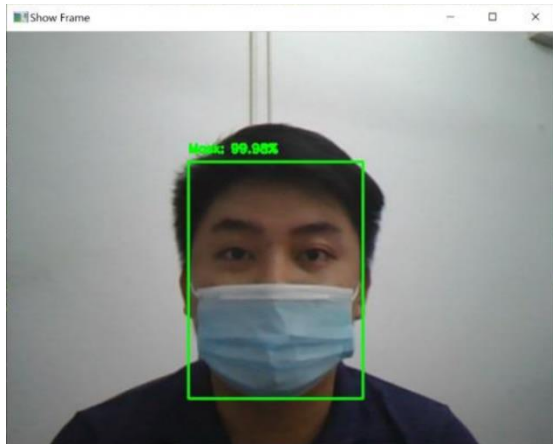
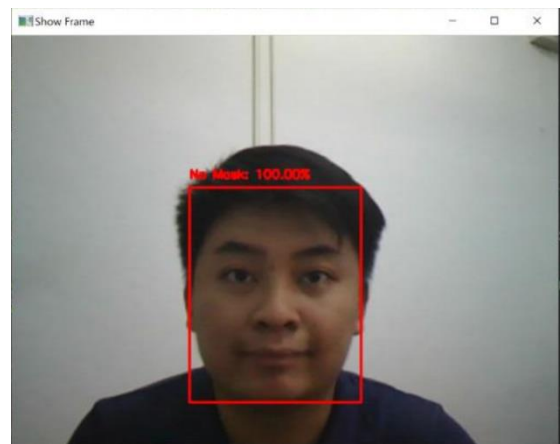


ຂັ້ນຕອນການຕິດຕັ້ງວິທີໃຊ້ການກວດຈັບ ຄົນໃສ່ຜ້າອັດປາກ ແລະ ຄົນບໍ່ໃສ່ຜ້າອັດປາກ ໂດຍໃຊ້ພາສາ Python.



ຄົນໃສ່ຜ້າອັດປາກ

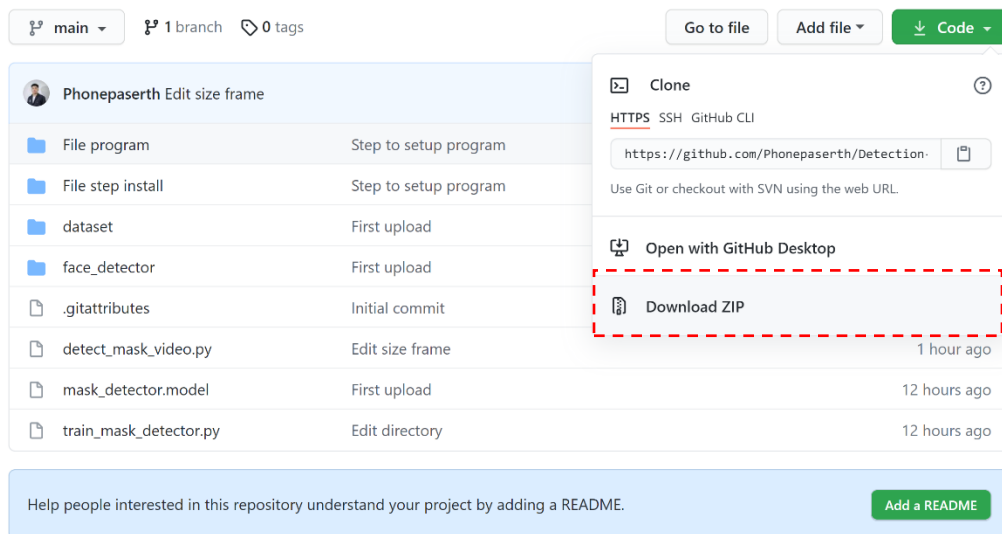


ຄົນບໍ່ໃສ່ຜ້າອັດປາກ

ໃນບົດການທົດລອງ ການກວດຈັບ ຄົນໃສ່ຜ້າອັດປາກ ແລະ ຄົນບໍ່ໃສ່ຜ້າອັດປາກ ໂດຍໃຊ້ພາສາ Python, ເປັນພາສາ ຫຼັກໃນການປະມວນຜົນ ແລະ ອຸປະກອນທີ່ໃຊ້ຮ່ວມຄື: Keras, Tensorflow, MobileNet ແລະ OpenCV.

