



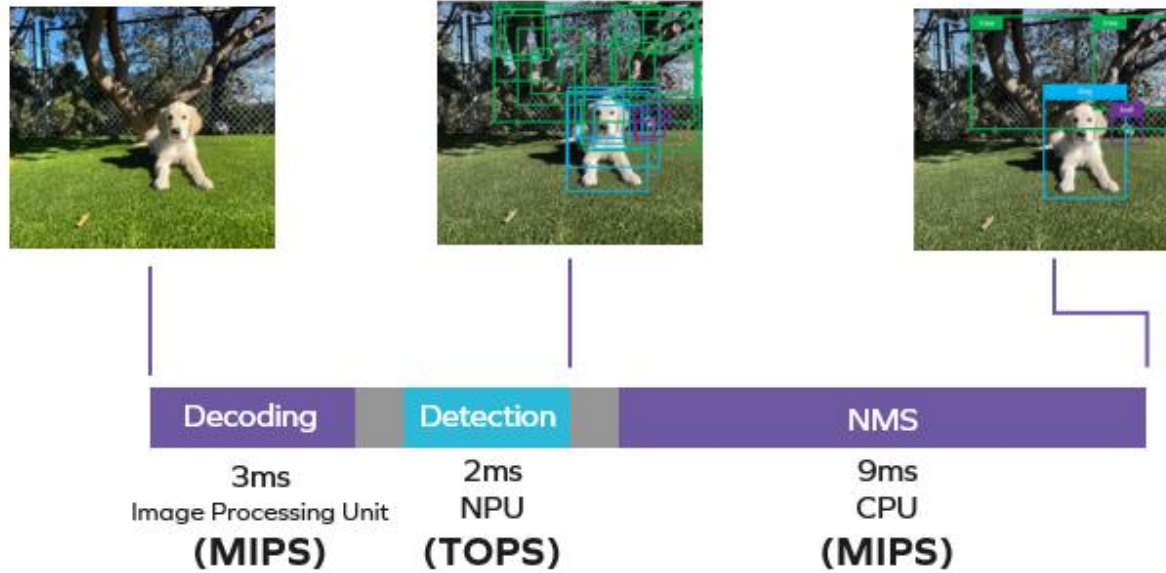
Accelerate All Your Algorithms with the Quadric q16 Processor

Daniel Firu
Co-founder & CPO
quadric.io

Software 2.0 – ML Inference + Traditional Code



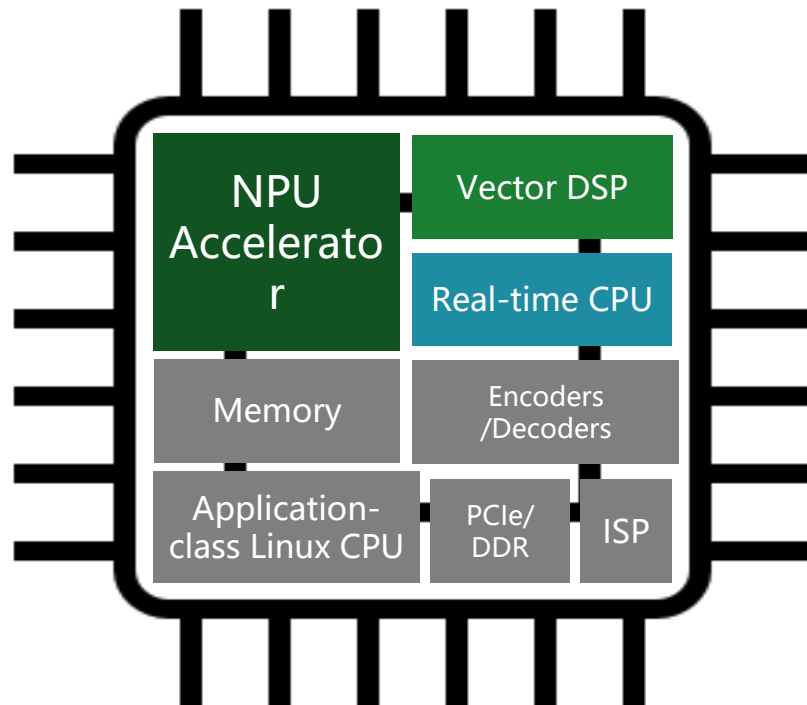
- Machine Learning is infiltrating nearly all applications
- Augmenting, not replacing, traditional application code
- Example: object detection pipeline:



- Almost everywhere DSPs are traditionally used today – vision, audio/sound, communications, sensors/radar



