PROJECT ELM

FINAL DESIGN

SDMay25-01

James Minardi, Eli Ripperda, Lindsey Wessel, Mason Inman

SDMay25-01

James Minardi [CprE]

Eli Ripperda [CprE]

Lindsey Wessel [SE]

Mason Inman [SE]

JR Spidell [Client]

Dr. Zambreno

Dr. Jones

OUR CLIENT

JR SPIDELL

- Sr. Principal Systems Engineer
- This is not affiliated with his work – this is an independent project.
- Formerly volunteered to help with individuals with cerebral palsy and is motivated to help them further.
- **Wants** to develop **assistive wheelchair tech** with features including mobility assistance **and real-time seizure detection.**

PROBLEM

- People with **mobility** and **cognitive impairments** face many challenges including maintaining **independence** and **safety**
- Lack of advanced wheelchair technologies leads to **gaps in autonomy** , communication, etc

OBJECTIVE

Develop a fast and accurate algorithm inference machine learning computer vision subsystem on an FPGA board to support our client's vision of advanced assistive technologies.

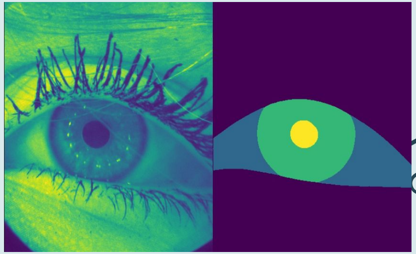
PROJECT OVERVIEW

SYSTEMS

- ❖ Camera
- ❖ Vision-Based ML algorithm
- ❖ Non-Vision-Based ML model
- ❖ Ultra96 v2 FPGA development board
- ❖ Display

REQUIREMENTS

- ❖ Real-time
- ❖ Accurate and performant to [NDA] fps
- ❖ Display model outputs and debugging information



**SDMay25-01 is under an NDA that prevents openly sharing specific metrics (eg. requirements).*

CLIENT PERSONA

INTERESTS & GOALS

- Helping people in wheelchairs
- Providing opportunities to SE & CprE Students

NEEDS

- Decrease latency in detecting health problems
- Iterative development of overall project

CHALLENGES

- Lacks abundant free time

CLIENT NEEDS

PRODUCT

- Embedded vision & non-vision ML algorithms
- Affordable System
- A fast and accurate system

PROGRESS

- Each team needs to improve upon their predecessors

COLLABORATION

- One of many teams
- Learn from previous teams
- Teach to future teams

REQUIREMENTS

REQUIREMENTS

TECHNICAL

- [NDA] FPS
- [NDA]% accuracy
- FPGA: Ultra96v2
- Tensil.ai to generate FPGA based ML accelerator

