VeriSilicon

LEARNING AT THE SPEED OF SIGHT

Paul Werp

Shang-Hung Lin

Mahadev S. Kolluru

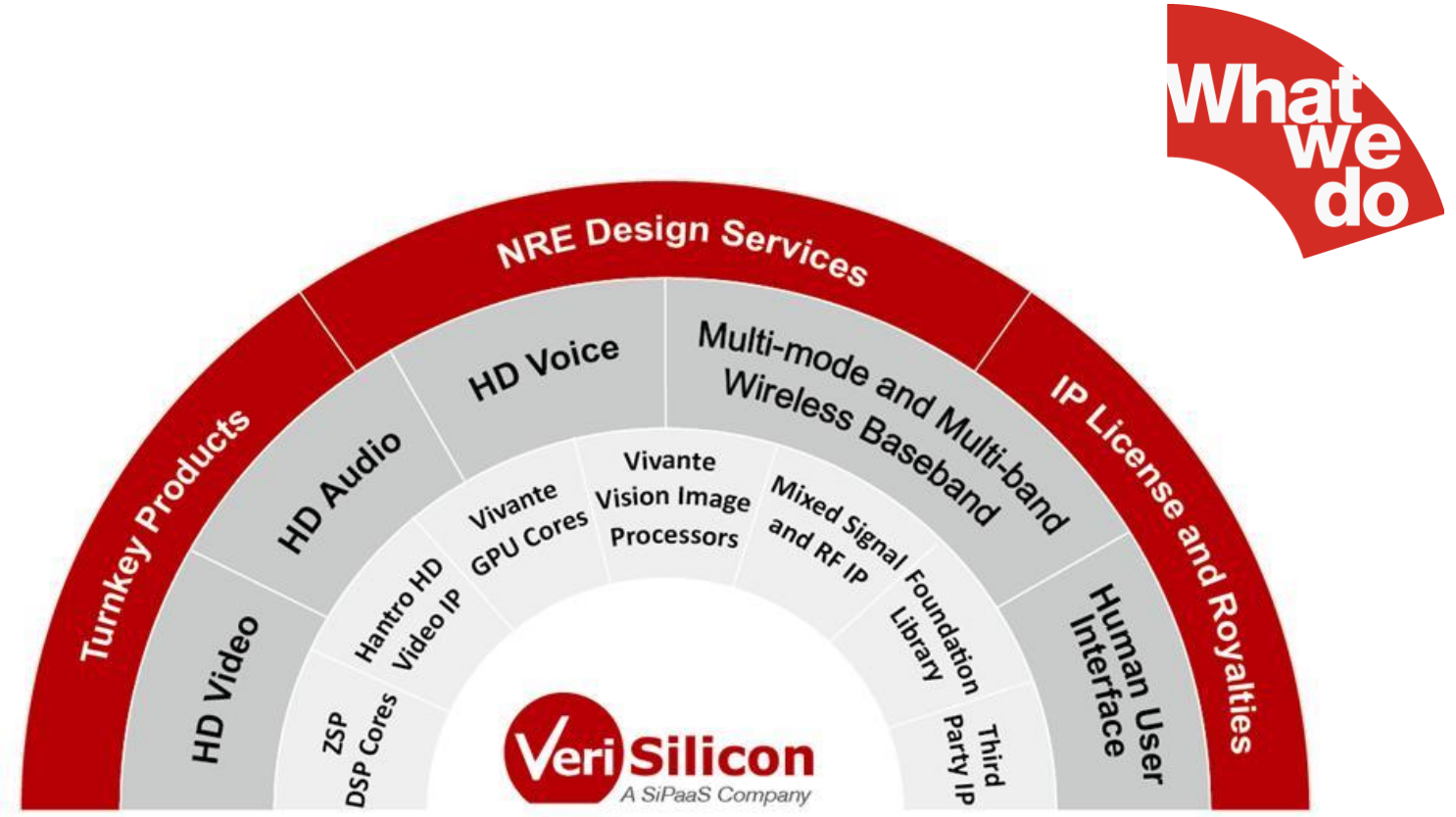
(mahadev.kolluru@verisilicon.com)

10/19/2016

VeriSilicon Global Operations

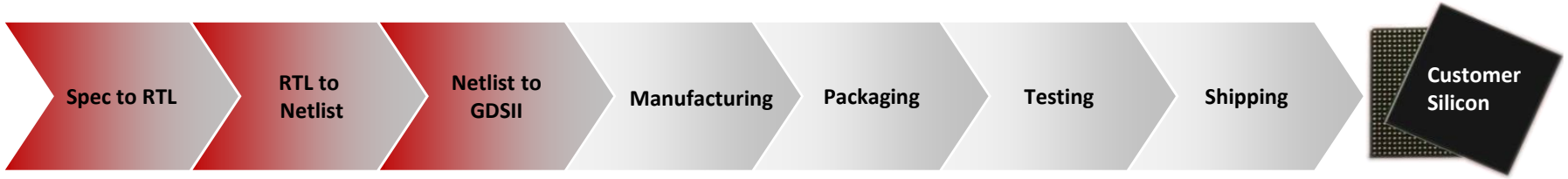
- Founded in 2001, currently 650+ employees
- 70% dedicated to R&D

IP-centric, Platform-Based, End-to-End Turnkey Service

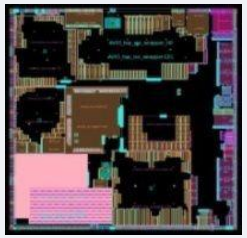


End-to-end Turnkey Service

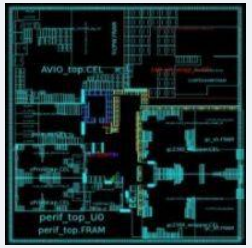
- Tape-out one chip a week; 50 chips a year
- Foundry neutral
- 98% first silicon success



