EE/CprE/SE 492

HAML: Heterogeneous and Accelerated Computing for Machine Learning

Semester 2 Week 5-6 Report

9/20/24 - 10/3/24

Faculty Advisor *Phillip Jones*
Client *JR Spidell*

Team Members:

Jonathan Tan	- Memory Affinity, Kria Board Manager
Josh Czarniak	- DPU Control Developer
Justin Wenzel	- Multi-threaded Developer
Kai Heng Gan	- Image Processing/Semantic Segmentation Developer
Santiago Campoverde	- Model Analytics

Summary for Progress These Two Weeks

Over the past few weeks, we have been working towards our 'Milestone 3' goal of running all three models on the Kria KV260 board within our multi-threaded program. This has involved migrating the program from our development environment to the board, integrating model interaction with the multithreaded system, creating a DPU handler to manage thread resource control, and setting up PetaLinux and profiling tools in preparation for testing in the next milestone.

These Two Weeks' Individual Contributions

- Justin
  - Moved multi-threaded application to the Kria board and ran a mock model that follows the model class format defined in the C++ implementation that all models can follow. Creating a standardized interaction for threads with models.
    - The model class consists of four functions to allow threads to interact with each different model in a standardized format. The functions are load model, preprocess, run inference, and output process
    - Began writing the blink model to follow the defined model class in the multi-threaded application to allow threads to easily interact with the model.
  - Use a previous group's Python implementation of the blink model to improve the accuracy of our C++ implementation by adjusting how we preprocess and handle the data, without modifying the model itself.
- Jonathan
  - Rebuilt Petalinux OS with updated components.
  - Debug error when using vaitrace, currently, profiling is not working due to a divide by zero error, which I believe is caused by the profiling program (vaitrace) failing to detect events.
    - Things tried (since last report):

- Went down a rabbit hole fixing the path of `zyxclmm_drm/ip_layout`. I thought that after the DPU IP is initialized, its debugfs is mounted in the wrong folder path, causing vaitrace to be unable to find the file (as evident from the warning "`WARNING:root:Cannot open 'zyxclmm_drm/ip_layout'`"). However, that turns out to not be the case.
- Now, I found that the ip_layout file path is correct. However, its size is 0, which is wrong. Also, when running dmesg, I see this error: `zocl-drm axi:zyxclmm_drm: IRQ index 0 not found`.

- Josh
  - Worked off of new system diagram with DPU implemented
    - Helped visualize what DPU will be working with
  - Adjusted DPU API examples to match our implementation of DPU core
    - The model in article showed what the DPU implementation will look like
  - Began work on DPU code
    - Debated on creating DPU runner every switch or only implementing once.

- Kai
  - Tried to quantized the Pytorch model for conversion to Xmodel that fit to the DPU.
    - I encountered an "Out of Memory" error while my Debian Linux VM has 64 GB RAM.
    - I tried to allocate 32GB memory to Vitis-AI docker container and it didn't resolve the issue.
    - I have set up a 1:1 meeting with our client to ensure that I was quantizing the model correctly.
- Santiago
  - Moved out from getting the Vitis AI Docker to run the models for testing the accuracy.
    - Found out that the models cannot run other than on the board.
  - Determined that it is feasible to use the library tools instead of creating new ones to test the accuracy of the model.
    - Performance will not be affected as these tools are run as a separate program on the board from the main project and not at the same time.

Team Member	These Two Weeks' Task	Completion Date	Hours Took	These Two Weeks' Hours	Total Project Hours
Justin Wenzel	Attended meetings	NA	3	9	117.5
	Transfer multi-threaded program from docker container environment to the Kria board	9/28	4		

	Ran and reviewed a previous teams Python script to run blink model, will use this script to adjust the C++ implementation to improve accuracy	9/16	2		
Jonathan Tan	Attended meetings	NA	3	10	137.5
	Debug error when running profiler on the board (vaitrace)	On-going	7		
Josh Czarniak	Attended meetings	NA	1	5	100
	Adjusted DPU API examples to match our implementation of DPU core	9/28	3		
	Began work on DPU code	10/1	1		
Kai Heng Gan	Attended meetings	NA	3	13	135.5
	Tried to quantized the Pytorch model for conversion to Xmodel that fit to the DPU.	On-going	10		
Santiago Campoverde	Attended meetings	NA	2	4	88
	Moved out from getting the Vitis AI Docker to run the models for testing the accuracy	9/21	1		
	Determined that it is feasible to use the library tools instead of creating new ones to test the accuracy of the model.	9/19	1		

Note: 1. This is per week hours, Σ "hours taken" = "week hours". 2. Due to multiple meeting times, meetings' "completion date" are "NA".

Plans for Coming Two Weeks

Team Member	Plans for Coming Week	Planned Completion	Planned Hours Required
Justin Wenzel	Implement eye track functions in the model class format defined in the multi-threaded application to	10/11-10/14	3

	interact allow threads to interact with the eye track model, and perform inference with DPU		
	Work with Josh to get a DPU handler for the multi-threaded application for the models to interact with	10/6	2
	Work with Kai to implement semantic segmentation in the model class format defined in the multi-threaded application to interact with the semantic segmentation model, and perform inference with the DPU	10/7-10/11	2
	Research into thread affinity to understand how the pthread.h library functions interact with our program, from the high-level usage to the low-level system calls and verifying thread affinity is working.	10/8	2
Jonathan Tan	Continue debug error with vaitrace	9/25	5
	Start looking into implementing memory affinity on the Kria board.	10/10	12
Josh Czarniak	Make a slide deck on DPU implementation	10/6	2
	Implement DPU code	Ongoing	5
	Meet with Justin to get a DPU handler for the multi-threaded application	10/5	2
Kai Heng Gan	Continue working and testing on preprocessing cpp code that will run on the Kria KV260. Resolve the invalid output from the xmodel.	ongoing	10
	Re-quantize and re-compile the semantic segmentation xmodel.	ongoing	10
	Work with Justin to implement semantic segmentation in the model class format defined in the multi-threaded application to interact with the semantic segmentation model, and perform inference with the DPU	NA	2
Santiago Campoverde	Test model on the board	10/5	1
	Determine the usability of output from the accuracy tools	10/6	2