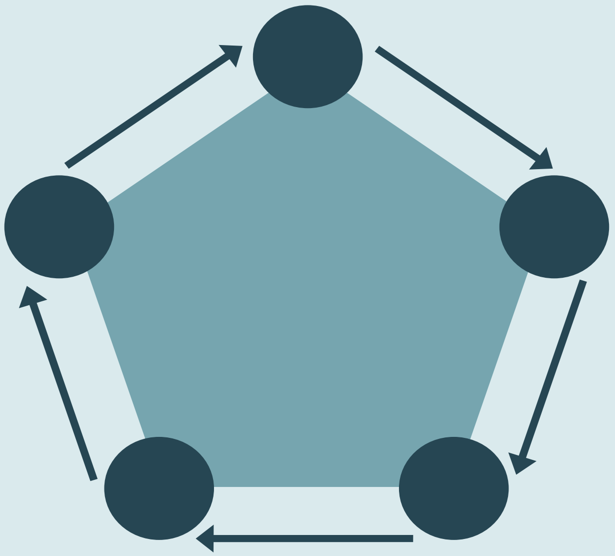
TRANSITION

- Aid in the handoff process to help future teams
- Well-documented work

CLIENT

- Low latency
- High accuracy
- Improve care of wheelchair bound individuals

**SDMay25-01 is under an NDA that prevents openly sharing specific metrics (eg. requirements).*



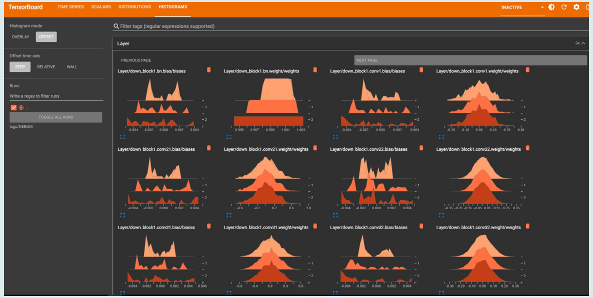
PROJECT MANAGEMENT

PROJECT MANAGEMENT STYLE

Agile

- ❖ Weekly client, team and advisor meetings.
- ❖ Requirements are flexible and changing per client request.
  - ➢ Concurrent work with other Sr. Design teams.

MILESTONES



Milestone 1

Obtain Baseline Metrics

Obtain baseline metrics for latency, accuracy, and FPS from client-provided model.

Milestone 2

Optimize

Train the ML model and gather more metrics.

Milestone 3

Port to FPGA

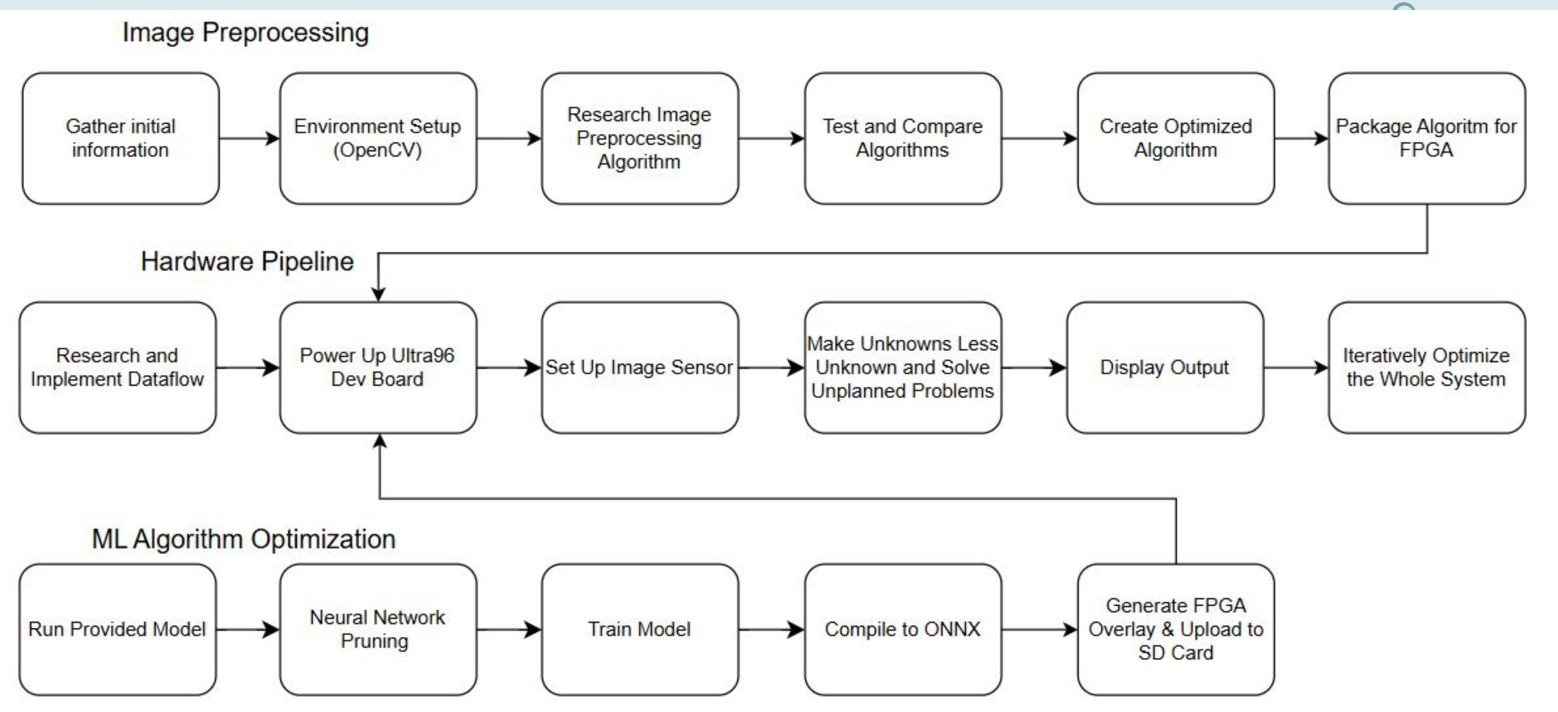
Use Tensil.ai framework to compile the model onto the FPGA.