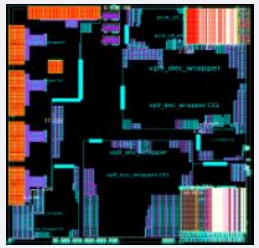
TSMC 28nm LP



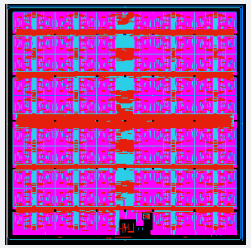
TSMC 28nm HPM



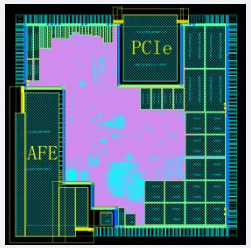
GF 28 nm SLP



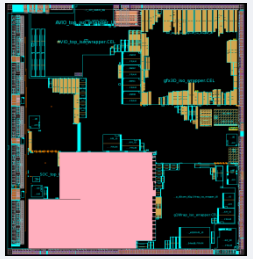
GF 28nm HPM



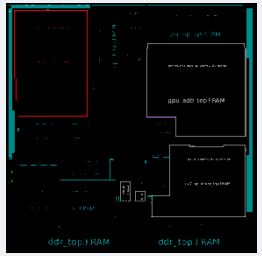
SEC 28nm LPP



UMC 28nm LP



SMIC 28nm HPM



Diversified Customer Base for Growing End Markets

Established Fabless and IDM Semiconductor Companies

Emerging Fabless Semiconductor Companies

OEMs

ODMs

Large Internet Platform Companies

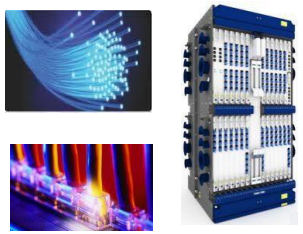
VeriSilicon

Consumer Electronics

Mobility

Wearables

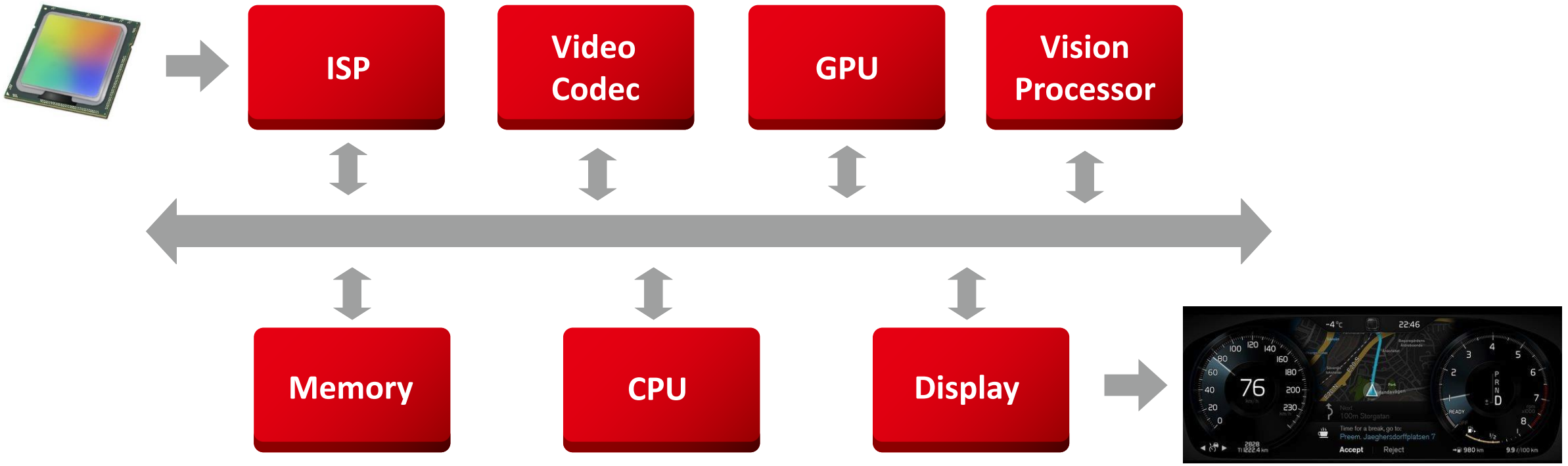
Networking



Automotive

VeriSilicon

Silicon Platform as a Service (SiPaaS) Example



VeriSilicon

Movement of Intelligent Devices

▲**Multi-Media**

- ▶ Graphics, Video, Audio, Voice

Movement for next decade

▲**Natural User Interface**

- ▶ VISION
- ▶ Natural Language Processing
- ▶ Sensors

VeriSilicon

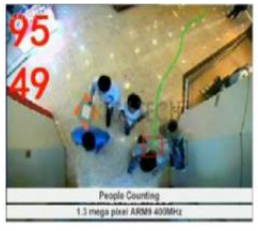
Vision Processing

- Wide range of applications, thousands of algorithms
- New applications come up everyday
- Compute intensive – need HW acceleration
- Best algorithm keeps changing – need programmable solution

