

Manual Book



Mask Detection



Annisa Aulia Nadhila (2041720023)
Yudas Malabi (2041720054)

DAFTAR ISI

DAFTAR ISI.....	1
1. INSTALASI TOOLS.....	2
1.1 INSTALASI PHYTON.....	2
Langkah 1 : Memilih Versi Python yang Akan Diinstall.....	2
Langkah 2 : Download Python Executable Installer.....	2
Langkah 3 : Menjalankan Executable Installer.....	2
Langkah 4 : Verifikasi Instalasi Python.....	3
Langkah 5 : Verifikasi Instalasi pip.....	4
1.2 INSTALASI VISUAL STUDIO.....	5
Langkah 1 : persiapan alat dan bahan.....	5
Langkah 2 : langkah install visual studio code.....	5
2. INSTALASI LIBRARY.....	7
3. KODE PROGRAM.....	7
4. PENGGUNAAN.....	12

1. INSTALASI TOOLS

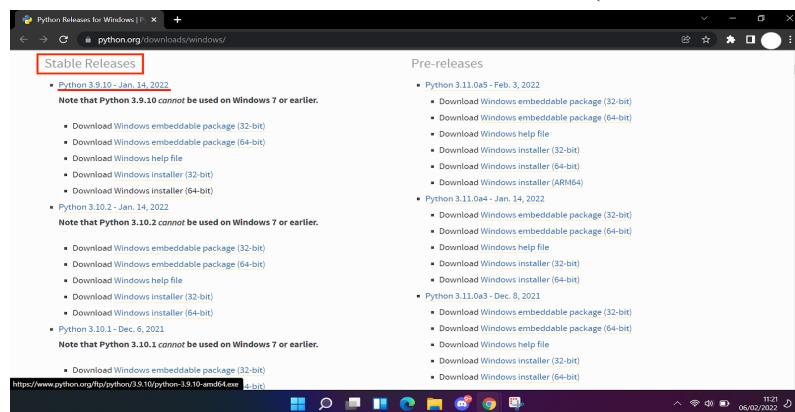
1.1 INSTALASI PHYTON

Langkah 1 : Memilih Versi Python yang Akan Diinstall

Untuk keperluan belajar menggunakan Python, direkomendasikan untuk menginstall Python versi 2 atau 3. Tapi pada panduan ini, Python yang akan diinstall adalah Python versi 3.

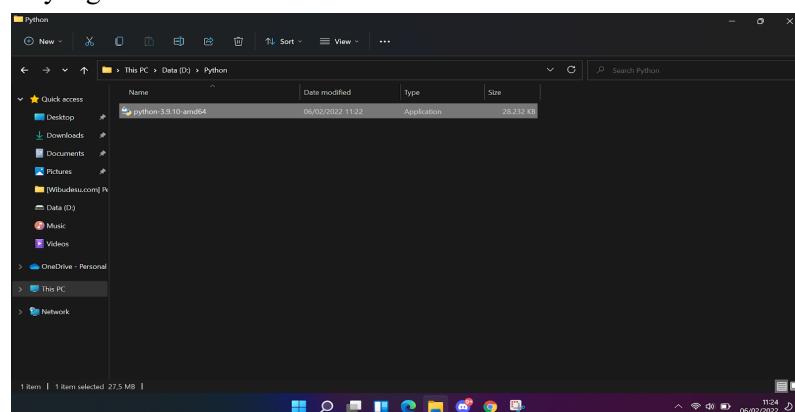
Langkah 2 : Download Python Executable Installer

1. Bukalah web browser Anda dan navigasi ke "[Downloads for Windows](#)" pada [website official Python](#)
2. Carilah versi Python yang akan diinstall. Pada saat ini, versi paling akhir yang sudah dirilis adalah versi 3.10.2. Akan tetapi, kita akan memilih versi stable release pada instalasi kali ini. Jadi, kita akan memilih versi 3.9.10.
3. Pilihlah link untuk melakukan download executable installer(32-bit atau 64-bit)