Milestone 4

Run Real-Time System

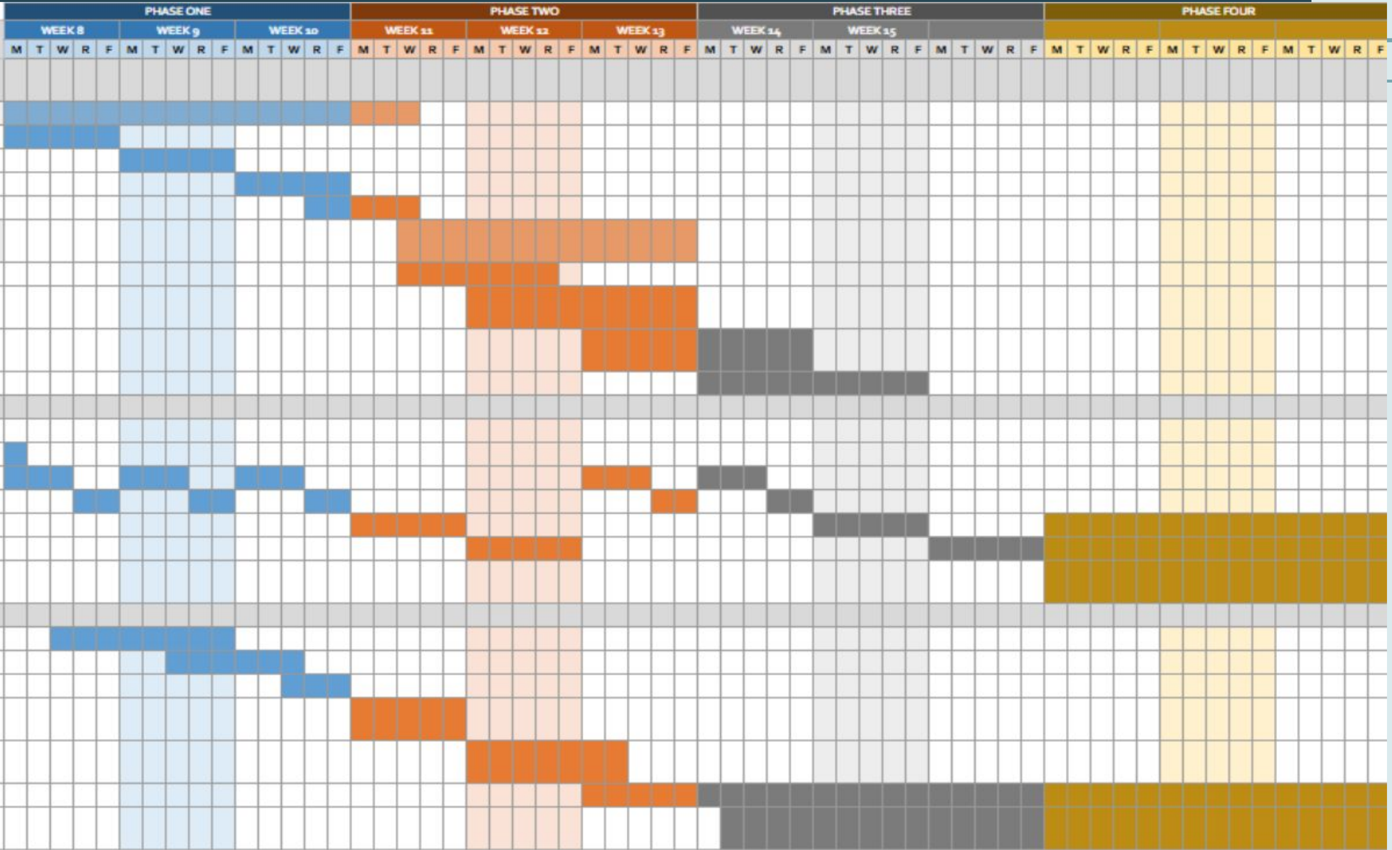
Real time video preprocessing and ML algorithm execution

