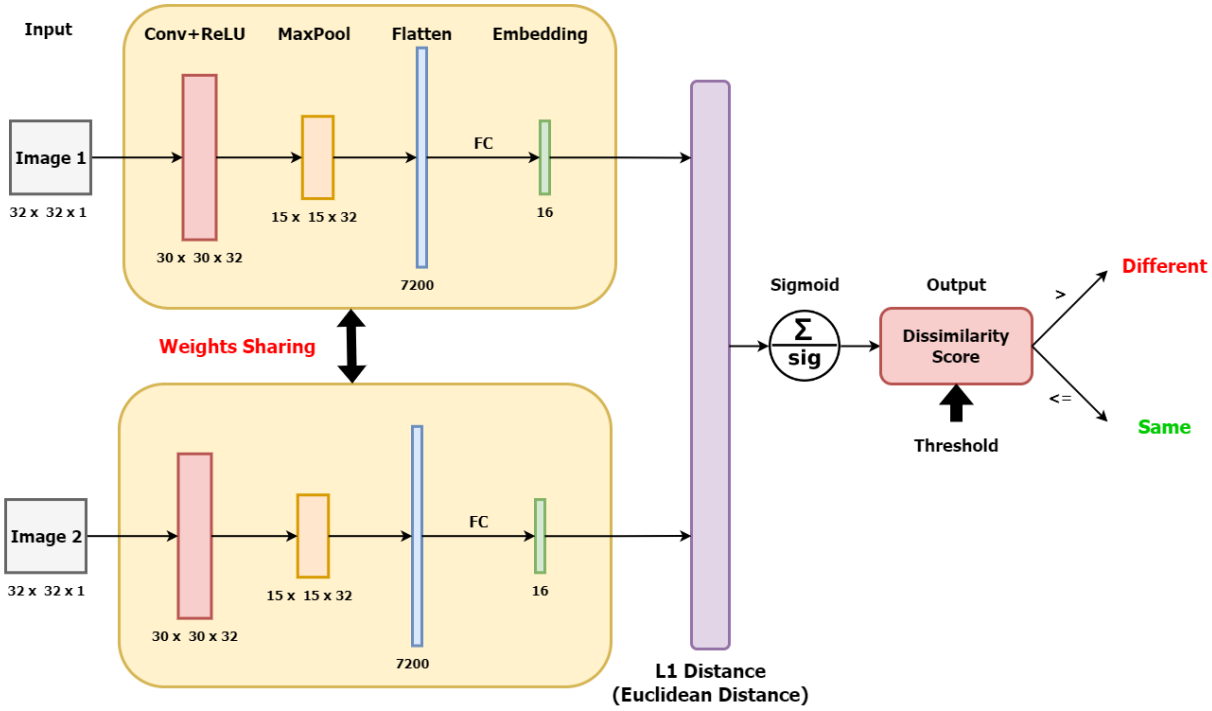
2023 Spring NYCU-EE Machine Learning Intelligent Chip Design – Final Project
SystemC Implementation of a NoC for Siamese Neural Network

IEE 陳冠璋 310510221 / Jun. 11, 2023

1. Siamese Neural Network

Final Project 主要想使用 SystemC 實現 NoC 並在上面運行 DNN 模型，我選擇實作的網路架構是 Siamese Neural Network，並使用 PyTorch 來訓練。Siamese Neural Network(孿生神經網路)是一種特殊的神經網路的架構，用來判斷兩個不同輸入之間的相似度，它的特徵是有兩個輸入與兩個完全相同的子網路所組成，這兩個子網路具有相同的架構與權重。每個子網路接收一個輸入，並將兩個子網路的輸出進行比較來計算它們的相似性，這種計算通常使用 Euclidean Distance (L1 or L2)來完成，如果兩個結果距離接近，表示兩個輸入相似。Siamese Neural Network 可以應用在人臉識別、手寫簽字識別等領域。因此，本次實作的數據集是使用 AT&T 人臉數據集與 EMNIST 手寫英文字母數據集。



