.putty.org/>
  + Insert the SD card and power on your Raspberry Pi
  + Open PuTTY and write in as below

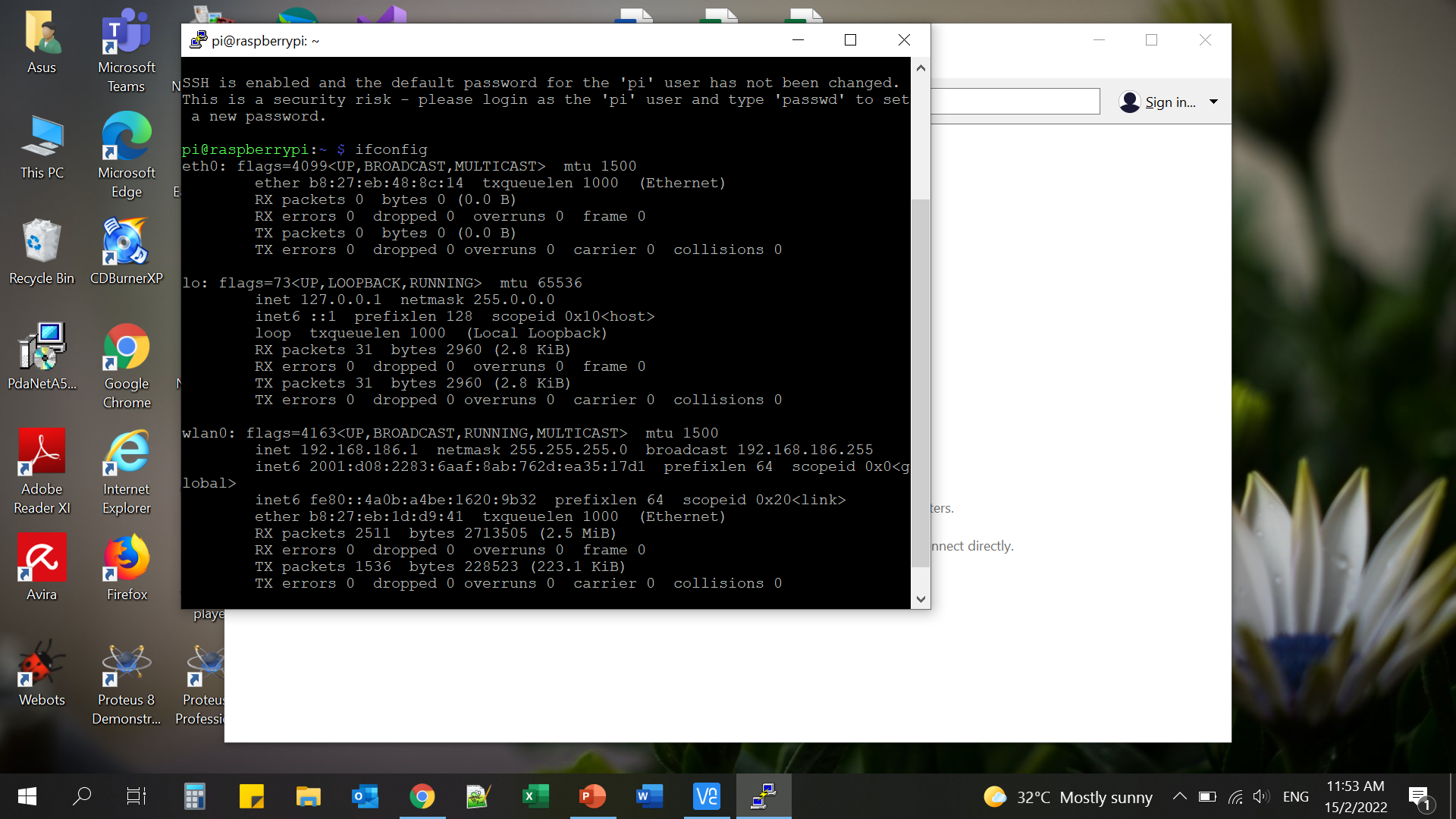


* + Enter default username and password

login as: pi

password: raspberry

* Install VNC (to access full desktop environment)
  + sudo apt update
  + sudo apt install realvnc-vnc-server realvnc-vnc-viewer
* Enabling VNC Server
  + sudo raspi-config
  + Navigate to Interface Options > VNC > Yes
  + Then select Finish
* Connecting to Raspberry Pi desktop
  + Download VNC Viewer in the remote desktop
  + On SSH terminal, run “ifconfig” to discover your private IP address



* + Enter your Raspberry Pi’s private IP address into VNC Viewer
  + Done!

1. To install OpenCV, open terminal, then follow steps below:

* sudo apt-get update / sudo apt-get --allow-releaseinfo-change update
* sudo apt-get upgrade
* pip3 install --upgrade pip
* sudo pip3 install numpy
* sudo pip3 install scipy
* sudo pip3 install scikit-image
* sudo pip3 install RPi.GPIO
* sudo apt-get install build-essential cmake unzip pkg-config
* sudo apt-get install libjpeg-dev libpng-dev libtiff-dev
* sudo apt-get install libavcodec-dev libavformat-dev libswscale-dev libv4l-dev
* sudo apt-get install libxvidcore-dev libx264-dev
* sudo apt-get install libgtk-3-dev
* sudo apt-get install libcanberra-gtk\*
* sudo apt-get install libatlas-base-dev gfortran
* sudo apt-get install python3-dev
* sudo apt-get install python3-opencv
* sudo apt-get install idle3( just to run the idle application)
* sudo idle
* In IDLE, type in:

import cv2

print(cv2.\_\_version\_\_)

* If OpenCV version is successfully print out, then the package is successfully installed

1. Enable camera

To do this, enter sudo raspi-config at a terminal window and then choose Interface Options, Camera and Yes.