TASK DECOMPOSITION



GANTT CHART

WBS NUMBER	TASK TITLE	TASK OWNER
2	**Run Compiled ML Model on FPGA**	
2.1	Create Compiled ML Model	James
2.1.1	Research Tensil	Eli
2.1.2	Make first Compilation	James
2.1.3	Compile Mason's Model	James
2.2	Download Model onto FPGA	James
2.3	Connect & Setup I/O for ML Model	Eli
2.3.1	Connect & Setup Webcam	Eli
2.3.2	Understand & Define ML Model Output	James
2.3.3	Determine How to Display Output	Eli
2.3.4	Display Output	James
3	**Image Preprocessing**	
3.1	Gather Initial Understanding	Lindsey
3.2	Environment Setup	Lindsey
3.3	Research	Lindsey
3.3.1	Compare algorithms	Lindsey
3.3.2	Test model	Lindsey
3.3.3	Compare models	Lindsey
3.4	Create optimized solution with research	Lindsey
4	**ML Algorithm Optimization**	Mason
4.1	Research	Mason
4.2	Environment Setup	Mason
4.3	Obtain Training Data	Mason
4.4	Obtain baseline model to put on board	Mason
4.5	Identify/Research points of optimization	Mason
4.6	Implement Optimizations	Mason
4.7	Train model with Optimizations	Mason



METRICS & EVALUATION CRITERIA

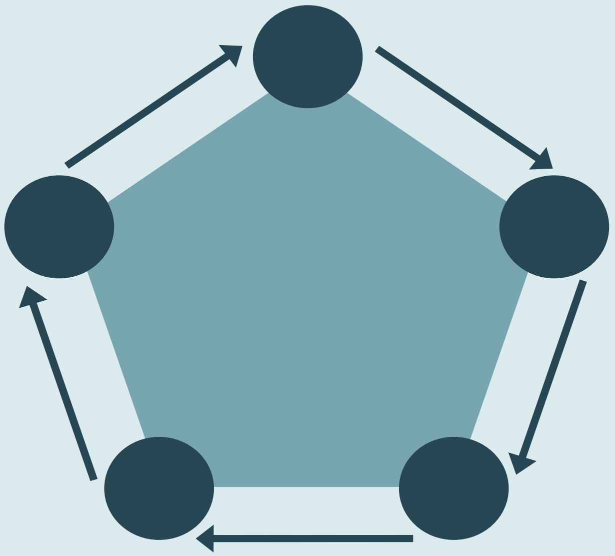
METRICS

- Test Latency Script (ms)
- Compare to Training Data
- Compare embedded ML prediction to ground truth
- Track FPS

EVALUATION CRITERIA

- Latency (ms)
- Accuracy (IoU values)
- Speed (FPS)

