

รายงาน

เรื่อง

Crop image to classify by python programing

จัดทำโดย

นางสาวจุฑาภรณ์ สิมมะลี	600510537
------------------------	-----------

นายณัฐกร เมษพันธุ์	600510545
--------------------	-----------

นำเสนอ

ผศ.ดร.วาริน เชาวทัต

รายงานนี้เป็นส่วนหนึ่งของวิชา 204382 คอมพิวเตอร์กราฟฟิก (Computer Graphics)

ภาคเรียนที่ 2 ปีการศึกษา 2562

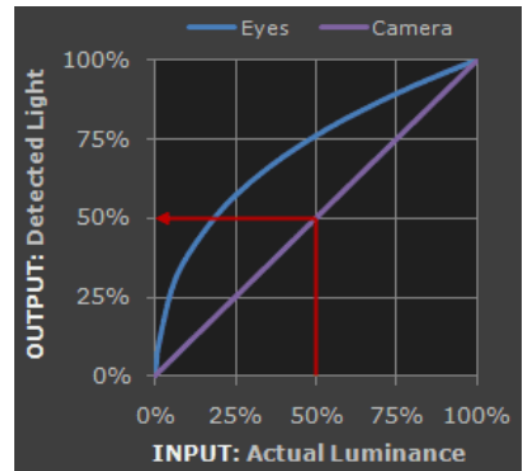
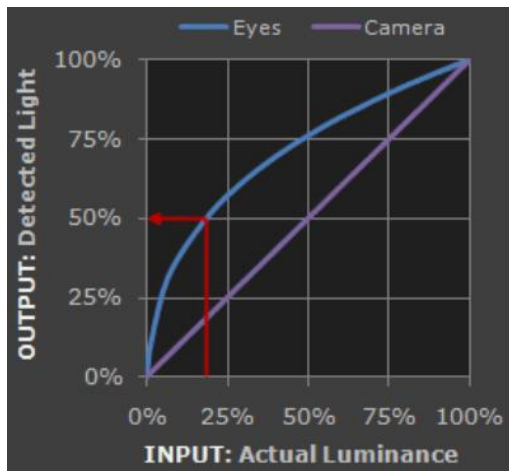
มหาวิทยาลัยเชียงใหม่

Crop image to classify by python programming

ใช้เทคนิค Gamma Correction, rescale image และ CNN Image Classification using CIFAR-10 dataset

Gamma Correction

ตามความเป็นจริงของ แคลสสีที่ถูกกล้องดิจิตอลจับไว้ได้ จะเห็นไม่ตรงกับที่ตาเรา(หรือในจอมอนิเตอร์) จึงต้องมีระบบที่เรียกว่า gamma correction ในการปรับแคลสสีให้ใกล้เคียงกับสีที่มนุษย์มองเห็นดังนี้



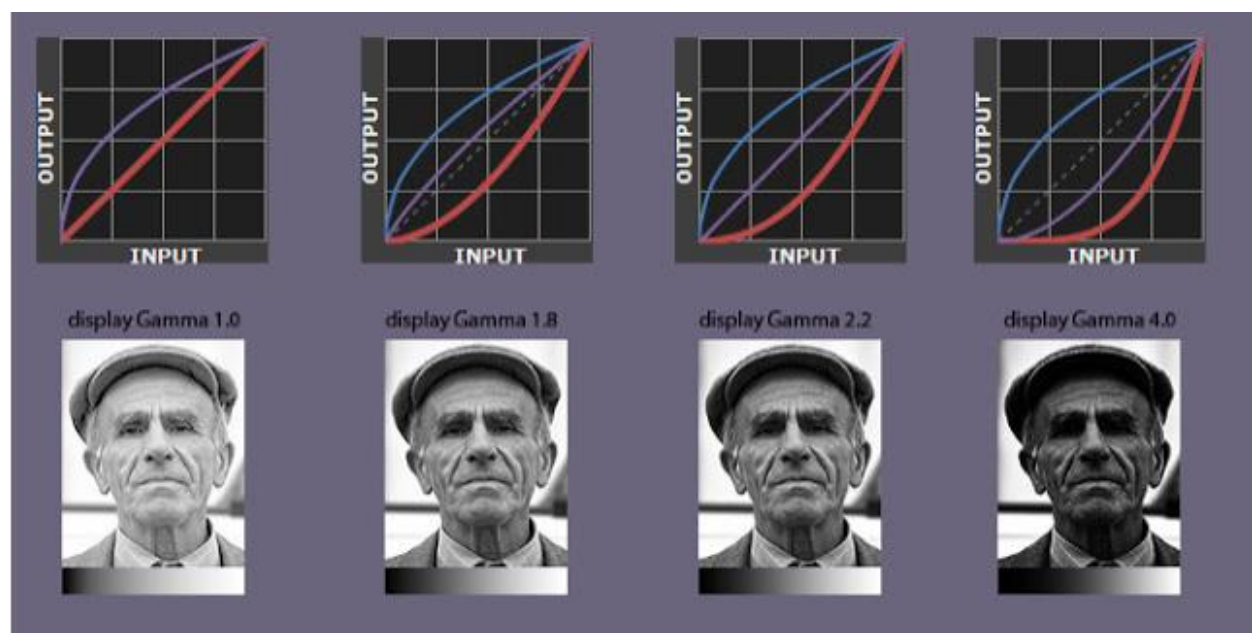
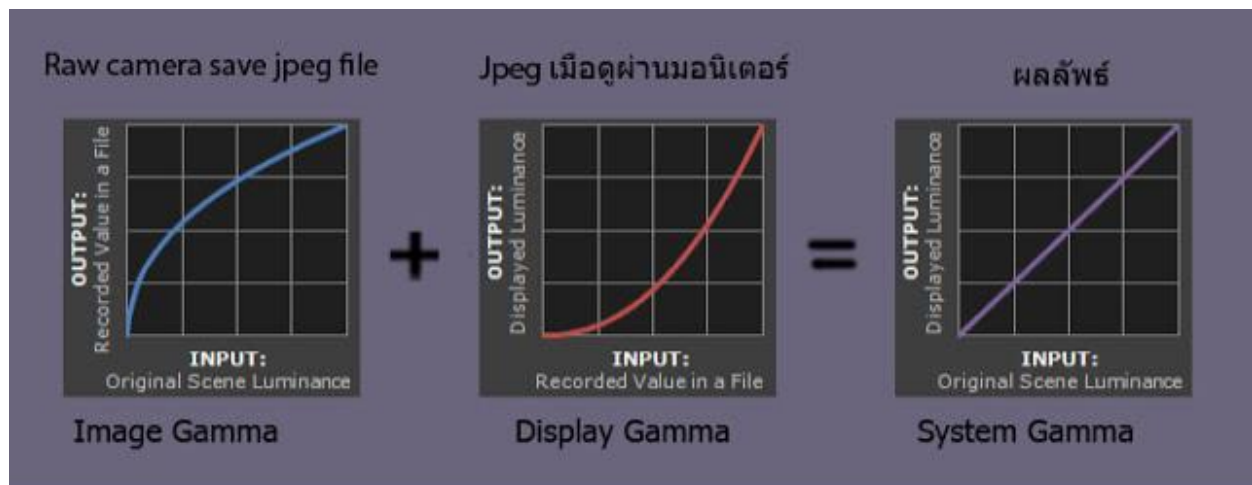
สีที่ตาเห็น