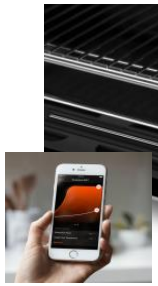












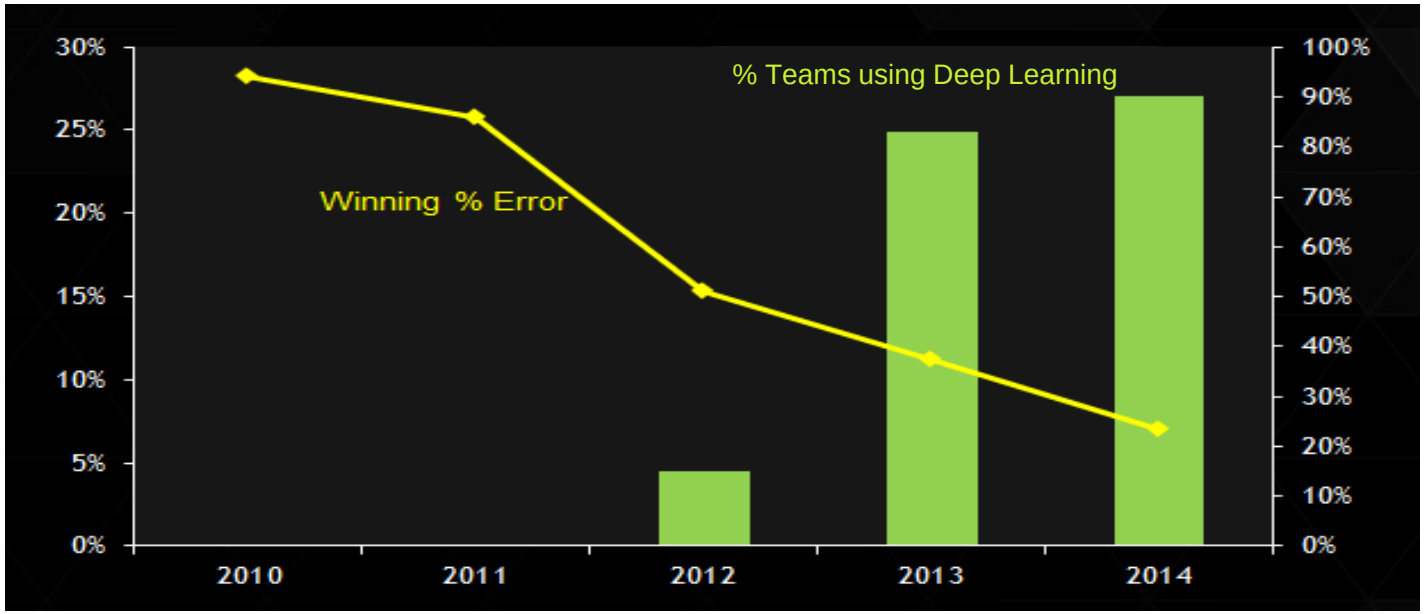


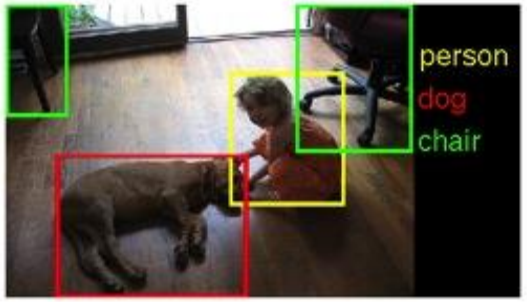


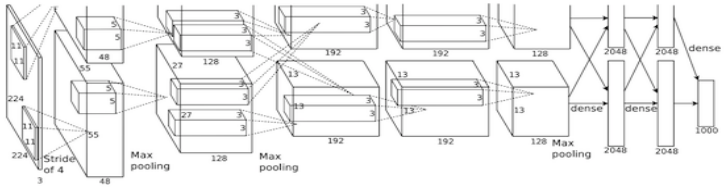
Deep Learning: Neural Networks Return with Vengeance

- With big data and high density compute, deep neural networks can be trained to solve "impossible" computer vision problems

IMAGENET



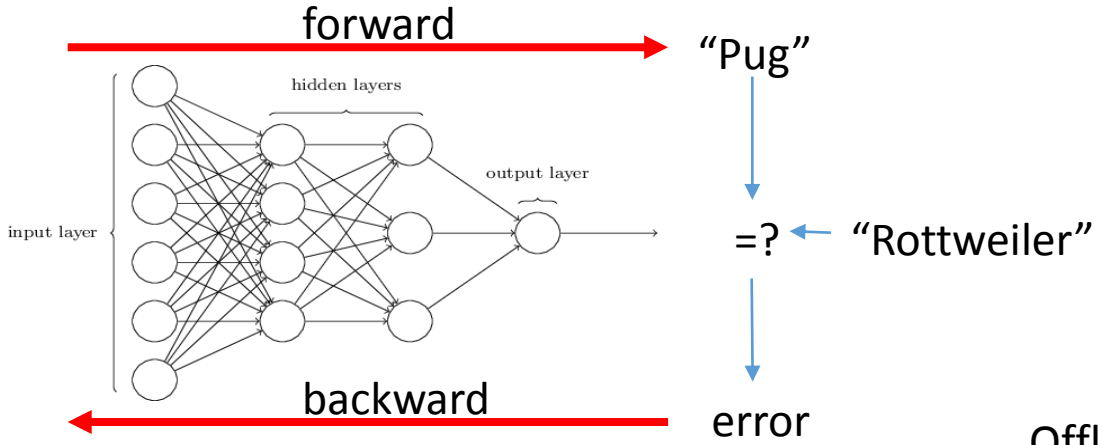




VeriSilicon

The Basics of Deep Neural Networks (DNN)

▲Training: 10^{16}-10^{22} MACs/dataset

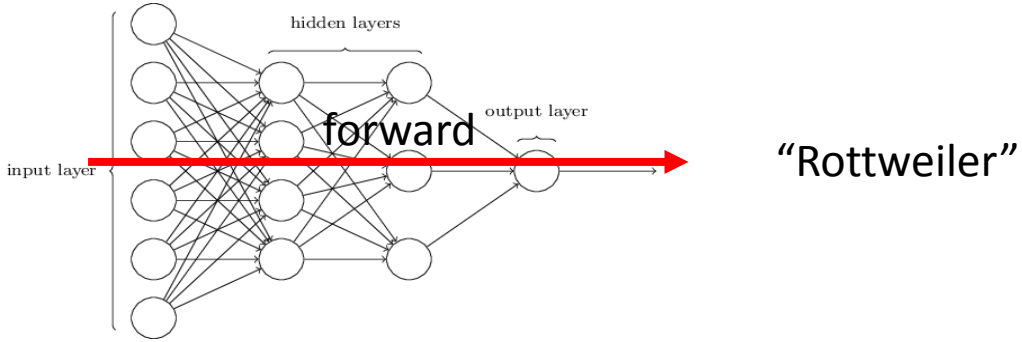


100K-100M Network Coefficients

Offline (mostly)