2. Model Training

(1) Requirement

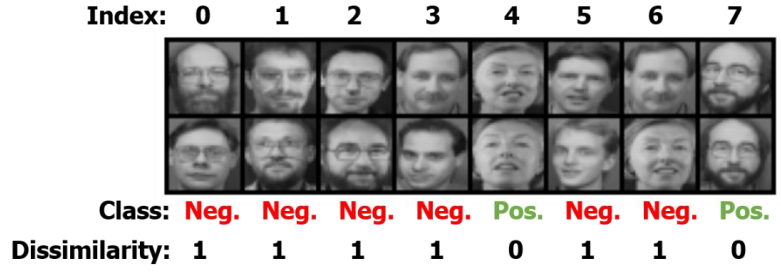
- Python 3.7
- PyTorch 1.13.1

(2) Dataset

- AT&T Database of Faces
- EMNIST

(3) Custom Data Loader

- Positive & Negative samples
- Batch size: 8
- Resize: 32 x 32 x 1



(4) Loss Function

- **Contrastive loss**

$$L = \frac{1}{2N}\left(\sum_{n=1}^{N} y_n {d_n}^2 + (1 - y_n)\max(margin - d_n, 0)^2\right)$$

3. Model Architecture

Layer No.	Type	Input Size	Stride	#Kernels	Output Size	#Weights	#Biases	#Params.
1	Conv2d	32 x 32 x 1	1	3 x 3 x 32	30 x 30 x 32	288	32	340
2	ReLU	30 x 30 x 32			30 x 30 x 32			
3	MaxPool2d	30 x 30 x 32	2	2 x 2	15 x 15 x 32			
4	Flatten	15 x 15 x 32			7200 x 1			
5	Fully Connect	7200 x 1			16 x 1	115,200	16	115,216
6	Sigmoid	16 x 1			16 x 1			
7	L1 Distance	16 x 1			1			
8	Sigmoid	1						
Total Parameters		115,556						

4. Input data & NN weights

Image	Size	#Pixels
1	32 x 32 x 1	1024
2	32 x 32 x 1	1024

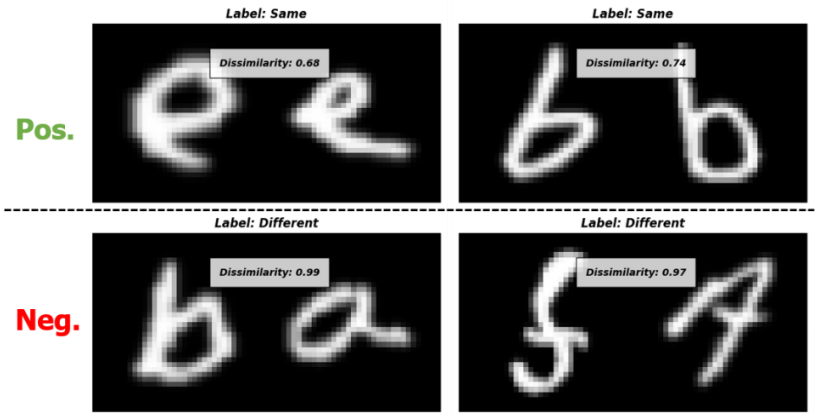
Layer No.	Type	#Weights	#Biases	#Params.
1	Conv2d	288	32	320
2	ReLU			
3	MaxPool2d			
4	Flatten			
5	Fully Connect	115,200	16	115,216
Total Parameters		115,536		

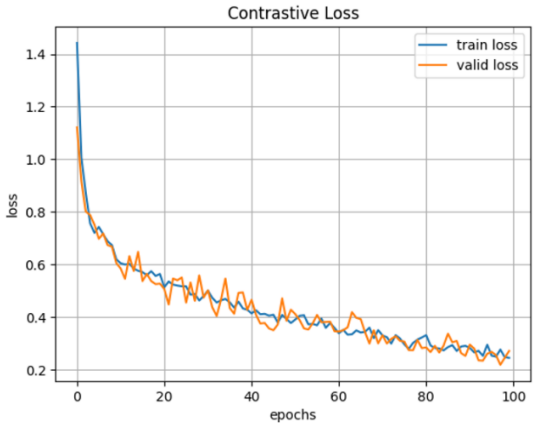
Index	Data
0	Image1 Pixel 1
1	Image1 Pixel 2
⋮	⋮
1023	Image1 Pixel 1024
1024	Image2 Pixel 1
2047	Image2 Pixel 1024
2048	Conv2d Weight 1
117,583	Fully Connect Bias 16

5. Training Result

(1) Character Verification (EMNIST)