สีที่กล้องจับได้



ที่ 50% หรือสีเทาตามมนุษย์จะเห็นแสงได้มืดกว่ากล้องถ่ายรูปอย่างเห็นได้ชัดเทียบกันกับกล้องแล้วตามนุษย์อ่อนไหวต่อแสงในส่วนโทนที่มืดได้มากกว่า จากเหตุผลในข้อนี้ทำให้ทัศนวิสัยในการมองของเราจะมีช่วงรับแสงที่กว้างกว่า ไม่อย่างนั้นความสว่างในที่โล่งแจ้งจะมากเกินไปกว่าตาจะรับได้



จากรูป เส้นสีฟ้าคือค่ามาตรฐาน Gamma, เส้นสีแดงคือการปรับ display ให้จอภาพ, เส้นสีม่วงคือผลที่เกิดขึ้น

ที่มา: [https://alwayyours.blogspot.com/2013/02/gamma-correction-](https://alwayyours.blogspot.com/2013/02/gamma-correction-22.html?fbclid=IwAR1Pjyv3xd0x5QFwWRt71So_plFOH07XhGr1mGiUJy7eAYraauO6ndvsVnM)

22.html?fbclid=IwAR1Pjyv3xd0x5QFwWRt71So_plFOH07XhGr1mGiUJy7eAYraauO6ndvsVnM

Module/ Library

Module หรือ library ที่จำเป็นสำหรับการติดตั้งเพื่อให้โปรแกรมสามารถใช้งานได้

1. tkinter

- ใช้สำหรับเขียนโปรแกรมเพื่อสร้าง Graphical User Interface (GUI) program
- **install packages in command line:** pip install tk-tools

2. numpy

- ใช้สำหรับเขียนโปรแกรมในคำนวณทางคณิตศาสตร์และวิทยาศาสตร์ เช่น การคำนวณ matrix เป็นต้น
- **install packages in command line:** pip install numpy

3. OpenCV (cv2)

- ใช้สำหรับเขียนโปรแกรมหรือพัฒนาซอฟต์แวร์ให้สามารถประมวลผลภาพได้ ยกตัวอย่าง เช่น ระบบตรวจจับใบหน้า (Face Detection) หรือการใช้ในด้านของ Machine Learning เป็นต้น
- **install packages in command line:** pip install opencv-python

4. keras

- ใช้สำหรับการเทรนดัดข้อมูลในการ classifiy image
- **install packages in command line:** pip install keras

5. matplotlib

- ใช้สำหรับการ plot รูปภาพให้ออกมา
- **install packages in command line:** pip install matplotlib

6. skimage (transform import resize)

- ใช้สำหรับการ resize รูปภาพไม่ให้ใหญ่จนเกินไป
- **install packages in command line:** pip install scikit-image

7. pandas

- มีความสามารถสำหรับจัดเตรียม ทำความสะอาด เก็บกวาดข้อมูลไว้สำหรับการทำ Data Visualization และสร้าง Model ต่อไป
- **install packages in command line:** pip install pandas

8. sklearn

- เป็นโมดูลย่อยของโมดูล Scikit-learn โดย Scikit-learn เป็นโมดูลสำหรับใช้ทำ Machine Learning และ Data Mining ในภาษา Python
- **install packages in command line:** pip install sklearn

9. tensorflow

- ใช้เพื่อพัฒนา machine learning และ deep learning โดยเฉพาะ
- **install packages in command line:** pip install tensorflow

การติดตั้งและใช้งานโปรแกรม

1. วิธีการติดตั้งโปรแกรม Python

- Version requirements: python: 3.5 – 3.7
- ดาวน์โหลดไฟล์ติดตั้ง: <https://www.python.org/downloads/>

2. วิธีการติดตั้ง Module/ Library ของ python

- เปิด Command line (cmd)
- เข้าไปยัง Path: C:\Users\Admin\AppData\Local\Programs\Python\Python37\Scripts หรือโฟลเดอร์ปลายทางที่ติดตั้งโปรแกรม
- พิมพ์ชื่อ Package ที่จำเป็นสำหรับการติดตั้งเพื่อให้โปรแกรมสามารถใช้งานได้
- Syntax: “pip install <Name package>”

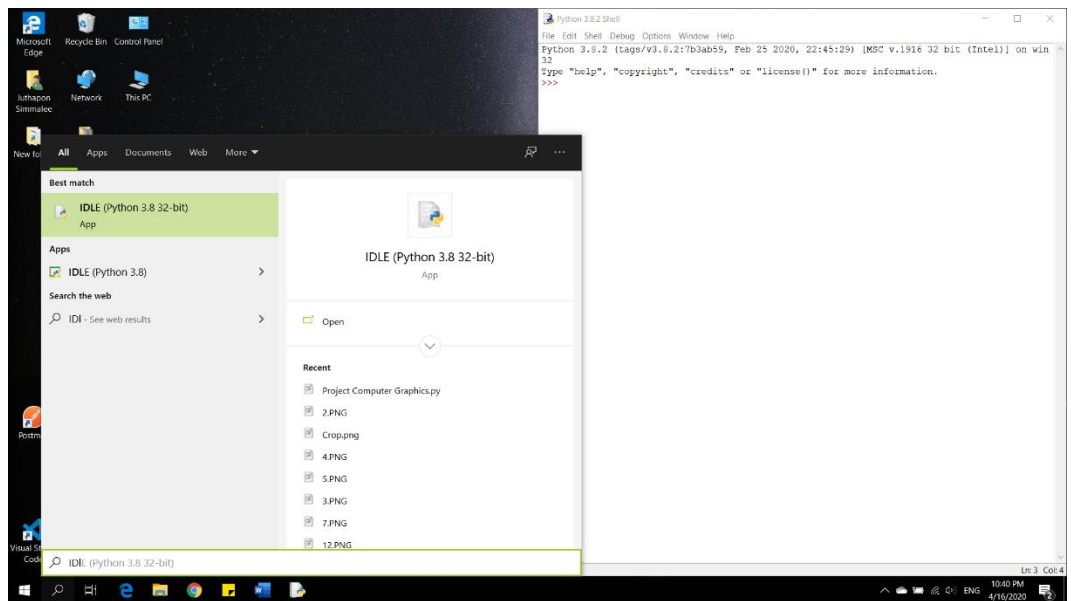
ตัวอย่างดังรูป

```
C:\Users\Admin\AppData\Local\Programs\Python\Python37\Scripts>pip install tensorflow
```