On-the-fly

▲Inference: 10^{6}-10^{11} MACs/image



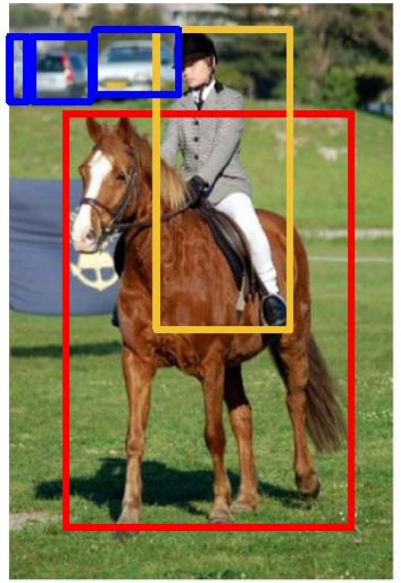
VeriSilicon

Computer Vision Problems That DNN is Good at Solving

Classification

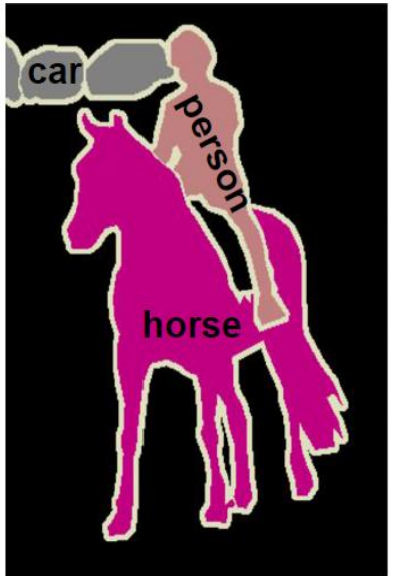
- ☐ dog
- ☑ car
- ☑ horse
- ☐ bike
- ☐ cat
- ☐ bottle
- ☑ perso

Detection



car person horse

Segmentation



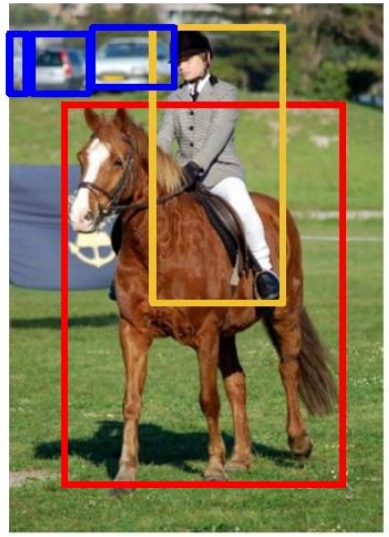
VeriSilicon

Challenges of Real-Time Inference

Classification

100G MAC/s

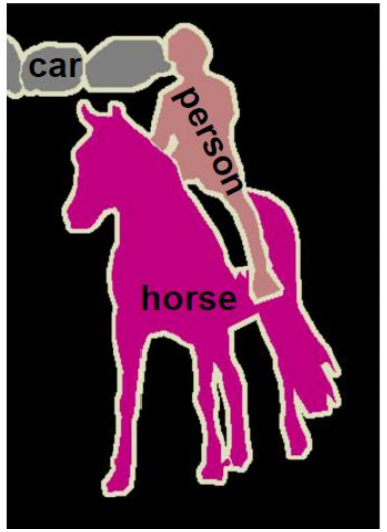
Detection



car person horse

1T MAC/s

Segmentation



10T MAC/s

- **Majority of deep neural net computation is Multiply-Accumulate (MAC) for convolutions and inner products**
- **Memory bandwidth is a bigger challenge**
  - Convolution in deep neural nets works with not just 2D images, but high-dimensional arrays (i.e. “tensor”)
  - Example: AlexNet (classification) has 60M parameters (240MB FP32), DDR BW with brute force implementation: 35GB/s

VeriSilicon

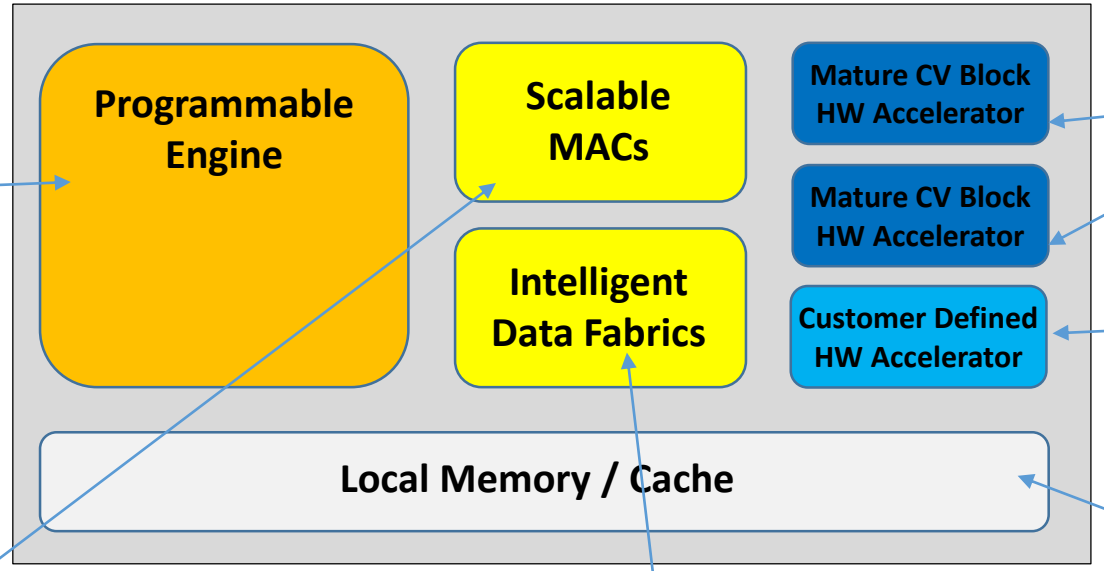
Embedded Vision: Common Acceleration Strategies

	Description	Pro's	Con's	Notes
Fixed Function HW	Specialty Logic that performs a single set of vision functions	Small / Fast	Not flexible. Algorithm stuck in time.	Vision algorithms evolve very quickly
Proprietary Vision Engine	Group of DSP devices that perform 2D calculates quickly	Power Efficiency	Algorithms are not portable, Architecture is usually also proprietary	Initial algorithm optimization commitment forces long term investment in a single HW Vendor
GPU	Leverage the ~10x compute power of a GPU for general compute	High Performance Availability Code Portability	Power Consumption Not efficient for sparse compute	Usually requires a CPU to perform control and classification tasks
CPU	Quickly leverage the CPU to perform the vision algorithm from top to bottom.	Code Portability Time to market Well understood	Lower compute throughput Shared with OS and other apps	Performance bound and not ideal for dense compute
CPU SIMD	CPU SIMD instruction set utilized for the higher compute performance	Available Speed increase in performance	SIMD setup overhead is high Performance is still bound by CPU being shared by OS and other apps	Useful, but reduces algorithm flexibility because SIMD instruction intrinsic are not well supported in general compilers