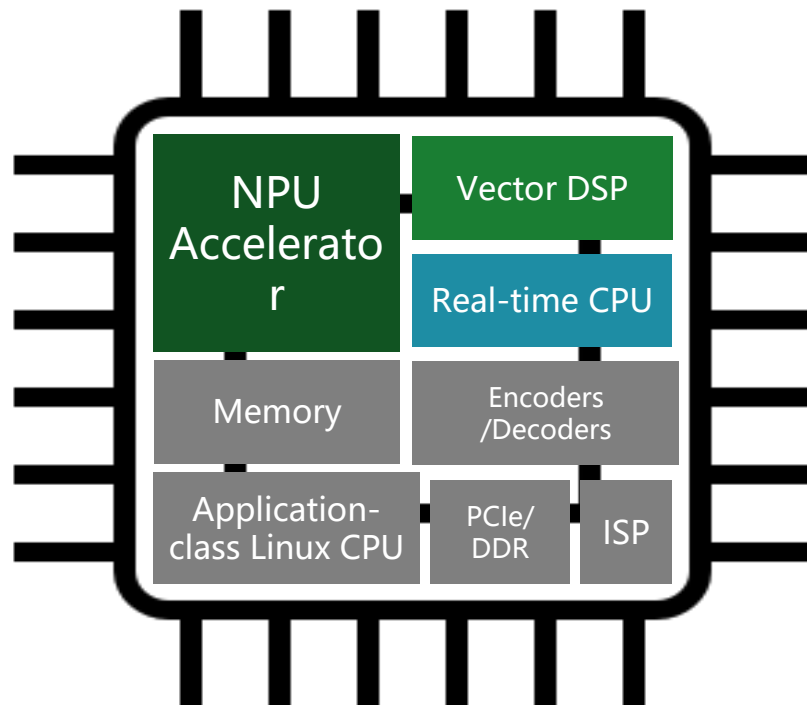
Heterogenous Multicore SoCs



Conceptual block diagram of a Smart Camera SoC

- NPU “Accelerators” (Matrix)
 - High efficiency
 - Limited operator support
- DSPs (Vector)
 - Good math kernel performance. Run traditional DSP code & some NN operators
 - Often weak for control code
 - Unfamiliar programming environment
- Embedded Control CPU (Scalar)
 - Orchestration of the real-time subsystem
 - “Operator fallback” – Easy to program
 - The failsafe to future-proof the system as NNs evolve
 - But ***very low*** ML performance

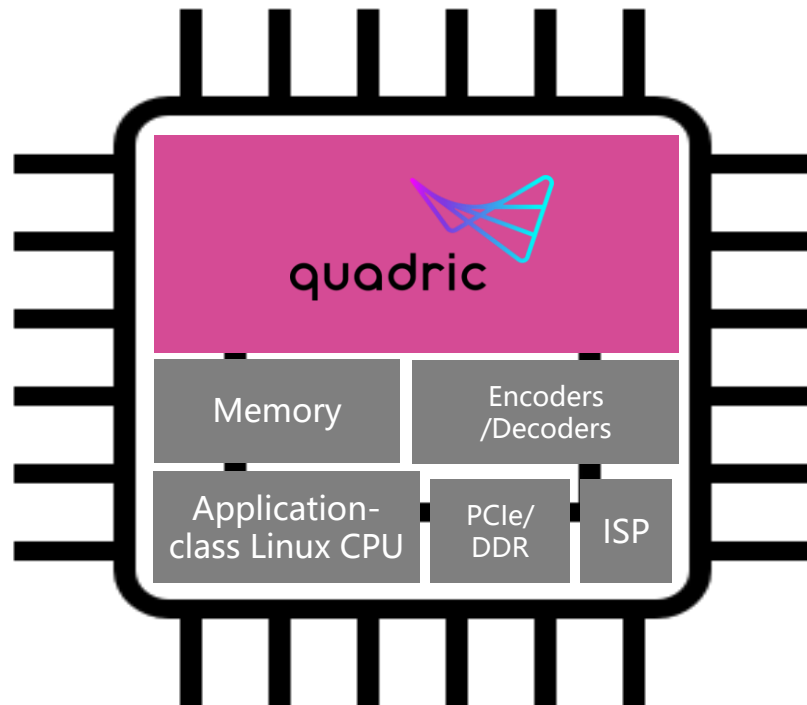
Heterogenous Multicore SoCs



Conceptual block diagram of a Smart Camera SoC

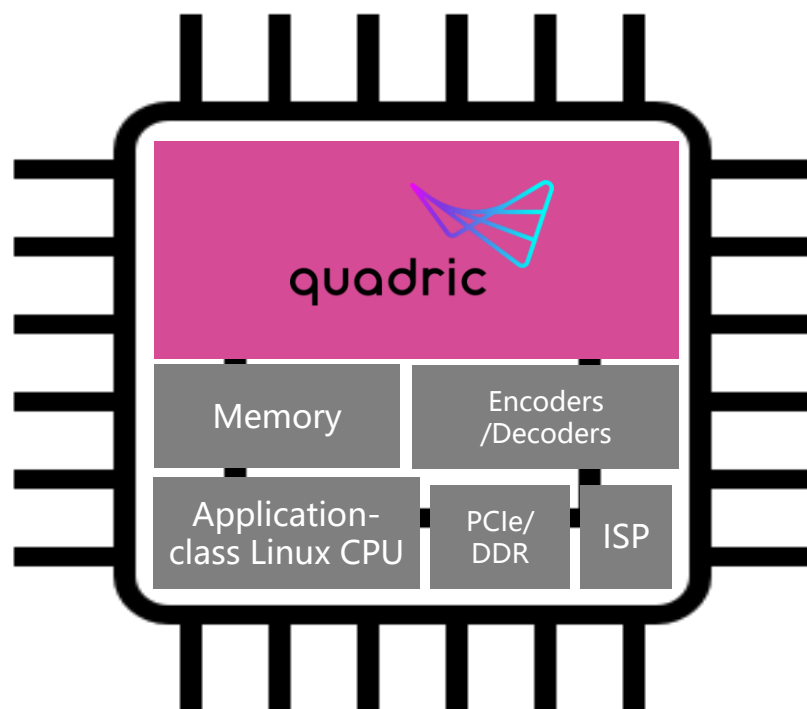
- System complexity / power
 - Hardware integration of multiple cores
 - Memory bandwidth sharing across cores
 - System power waste – data paging through external memory
- Programming complexity
 - Synchronization between cores
 - 2 or 3 toolchains (debug, performance tuning)
 - End OEM programmer accesses only the Real-Time CPU (under-used potential)
- Accelerator “brittleness”
 - Many NN accelerators have limited operator support
 - Leads to lower-performance “fallback” onto the DSP or CPU

Quadric GPNPU: Hybrid DSP + NN Graph Processor



- One architecture for inference plus pre- & post-processing
- NN graphs and C++ code merged into a single code stream
- Matrix + vector + scalar code in a single, unified architecture