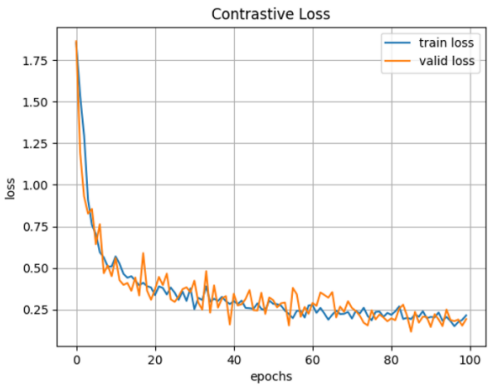
- Test Samples: 50
- Test Accuracy: 96%





(2) Human Face Verification (AT&T)

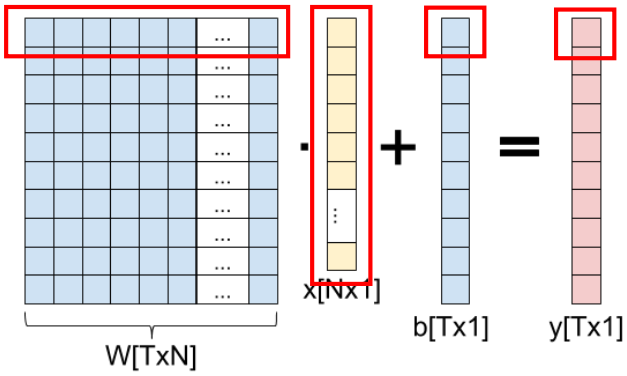
- Test Samples: 50
- Test Accuracy: 94%



6. Architecture

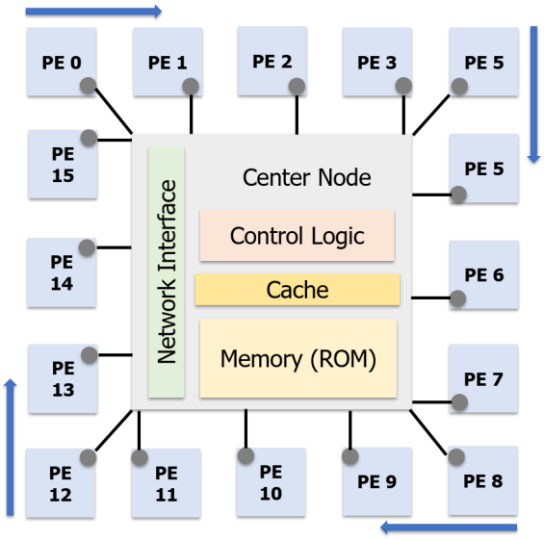
(1) Processing Element

由於 Fully Connected 的運算量龐大，所以 PE 的設計針對矩陣乘法做運算來設計。因為，設計 16 個 PE，並使用星狀(star)的拓樸來實現 NoC。每個 PE 會計算權重的 row vector 與 activation 內積的值，最後再加上 bias，16 個 PE 將 16 個 row vector 順時針丟到對應的 PE 上，即可計算出矩陣乘法的結果。



(2) NoC

- **Topology**: Star
- **Always clockwise**



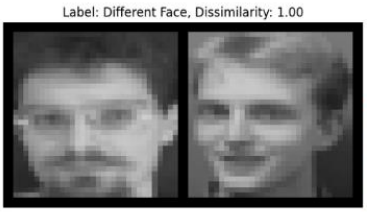
7. Simulation Result

PyTorch

SystemC

```
Starting Simulation...
Result:
Result:
Dissimilarity: 0.535529
Predict: Same
```

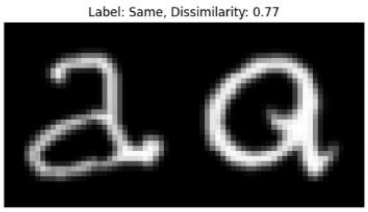
PyTorch



SystemC

```
Starting Simulation...
Result:
Result:
Dissimilarity: 0.997747
Predict: Different
```

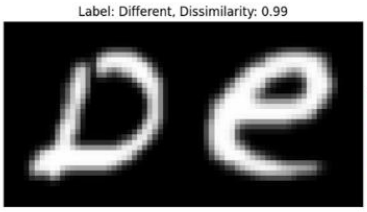
PyTorch



SystemC

```
Starting Simulation...
Result:
Result:
Dissimilarity: 0.768814
Predict: Same
```

PyTorch



SystemC

```
Starting Simulation...
Result:
Result:
Dissimilarity: 0.994181
Predict: Different
```