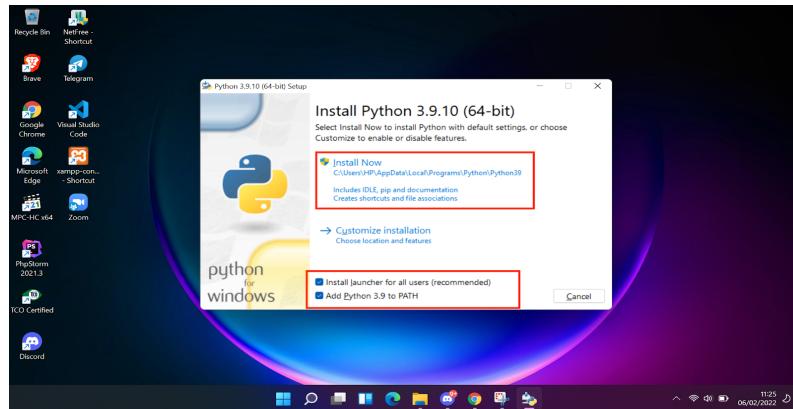
Langkah 3 : Menjalankan Executable Installer

1. Jalankan installer yang sudah selesai di-download

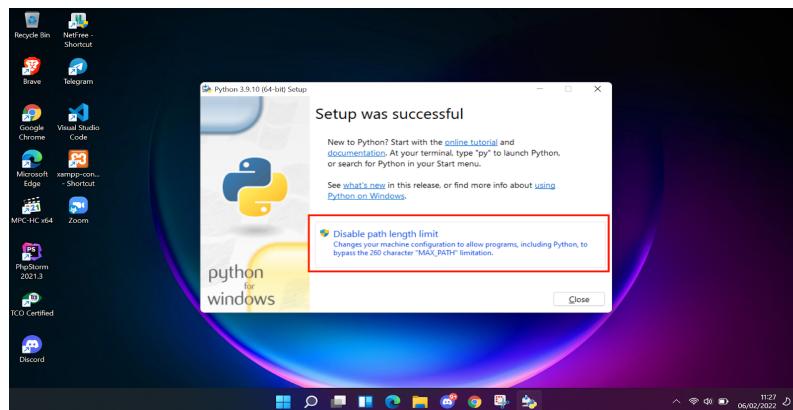


2. Pastikan Anda memilih **Install launcher for all users** dan **Add Python 3.9 to PATH** pada checkbox yang tersedia dan pilih **Install Now**.

Untuk semua versi Python akhir-akhir ini, pada proses instalasinya akan merekomendasikan opsi instalasi termasuk pip dan IDLE.



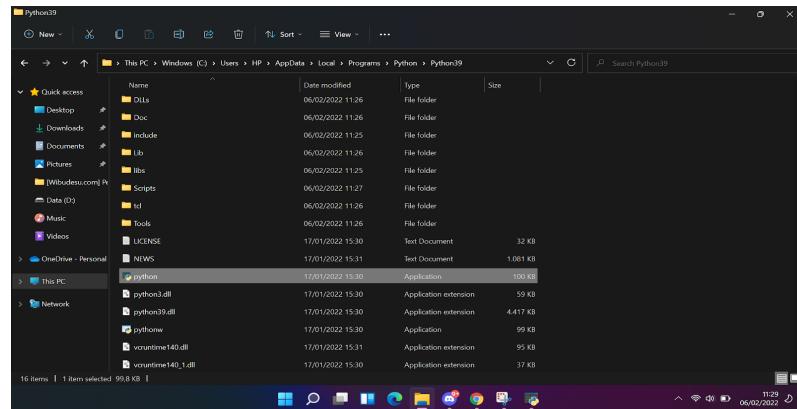
3. Dialog selanjutnya yang muncul akan memberikan pilihan kepada Anda untuk mematikan path length limit atau tidak. Dengan memilih ini, Python akan dapat melakukan bypass 260-character MAX_PATH limit. **Disable path length limit** tidak akan mempengaruhi pengaturan sistem yang lain.