- ຂັ້ນຕອນທີ 1

ດາວໂຫຼດຝາຍທີ່ຈະຕ້ອງກະກຽມໃນ Github: <https://github.com/Phonepaserth/Detection-Face-Mask>



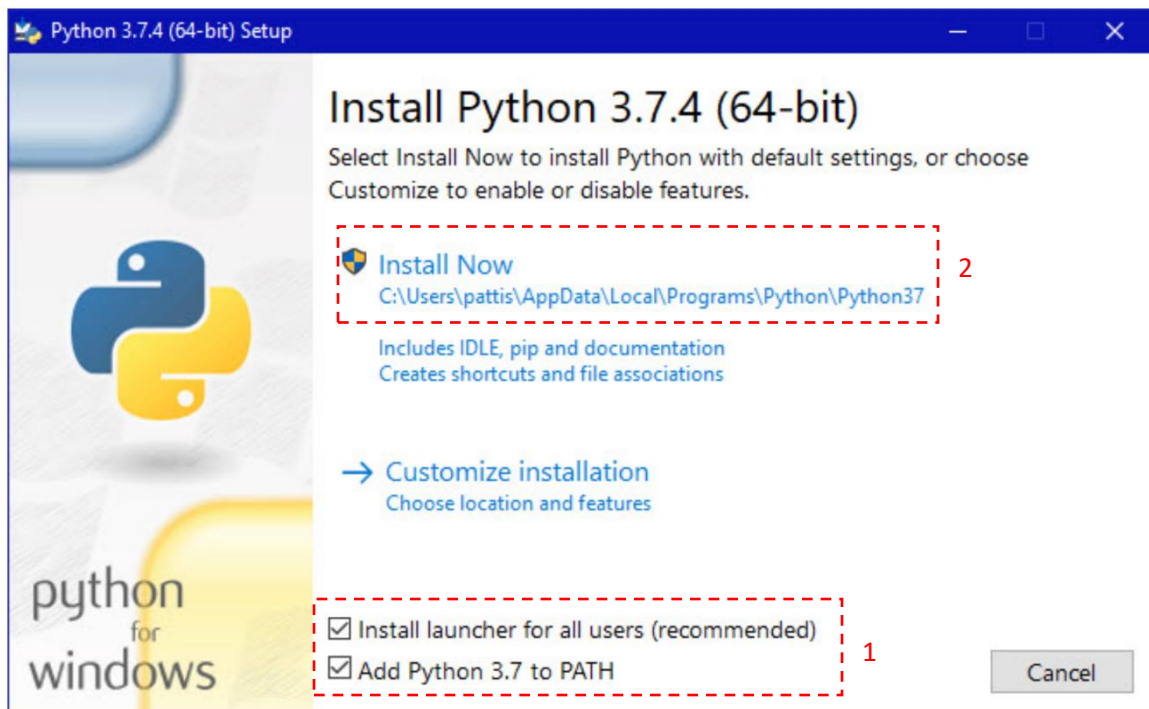
- ຂັ້ນຕອນທີ 2

ຝາຍຫຼັງ ດາວໂຫຼດຝາຍເປັນທີ່ຮຽບຮ້ອຍໃຫ້ແຕກຝາຍໄວໃນບ່ອນເກັບທີ່ຕ້ອງການ

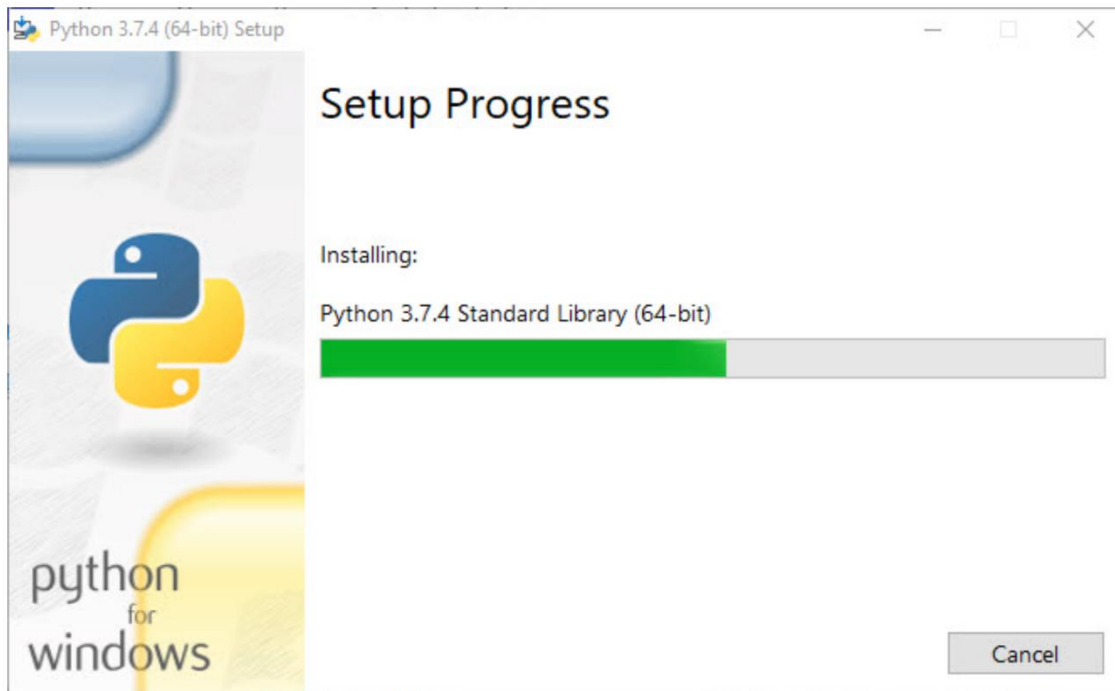
Name	Date modified	Type	Size
.git	6/21/2021 12:51 PM	File folder	
dataset	6/21/2021 1:31 AM	File folder	
face_detector	6/21/2021 1:31 AM	File folder	
File program	6/21/2021 2:05 AM	File folder	
