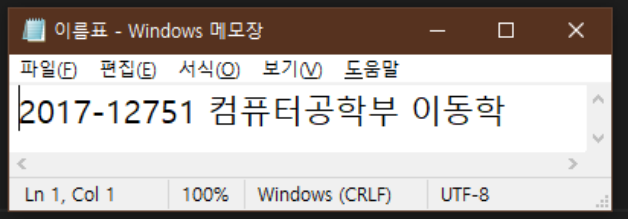
하드웨어 시스템 설계 7주차 실습 보고서

2017-12751 컴퓨터공학부 이동학

Goal: Implement convolution lowering in C++

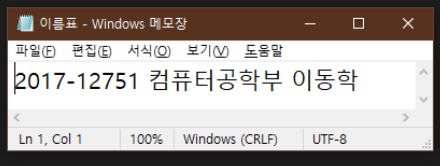
Code: src/fpga_api.cpp & src/fpga_api_on_cpu.cpp

```
for(int i = 0 ; i < conv_channel ; i++)
  for(int j = 0 ; j < input_channel ; j++)
    for(int k = 0 ; k < conv_height ; k++)
      for(int l = 0 ; l < conv_width ; l++)
        new_weights[i][(j*conv_height + k)*conv_width + l] = cnn_weights[i][j][k][l];

int row_filter = input_height - conv_height + 1;
int col_filter = input_width - conv_width + 1;

for(int i = 0 ; i < input_channel ; i++)
  for(int j = 0 ; j < conv_height ; j++)
    for(int k = 0 ; k < conv_width ; k++)
      for(int m = 0 ; m < row_filter ; m++)
        for(int n = 0 ; n < col_filter ; n++)
          new_inputs[(i*conv_height + j)*conv_width + k][m*col_filter + n] = inputs[i][j+m][k+n];
```



Result:

```
root@a579e75b7786:~/hsd20_lab07# bash benchmark.sh
[*] Arguments: Namespace(m_size=64, network='mlp', num_test_images=100, run_type='cpu', v_size=64)
[*] Read MNIST...
[*] The shape of image: (100, 28, 28)
[*] Load the network...
[*] Run tests...
[*] Statistics...
{'accuracy': 0.97,
 'avg_num_call': 627,
 'm_size': 64,
 'total_image': 100,
 'total_time': 0.3991730213165283,
 'v_size': 64}

=> Accuracy should be 0.97

[*] Arguments: Namespace(m_size=64, network='mlp', num_test_images=100, run_type='fpga', v_size=64)
[*] Read MNIST...
[*] The shape of image: (100, 28, 28)
[*] Load the network...
[*] Run tests...
benchmark.sh: line 8:  1694 Segmentation fault      (core dumped) python eval.py --num_test_images 100 --m_size 64 --v_size 64 --network mlp --run_type fpga

=> Accuracy should be 0.97

[*] Arguments: Namespace(m_size=64, network='cnn', num_test_images=100, run_type='cpu', v_size=64)
[*] Read MNIST...
[*] The shape of image: (100, 28, 28)
[*] Load the network...
[*] Run tests...
[*] Statistics...
{'accuracy': 1.0,
 'avg_num_call': 741,
 'm_size': 64,
 'total_image': 100,
 'total_time': 0.3662428855895996,
 'v_size': 64}

=> Accuracy should be 1.0

[*] Arguments: Namespace(m_size=64, network='cnn', num_test_images=100, run_type='fpga', v_size=64)
[*] Read MNIST...
[*] The shape of image: (100, 28, 28)
[*] Load the network...
[*] Run tests...
benchmark.sh: line 16:  1751 Segmentation fault      (core dumped) python eval.py --num_test_images 100 --m_size 64 --v_size 64 --network cnn --run_type fpga

=> Accuracy should be 1.0
```



Explain: convolution lowering을 통해 cnn_weights를 new_weights로, inputs를 new_inputs로 변환시키는 코드입니다. 배웠던 그대로 구현하였으나 약간의 차이점이 있다면 new_inputs에 inputs의 값들을 뒤집어서 넣지 않고 똑바로 넣었다는 점입니다.

Discussion: fpga board가 아니라 pc로 구동했기 때문에 –run_type fpga인 경우에는 segmentation fault가 떴습니다. 그 외에는 benchmark.sh의 요구사항대로 잘 구현되었습니다.