|  |  |
| --- | --- |
| **Name:** | *Sheng-Wei Huang* |
| **NetID:** | *Swhuang3* |
| **Section:** | *AL2* |

**ECE 408/CS483 Milestone 2 Report**

|  |
| --- |
| 1. Show output of rai running Mini-DNN on the basic GPU convolution implementation for batch size of 1k images. This can either be a screen capture or a text copy of the running output. Please do not show the build output. (The running output should be everything including and after the line "*Loading fashion-mnist data...Done*"). |
|  |
| 1. For the basic GPU implementation, list Op Times, whole program execution time, and accuracy for batch size of 100, 1k, and 10k images. |
| |  |  |  |  |  | | --- | --- | --- | --- | --- | | Batch Size | Op Time 1 | Op Time 2 | Total Execution Time | Accuracy | | 100 | *0.176298ms* | *0.635443ms* | *1.156 s* | *0.86* | | 1000 | *1.62754ms* | *6.26372ms* | *9.761 s* | *0.886* | | 10000 | *16.0021ms* | *62.2321ms* | *95.227 s* | *0.8714* | |
| 1. List all the kernels that collectively consumed more than 90% of the kernel time and what percentage of the kernel time each kernel did consume (start with the kernel that consumed the most time, then list the next kernel, until you reach 90% or more). |
| *conv\_forward\_kernel: 100.0% time* |
| 1. List all the CUDA API calls that collectively consumed more than 90% of the API time and what percentage of the API time each call did consume (start with the API call that consumed the most time, then list the next call, until you reach 90% or more). |
| *cudaMemcpy: 84.0%*  *cudaMalloc: 10.8%* |
| 1. Explain the difference between kernels and CUDA API calls. Please give an example in your explanation for both. |
| *Cuda kernels are user defined function with \_\_global\_\_ prefix, which can directly access GPU memory and use GPU to calculate, like \_\_global\_\_ conv\_forward\_kernel() we defined in this milestone.*  *Cuda API calls are the api cuda provided, which allow host to allocate or copy data to GPU device, like cudaMemcpy, cudaMalloc.* |
| 1. Show a screenshot of the GPU SOL utilization |
|  |