File step install	6/21/2021 2:06 AM	File folder	
.gitattributes	6/21/2021 1:39 AM	Text Document	1 KB
detect_mask_video.py	6/21/2021 12:47 PM	Python File	5 KB
mask_detector.model	6/21/2021 12:07 AM	MODEL File	11,215 KB
train_mask_detector.py	6/21/2021 1:48 AM	Python File	5 KB

- ຂັ້ນຕອນທີ 3

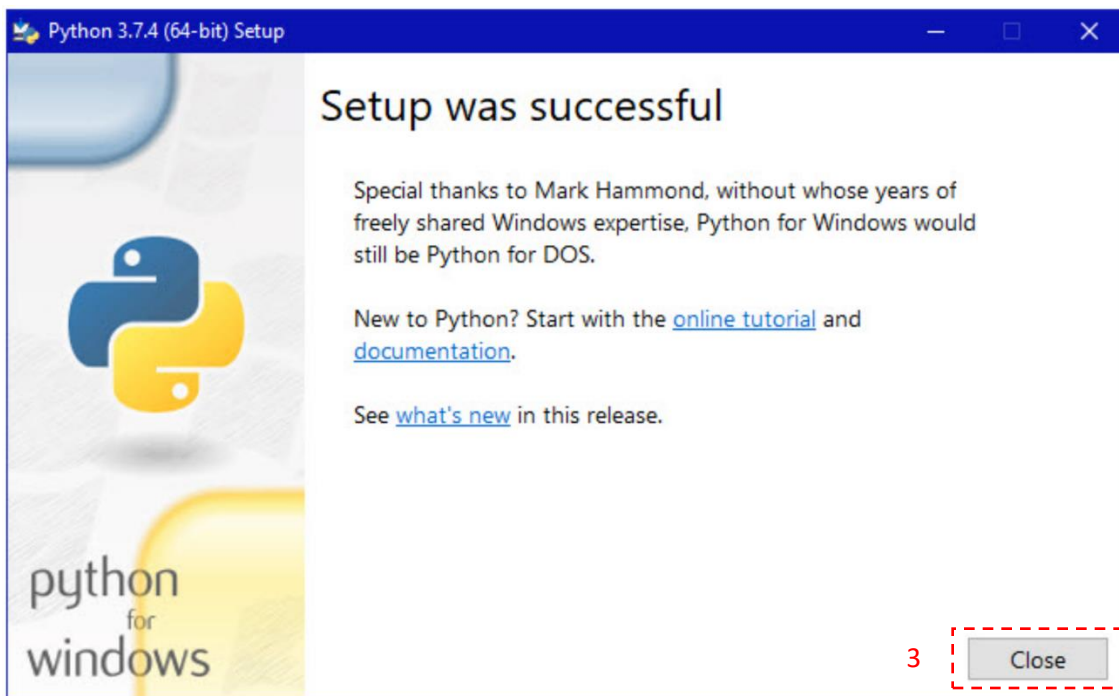
ຕິດຕັ້ງ Python ລົງໃນອຸປະກອນຂອງທ່ານ ໄປທີ່ File program ແລ້ວກົດຕິດຕັ້ງ python-3.9.5-amd64.exe



