Computer System Engineering 302
Python Project

Powered by Team HANDOCR
Group 8

KUN WANG
kwan574
kwan574@aucklanduni.ac.nz
Computer System Engineering
C/C++, VHDL, MATLAB, Python

SIWEI YANG
syan553
syan553@aucklanduni.ac.nz
Computer System Engineering
C/C++, VHDL, MATLAB, Python

GitHub Repository Link: https://github.com/UOA-CS302-2020/CS302-Python-2020-Group8

Purpose of the system

Background

As early as the 1960s and 1970s, the study of OCR had launched around the world. At the beginning of the research, the recognised characters were printed numbers and which range was only from 0 to 9.
Back to nowadays, with the continuous development of science and technology, the research of OCR also has significant progress in speed and accuracy.
The topic of our project is OCR.

Reason to Develop

- Simple to develop
- Multiple existed examples
- Reachable resources online
- Easy to verify the result
- Low training difficulty

Benefit to society

Handwriting recognition can help to distinguish the letters from handwriting. It can be used such as changing signatures into text format or transfer handwriting material to a text file. Using systems instead of humans to do recognition tasks can save human force and also increase the efficacy of work.

The Goal

The goal of our project is to find and recognise the English alphabets and number digits in the various images, and the characters in the images are not limited to printed with different fonts or handwritten.

Database to use

About the database

EMNIST (https://www.westernsydney.edu.au/bens/home/reproducible_research/emnist)
Download Sources: http://www.itl.nist.gov/iaui/vip/cs_links/EMNIST/gzip.zip
https://cloudstor.aarnet.edu.au/plus/s/ZNmuFiuQTqZlu9W/download

Details of the database

Extended MNIST(EMNIST) is a database that contains a large group of handwriting image samples which include lowercase and uppercase handwritten letters and numbers. It is derived from NIST Special Database 19 and converted to a 28*28 pixel image which matches the structure of MNIST dataset. EMNIST contains 6 different splits: by class, by merge,balanced, letters, digits and mnist and the config of each split is (NIST. (2017)):

- EMNIST ByClass: 814,255 characters. 62 unbalanced classes.
- EMNIST ByMerge: 814,255 characters. 47 unbalanced classes.
- EMNIST Balanced: 131,600 characters. 47 balanced classes.
- EMNIST Letters: 145,600 characters. 26 balanced classes.
- EMNIST Digits: 280,000 characters. 10 balanced classes.
- EMNIST MNIST: 70,000 characters. 10 balanced classes.

Algorithm to use

We selected four models to use: VGG-19, Inception V3, GoogLeNet, ResNetXt. In our project, both input and output format is constant, and each layer of convolution is not sensitive with the previous one, thus the backpropagation is not important, and this is the main reason we choose CNN models.

Main models to use

- VGG-19
- Inception V3
- GoogLeNet
- ResNetXt

Alternative models or methods

We will test more models depending on the selected models' performance and the result of loss and accuracy after training. We are also going to modify the existing models (i.e increasing the dimensions or the convolution layers e.t.c) to test to optimise the performance.

Testing plan

Test environment

Coding/Pre-Test Machine:

- i5-4210M 2.6GHz 2C4T 16G DDR3

Training Machines:

- i7-7700HQ 8GB DDR4 GTX1060 6GB DDR5 VRAM (CUDA with cuDNN)
- Xeon E3-1230 v3 3.3GHz 4C8T 32GB DDR3
- Xeon Gold 6142 2.6GHz 16C32T(4C8T used) 32GB DDR4

Database usage ratio (Training/Developing/Testing)

Traditionally, researchers said 60-20-20, 80-10-10, but recently famous researchers say 90-5-5 or 96-2-2.

We have four main models to use, we will test each of them with the following database usage ratio:
(Training : Developing : Testing)

- 3 : 1 : 1
- 8 : 1 : 1
- 18 : 1 : 1
- 48 : 1 : 1

Expected results

We expected the result to be the letters that were recognised from the images by the system. We will plot the training accuracy, training loss, validation accuracy and validation loss for different times of training and compare them using graphs. We are expected to see the accuracy will increase when we do more training.

Foreseen challenges/risks

One challenge we are facing is the computers we are using do not support high performance. The available working capacity in our hardware is limited therefore we are not able to use big datasets when training the machine. Another problem caused by the equipment is the CPU and GPU of our hardware limits the processing speed, a relatively big dataset will take a long time to process and it will slow our progress for the project.

The training and database usage could be another factor of the challenge as if the training is not enough or the size of datasets is too small, the accuracy will not be ideal, moreover, if the system is overtrained, it will cause the overfitting problem or be stuck in the local optimum. So in order to balance the system, the choice of how deep the learning should apply and the usage of datasets need to be considered carefully.

We are under the risk of lack or miss of communication within the group and with course staff due to we are not able to participate in physical meetings and lectures or labs, and Kun is facing the problem of unstable internet connection which also might cause the communication problem. The images might not be recognised accurately due to the quality of the input photos or documents that are not guaranteed in a standard condition.

Role of each student

Siwei Yang: Designing, Coding, Documenting, Data Collecting, Training, Validating, Optimising

Kun Wang: Designing, Coding, Documenting, Data Collecting, Training, Validating, Optimising

Reference

NIST. (2017). The EMNIST Dataset.Retrieved from:

https://www.nist.gov/itl/products-and-services/emnist-dataset

COMPSYS302 Project Plan

TASK NAME	START DATE	END DATE	DURATION (WORK DAYS)	TEAM MEMBER	PERCENT COMPLETE
Design Document(10%)					
Purpose of System	3/16	4/10	24	Siwei	100%
Database proposal	3/16	4/10	24	Siwei, Kun	100%
Project Algorithm	3/16	4/10	24	Siwei	100%
Testing Plan	3/16	4/10	24	Siwei	100%
Foreseen challenges/risks	3/16	4/10	24	Siwei, Kun	100%
Gantt chart	3/16	4/10	24	Siwei, Kun	100%
Project Code(5%)					
Github Repository	4/3	4/3	0	Siwei	100%
Program Code Design	4/10	4/20	10	Siwei, Kun	0%
Model Training	4/20	4/24	4	Siwei, Kun	0%
Model Validating	4/21	4/26	5	Siwei, Kun	0%
Model Debugging and Optimising	4/10	4/26	16	Siwei, Kun	0%
Final Model Review	4/26	5/1	5	Siwei, Kun	0%
Final Project Demo(15%)					
Individual interview preparation	5/1	5/6	5	Siwei,Kun	0%
Question & Anwer(5%)					
Q&A preparation	5/1	5/6	5	Siwei,Kun	0%
Final Project Report(10%)					
Abstract	4/26	5/6	10	Siwei,Kun	0%
Introduction	4/12	4/12	0	Siwei, Kun	0%
Background and Literature Review	4/13	4/15	2	Siwei, Kun	0%
Methodology	4/14	5/3	19	Siwei, Kun	0%
Evaluation	4/21	5/3	12	Siwei, Kun	0%
Result and Future Work	4/25	5/8	13	Siwei, Kun	0%

WEEK 1	WEEK 2	WEEK 3	WEEK 4	WEEK 5	WEEK 6	WEEK 7	WEEK 8	WEEK 9	WEEK 10	WEEK 11	WEEK 12
M T W Th F	M T W Th F	M T W Th F	M T W Th F	M T W Th F	M T W Th F	M T W Th F	M T W Th F	M T W Th F	M T W Th F	M T W Th F	M T W Th F