VeriSilicon

Efficient Processor Architecture to Enable DNN for Embedded Vision

Unified programming model to synchronize all blocks under industry standard API



- Programmable engine to cope with new CV algorithms
- Future new NN layers can be implemented here
- **Scalable** architecture to support different PPA (Performance, Power, Area) requirements

- High utilization MACs for DNN
- **Scalable** architecture to support different PPA (Performance, Power, Area) requirements

- Handles other DNN functions (normalization, pooling,…)
- Handles pruning, compression, batching… to increase MAC utilization and decrease DDR BW

Custom RTL to accelerate mature CV algorithms for 24/7 low power operation

I/F to extend HW acceleration for application specific purpose

For data synchronization between threads/blocks and minimize DDR BW

VeriSilicon

VeriSilicon Vision Image Processor (VIP)

Vision & Image Processor
VIP Series

OpenVX

OpenCL

VIP 8000

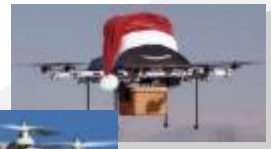
VIP Nano

Automotive
ADAS, In-Car Vision





Drones



Surveillance


IoT




1 Core

2 Cores

4 Cores

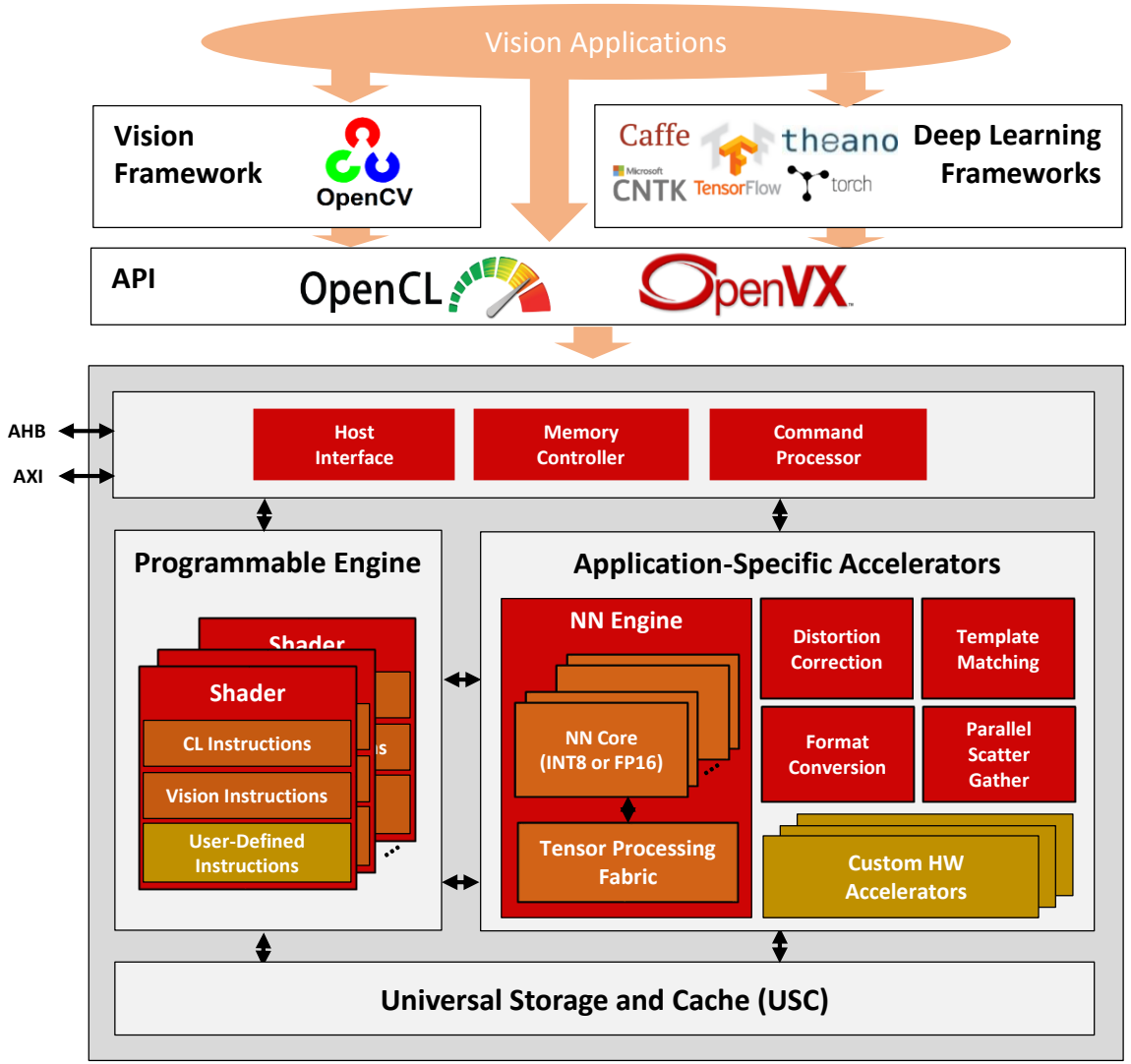
8 Cores

16 Cores

VeriSilicon

VIP8000 Architecture

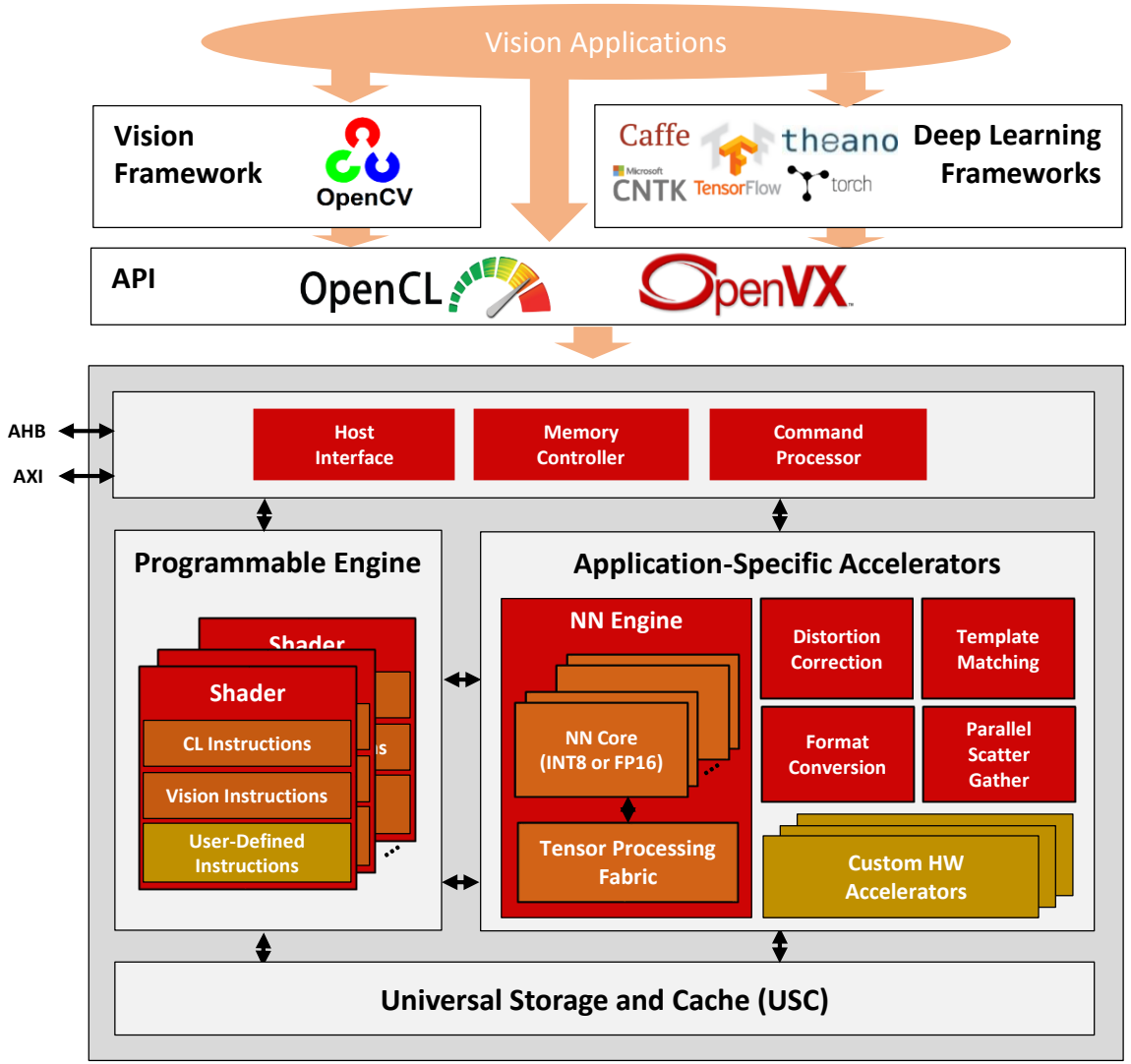
- **Programmable Engine**
  - 128-bit vector processing unit (shader)