1. ກົດເລືອກ Install launcher for all users (recommended) ແລະ Add Python to PATH.
2. ກົດທີ່ Install Now ແລ້ວທຳການລໍຖ້າ.



3. ພາຍຫຼັງຕິດຕັດເປັນທີ່ຮຽບຮ້ອຍໃຫ້ກົດປຸ່ມ Cancel.



ໝາຍເຫດ: ຂັ້ນຕອນໂດຍລະອຽດ Link:

<https://www.ics.uci.edu/~pattis/common/handouts/pythoneclipsejava/python.html>

- **ຂັ້ນຕອນທີ 4**

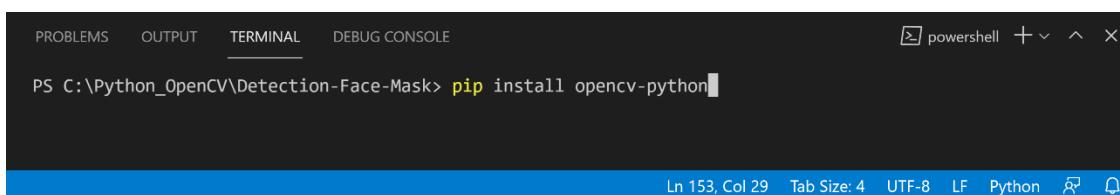
ຕິດຕັ້ງອຸປະກອນ Library ຕ່າງໆໂດຍໃຊ້ VS Code ແລະ ໃຊ້ຄໍາສັ່ງຜ່ານ Command ຄື pip install (library ທີ່ຕ້ອງການ).

ໄປທີ່ Directory: File setup install ແລ້ວໄປທີ່ Setup.txt ໃຊ້ command ຕາມຂັ້ນຕອນ.

ລາຍການທີ່ຕ້ອງຕິດຕັ້ງ

- opencv-python
- keras
- imutils
- numpy
- matplotlib
- scipy
- tensorflow
- wav-win-sound

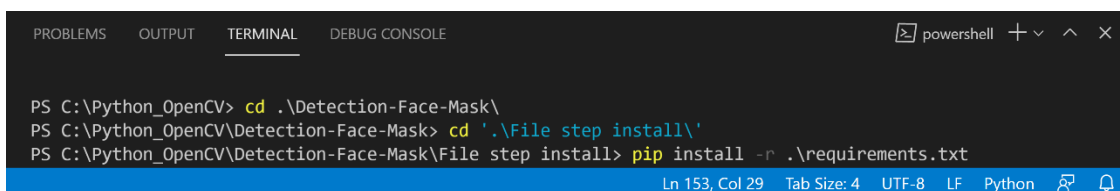
ຕົວຢ່າງ: ການໃຊ້ command ໃນການຕິດຕັ້ງ pip install opencv-python



```
PROBLEMS OUTPUT TERMINAL DEBUG CONSOLE powershell + ^ x
PS C:\Python_OpenCV\Detection-Face-Mask> pip install opencv-python
```

ໝາຍເຫດ: ການຕິດຕັ້ງໃນຄັ້ງດຽວ

ໄປທີ່ Directory: File setup install ແລ້ວໃຊ້ Command: pip install -r .\requirements.txt



```
PROBLEMS OUTPUT TERMINAL DEBUG CONSOLE powershell + ^ x
PS C:\Python_OpenCV> cd .\Detection-Face-Mask\
PS C:\Python_OpenCV\Detection-Face-Mask> cd '.\File step install\'
PS C:\Python_OpenCV\Detection-Face-Mask\File step install> pip install -r .\requirements.txt
```