Quadric GPNPU Advantages

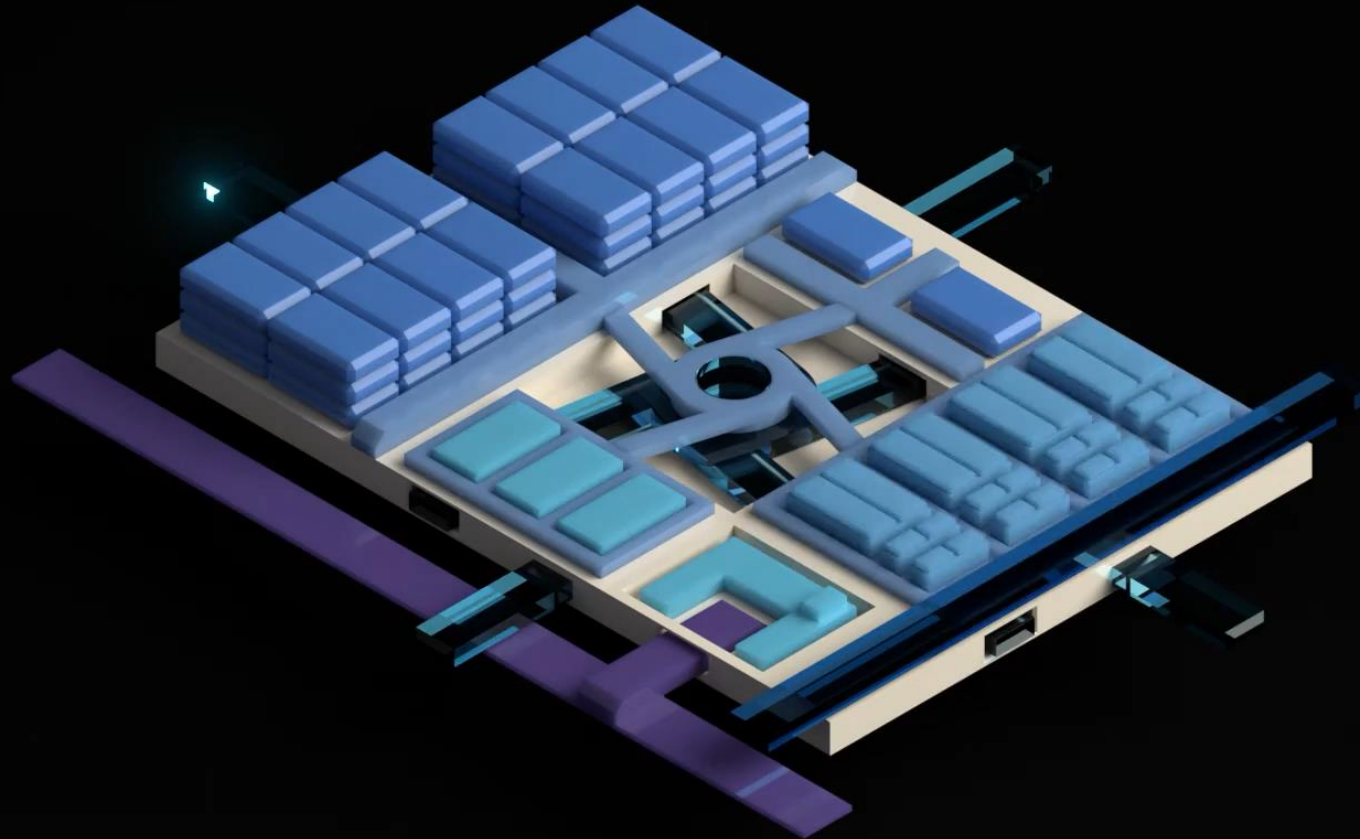


- System Simplicity
 - Only one multipurpose core to integrate
 - Memory bandwidth optimized by single unified compiler / scheduler
 - Lower power – intermediate data sharing between C++ and NN code sections
- Programming Simplicity
 - One toolchain for scalar+vector+matrix
 - Single debug of C++ and NN graph code
 - End-OEM programmer accesses 100% of the machine compute power
- Ultimate Future Proof Flexibility
 - New NN graph operators added with C++ kernels
 - New DSP libraries added w/ C++ code
 - Anything you can write in C++ will run

Architecture

The image features a teal background with various geometric elements. In the top right, there are overlapping rectangular shapes in shades of blue and teal, with a prominent red trapezoidal shape. A horizontal line of five small orange squares is positioned below the red shape. In the bottom right, there is a series of parallel lines and a dotted line pattern. The word "Architecture" is centered in a white, bold, sans-serif font.

