TEXAS
The University of Texas at Austin

MobileNet DPU Accelerated Inference System

EE382N-4 Advanced Embedded Systems Spring 2024

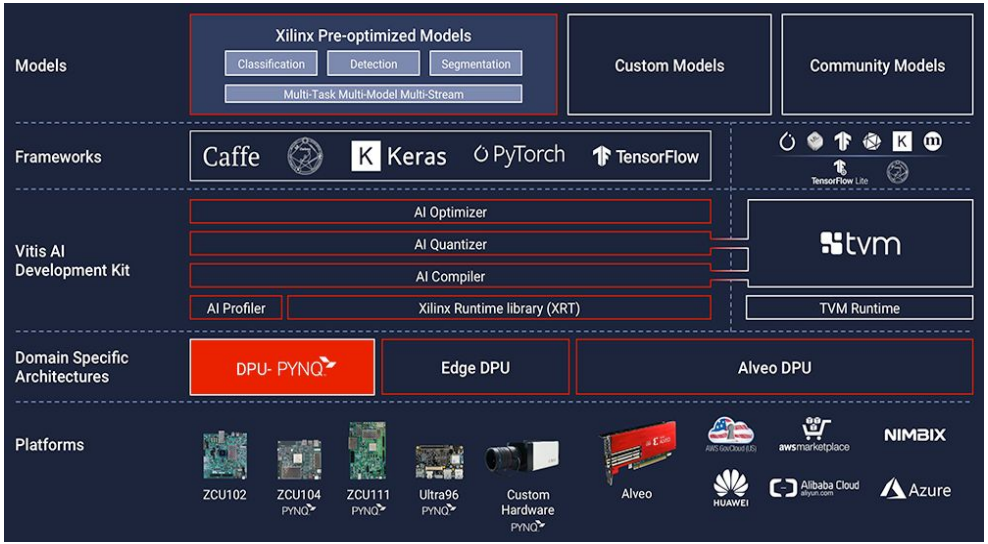
Christian Lancaster, Reva Vaidya, William Avery

Problem Statement

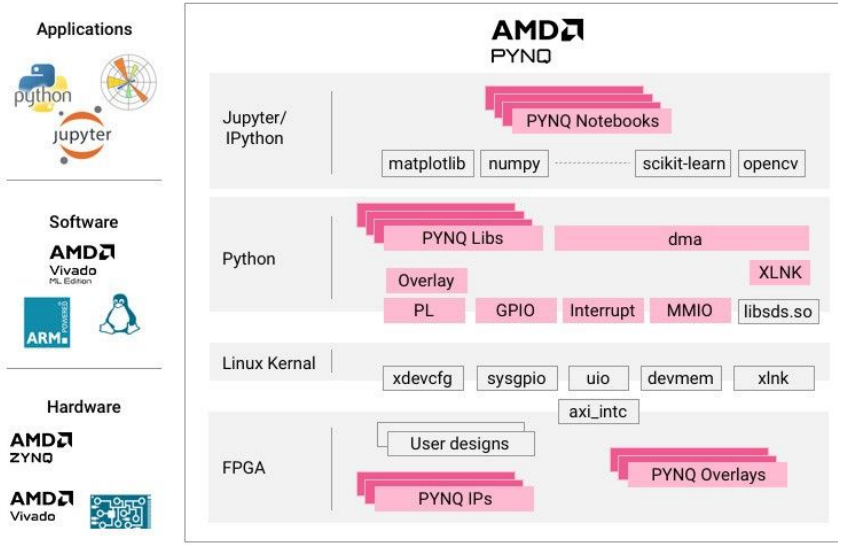
- Accelerating deep neural networks for pose estimation using the Zynq Deep Learning Processing Unit (DPU)
  - Optimizing for the split between CPU and DPU utilization