  - OpenCL shader instruction set
  - Enhanced vision instruction set (EVIS)
  - INT 8/16/32b, Float 16/32b
- **NN Engine**
  - Convolution and inner products
  - 128 MACS/cycle per core
  - INT8 or Float16
- **Tensor Processing Fabric**
  - Data shuffling, normalization, pooling/unpooling, LUT, etc.
  - Network pruning support, zero skipping, compression
  - Accepts INT8 and Float16 (Float16 internal)
- **USC**
  - Configurable Local memory and L1 cache to pass data among shaders, NN, and HW accelerators
- **Number of shader cores and NN cores can be independently configured**



VeriSilicon

VIP8000 Features

- **Unified Programming Model**
  - OpenCL, OpenVX
  - Parallel processing between shaders and HW accelerators
  - EVIS Built-in functions to expose HW acceleration to programmers
  - Support popular vision and deep learning frameworks
- **Scalability**
  - Number of shaders and NN cores can be configured independently
  - Same OpenVX/OpenCL code runs on all processor variants; scalable performance
- **Extendibility**
  - VIP-Connect™: HW and SW I/F protocol to plug in customer HW accelerators and expose functionality to programmers via EVIS Built-in
  - Reconfigurable EVIS for user to define own instructions
- **24/7 Low Power Operations**
  - Optimized RTL for mature and commonly used CV functions