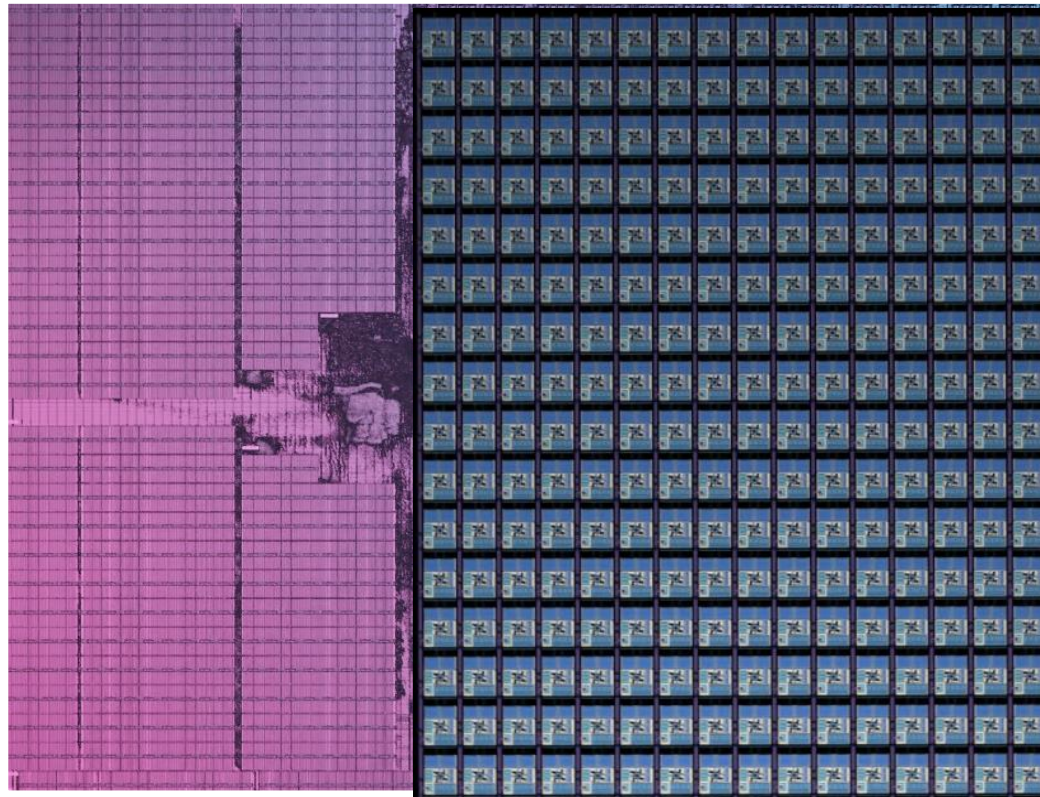
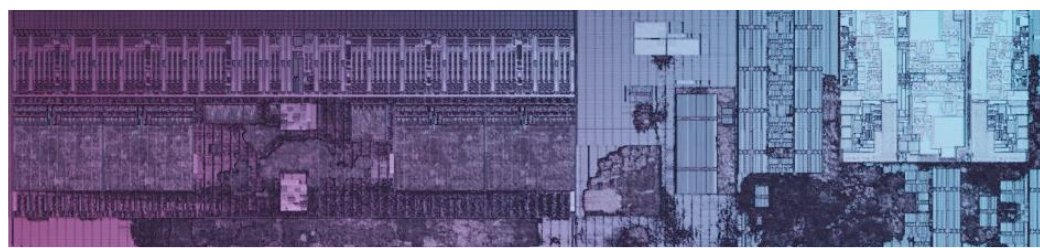
Quadric GPNPU Processor Architecture