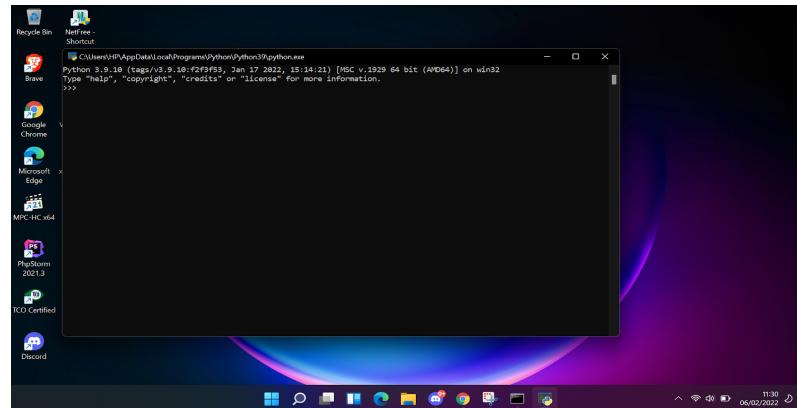
Langkah 4 : Verifikasi Instalasi Python

Untuk memastikan apakah Python sudah berhasil terinstal pada perangkat, langkah-langkah yang dapat dilakukan adalah sebagai berikut.

1. Navigasi ke directory atau folder dimana python terinstall. Sebagai contoh seperti yang terlihat pada screenshot, Python terinstall pada path **C:\Users\HP\AppData\Local\Programs\Python\Python39**

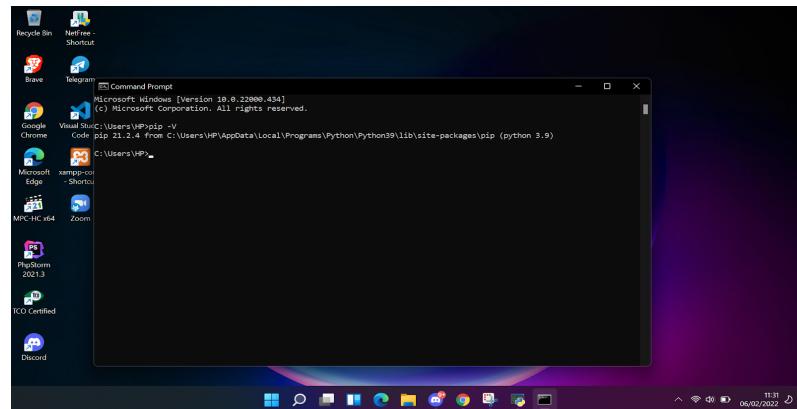


- Double-click pada python.exe dan tampilan pada layar Anda seharusnya kurang-lebih mirip seperti yang tertera pada screenshot.



Langkah 5 : Verifikasi Instalasi pip

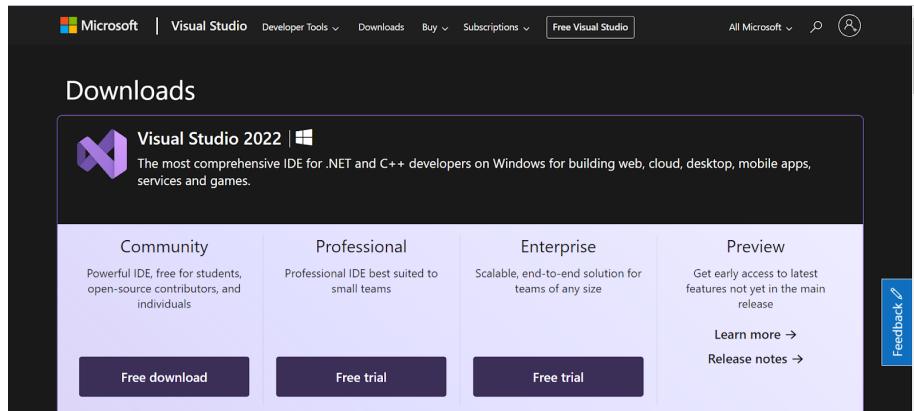
- Buka menu windows dan ketikkan "cmd".
- Pilih aplikasi Command Prompt.
- Ketikkan **pip -V** pada console. Jika pip sudah berhasil terinstall, maka output yang akan dihasilkan dapat dilihat pada screenshot berikut.