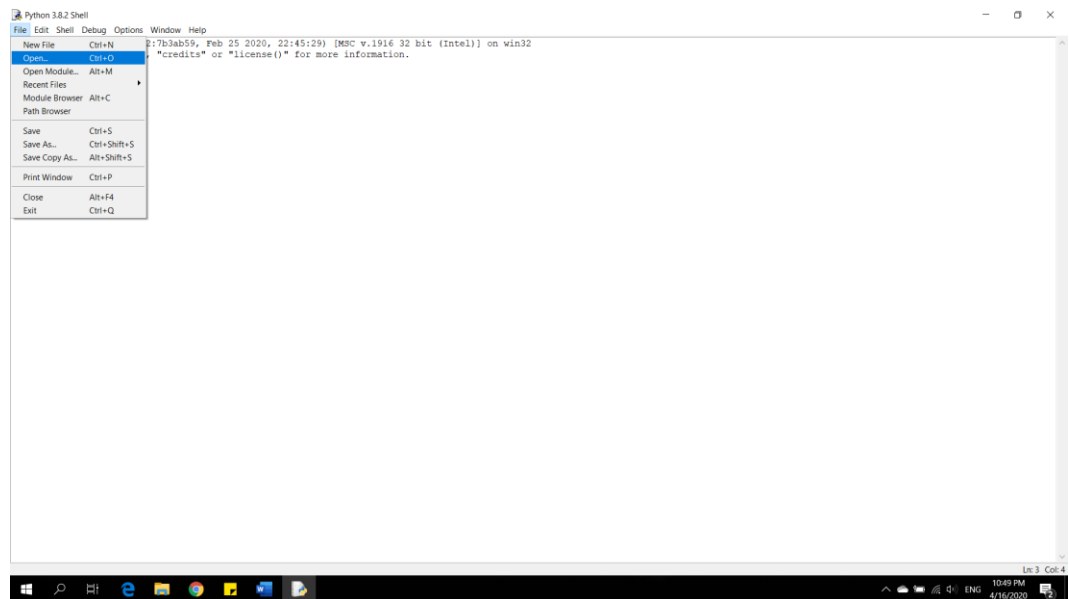
- กด Enter เพื่อทำการติดตั้ง Package
- เพิ่มเติม: วิธีการ check package ให้พิมพ์ “pip list”

3. วิธีการใช้งานโปรแกรม

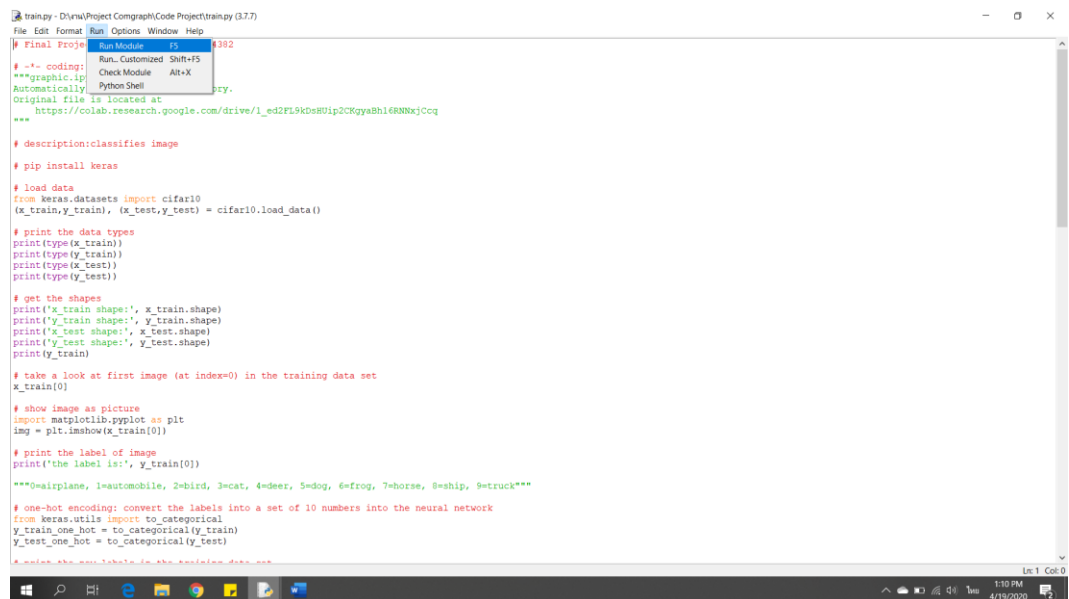
3.1. เปิด source code ของ project จากโปรแกรม IDLE (Python) หรือ Text editor อื่นๆ ที่สามารถรันภาษา python ได้ ซึ่งในการยกตัวอย่างนี้เราจะทำการรันตัว source code จาก IDLE Python



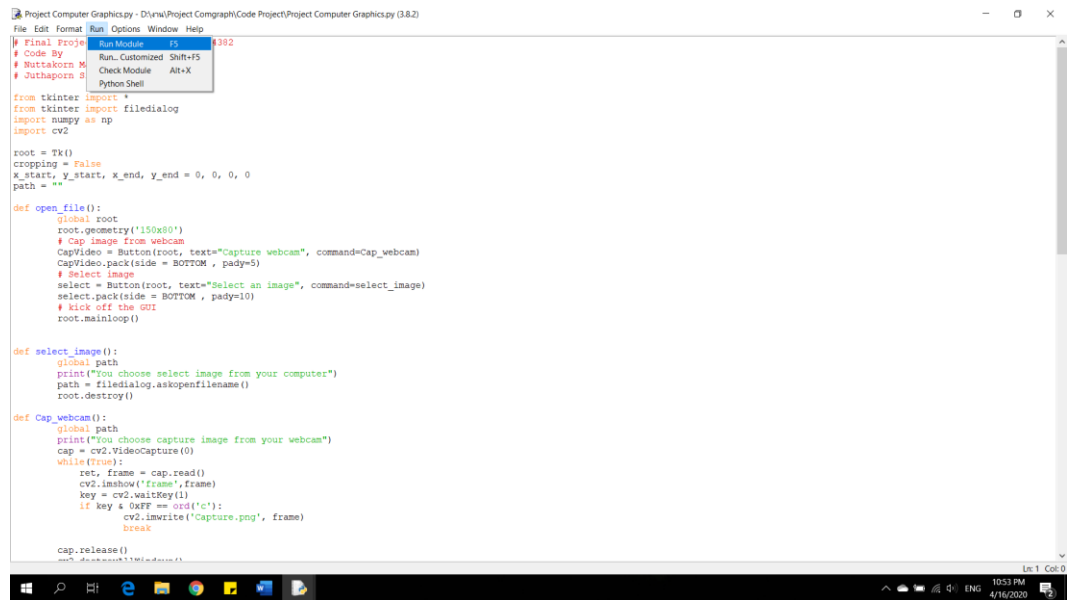
3.2. ทำการเปิดไฟล์ source code ของ project บน IDLE Python