**SDMay25-01 is under an NDA that prevents openly sharing specific metrics (eg. requirements).*



OUR DESIGN

MAJOR COMPONENTS

01 ML Algorithm Optimization

Team member: Mason

02 ML Image Processing

Team member: Lindsey

03 ML Acceleration: Tensil

Team member: Eli

04 Ultra96v2 Board

Team member: James

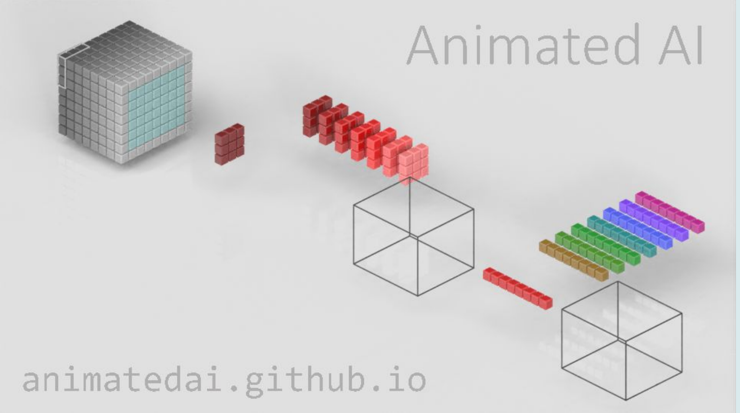
01

ML Algorithm Optimization

OPTIMIZATION STRATEGIES

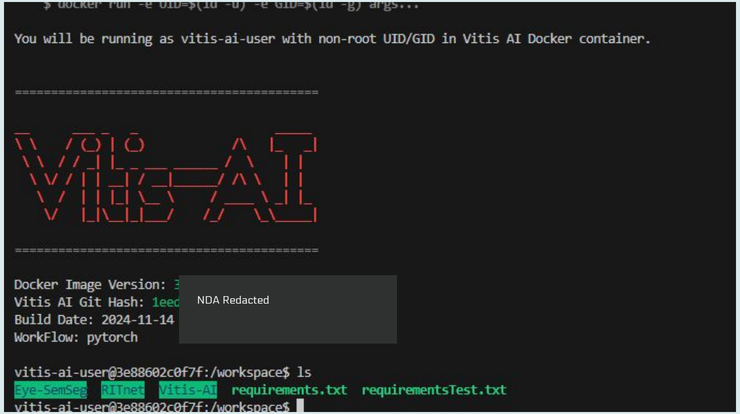
DEPTHWISE SEPARABLE CONVOLUTION

❖ Significant **parameter reduction** through smaller convolutions.

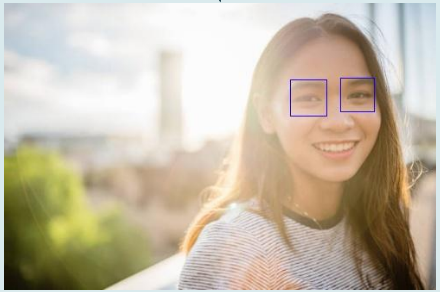


NEURAL NETWORK PRUNING

❖ Implement Vitis-AI tooling to use **quantization aware training.**



Test & Choose an Algorithm



* Algorithm 1 Result

* Algorithm 2 Result

02

Image PreProcessing ML Algorithm

CHOOSING AN ALGORITHM

Input	Algorithm 1	Algorithm 2

ACCURACY & TIME [s]