1.2 INSTALASI VISUAL STUDIO

Langkah 1 : persiapan alat dan bahan

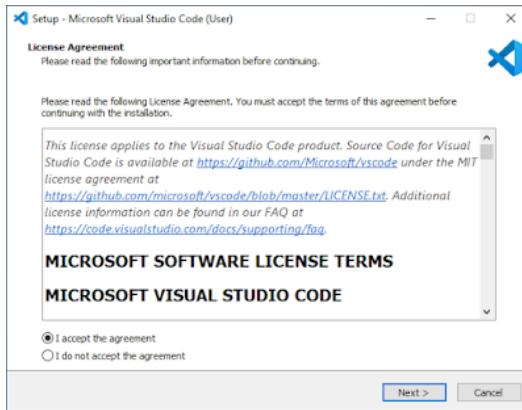
1. Kunjungi web [visual studio code](#), lalu pilih download for Windows



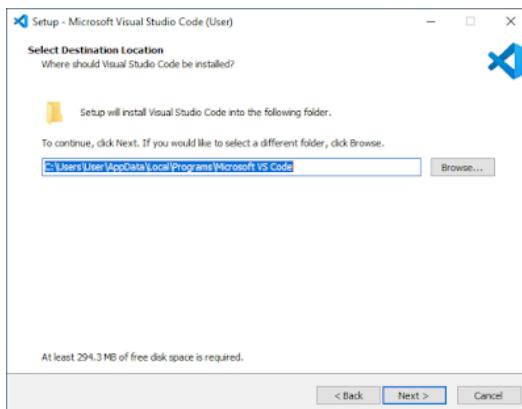
2. Jika buru-buru, silahkan langsung download saja aplikasi Visual studio Code Windows 10 [disini](#)

Langkah 2 : langkah install visual studio code

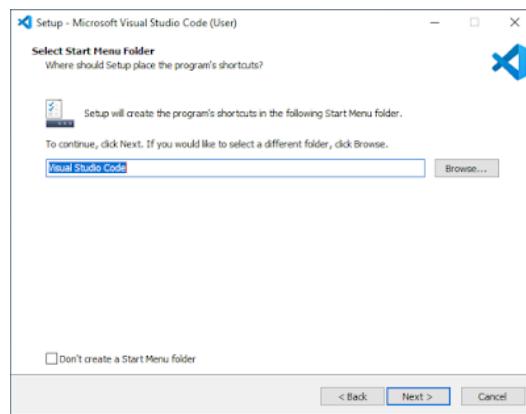
1. Buka folder hasil download Visual Studio Code
2. **Double klik** setup Visual Studio Code
3. Klik "**I accept the agreement**" dan klik Next



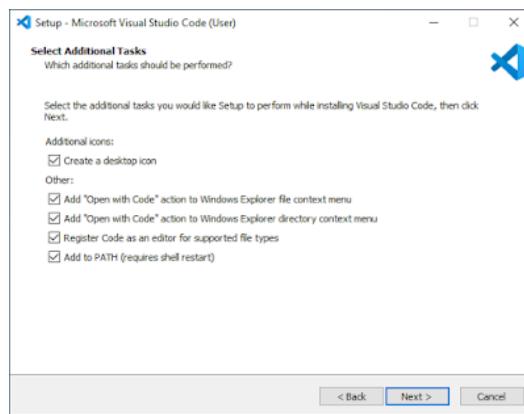
4. **Select destination location**, biarkan default. Klik Next



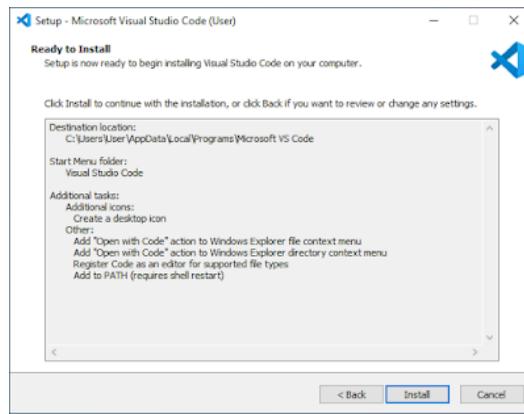
5. Select Start menu folder dan klik Next