3.3. เปิดไฟล์ train.py เพื่อทำการโหลดข้อมูลและเทรนดโมเดล CNN ในการ classify image รวจนกระทั่งเสร็จ (เฉพาะเมื่อต้องการเพิ่มการเทรนข้อมูลใหม่) แต่ถ้าหากมี model ที่ทำการเทรนไว้แล้วก็ไม่จำเป็นต้อง train โปรแกรม



- 3.4. เมื่อเปิดไฟล์ “Project Computer Graphics” แล้วให้คลิกที่ Tab menu > Run > Run Module หรือกดปุ่ม F5 เพื่อรันตัว source code ให้โปรแกรมทำงาน



The screenshot shows a Python IDE window titled "Project Computer Graphics.py - D:\na\Project Comgraph\Code Project\Project Computer Graphics.py (3.8.2)". The "Run" menu is open, showing options: "Run", "Run Customized Shift+F5", "Check Module Alt+X", and "Python Shell". The source code in the background includes imports for tkinter, filedialog, numpy, and cv2. It defines functions for opening a file, selecting an image, and capturing a webcam image. The main loop calls these functions.

```
File Edit Format Run Options Window Help
Run Customized Shift+F5
Check Module Alt+X
Python Shell

from tkinter import *
from tkinter import filedialog
import numpy as np
import cv2

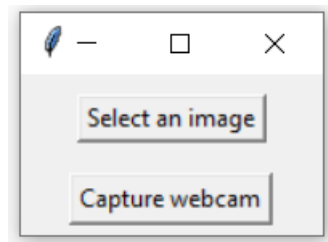
root = Tk()
cropping = False
x_start, y_start, x_end, y_end = 0, 0, 0, 0
path = ""

def open_file():
    global root
    root.geometry('150x80')
    # Cap image from webcam
    CapVideo = Button(root, text="Capture webcam", command=Cap_webcam)
    CapVideo.pack(side = BOTTOM , pady=5)
    # Select image
    select = Button(root, text="select an image", command=select_image)
    select.pack(side = BOTTOM , pady=10)
    # Kick off the GUI
    root.mainloop()

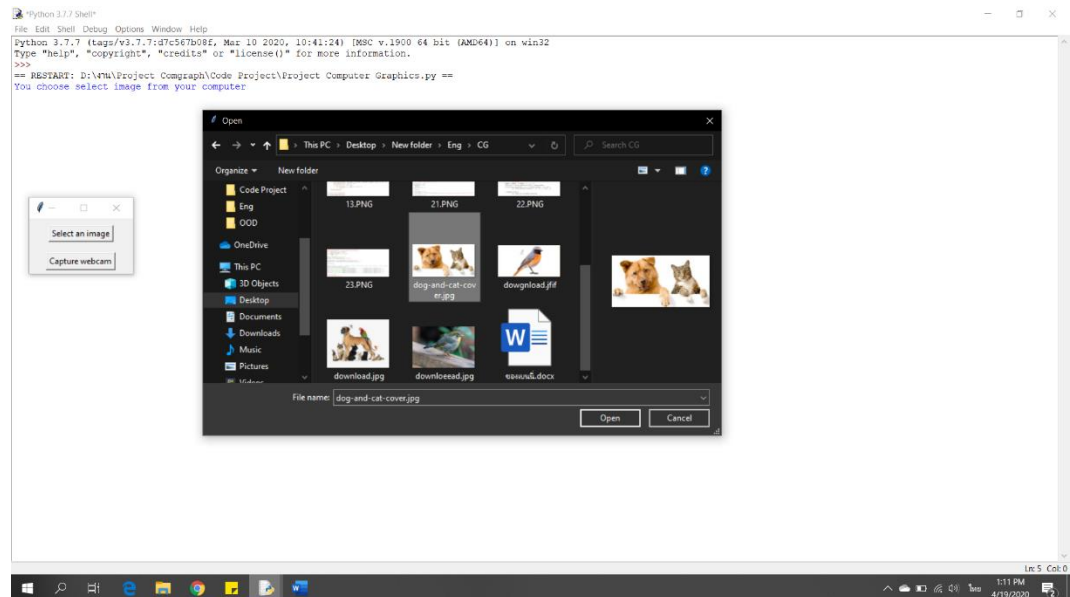
def select_image():
    global path
    print("You choose select image from your computer")
    path = filedialog.askopenfilename()
    root.destroy()

def Cap_webcam():
    global path
    print("You choose capture image from your webcam")
    cap = cv2.VideoCapture(0)
    while(True):
        ret, frame = cap.read()
        cv2.imshow('frame',frame)
        key = cv2.waitKey(1)
        if key & 0xFF == ord('c'):
            cv2.imwrite('Capture.png', frame)
            break
    cap.release()
    root.destroy()
```

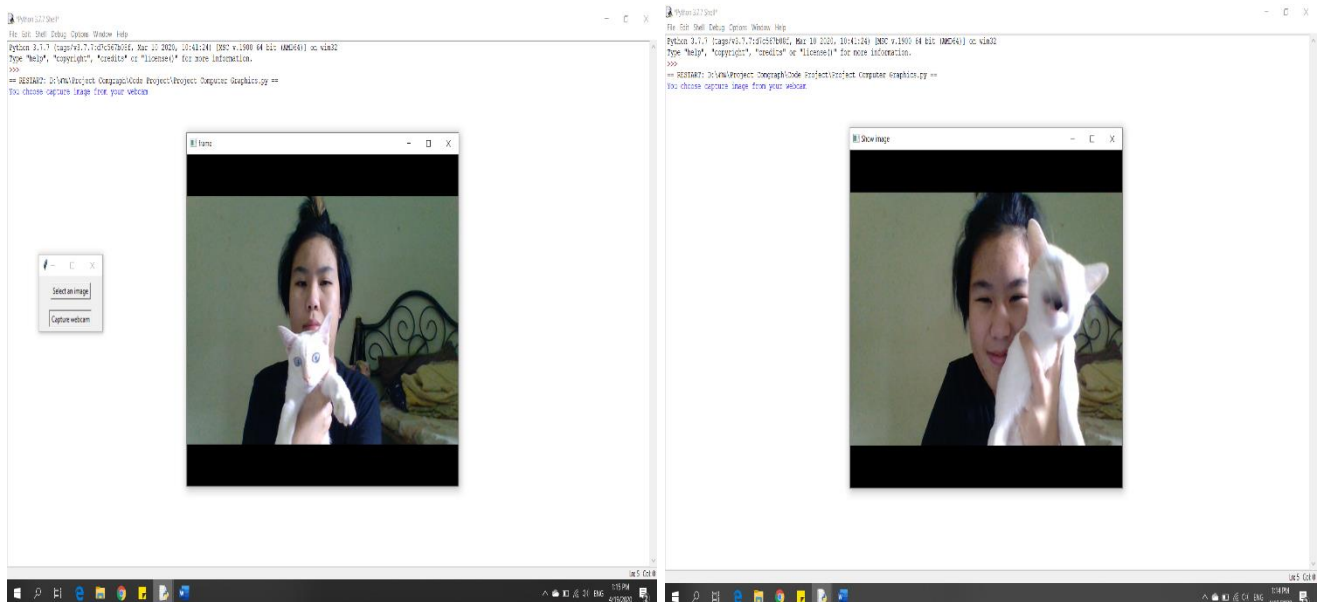
- 3.5. หน้าตัวโปรแกรมเมื่อเปิดขึ้นมาจะมีให้ผู้ใช้ (User) เลือกฟังก์ชันการใช้งาน 2 อย่างคือ Select an image และ Capture webcam ดังรูป