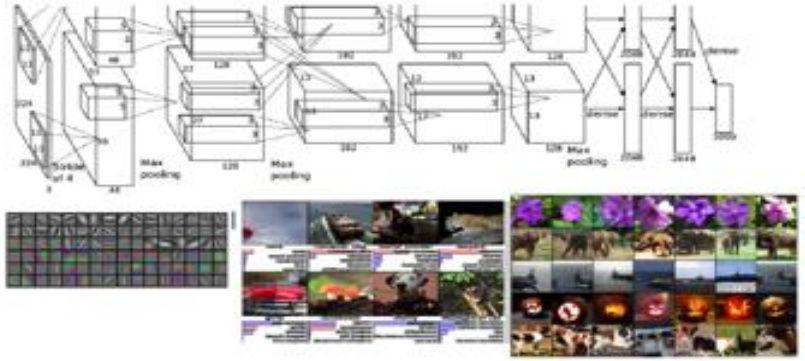


VeriSilicon

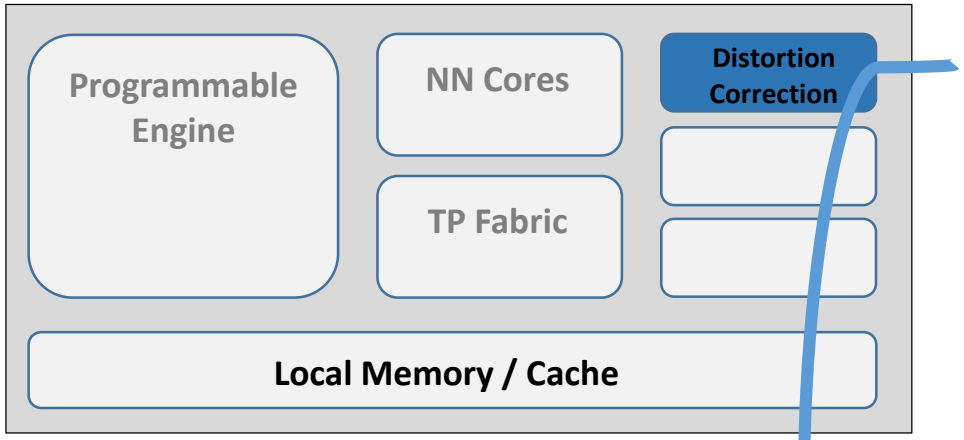
Scalable Performance from VIP8000

VIP8000 Series	VIP Nano	VIP8000UL	VIP8000L	VIP8000
Number of Execution cores	1	2	4	8
Clock Frequency SVT @WC125C (MHz)	800	800	800	800
HD Performance@800MHz				
Perspective Warping	60 fps	120 fps	240 fps	480 fps
Optical Flow LK	30 fps	60 fps	120 fps	240 fps
Pedestrian Detection	30 fps	60 fps	120 fps	240 fps
Convolutional Neural Network (AlexNet)	125 fps	250 fps	500 fps	1000 fps

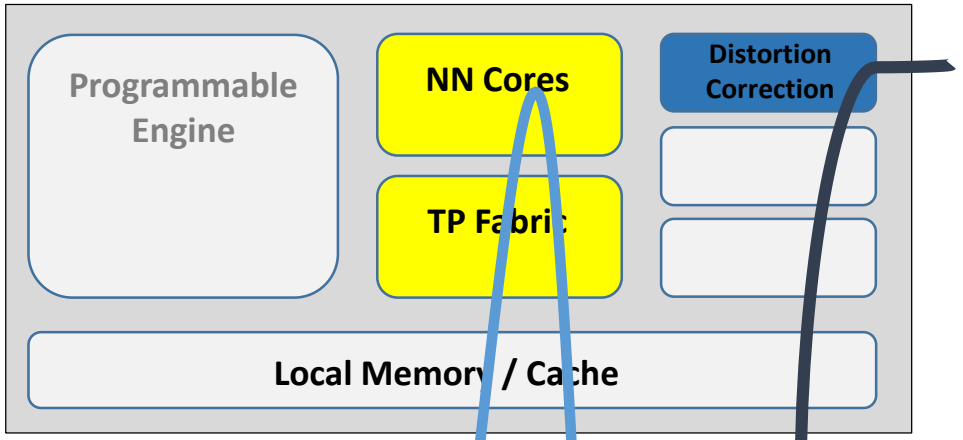
▲Scalable performance with SAME application code on different processor variants



Use Case Study: Always-On Camera

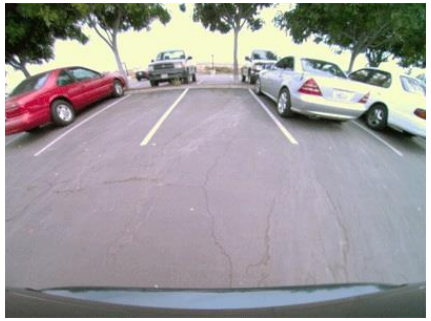


Dewarp and scale input frame,
Store result to on-chip SRAM



Run classification or detection
Store result to on-chip SRAM

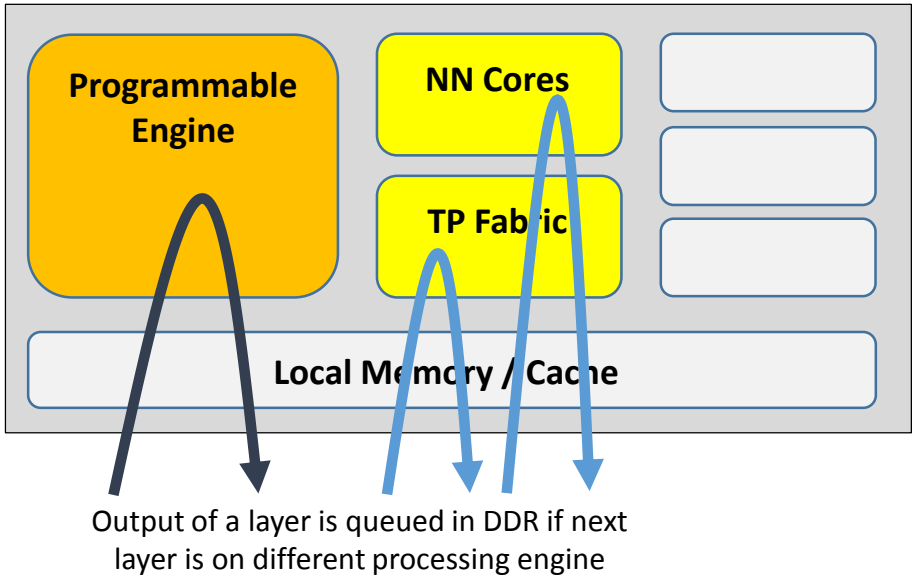
Dewarp and scale next frame
simultaneously

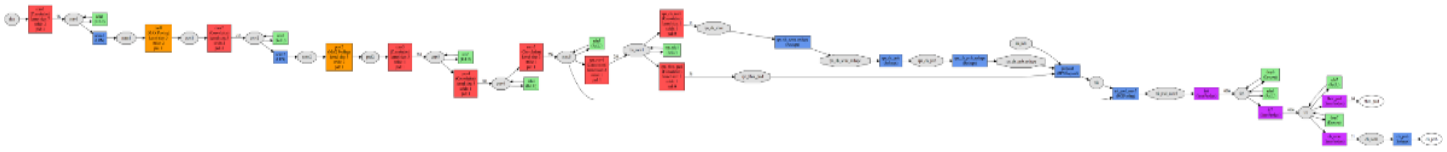


VeriSilicon

Use Case Study: Faster RCNN

- **One of state-of-the-art object detection CNNs**
  - Network configuration
    - 300 region proposals
    - 60M parameters, 30G MACs/image
- **Collaborative computing in VIP**
  - NN Engine: number crunching
    - Convolution layers, full connected layers
  - Tensor Processing Fabric: data rearrangement
    - LRN, max pooling, ROI pooling
  - Programmable Engine: precision compute
    - RPN, NMS, softmax
- **Real-time performance (30fps HD) under 0.5W @ 28nm HPM**