Technologies Used: Vitis AI & AMD Pynq

Vitis AI

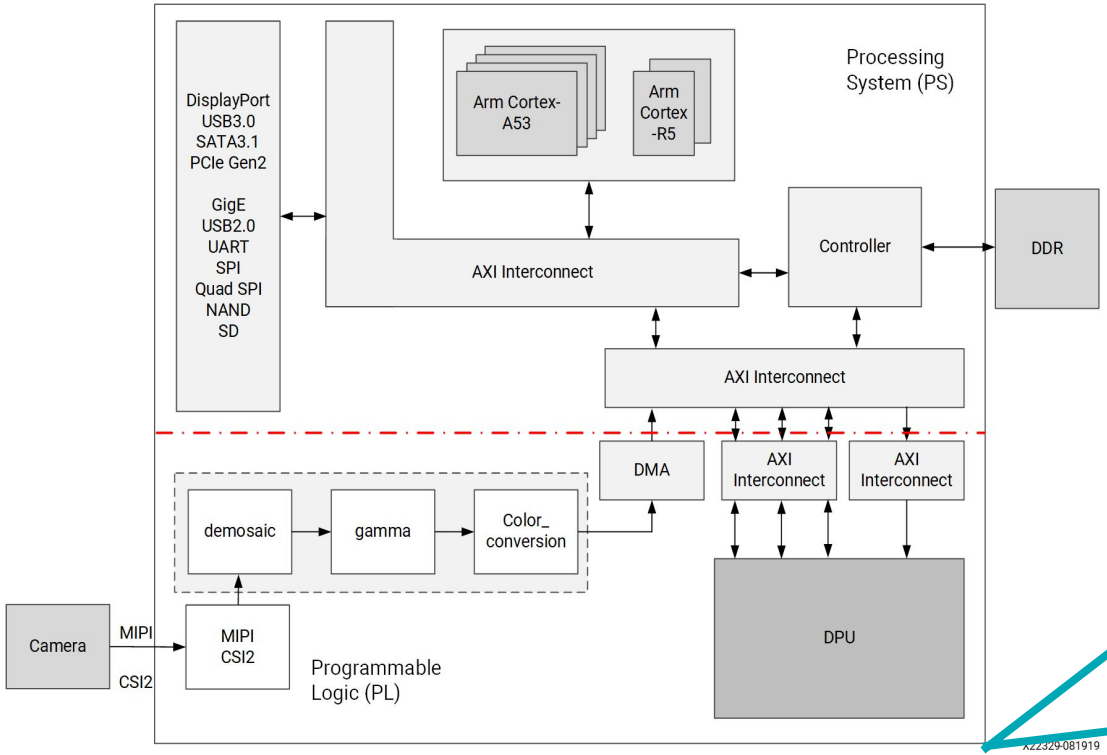


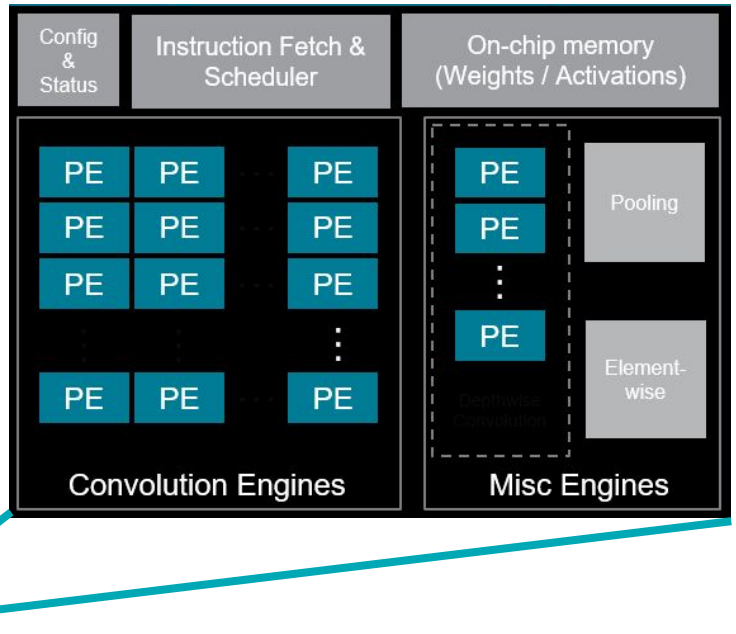
PYNQ



https://www.xilinx.com/applications/medical/healthcare-ai.html

Deep Learning Processing Unit



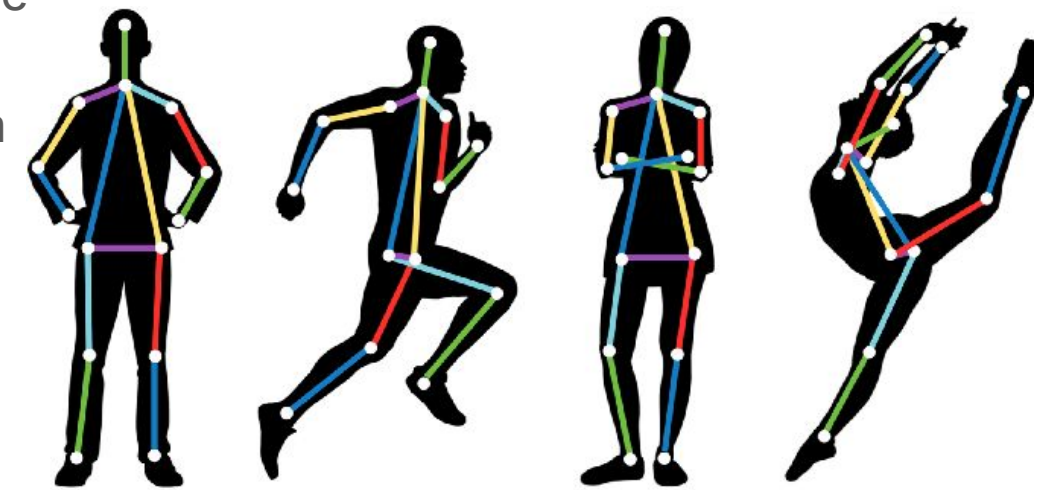


https://docs.amd.com/r/en-US/pg338-dpu/MPSoC-Integration-Example

https://xilinx.github.io/Vitis-AI/3.0/html/docs/workflow-system-integration.html

Mobile Pose Inference

- MobileNetV2 backbone with dense upsampling convolutions
- Downstream pose estimation with dense upsampling convolutions and differentiable spatial to numerical transformations
- Trained using the MobilePose library and the MPII human pose dataset on a single A100 GPU



https://www.analyticsvidhya.com/blog/2021/10/human-pose-estimation-using-machine-learning-in-python/

Key Results

Hardware	Static INT8 Quantization	Euclidean Loss (Pixels)	Jenson-Shannon Divergence	Framerate (FPS)
DPU (Vitis-AI) + CPU (ONNX)	Y+Y	0.093	0.26	3.1383
DPU (Vitis-AI) + CPU (ONNX)	Y+N	0.093	0.26	7.4450
CPU (ONNX)	Y	0.092	0.25	9.7242
CPU (ONNX)	N	0.090	0.25	11.6259

- Euclidean Loss: distance between predicted pose and true pose
- Jenson-Shannond Divergence: similarity between heatmaps

What went well

- Remote training and inference system
  - We were able to work effectively over the internet using own resources