3.5.1. ถ้าผู้ใช้เลือกปุ่ม Select an image ตัวโปรแกรมจะแสดง Open file dialog เพื่อให้ผู้ใช้สามารถเลือกรูปภาพจากคอมพิวเตอร์ของผู้ใช้ได้

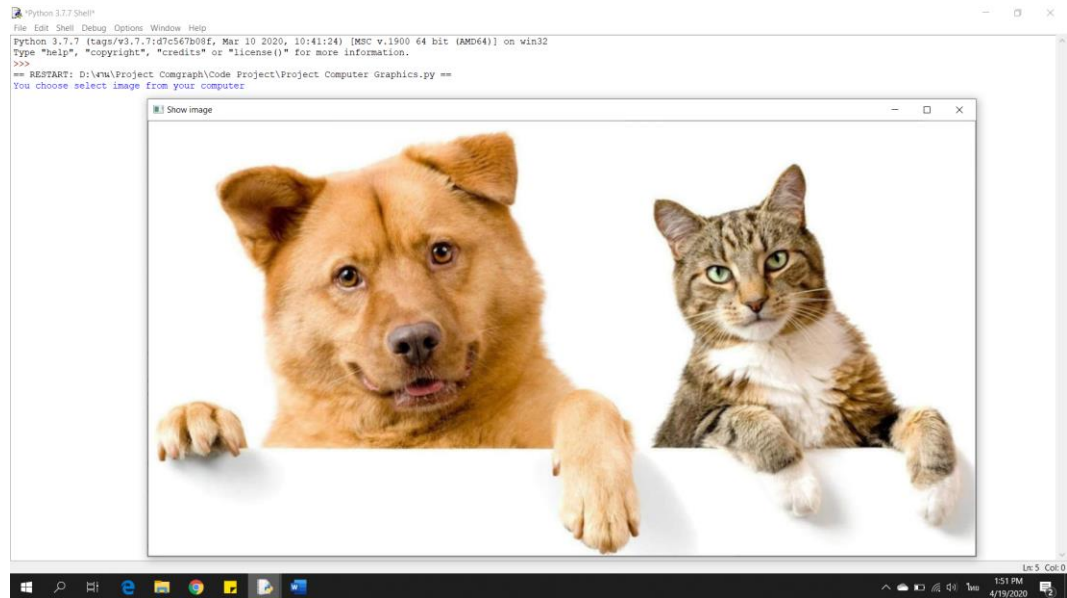


3.5.2. ถ้าผู้ใช้เลือกปุ่ม Capture webcam ตัวโปรแกรมจะเปิดกล้องจากคอมพิวเตอร์ของผู้ใช้และแสดงหน้าต่างขณะกล้องกำลังเปิดอยู่ หากผู้ใช้ต้องการถ่ายภาพให้กดคีย์บอร์ดอักษรตัว 'c' เพื่อทำการถ่ายรูปจากกล้องคอมพิวเตอร์ของผู้ใช้

