

Tuan2\_BatoanDulieuMNIST.ipynb

File Edit View Insert Runtime Tools Help

Comment Share

RAM Disk

+ Code + Text

21022008\_NguyenHuuTho

CHẮC CÓ KIỂM TRA

Nét chữ Kiểu chữ Góc chụp

Train : Huấn luyện Test : Kiểm thử Location

MNIST tập dữ liệu: Áo quần, chữ số => Tùy nhu cầu

[18]

from google.colab import drive  
drive.mount('/content/gdrive')

Mounted at /content/gdrive

[21]

%cd /content/gdrive/MyDrive/AI\_1333/AI\_Tuan2/Tuan2\_BatoanDulieuMNIST

/content/gdrive/MyDrive/AI\_1333/AI\_Tuan2/Tuan2\_BatoanDulieuMNIST

[22]

!pwd

/content/gdrive/MyDrive/AI\_1333/AI\_Tuan2/Tuan2\_BatoanDulieuMNIST

[ ]

#pip install opencv-python keras tensorflow

Xây dựng model nhận diện, dữ liệu được xây dựng training model là mnist Đầu tiên load dataset và load các thư viện cần dùng:

[15]

import cv2  
import numpy as np  
from skimage.feature import hog # Nền tảng AI, máy học: đưa sao học sau ra vậy, chưa học chưa ra được. Thuật toán: HOG, SVC (Machine learning), trích xuất đặc trưng ra một vector  
from sklearn.svm import LinearSVC  
from sklearn.metrics import accuracy\_score # accuracy\_score : Độ chính xác, độ tin cậy  
from keras.datasets import mnist # Deep learning, huấn luyện mô hình, train module, thư viện mnist  
!pip install --upgrade tf-nightly  
from google.colab.patches import cv2\_imshow

Tiếp theo ta sẽ tính hog. Ta dùng orientations=9,pixels\_per\_cell=(14,14),cells\_per\_block=(1,1).

[7]

(X\_train,y\_train),(X\_test,y\_test) = mnist.load\_data()  
  
X\_train\_feature = []  
for i in range(len(X\_train)):  
 feature = hog(X\_train[i],orientations=9,pixels\_per\_cell=(14,14),cells\_per\_block=(1,1),block\_norm="L2")  
 X\_train\_feature.append(feature)  
X\_train\_feature = np.array(X\_train\_feature, dtype = np.float32)  
  
#cho x\_test  
X\_test\_feature = []  
for i in range(len(X\_test)):  
 feature = hog(X\_test[i],orientations=9,pixels\_per\_cell=(14,14),cells\_per\_block=(1,1),block\_norm="L2")  
 X\_test\_feature.append(feature)  
X\_test\_feature = np.array(X\_test\_feature, dtype=np.float32)  
  
Downloading data from <https://storage.googleapis.com/tensorflow/tf-keras-datasets/mnist.npz>  
11490434/11490434 [=====] - 1s 0us/step

Tiếp theo ta build model vào predict

predict Tính toán độ chính xác mô hình

[8]

model = LinearSVC(C=10)  
model.fit(X\_train\_feature,y\_train)  
y\_pre = model.predict(X\_test\_feature)  
print(accuracy\_score(y\_test,y\_pre)) # In ra độ chính xác mô hình (độ chính xác của mô hình vừa huấn luyện)  
  
0.8988  
/usr/local/lib/python3.8/dist-packages/sklearn/svm/\_base.py:1206: ConvergenceWarning: Liblinear failed to converge, increase the number of iterations.  
warnings.warn()

Predict trên ảnh


[29]