Silicon Proven GPNPU: Quadric Processor

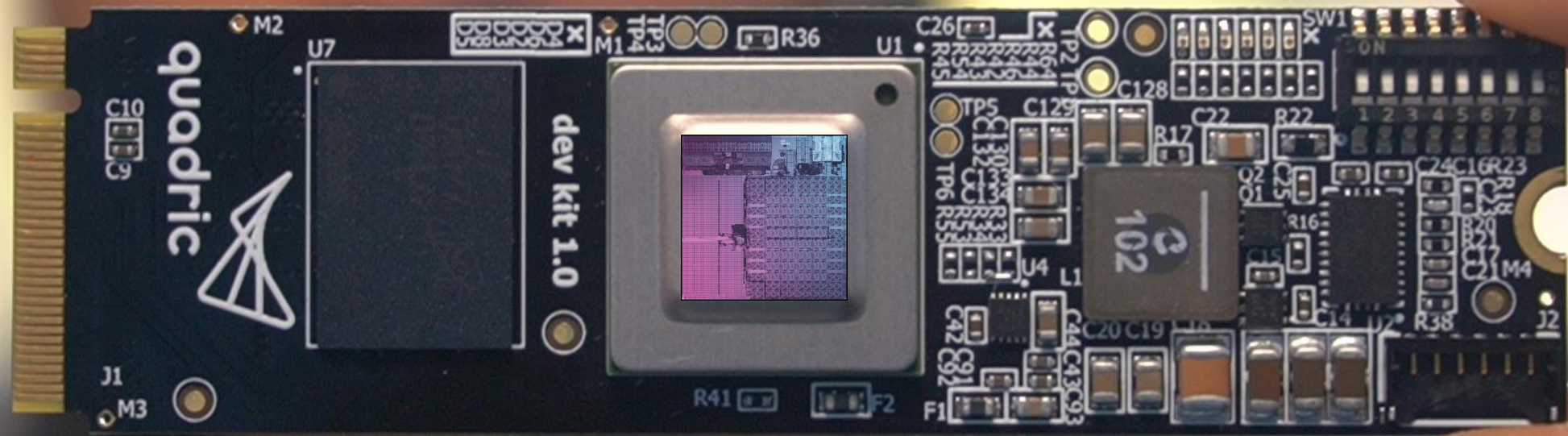


LPDDR4 and PCIE
Interfaces



256 Vortex Core
GPNPU Instance

Quadric Developer Kit

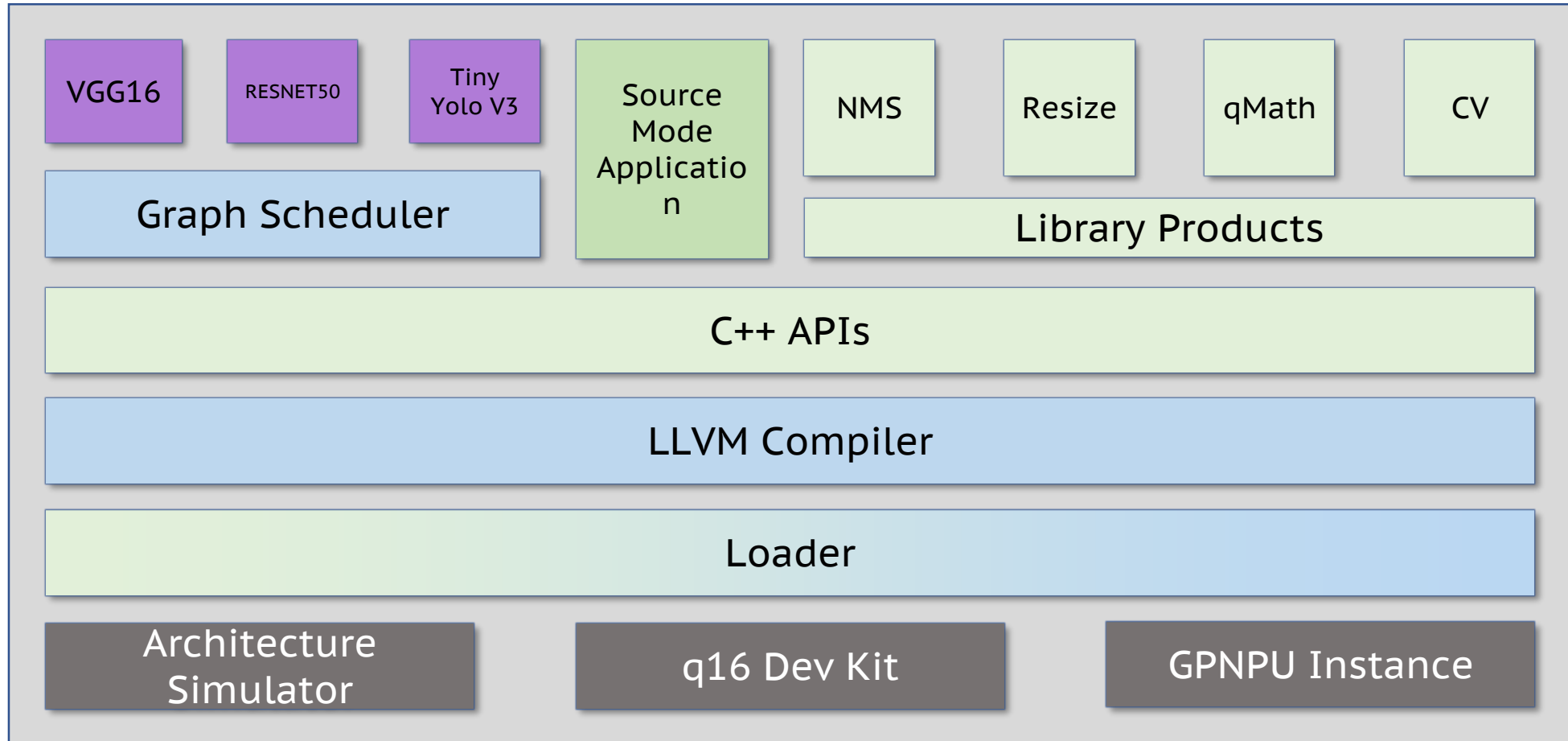


Silicon Proven
available for evaluation now

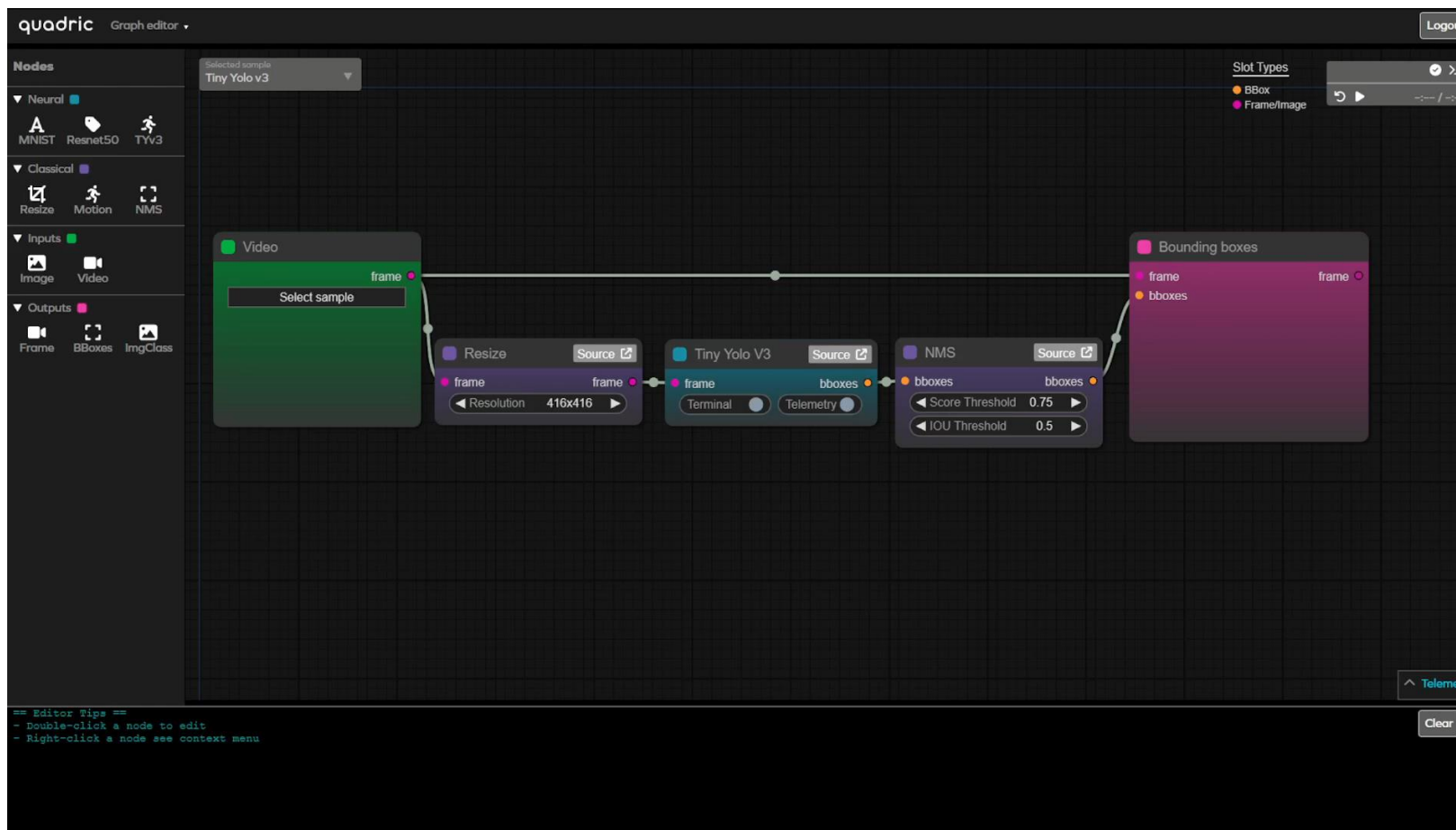
Software

The background is a vibrant teal color with various geometric elements. In the top left, there are several thin, parallel lines in yellow and light green. On the right side, there are layered, semi-transparent shapes in shades of teal and blue, with a prominent orange-red trapezoidal shape. In the bottom right corner, there is a series of small, dark blue squares arranged in a diagonal line. The overall aesthetic is modern and digital.