- ຂັ້ນຕອນທີ 5

ກ່ອນທີ່ຈະ RUN System ຕ້ອງທຳການສ້າງ Model ເປີດ File: train_mask_detector.py

ແກ້ໄຂ Code ໂດຍໄດ້ທຶດແຖວທີ 29 ທຳການແກ້ໄຂ Directory ທີ່ຢູ່ dataset ໃຫ້ຕົງກັບຕຳແໜ່ງຂອງ dataset ໃນອຸປະກອນຂອງທ່ານ.

ຕົວຢ່າງ: ຕຳແໜ່ງຂອງ dataset

- ໃນອຸປະກອນຕົວຢ່າງແມ່ນຢູ່ Directory: C:\Python_OpenCV\Detection-Face-Mask\dataset
- ຫຼັງຈາກນັ້ນແມ່ນໃຊ້ Command: python .\train_mask_detector.py ເພື່ອໃຫ້ອຸປະຮຽນຮູ້ຈາກຮູບພາບທີ່ຢູ່ໃນ dataset ແລ້ວສ້າງເປັນ File model: mask_detector.model

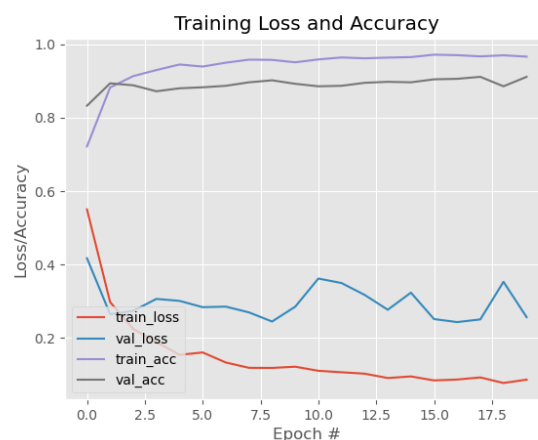
```

21 import os
22
23 # initialize the initial learning rate, number of epochs to train for,
24 # and batch size
25 INIT_LR = 1e-4
26 EPOCHS = 20
27 BS = 32
28
29 DIRECTORY = r"C:\Python_OpenCV\Detection-Face-Mask\dataset"
30 CATEGORIES = ["with_mask", "without_mask"]
31
32 # grab the list of images in our dataset directory, then initialize
33 # the list of data (i.e., images) and class images
34 print("[INFO] loading images...")
35
36 data = []
37 labels = []
38

```

PS C:\Python_OpenCV\Detection-Face-Mask> python .\train_mask_detector.py

- ເມື່ອອຸປະກອນສ້າງເປັນ model ເປັນທີ່ຮຽບຮ້ອຍ ຈະສະແດງຜົນການຮຽນຮູ້ຂອງອຸປະກອນ.