- Support of Xilinx PYNQ framework
  - Documentation
  - Jupyter Notebook programming interface
  - Built in DPU-PYNQ support for Ultra96V2
- Camera interfacing fully supported by PYNQ

What did not go well

Xilinx documentation was limited and varies between Vitis AI major versions

- Many operations not supported by the DPU
- Sparse documentation on extending compiler and DPU to support custom IP
- Validating the quantized model
  - This just took a long time due to discrepancies between Vitis AI documentation and what actually happens

What would we do differently

- Pivoted several times due to limitations of the Vitis AI ecosystem
  - Board support, custom op support, model support
  - We only found PYNQ at a late stage in the project
- Accelerate a proprietary model with cutting edge operations optimized for mobile architecture
  - Knowing what we have learned now, we could likely achieve this
- Possibly explore the design of a custom system using Vivado

References

[1] Vitis AI: https://www.xilinx.com/products/design-tools/vitis/vitis-ai.html

[2] AMD PYNQ: https://www.pynq.io/

[3] DPU-PYNQ: https://github.com/Xilinx/DPU-PYNQ

[4] MPII Human Pose Database: http://human-pose.mpi-inf.mpg.de/

[5] MobilePose: https://github.com/YuliangXiu/MobilePose

[6] MobileNetV2: https://arxiv.org/abs/1801.04381