Quadric SDK: Unified Software Toolchain



Quadric Developer Studio



studio.quadric.io

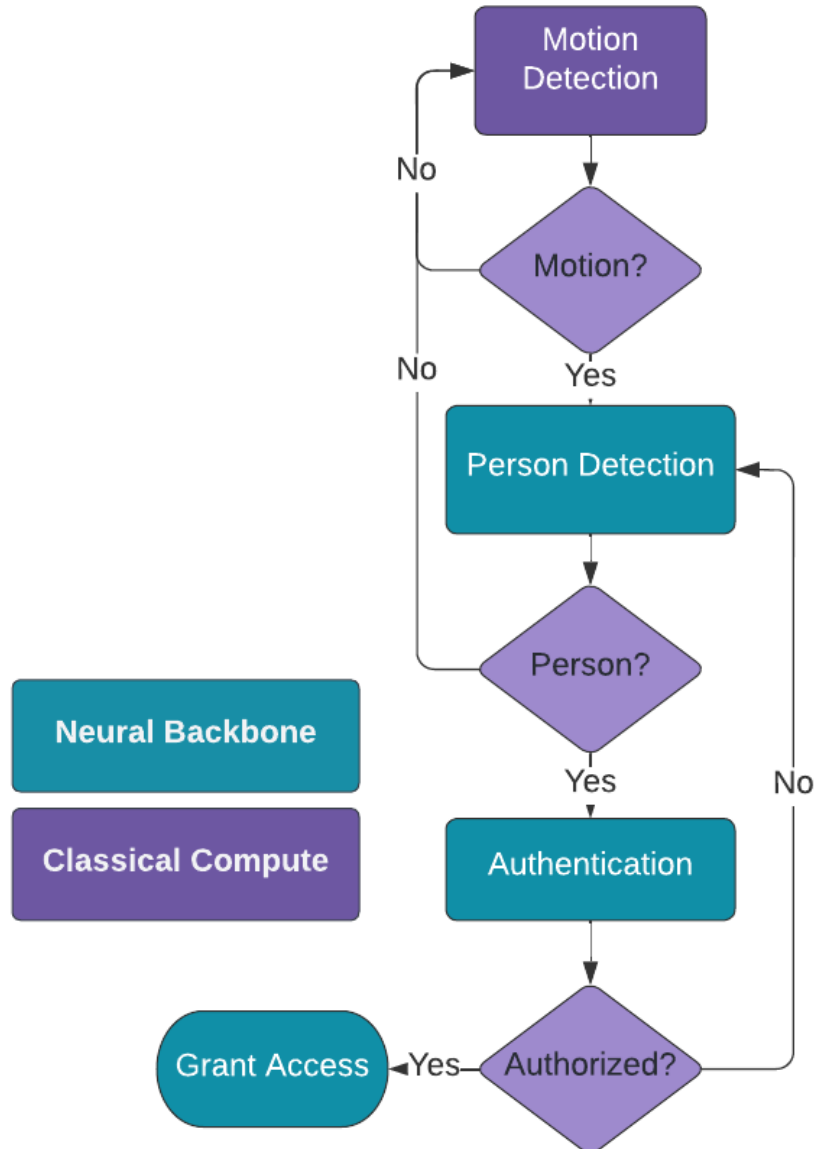
© 2022 quadric.io, Inc.



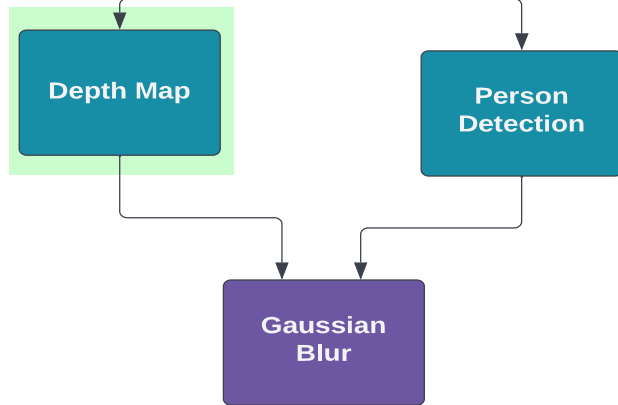
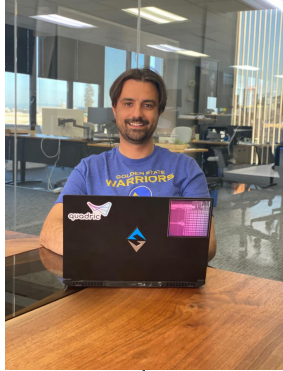
Practical Use Cases

The background features a teal gradient with various geometric elements. On the left, several thin yellow lines intersect. In the center, there are horizontal bars in shades of teal and a prominent yellow bar. On the right, there are layered geometric shapes, including a red trapezoid and a series of small orange squares.