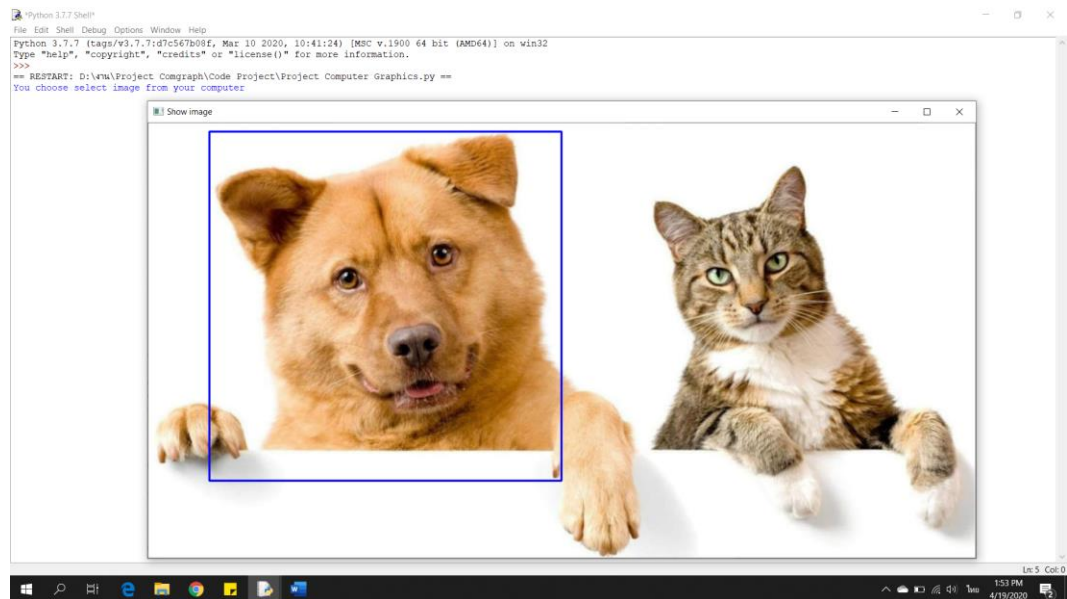


เพิ่มเติม: หากต้องการหยุดการทำงานของกล้องกด 'q' โปรแกรมจะหยุดทำงานทันที

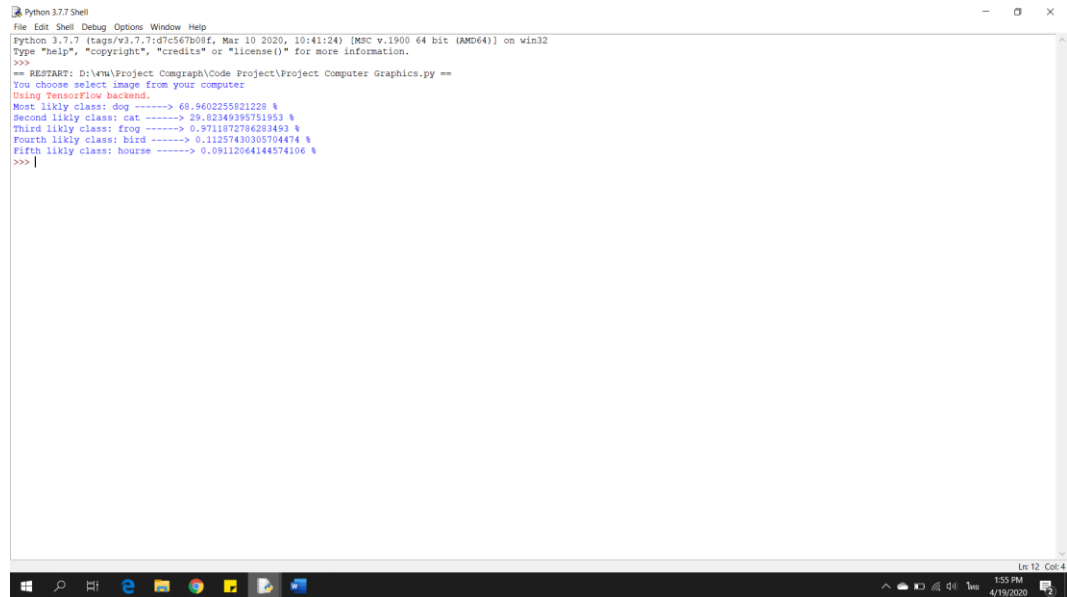
- 3.6. เมื่อผู้ใช้ทำการเลือกรูปภาพจากคอมพิวเตอร์ (Select an image) หรือจากการถ่ายรูปจากกล้องคอมพิวเตอร์ (Capture webcam) โปรแกรมจะทำการแสดงภาพนั้นขึ้นมา



- 3.7. ต่อมาผู้ใช้จะทำการ Crop ส่วนที่ต้องการในรูปภาพ โดยการใช้เมาส์คลิกลากกรอบสี่เหลี่ยมไปบนรูปภาพในโปรแกรมเพื่อ crop ส่วนที่ต้องการ



- 3.8. หลังจาก Crop ภาพส่วนที่ต้องการแล้วให้ผู้ใช้ต้องการถ่ายภาพให้กดคีย์บอร์ดอักขรตัว ‘q’ เพื่อทำการปิดโปรแกรมการครอบภาพ และรอจนกว่าตัวโปรแกรมจะทำงานเสร็จ จะพบผลลัพธ์การ classify image จากส่วนที่ครอบปรากฏขึ้น ว่ามีความน่าจะเป็นเท่าไรและในรูปอะไร



```
Python 3.7.7 Shell
File Edit Shell Debug Options Window Help
Python 3.7.7 (tags/v3.7.4:fc567b0ff, Mar 10 2020, 10:41:24) [MSC v.1900 64 bit (AMD64)] on win32
Type "help", "copyright", "credits" or "license()" for more information.
>>>
== RESTART: D:\venv\Project Comgraph\Code Project\Project Computer Graphics.py ==
You choose select image from your computer
Using TensorFlow backend.
Most likely class: dog -----> 60.9602255021220 %
Second likely class: cat -----> 29.82349395751953 %
Third likely class: frog -----> 0.9711872786283493 %
Fourth likely class: bird -----> 0.11257430305704474 %
Fifth likely class: house -----> 0.09112064144574104 %
>>> |
```

ข้อควรระวัง

- Path ที่ใช้ในการเลือกรูปภาพไม่ควรมีโฟลเดอร์ที่เป็นภาษาไทย เนื่องจากตัวโมดูลของ OpenCv จะไม่สามารถอ่าน path ที่มีภาษาไทยแล้วทำการ cv2.imread(path) ได้
- สามารถ classify เบื้องต้นได้ตามตารางด้านล่างนี้เท่านั้น หากอยากทำการ classify ให้หลากหลาย ต้องทำการหาข้อมูลแล้วเทรนดโมเดลในโปรแกรม train.py เพิ่ม

Source code

แสดง source code ใน Project Computer Graphics.py และอธิบายการทำงานในแต่ละส่วนของ

โปรแกรม

```
from tkinter import *
from tkinter import filedialog
import numpy as np
import cv2

root = Tk()
cropping = False
x_start, y_start, x_end, y_end = 0, 0, 0, 0
path = ""

def open_file():
    global root
    root.geometry('150x80')
    # Cap image from webcam
    CapVideo = Button(root, text="Capture webcam", command=Cap_webcam)
    CapVideo.pack(side = BOTTOM , pady=5)
    # Select image
    select = Button(root, text="Select an image", command=select_image)
    select.pack(side = BOTTOM , pady=10)
    # kick off the GUI
    root.mainloop()

def select_image():
    global path
    print("You choose select image from your computer")
    path = filedialog.askopenfilename()
    root.destroy()

def Cap_webcam():
    global path
    print("You choose capture image from your webcam")
    cap = cv2.VideoCapture(0)
    while(True):
        ret, frame = cap.read()
        cv2.imshow('frame', frame)
        key = cv2.waitKey(1)
        if key & 0xFF == ord('c'):
            cv2.imwrite('Capture.png', frame)
            break
        elif key & 0xFF == ord('q'):
            cv2.destroyAllWindows()
            exit()
            break
    cap.release()
    cv2.destroyAllWindows()
    path = 'Capture.png'
    root.destroy()
```

ส่วนที่ 1

ในการสร้างหน้าต่างโปรแกรมเพื่อสร้างปุ่มเพื่อเรียกใช้ในการเปิดไฟล์รูปภาพจากคอมพิวเตอร์ เราจะใช้โมดูลที่มีชื่อว่า tkinter โดยใช้คำสั่ง `“from tkinter import *”` เพื่อเรียกใช้ฟังก์ชันทั้งหมดของ tkinter และใช้คำสั่ง `“from tkinter import filedialog”` เพื่อเรียกใช้ฟังก์ชันในการเปิดกล่องสำหรับเรียกไฟล์ขึ้นมา (open file dialog)

ต่อมาเราจะใช้คำสั่ง `“root = Tk()”` เพื่อสร้างหน้าต่างโปรแกรมที่สามารถเรียกใช้คำสั่งในโมดูล tkinter ในส่วนของตัวแปรที่เป็นแบบ global คือ `“cropping = False”` หากยังไม่ได้ครอบรูปจะมีค่าเริ่มต้นเป็นเท็จ, `“x_start, y_start, x_end, y_end = 0, 0, 0, 0”` ค่าเริ่มต้นในการกำหนดขอบเขตของการครอบ และ `“path = ”` สำหรับเก็บที่อยู่ของไฟล์ที่เรียกในรูปแบบ String