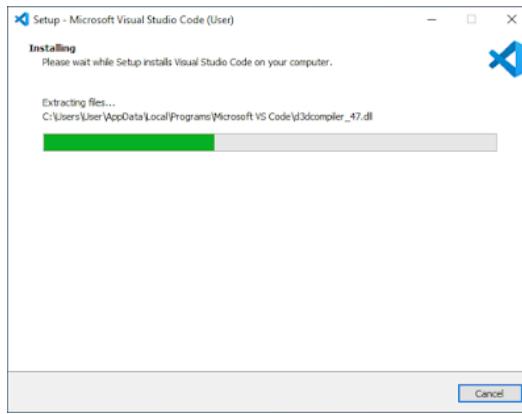
6. Select additional task, ceklist semua dan klik Next



7. Klik Install, Tunggu beberapa saat



8. Proses instalasi sedang berlangsung, jangan di cancel ya!



9. Klik Finish



2. INSTALASI LIBRARY

- a. pytsxs3 -> untuk convert text ke audio
- b. Opencv -> Pemrosesan Citra
- c. Numpy -> Komputasi Numerik
- d. gTTs -> Google text to speech, convert text ke audio namun secara online
- e. Tensorflow keras -> Analisa model deep learning

3. KODE PROGRAM

```
import numpy as np
import time
import cv2
import os

import pytsxs3
import time
from gtts import gTTS
from playsound import playsound
```

```

import hashlib
import threading
from keras.models import load_model
from tensorflow.keras.preprocessing.image import img_to_array
from tensorflow.keras.applications.mobilenet_v2 import preprocess_input

# class speech
class Speech:
    def __init__(self):
        self.die = False
        self.busy = False

        self.tts = pyttsx3.init()
        self.tts.setProperty('voice', 'indonesian')

    def say(self, txt, using_gtts=True):

        if self.busy:
            return

        self.busy = True

        if using_gtts:
            sfname = ''.join(x for x in txt if x.isalnum())
            sfname = sfname[0:3] + sfname[-3:] +
hashlib.md5(sfname.encode()).hexdigest()
            sfname = 'speech/{}.mp3'.format(sfname)
            if not os.path.exists(sfname):
                try:
                    speech = gTTS(txt, Lang='id')
                    speech.save(sfname)
                except:
                    os.unlink(sfname)
                    tts.say(txt)
                else:
                    playsound(sfname)
            else:

```

```

        playsound(sfname)
    else:
        self.tts.say(txt)
        self.tts.runAndWait()

    self.busy = False


def detect_and_predict_mask(frame, faceNet, maskNet):
    (h, w) = frame.shape[:2]
    blob = cv2.dnn.blobFromImage(frame, 1.0, (300, 300),
                                 (104.0, 177.0, 123.0))

    faceNet.setInput(blob)
    detections = faceNet.forward()

    # initialize list of prediction face mask network,
    faces = []
    locs = []
    preds = []

    # loop detection
    for i in range(0, detections.shape[2]):

        # extract the confidence (i.e., probability) associated with
        # the detection
        confidence = detections[0, 0, i, 2]

        # threshold score
        if confidence > 0.5: # "confidence":
            # compute the (x, y)-coordinates of the bounding box for
            # the object
            box = detections[0, 0, i, 3:7] * np.array([w, h, w, h])
            (startX, startY, endX, endY) = box.astype("int")

            # ensure the bounding boxes fall within the dimensions of
            # the frame
            (startX, startY) = (max(0, startX), max(0, startY))
            (endX, endY) = (min(w - 1, endX), min(h - 1, endY))