Smart Physical Access: On-Device Detection and Authorization



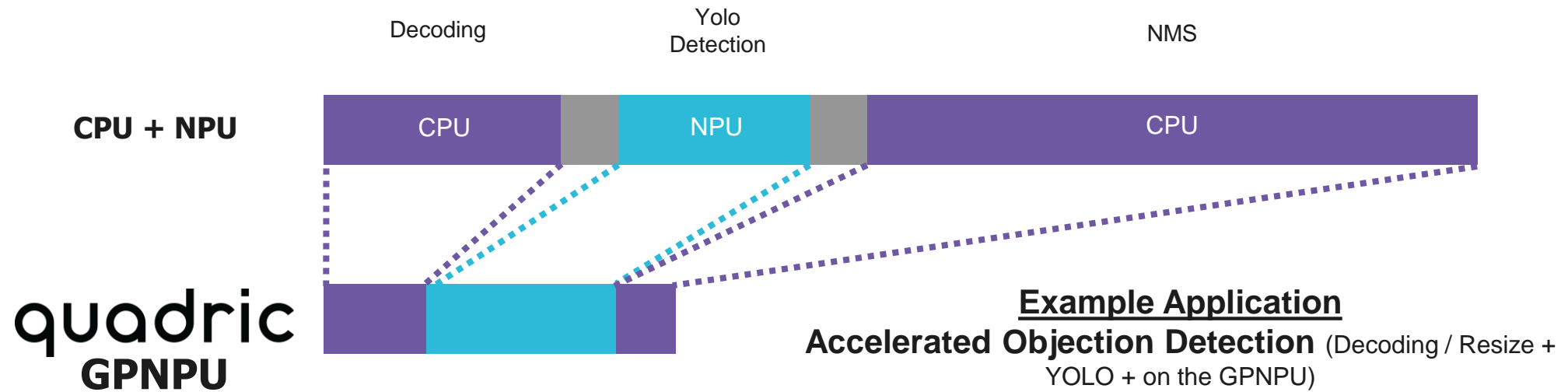
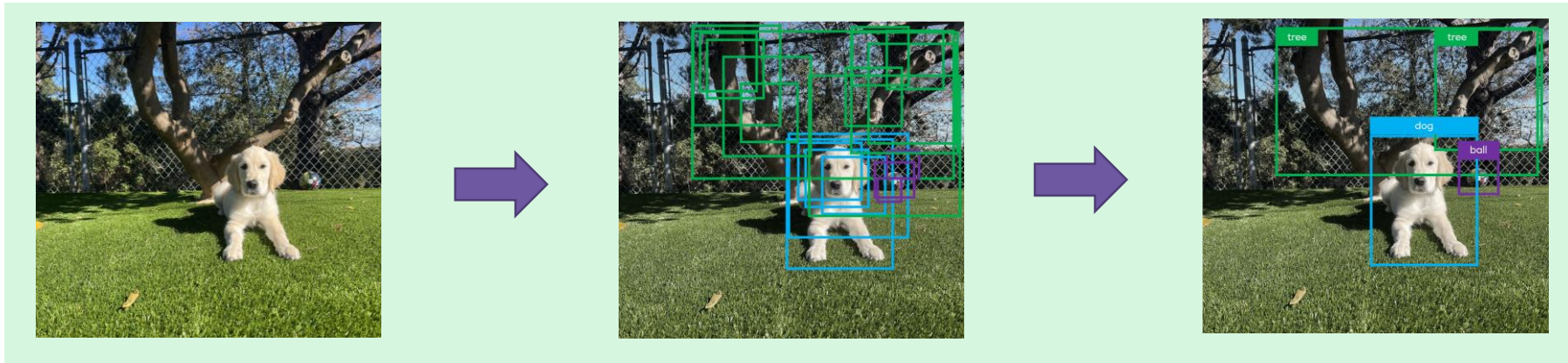
Computational On-Device Bokeh