- ฟังก์ชัน `“open_file()”` เพื่อทำการสร้างปุ่มในการเรียกฟังก์ชัน `“select_image()”` และ `“Cap_webcam()”` โดยใช้คำสั่ง `“command = (ชื่อฟังก์ชันที่ต้องการเรียก)”`
- ฟังก์ชัน `“select_image()”` เพื่อเปิดหน้าต่างสำหรับการเรียกไฟล์จากคอมพิวเตอร์โดยใช้คำสั่ง `“filedialog.askopenfilename()”`
- ฟังก์ชัน `“Cap_webcam()”` เพื่อใช้ในการเปิดตัวกล้องคอมพิวเตอร์หรือโน้ตบุ๊ก โดยใช้คำสั่ง `“cv2.VideoCapture(0)”` ความหมายของเลข 0 การเปิดกล้องแบบ real time ส่วนเลข 1 คือการเปิดกล้องโดยที่จะทำการจับภาพนิ่งแล้ว โดยการจะจับภาพจากกล้องจะใช้คำสั่ง `“if key & 0xFF == ord('c')”` เพื่อจับภาพและสร้างเป็นไฟล์รูปภาพขึ้นมาโดยใช้คำสั่ง `“cv2.imwrite('Capture.png', frame)”`

```

def mouse_crop(event, x, y, flags, param):
    # grab references to the global variables
    global x_start, y_start, x_end, y_end, cropping
    # if the left mouse button was DOWN, start RECORDING
    # (x, y) coordinates and indicate that cropping is being
    if event == cv2.EVENT_LBUTTONDOWN:
        x_start, y_start, x_end, y_end = x, y, x, y
        cropping = True

    # Mouse is Moving
    elif event == cv2.EVENT_MOUSEMOVE:
        if cropping == True:
            x_end, y_end = x, y

    # if the left mouse button was released
    elif event == cv2.EVENT_LBUTTONUP:
        # record the ending (x, y) coordinates
        x_end, y_end = x, y
        cropping = False # cropping is finished
        refPoint = [(x_start, y_start), (x_end, y_end)]

        if len(refPoint) == 2: # when two points were found
            roi = oriImage[refPoint[0][1]:refPoint[1][1], refPoint[0][0]:refPoint[1][0]]
            cv2.imwrite('Crop.png', roi)
            cv2.imshow("Cropped", roi)

```

ส่วนที่ 2

ฟังก์ชัน “`mouse_crop(event, x, y, flags, param)`” เป็นฟังก์ชันที่ถูกเรียกมาจากคำสั่ง “`cv2.setMouseCallback("Show image", mouse_crop)`” เพื่อเก็บขอบเขตในการครอบรูปลูกจากการกระทำของเมาส์

เงื่อนไขในแต่ละส่วน

- “`if event == cv2.EVENT_LBUTTONDOWN`”
คือถ้ามีการเริ่มกดที่เมาส์ซ้ายปุ่มซ้ายจะเริ่มการบันทึกพิกัด
- “`elif event == cv2.EVENT_MOUSEMOVE`”
คือถ้าเมาส์กำลังมีการเคลื่อนที่
- “`elif event == cv2.EVENT_LBUTTONUP`”
คือถ้าปุ่มซ้ายของเมาส์ถูกปล่อยออกจะทำการบันทึกพิกัดที่สิ้นสุด
และถ้าสองจุดมาพบกันจะทำการครอบรูปลูกภาพ

```
##### Open file #####

open_file()

##### crop function #####
image = cv2.imread(path)
oriImage = image.copy()
cv2.namedWindow("Show image")
cv2.setMouseCallback("Show image", mouse_crop)

while True:
    i = image.copy()
    if not cropping:
        cv2.imshow("Show image", image)
        # press key 'q' if close
        if cv2.waitKey(1) & 0xFF == ord('q'):
            break
    elif cropping:
        cv2.rectangle(i, (x_start, y_start), (x_end, y_end), (255, 0, 0), 2)
        cv2.imshow("Show image", i)
        # press key 'q' if close
        if cv2.waitKey(1) & 0xFF == ord('q'):
            break
    cv2.waitKey(1)

# Close all cv2 Windows
cv2.destroyAllWindows()
```

ส่วนที่ 3

คำสั่ง “open_file()” เพื่อเรียกใช้ฟังก์ชันในเปิดหน้าต่างเพื่อเลือกรูปภาพหรือแคปเจอร์จากกล้องคอมพิวเตอร์หรือโน้ตบุ๊ก

ต่อมาจะทำการอ่านรูปภาพจาก path โดยใช้คำสั่ง “cv2.imread(path)” และใช้คำสั่ง “cv2.setMouseCallback("Show image", mouse_crop)” เพื่อทำการครอบรูปภาพ

ถ้าหาก cropping ยังมีสถานะเป็น False ก็ยังจะเปิดหน้าต่างแสดงรูปภาพเพื่อที่จะสามารถเลือกครอบรูปภาพใหม่ได้ต่อไปจนกว่าจะได้รูปที่พอใจ

แต่ถ้า cropping ยังมีสถานะเป็น True ถ้าแสดงกรอบสีน้ำเงินในขณะที่กำลังคลิกเมาส์ครอบรูป แล้วเมื่อเมาส์ปล่อยก็จะทำการเปิดหน้าต่างรูปที่ครอบขึ้นมา

```
##### increase gamma #####
# from __future__ import print_function
import numpy as np
# import argparse
import cv2
def adjust_gamma(image, gamma=1.0):
    invGamma = 1.0 / gamma
    table = np.array([(i / 255.0) ** invGamma) * 255
                      for i in np.arange(0, 256)]).astype("uint8")
    return cv2.LUT(image, table)

class myImage:
    def __init__(self, img_name):
        self.img=cv2.imread(img_name)
        self.__name=img_name
    def __str__(self):
        return self.__name
    # example
    # x = MyImage('1.jpg')
    # str(x) => 1.jpg
    # x.img => numpy array store img

x = myImage("Crop.png")
if x.img.mean() > 127:
    save=0.0
    filename="Crop.png"
else:
    save=1.5

if x.img.shape[1] > 450:
    scale_percent= int((450*100)/x.img.shape[1])# percent of original size if width > 450 make to 450 & cal %
else:
    scale_percent = 100
width = int(x.img.shape[1] * scale_percent / 100)
height = int(x.img.shape[0] * scale_percent / 100)
dim = (width, height)
```