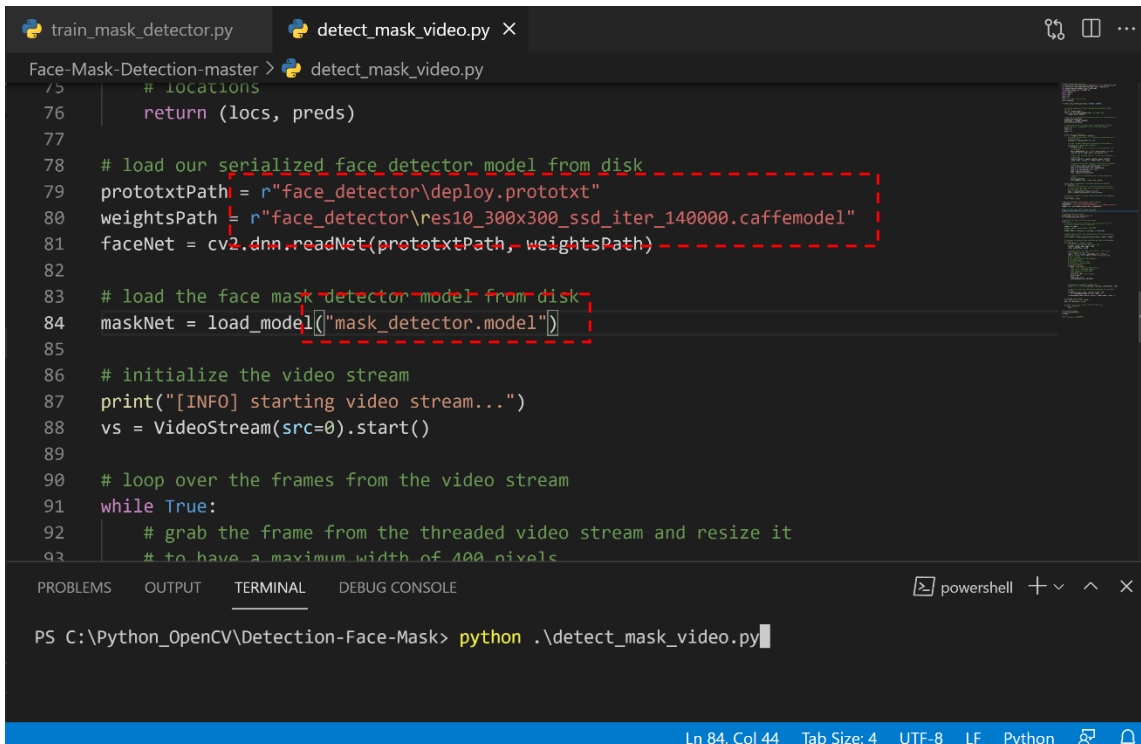
- **ຂັ້ນຕອນທີ 6**

ກ່ອນທີ່ຈະ RUN System ຫຼືເປີດໃຊ້ລະບົບ ເປີດ Directory: detect_mask_video.py

ແກ້ໄຂ Code ໂດຍໄດ້ທີ່ແຖວທີ 79 ໃຫ້ Link ກັບ deploy.prototxt ທີ່ຢູ່ໃນ Folder: face_detector

ແກ້ໄຂ Code ໂດຍໄດ້ທີ່ແຖວທີ 80 ໃຫ້ Link ກັບ res10_300x300_ssd_iter_140000.caffemodel ທີ່ຢູ່ໃນ Folder: face_detector

ແກ້ໄຂ Code ໂດຍໄດ້ທີ່ແຖວທີ 84 ໃຫ້ Link ກັບ mask_detector.model ທີ່ທ່ານສ້າງຂຶ້ນມາ.



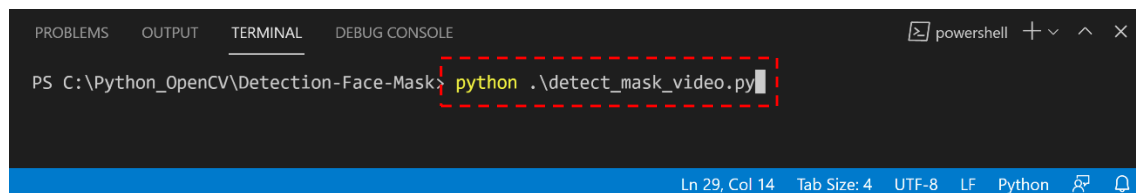
```
Face-Mask-Detection-master > detect_mask_video.py
75 # locations
76 return (locs, preds)
77
78 # load our serialized face detector model from disk
79 prototxtPath = r"face_detector\deploy.prototxt"
80 weightsPath = r"face_detector\res10_300x300_ssd_iter_140000.caffemodel"
81 faceNet = cv2.dnn.readNet(prototxtPath, weightsPath)
82
83 # load the face mask detector model from disk
84 maskNet = load_model(["mask_detector.model"])
85
86 # initialize the video stream
87 print("[INFO] starting video stream...")
88 vs = VideoStream(src=0).start()
89
90 # loop over the frames from the video stream
91 while True:
92     # grab the frame from the threaded video stream and resize it
93     # to have a maximum width of 400 pixels
```