Input

*input images to test accuracy

Algorithm 1

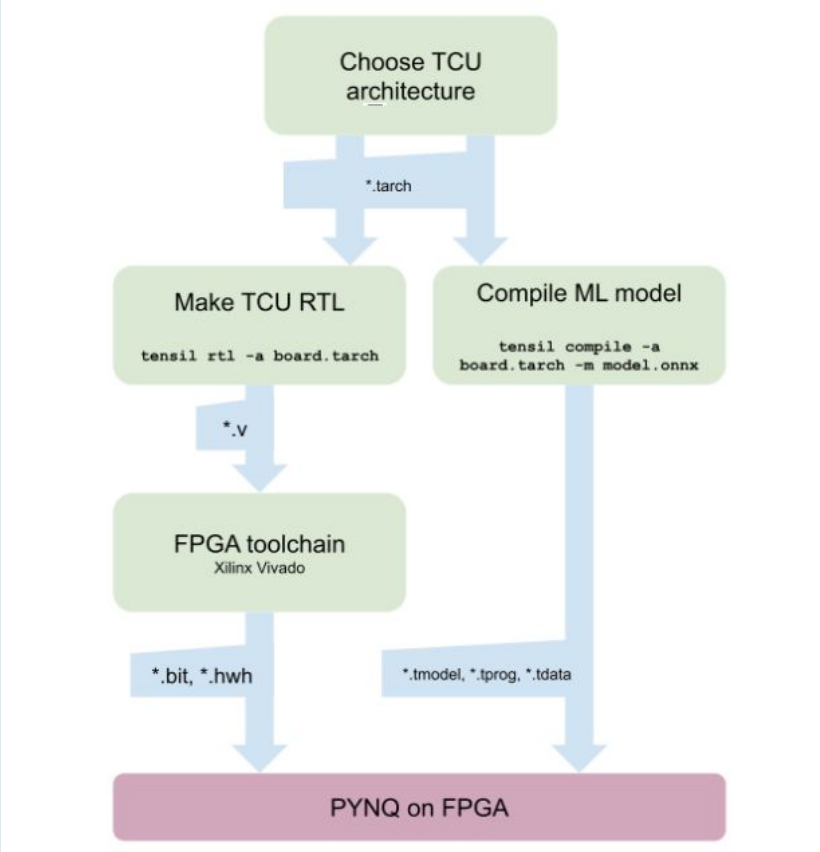
```
0.10858750343
0.11860680580
0.11986422538
0.10754251480
0.11414432525
0.10358309745
0.12024736404
0.11383533477
0.16153049468
0.11439013481
0.10884189605
0.10570573806
0.11348056793
0.16413927078
0.10365796089
0.10923194885
0.11400699615
0.10471320152
0.12034702301
```

*Algorithm 1 is fast but lacks some accuracy

Algorithm 2


```
0.33411288
0.45055270
0.47802901
0.34627008
0.19120693
0.20115184
0.21317386
0.32428646
0.18401265
0.24870061
1.61701750
0.40877246
0.31383681
0.54106521
0.30658364
0.48964858
```