```

```

# extract the face ROI, convert it from BGR to RGB channel
# ordering, resize it to 224x224, and preprocess it
face = frame[startY:endY, startX:endX]
face = cv2.cvtColor(face, cv2.COLOR_BGR2RGB)
face = cv2.resize(face, (224, 224))
face = img_to_array(face)
face = preprocess_input(face)

# add the face and bounding boxes to their respective
# lists
faces.append(face)
locs.append((startX, startY, endX, endY))

# only make a predictions if at least one face was detected
if len(faces) > 0:
    # for faster inference we'll make batch predictions on *all*
    # faces at the same time rather than one-by-one predictions
    # in the above `for` loop
    faces = np.array(faces, dtype="float32")
    preds = maskNet.predict(faces, batch_size=32)

# return a 2-tuple of the face locations and their corresponding
# locations
return (locs, preds)

# -----
if __name__ == '__main__':
    labels_dict={0:'without mask', 1:'mask'}
    color_dict={0:(0,255,0), 1:(0,0,255)}

    # load our serialized face detector model from disk
    print("[INFO] loading face detector model...")
    prototxtPath = os.path.sep.join(['models/face_detector', 'deploy.prototxt'])
    weightsPath = os.path.sep.join(['models/face_detector',
        'res10_300x300_ssd_iter_140000.caffemodel'])


```

```

faceNet = cv2.dnn.readNet(prototxtPath, weightsPath)
maskNet = load_model('models/mask_detector/mask_detector.model')

onmask = 0
speech = Speech()
#speech.start()

# print('starting video captures..')
webcam = cv2.VideoCapture(0) #Use camera 0

# resize image to speed analyse
resize = 1/4

# loop over the frames from the video stream
while True:

    # grab the frame from the threaded video stream and resize it
    (rval, im) = webcam.read()

    #resize and flip mirror
    im = cv2.flip(im, 1)

    (locs, preds) = detect_and_predict_mask(im, faceNet, maskNet)

    # loop over the detected face locations and their corresponding
    # locations
    for (box, pred) in zip(locs, preds):
        # unpack the bounding box and predictions
        (startX, startY, endX, endY) = box
        (mask, withoutMask) = pred

        # determine the class label and color we'll use to draw
        # the bounding box and text
        if mask > withoutMask:
            displayTxt = 'Masked'
            label = 0 #'mask'
            onmask += 1
            if onmask >= 10:

```

```

onmask = 0
txt = 'Silahkan Masuk'
sp = threading.Thread(target=speech.say, args=(txt,))
sp.start()
# repeat threading several times, does it eat more memories?
# inspect it or try another method

else:
    displayTxt = 'No Mask'
    label = 1 #'without mask'
    onmask -= 1
    if onmask <= -10:
        onmask = 0
        txt = 'Harap gunakan masker sekarang'
        sp = threading.Thread(target=speech.say, args=(txt,))
        sp.start()

displayTxt = "{}: {:.2f}%".format(displayTxt, max(mask, withoutMask)
* 100)

cv2.putText(im, displayTxt, (startX, startY - 10),
            cv2.FONT_HERSHEY_SIMPLEX, 0.9, color_dict[label], 2)
cv2.rectangle(im,(startX,startY),(endX,endY),color_dict[label],2)

# show the output frame
cv2.imshow("TI-3C Kelompok 7 Annisa_Yudas", im)

# if the `q` key was pressed, break from the loop
if cv2.waitKey(1) & 0xFF == ord("q"):
    break

try:
    sp.join()
except:
    pass

webcam.release()
cv2.destroyAllWindows()

```

4. PENGGUNAAN

Untuk menggunakan aplikasi langkah – langkah yang harus dilakukan adalah :

- a. Buka vscode, kemudian buka terminal.
- b. Ketik pada terminal: python3 <nama_file>.py , misal python3 mask_detect.py.
- c. Setelah itu, akan muncul window baru, dan kamera device akan menyala secara otomatis.
- d. pastikan wajah terdeteksi pada frame, kemudian deteksi akan segera berjalan.
- e. Jika tidak menggunakan masker, maka border akan berwarna merah, kemudian akan muncul suara “Harap gunakan masker sekarang”.
- f. Sebaliknya jika menggunakan masker, maka border akan berwarna hijau, kemudian akan muncul suara “Silahkan Masuk”.