Neural Backbone
Classical Compute



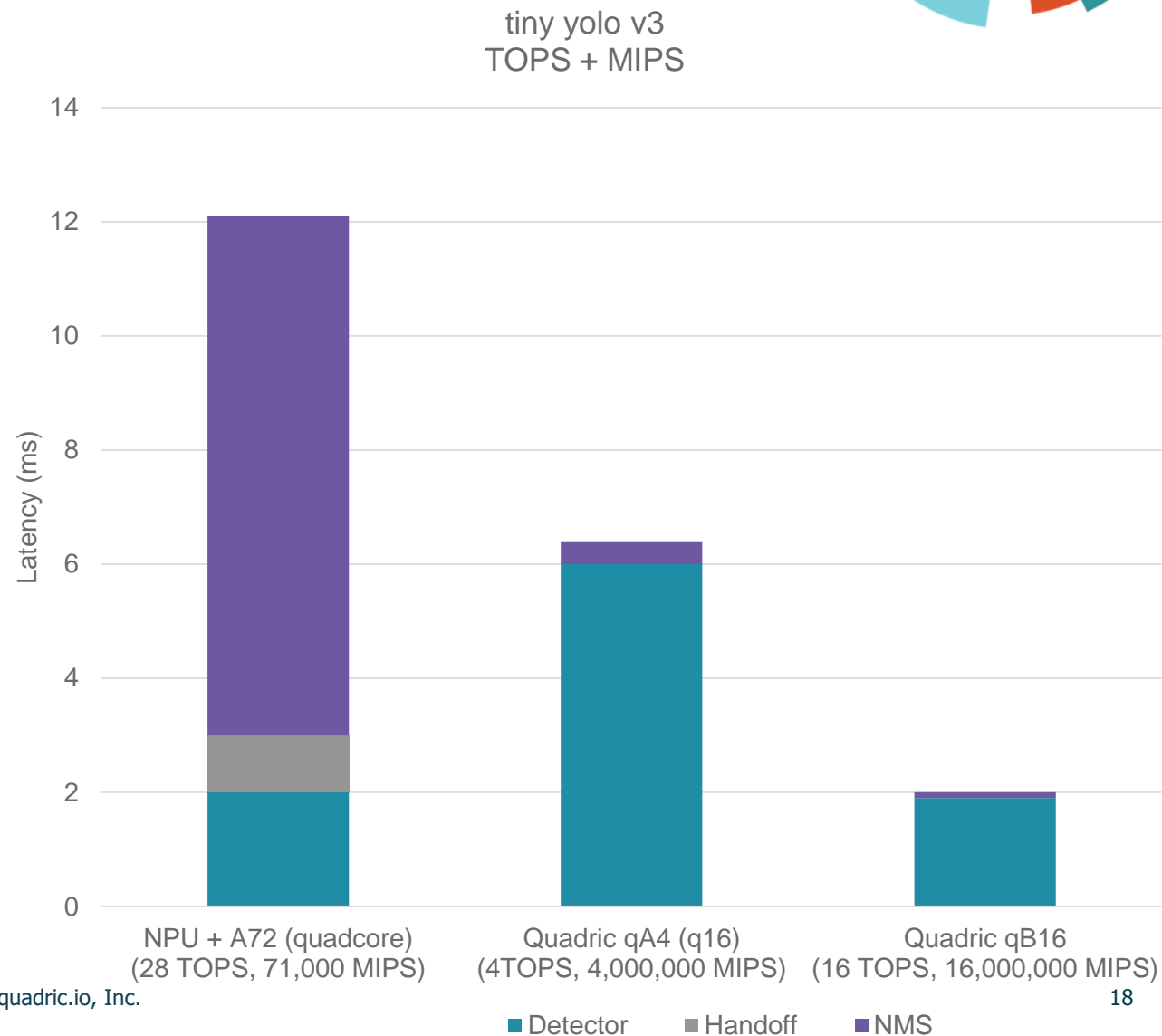
Tiny Yolo v3 Application Use Case



Tiny Yolo v3 Performance Comparison



- When deploying algorithms, total algorithm pipeline performance is key
- Quadric's GPNPU accelerates the entire pipeline, resulting in the best total performance

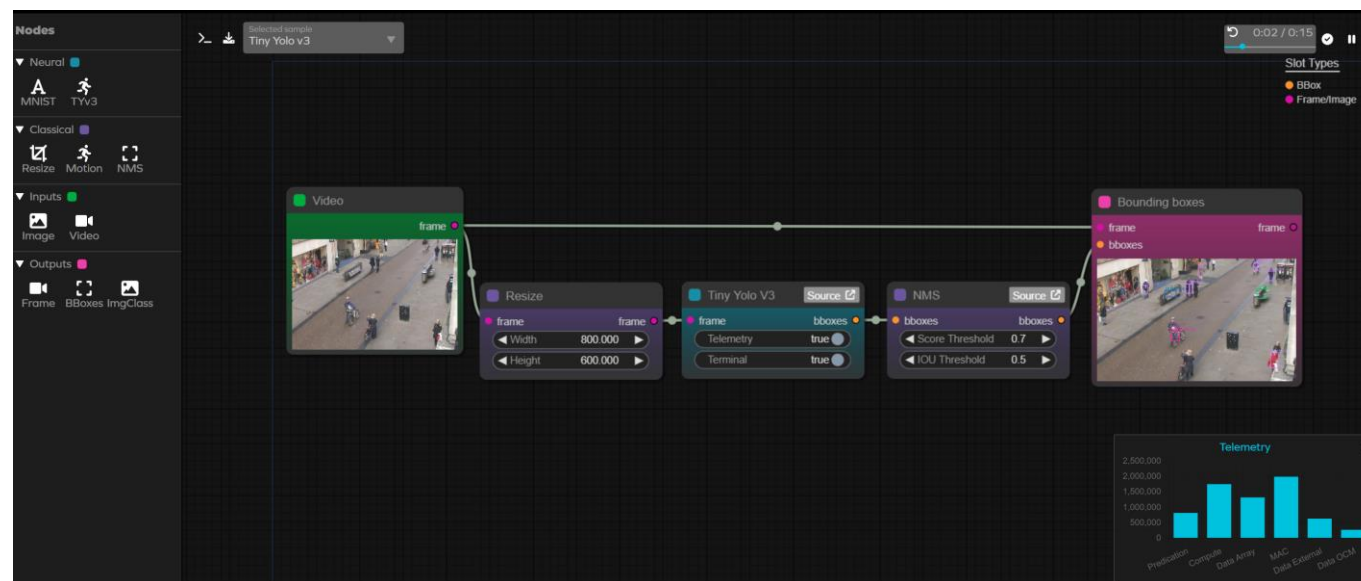


Conclusion



- Quadric develops GPNPU Processor IP technology
- The Quadric SDK enables the development of complex high-performance application pipelines that are a mix of both Neural and C++ Vector / Matrix Code
- Experience our technology on the Quadric Developer Studio!

- studio.quadric.io





HW / Product Resources

Architecture Information

quadric.io/technology/architecture

The Quadric Developer Studio

studio.quadric.io

Software Information

quadric.io/technology/sdk

docs.quadric.io

2022 Embedded Vision Summit

Visit us at our booth!

Booth #521