|  |  |
| --- | --- |
| **Name:** | *Haozhe Chen* |
| **NetID:** | *haozhe3* |
| **Section:** | *ZJ1* |

**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  (ms) | Op Time 2  (ms) | Total Execution Time (s) | Accuracy  (%) | | 100 | *0.479951* | *2.93208* | *1.169* | *0.86* | | 1000 | *4.65613* | *29.7385* | *9.719* | *0.886* | | 10000 | *46.1701* | *300.715* | *1m38.061s* | *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). |
| *Ordered result:*    Raw result from: *analysis\_file.ncu-rep*    *Raw result from: “nsys profile --stats=true ./m2”* |
| 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). |
| *Raw result from: “nsys profile --stats=true ./m2”* |
| 1. Explain the difference between kernels and CUDA API calls. Please give an example in your explanation for both. |
| *Kernels are the code that the GPU actually runs to realize certain functionalities. They will various significantly when the goals of projects change. For example, the function “conv\_forward\_kerne()l” used in m2.*  *CUDA API contains some functions that help to connect CPU (or your computer) and GPU. They do some basic things which make it possible for you to control the behavior of GPU, like allocate, free memory, copy data from one to another and Synchronize between devices. For example, the function “cudaDeviceSynchronize()” used in m2.* |
| 1. Show a screenshot of the GPU SOL utilization |
|  |

*Output of “nsys profile --stats=true ./m2”*

文本

描述已自动生成

文本

描述已自动生成

文本

描述已自动生成