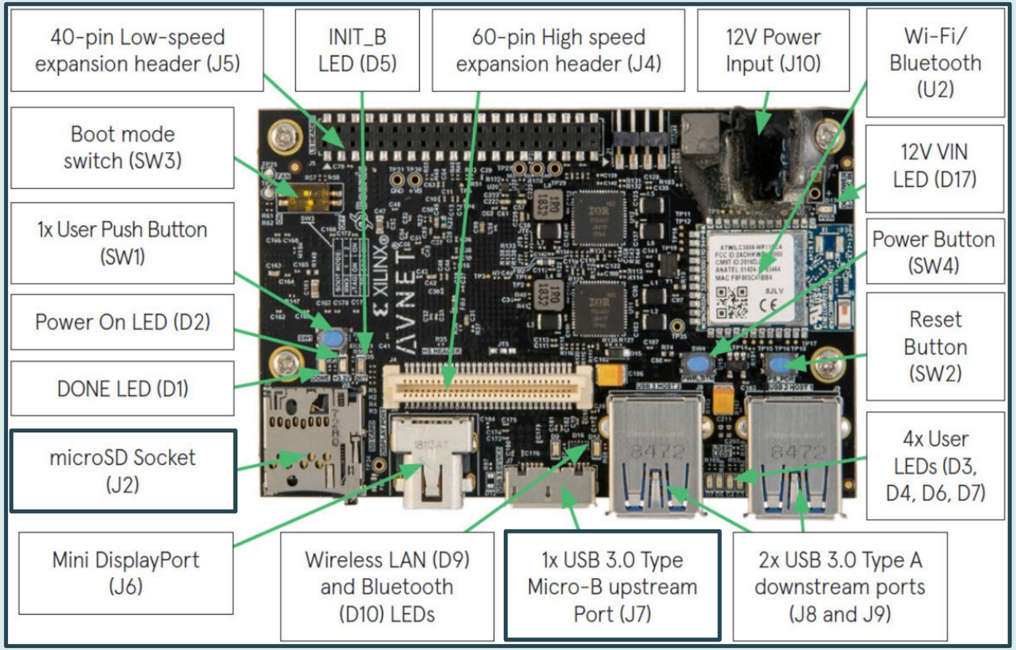
*Algorithm 2 is more accurate but slow



Choose TCU architecture
*.tarch
Make TCU RTL
tensil rtl -a board.tarch
Compile ML model
tensil compile -a board.tarch -m model.onnx
*.v
FPGA toolchain
Xilinx Vivado
*.bit, *.hwh
*.tmodel, *.tprog, *.tdata
PYNQ on FPGA