PROBLEMS OUTPUT TERMINAL DEBUG CONSOLE

PS C:\Python_OpenCV\Detection-Face-Mask> python .\detect_mask_video.py

Ln 84, Col 44 Tab Size: 4 UTF-8 LF Python

ເມື່ອແກ້ໄຂ Code ເປັນທີ່ຮຽບຮ້ອຍໃຫ້ໃຊ້ Command: python .\detect_mask_video.py

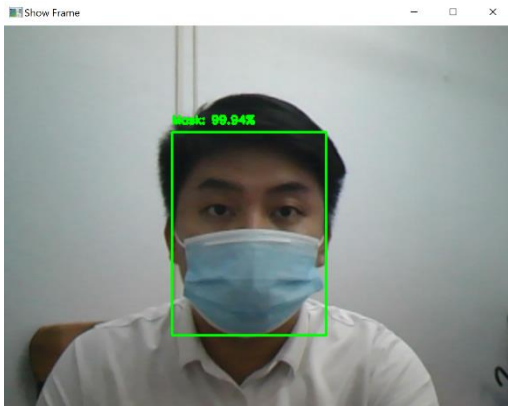


PROBLEMS OUTPUT TERMINAL DEBUG CONSOLE

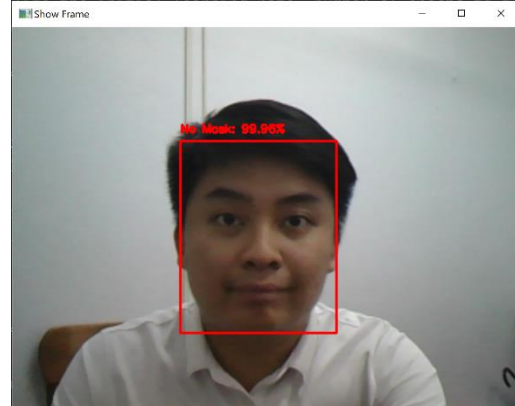
PS C:\Python_OpenCV\Detection-Face-Mask> python .\detect_mask_video.py

Ln 29, Col 14 Tab Size: 4 UTF-8 LF Python

- ຜົນໄດ້ຮັບເມື່ອ Run System



ຄົນໃສ່ຜ້າອັດປາກ



ຄົນບໍ່ໃສ່ຜ້າອັດປາກ

ຜົນໄດ້ຮັບແມ່ນ ອຸປະກອນຈະສາມາດຈຳແນກໄດ້ວ່າ ບຸກຄົນໃດໃສ່ຜ້າອັດປາກ ຫລືບໍ່ໃສ່ຜ້າອັດປາກ ເມື່ອຖ້າບໍ່ໃສ່ຈະມີສຽງໄວ້ແຈ້ງເຕືອນ (ສາມາດວິເຄາະໃບໜ້າພ້ອມກັນໄດ້ຫຼາຍກວ່າໜຶ່ງຄົນ).

ບົດການທົດລອງໃນຄັ້ງນີ້ແມ່ນສ້າງຂຶ້ນເພື່ອໃຊ້ເປັນປະໂຫຍດໃນຊ່ວງຂອງ COVID-19 ແລະຫວັງວ່າບົດການທົດລອງໃນຄັ້ງນີ້ຈະມີປະໂຫຍດຕໍ່ສັງຄົມລະທຸກຄົນ. ສຸດທ້າຍນີ້ ຖ້າຜິດພາດປະການໃດຂໍໂທດຂໍອະໄພ ແລະພ້ອມຮັບທຸກຄຳຕຳນິຕິຊົມ ຂໍຂອບໃຈ.

ໂດຍ ທ້າວ ພອນປະເສີດ ສີໃສແກ້ວ.

ນັກສຶກສາ: ມະຫາວິທະຍາໄລ ແຫ່ງຊາດລາວ National University of Laos; ຄະນະວິສະວະກຳສາດ Faculty of Engineering; ພາກວິຊາ ວິສະວະກຳຄອມພິວເຕີ ແລະ ເຕັກໂນໂລຊີຂໍ້ມູນຂ່າວສານ Department of Computer Engineering and Information Technology; ສາຂາ ວິສະວະກຳຄອມພິວເຕີ Major of Computer Engineering.

Facebook: Phonepaserth SISAYKEO

E-Mail: Phonepaserth.ssk@gmail.com

ກວດແກ້ໂດຍ ອາຈານ ແສງລັດສະໝີ ຈັນທະມິນາວົງ.

ຈົບແລ້ວ