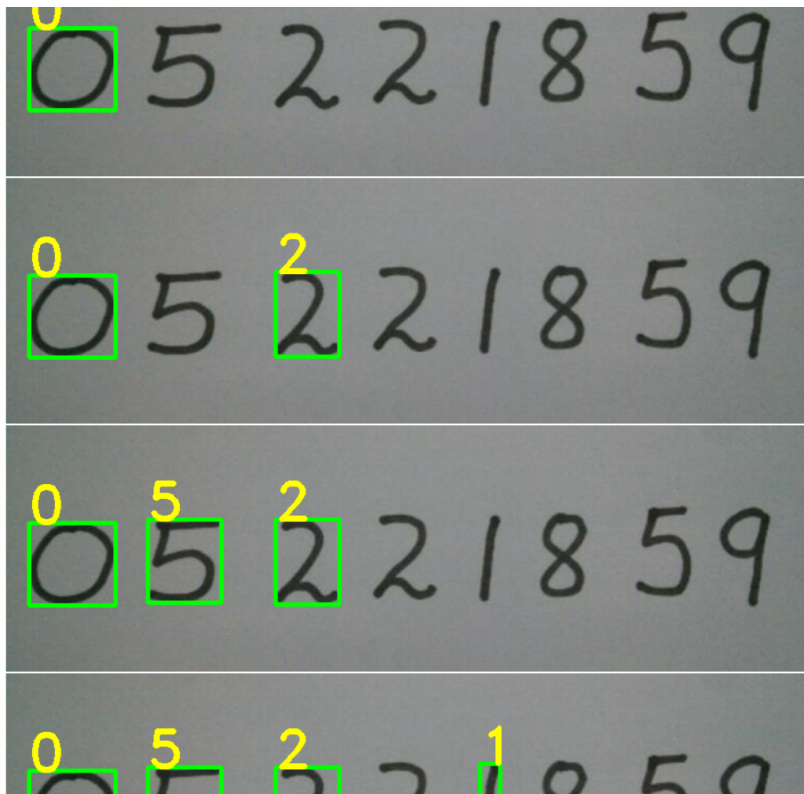
image = cv2.imread("/content/gdrive/MyDrive/AI\_1333/test\_2.png")  
# image = cv2.imread("/content/gdrive/MyDrive/AI\_1333/AI\_Tuan2/Tuan2\_BatoanDulieuMNIST/test\_img.png")  
im\_gray = cv2.cvtColor(image, cv2.COLOR\_BGR2GRAY)  
im\_blur = cv2.GaussianBlur(im\_gray, (5,5),0)  
im, thre= cv2.threshold(im\_blur,90,255,cv2.THRESH\_BINARY\_INV)  
contours, hierarchy = cv2.findContours(thre, cv2.RETR\_EXTERNAL, cv2.CHAIN\_APPROX\_SIMPLE) # Đi tìm ra những nét của chữ số, canny, edge: phát hiện cạnh biên  
rects = [cv2.boundingRect(cnt) for cnt in contours] #

Sau đó predict digit của mỗi box.

[28]

for i in contours:  
 (x,y,w,h) = cv2.boundingRect(i)  
 cv2.rectangle(image,(x,y),(x+w,y+h),(0,255,0),3)  
 roi = thre[y:y+h,x:x+w]  
 roi = np.pad(roi,(20,20),'constant',constant\_values=(0,0))  
 roi = cv2.resize(roi,(28,28),interpolation=cv2.INTER\_AREA)  
 roi = cv2.dilate(roi,(3,3))  
  
 # Calculate the HOG features - Tính toán độ chính xác mô hình  
 roi\_hog\_fd = hog(roi,orientations=9,pixels\_per\_cell=(14,14),cells\_per\_block=(1,1),block\_norm="L2")  
 nbr = model.predict(np.array([roi\_hog\_fd], np.float32)) # np.array([roi\_hog\_fd], np.float32) => Hình đang xử lý  
 cv2.putText(image, str(int(nbr[0])), (x, y), cv2.FONT\_HERSHEY\_DUPLEX, 2, (0, 255, 255), 3) # putText : Vẽ chữ lên ảnh, putText(Ảnh vẽ đè lên, str(int(nbr[0])) - để số 9 lên ảnh, (x, y) - tọa độ vẽ lên, cv2.imshow(image)  
 cv2.imwrite("image\_pand.jpg",image) # để lưu ảnh  
cv2.waitKey()  
cv2.destroyAllWindows()





### Bài tập nâng cao

- Anh chị hãy in ra độ chính xác của mô hình vừa huấn luyện? - Xong
- Anh chị chỉ hiển thị những chữ số có độ chính xác trên 98%?
  - Tìm ra độ chính xác từng số
  - bỏ vào hàm if