DIGITAL IMAGE PROCESSING COURSE - 505060
PRACTICE LABS

LAB 08. Face Detection with OpenCV and Deep learning

Requirements

(1) Follow the instructions with the help from your instructor.
(2) Finish all the exercises in class and do the homework at home. You can update your solutions after class and re-submit all your work together with the homework.
(3) Grading

Total score = 50% * Attendance + 50% * Exercises

Rules:

- If the number of finished exercises is less than **80% total number of excercises**, you will get **zero** for the lab.
- Name a source file as “**src_XX.py**” where XX is the exercise number, for ex., “src_03.py” is the source code for the Exercise 3.
- Add the text of your Student ID to each of the output image.
- Name an output image as “**image_XX_YY.png**” where XX is the exercise number and YY is the order of output images in the exercise, for ex., “image_03_02.png” is the second output image in the Exercise 3.
- Submit the source code and output image files directly to Google classroom assignment, donot compress the files.

If you submit the exercises with wrong rules, you will get **zero** for the lab or the corresponding exercises.

(4) Plagiarism check

If any 2 of the students have the same output images, then all will get zero for the corresponding exercises.

INTRODUCTION

In this Lab, you will apply some image processing and deep learning techniques to build an application for detecting faces in images, such as:

- Cascade Classifier
- Multi-task Cascade Convolutional Neural Network

INSTRUCTIONS

Look at articles in the reference links below to learn more about the steps to detect faces using OpenCV and deep learning.

1. OpenCV

https://www.geeksforgeeks.org/opencv-python-program-face-detection/?ref=lbp

2. Deep learning

https://machinelearningmastery.com/how-to-perform-face-detection-with-classical-and-deep-learning-methods-in-python-with-keras/

3. Setup guide:

- python -m pip install **tensorflow==2.0**
- python -m pip install **mtcnn**

If there exists DLL file errors, install **Microsoft Visual C++ 2015 Redistributable Update 3** from https://www.microsoft.com/en-us/download/details.aspx?id=53840

If you cannot install those above packages, you can use **Google Colab**:
Insert the following codes into your Colab file:

```
!pip install mtcnn
```

```
!wget 'https://machinelearningmastery.com/wp-content/uploads/2019/03/test1.jpg&#39;
!wget 'https://machinelearningmastery.com/wp-content/uploads/2019/03/test2.jpg&#39;
```

and then change a little the method of loading image:

```
# load image from file
pixels = pyplot.imread("test1.jpg")
```

EXERCISES

Ex10.1. Build the applications to detect faces as in the two above articles in the INSTRUCTIONS section.

Ex10.2. Extract all information of bar codes in the following image (position and thickness of each line in a bar code):

ISBN 978-7-5643-5084-0
9 787564 350840 >
定价: 49.00元

Materials Science / Engineering

ISBN 978-3-319-20006-4
9 783319 200064
▸ springer.com

http://www.ndip.cn
ISBN 978-7-118-08147-3
9 787118 081473 >
定价：65.00 元

定价：48.00元
ISBN 978-7-122-27612-4
9 787122 276124 >

Submit your code in practice class.

REFERENCES

1. https://docs.opencv.org/3.4/db/d28/tutorial_cascade_classifier.html
2. https://www.geeksforgeeks.org/face-detection-using-cascade-classifier-using-opencv-python/