ส่วนที่ 4

ส่วนนี้เป็นการเขียน function, การปรับสี Gamma, โหลดรูปภาพที่ได้จากการ crop แล้วที่ชื่อ “Crop.png” โดยเช็คขนาดภาพแล้วกำหนด scale ที่ต้องการก่อน ต่อมาจึงกำหนดค่า Gamma ที่ต้องการเซฟรูปจากการเช็คว่ารูปร่างที่ใส่เข้ามานั้นมีค่าเฉลี่ยของ Pixel ทั้งหมดมากกว่า 127 หรือไม่ ถ้ามากกว่าแสดงว่าภาพที่ใส่เข้ามาเป็นภาพที่มีความสว่างเพียงพอ ถ้าน้อยกว่าแสดงว่าภาพที่ใส่เข้ามานั้นมีความสว่างน้อยจึงกำหนดค่า Gamma ที่ต้องการเพิ่มเท่ากับ 1.5


```
# resize image
resized = cv2.resize(x.img, dim, interpolation = cv2.INTER_AREA)
original = resized

for gamma in np.arange(0.0, 2.0, 0.5):
    if gamma == 1:
        continue
    gamma = gamma if gamma > 0 else 0.1
    adjusted = adjust_gamma(original, gamma=gamma)
    cv2.putText(adjusted, "g={}".format(gamma), (10, 30),
                cv2.FONT_HERSHEY_SIMPLEX, 0.8, (0, 0, 255), 3)

    if (gamma==save):
        cv2.imwrite(filename,adjusted)
```

ส่วนที่ 5

หลังจากกำหนด scale ของรูปภาพแล้วก็จะทำการปรับขนาดรูปภาพให้เป็นไปตาม scale ที่กำหนดไว้ และนำรูปภาพที่ปรับ scale แล้วไปปรับสี Gamma ของภาพต่อ หลังจากนั้นก็จะทำการเพิ่ม Gamma และเซฟรูปภาพที่มีค่า Gamma = 1.5

```
##### Classify #####
from keras.models import load_model
import matplotlib.pyplot as plt
from skimage.transform import resize
import numpy as np

model = load_model('my_model.h5')
my_image = plt.imread(filename)# read file

my_image_resize = resize(my_image, (32,32,3))# resize
probabilities = model.predict(np.array([my_image_resize,] ))

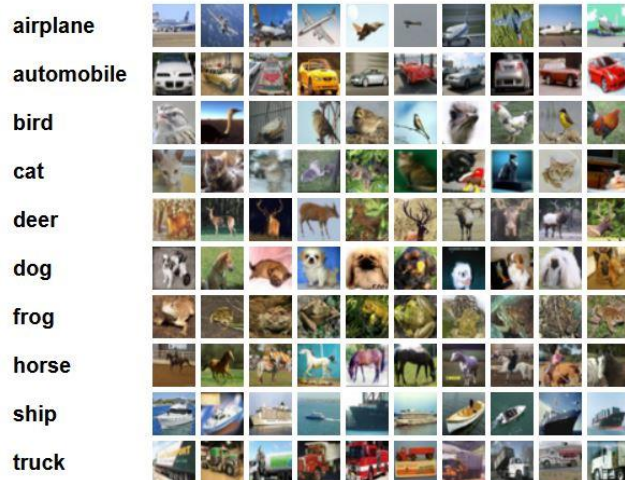
probabilities

number_to_class = ['airplan', 'automobile', 'bird', 'cat', 'deer', 'dog', 'frog', 'hourse', 'ship', 'truck']
index = np.argsort(probabilities[0,:])
print('Most likly class:',number_to_class[index[9]], '----->', probabilities[0, index[9]]*100, '%')
print('Second likly class:',number_to_class[index[8]], '----->', probabilities[0, index[8]]*100, '%')
print('Third likly class:',number_to_class[index[7]], '----->', probabilities[0, index[7]]*100, '%')
print('Fourth likly class:',number_to_class[index[6]], '----->', probabilities[0, index[6]]*100, '%')
print('Fifth likly class:',number_to_class[index[5]], '----->', probabilities[0, index[5]]*100, '%')

# save the model
# model.save('my_model.h5')
```

ส่วนที่ 6

เมื่อได้ภาพที่มีความสว่างเพียงพอแล้ว ขั้นตอนต่อมาคือการนำภาพนั้นไปใส่ในโมเดลที่เทรนด้วยข้อมูลไว้ล่วงหน้าเพื่อ classify image ว่ารูปที่ใส่เข้ามานั้นเป็นรูปอะไร โดยแสดงค่าความน่าจะเป็นและผลลัพธ์ที่คำนวณออกมา 5 อันดับแรก โดยขอบเขตของการ classify มีดังนี้



การอ้างอิงถึง Source code ที่นำมาประยุกต์ใช้ใน Term project

1. Open File Dialog

การเขียนโปรแกรมเพื่อทำการสร้าง Open file dialog สำหรับการเลือกรูปภาพจากเครื่องคอมพิวเตอร์ของตนเอง โดยใช้ Module ที่มีชื่อว่า “tkinter”

ที่มา: <https://www.youtube.com/watch?v=iUmqLGUktek>

2. Capture Video from Camera

การเขียนโปรแกรมโดยใช้ Module ที่มีชื่อว่า “OpenCV” ในการเปิดตัวกล้อง (Webcam) เพื่อทำการบันทึกภาพลงในคอมพิวเตอร์

ที่มา:

https://github.com/cuicaihao/Webcam_QR_Detector/blob/master/Lab_02_QR_Bar_Code_Detector/Webcam.ipynb

3. Click and Crop Image

การเขียนโปรแกรมโดยใช้ Module ที่มีชื่อว่า “OpenCV” ในการแสดงรูปภาพและการ click mouse เพื่อทำการ crop รูปที่ต้องการ ร่วมกับ module ที่มีชื่อว่า “numpy” ในการคำนวณขอบเขตเพื่อให้ได้รูปภาพที่มาจาก การ crop จาก click mouse

ที่มา: <https://www.life2coding.com/crop-image-using-mouse-click-movement-python/>

4. Gamma Correction

การเขียนโปรแกรมโดยใช้ Module ที่มีชื่อว่า “OpenCV” ในการเพิ่มความสว่างให้แก่รูปภาพโดยใช้วิธีการของ image processing และใช้ร่วมกับ Module “skimage” ในการปรับขนาดรูปภาพ

ที่มา: <https://www.pyimagesearch.com/2015/10/05/opencv-gamma-correction/>

5. CNN Image Classification

การเขียนโปรแกรมโดยใช้ Module “Tensorflow, pandas, sklearn, keras” ในการทำ machine learning โดยการดึงข้อมูลจาก CIFAR-10 dataset มาใช้การการเทรนดโมเดล

ที่มา: <https://mc.ai/cnn-image-classification-using-cifar-10-dataset-on-google-colab-tpu/>

6. Resize

การเขียนโปรแกรมโดยใช้ Module “cv2” ในการปรับขนาดรูปภาพและใช้ร่วมกับ Module “matplotlib” ในการแสดงผลรูปภาพ

ที่มา: <https://www.tutorialkart.com/opencv/python/opencv-python-resize-image/>