03

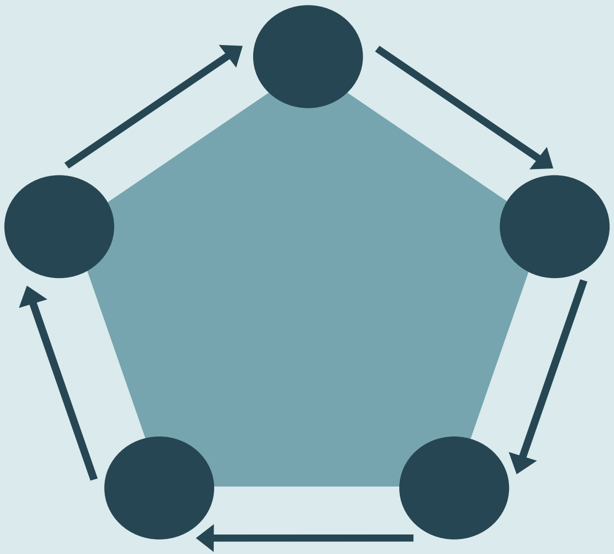
Tensil.AI



Topology

04

Ultra96v2

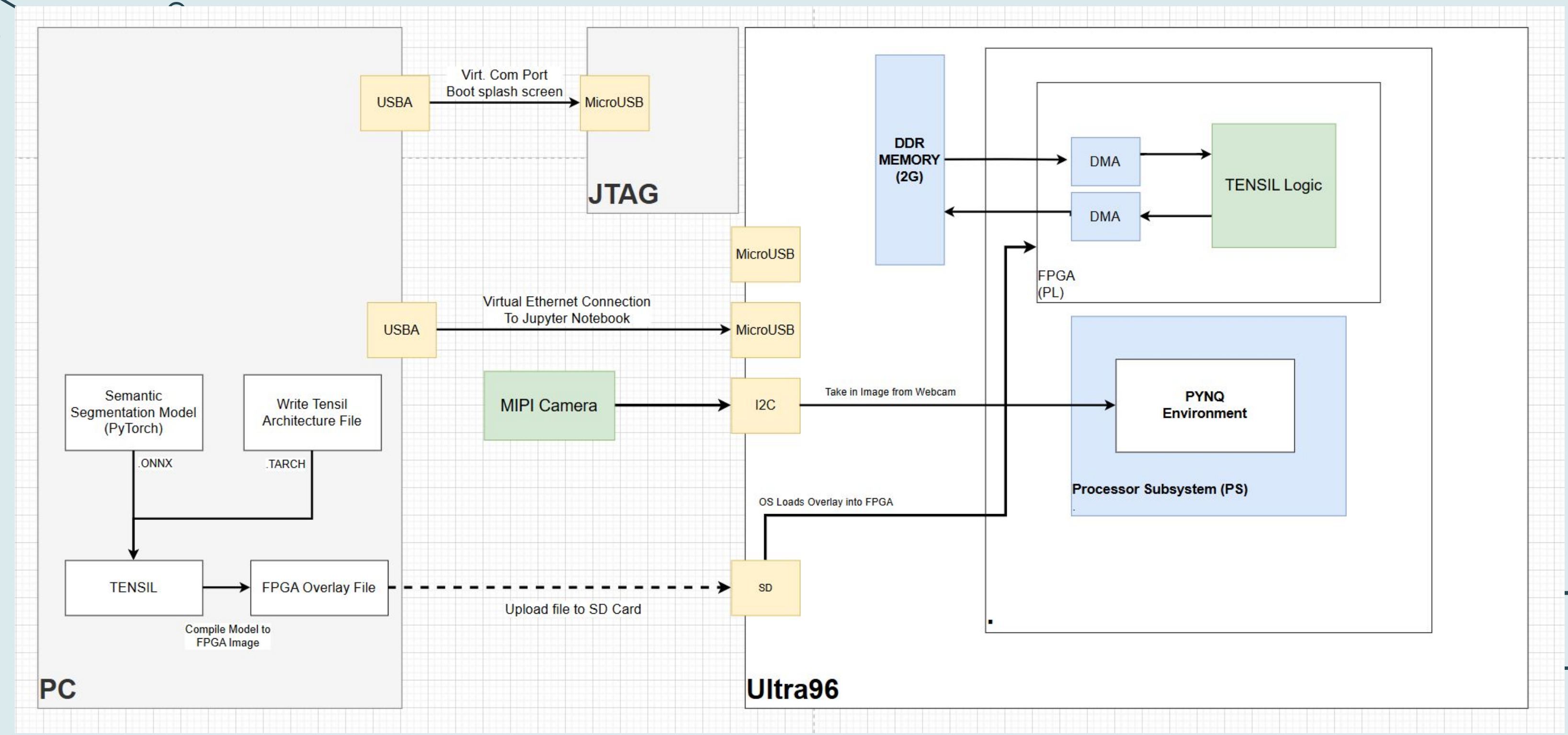


DESIGN & VISUALS

SYSTEM FLOW DIAGRAM

This represents the flow of data (an image/frame) will go through in our system.

SYSTEM BLOCK DIAGRAM



SYSTEM BLOCK DIAGRAM DESCRIPTION

ULTRA 96v2 Board

- AMD Zynq UltraScale+ processor
- 2GB DDR Memory
- PYNQ OS
  - Jupyter Notebook-based OS with Python APIs
  - Runs the Image preprocessing algorithm
- Uses AXI to send video data between the OS and model on the FPGA

THE MODEL

- Trained Model asynchronously compiled through Tensil-AI
- Model can now be used to generate and FPGA overlay for the board

I/O Devices

- MIPI Camera
- Ethernet connection via micro-USB for Jupyter notebook development
- Displayport output to monitor

Ethics

Ethical Concerns

Bias

- Will the Image Processing algorithm have unintentional gender or ethnicity bias?
- Training data must be diverse.

Liability

- Our system in the long-term may be used to determine life saving decisions.
- We must produce and be confident on the metrics we present.

Contextual Limitation

- How will the system's performance (accuracy) be impacted in non-ideal environments?
- Thoroughly test sub-systems in different environments.

CONCLUSION

As a result

Of our given problem and considerations of our project

We will

Increase the performance of an existing FPGA system

To achieve

Throughput high enough to make real-time decisions.

Linking to Our Client's Problem

This increase in data throughput will supplement our client's system, unlocking the ability to predict when end-users might have health-affecting events such as a seizure.

Thanks!

Any questions?