1. Enable Glamor

On Pi3 and earlier devices running **Bullseye** you need to re-enable Glamor in order to make the X-Windows hardware accelerated preview window work. To do this at the configuration window, choose Advanced Options, Glamor  and Yes. Finally quit raspi-config and let it reboot your Pi.

1. To test your camera, open terminal, then type in:

libcamera-hello (for *Bullseye*)

Note: If you got error “the system appears to be configured for the legacy camera stack” please use “raspistill” command instead.

Or

raspistill -o Desktop/image.jpg

1. Try the code in Python 3 IDE:

import cv2

# define a video capture object

vid = cv2.VideoCapture(0)

while(True):

    # Capture the video frame by frame

    ret, frame = vid.read()

    # Display the resulting frame

    cv2.imshow('frame', frame)

    # the 'q' button is set as the

    # quitting button you may use any

    # desired button of your choice

    if cv2.waitKey(1) & 0xFF == ord('q'):

        break

# After the loop release the cap object

vid.release()

# Destroy all the windows

cv2.destroyAllWindows()

**How to manually set time and date:**

sudo date -s "Mon Aug 12 20:14:11 UTC 2014"

**Swap memory.**

The next step is to increase your swap space. OpenCV needs a lot of memory to compile. The latest versions want to see a minimum of 6.5 GB of memory before building. Your swap space is limited to 2048 MByte by default. To exceed this 2048 MByte limit, you will need to increase this maximum in the /sbin/dphys-swapfile. Everything is demonstrated in the slideshow below.

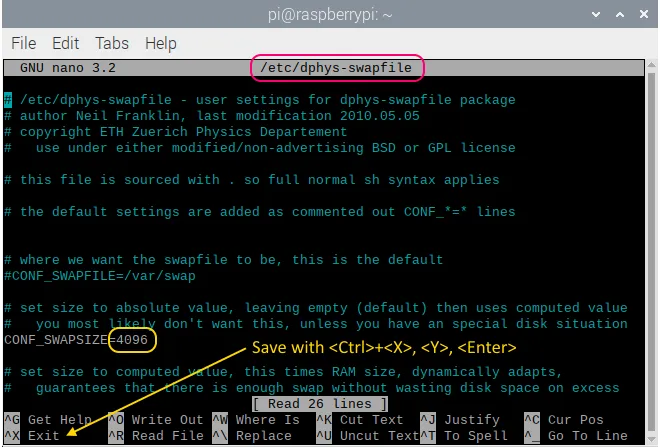
# edit the swap configuration

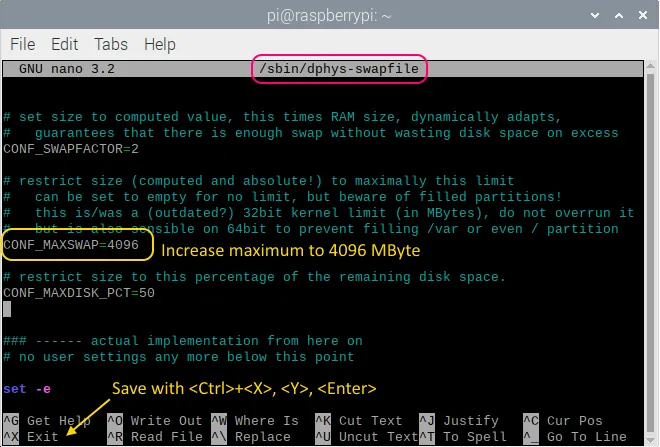
$ sudo nano /sbin/dphys-swapfile

$ sudo nano /etc/dphys-swapfile

# reboot

$ sudo reboot





❮❯

**Installation script.**

Installing OpenCV on your Raspberry Pi is not that complicated. With its 60 command lines, it is more of an administrative task. That's why we created an installation script that executes all commands in this guide at once. Use it if you want, it shouldn't cause any problems. The whole installation will take one and a half hour to complete. It starts with the installation of the dependencies and ends with the ldconfig.

If you want to beautify your OpenCV with the Qt5 GUI, follow the instructions on the [GitHub page](https://github.com/Qengineering/Install-OpenCV-Raspberry-Pi-32-bits), or in the section below.

OpenCV 4.5.5 (must update the os 1st using sudo raspi-config)

# check your memory first  
$ free -m  
# you need at least a total of 6.5 GB!  
# if not, enlarge your swap space as explained in the guide  
$ wget https://github.com/Qengineering/Install-OpenCV-Raspberry-Pi-32-bits/raw/main/OpenCV-4-5-5.sh  
$ sudo chmod 755 ./OpenCV-4-5-5.sh  
$ ./OpenCV-4-5-5.sh

Line tracking references

1. [Line Following Robot Using OpenCV - YouTube](https://www.youtube.com/watch?v=tlkWX7R-NHE) (<https://youtu.be/tlkWX7R-NHE>)
2. https://github.com/Vivek-Nehra/Line-Follower/blob/master/Final\_Line\_Test.py