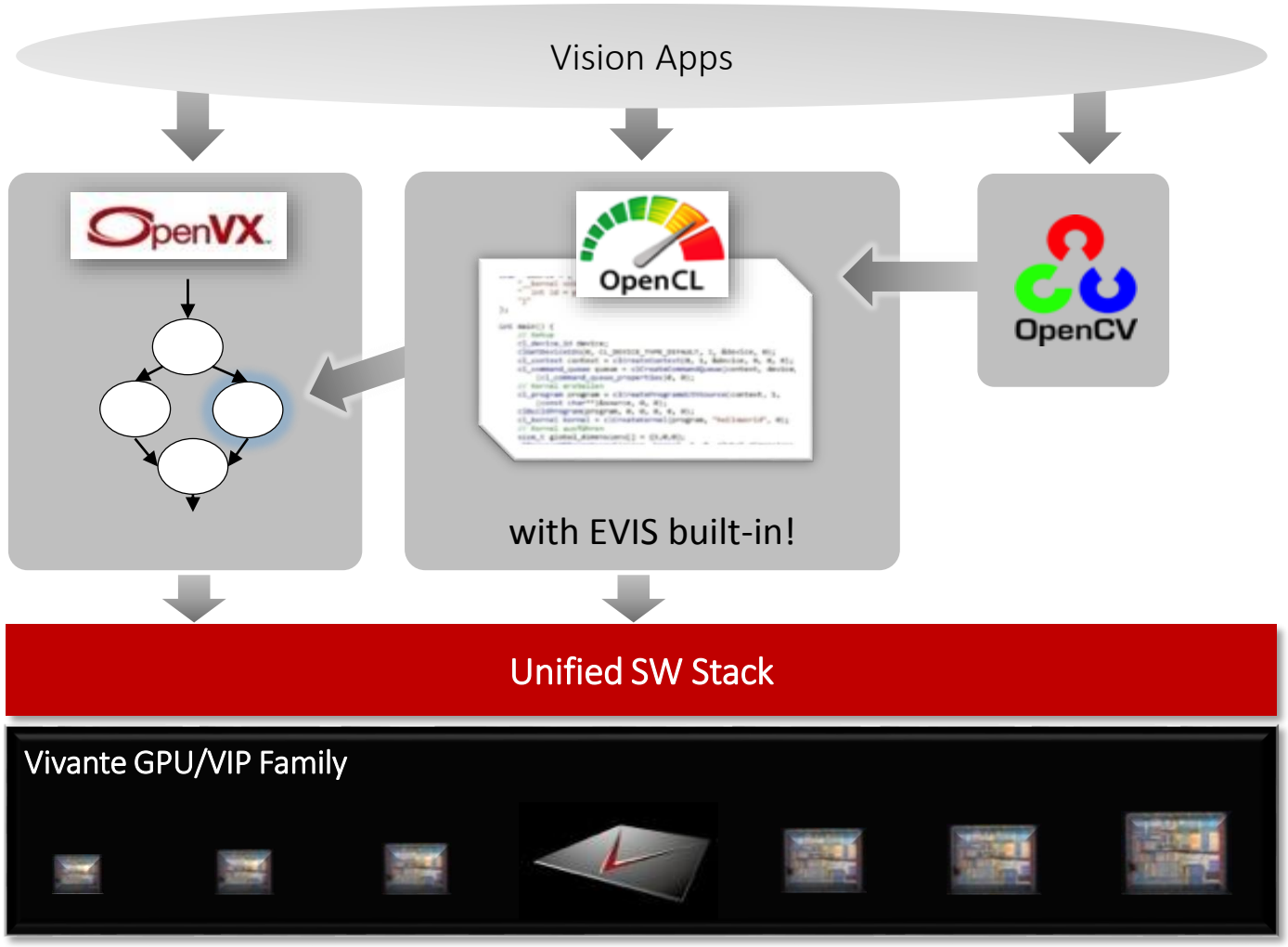




VeriSilicon

Various API Choices for Creating Vision Apps

- **API Supported for VIP acceleration:**
  - OpenCL
  - OpenVX
  - OpenCV
- **Wide Range of OS Supported:**
  - Android
  - Linux
  - Chrome
  - Ubuntu
  - Windows
  - Wind River
  - QNX
  - Green Hills



VeriSilicon

VIP Program Example

```
#include "cl_viv_vx_ext.h"

    _viv_uniform int height;
    __kernel void gaussian3x3
    (
        __read_only image2d_t in_image,
        __write_only image2d_t out_image
    )
    {
        int2 coord = (int2)(get_global_id(0), get_global_id(1));
        int2 coord1 = coord + (int2)(-1, -1);
        vxc_uchar16 v0, v1;
        VXC_ReadImage(v0, in_image, coord1, 0, VXC_MODIFIER(0, 15, 0, VXC_RM_TowardZero, 0));
        int2 coord2 = coord + (int2)(-1, 0);
        VXC_ReadImage(v1, in_image, coord2, 0, VXC_MODIFIER(0, 15, 0, VXC_RM_TowardZero, 0));

        int info = VXC_MODIFIER(0, 13, 0, VXC_RM_TowardZero, 0);
        int infox = VXC_MODIFIER_FILTER(0, 13, 0, VXC_FM_Guassian, 0);

        do
        {
            int2 coord3 = coord + (int2)(-1, 1);
            vxc_uchar16 v2;
            VXC_ReadImage(v2, in_image, coord3, 0, VXC_MODIFIER(0, 15, 0, VXC_RM_TowardZero, 0));
            vxc_uchar16 p_out;
            VXC_Filter(p_out, v0, v1, v2, infox);
            VXC_WriteImage(out_image, coord, p_out, info) ;
            v0 = v1;
            v1 = v2;
            coord.y++;
        }
        while(coord.y<height);
    }
```

▲OpenCL Programming

- ▶ Simple control; multi-thread data read/write scheduled by HW
- ▶ Automatic image border handling

▲EVIS Built-In

- ▶ Intrinsic functions for vision acceleration
- ▶ Ex: VXC_Filter() produces 14 filtered pixel outputs (per processor core) in one cycle

▲Scalability

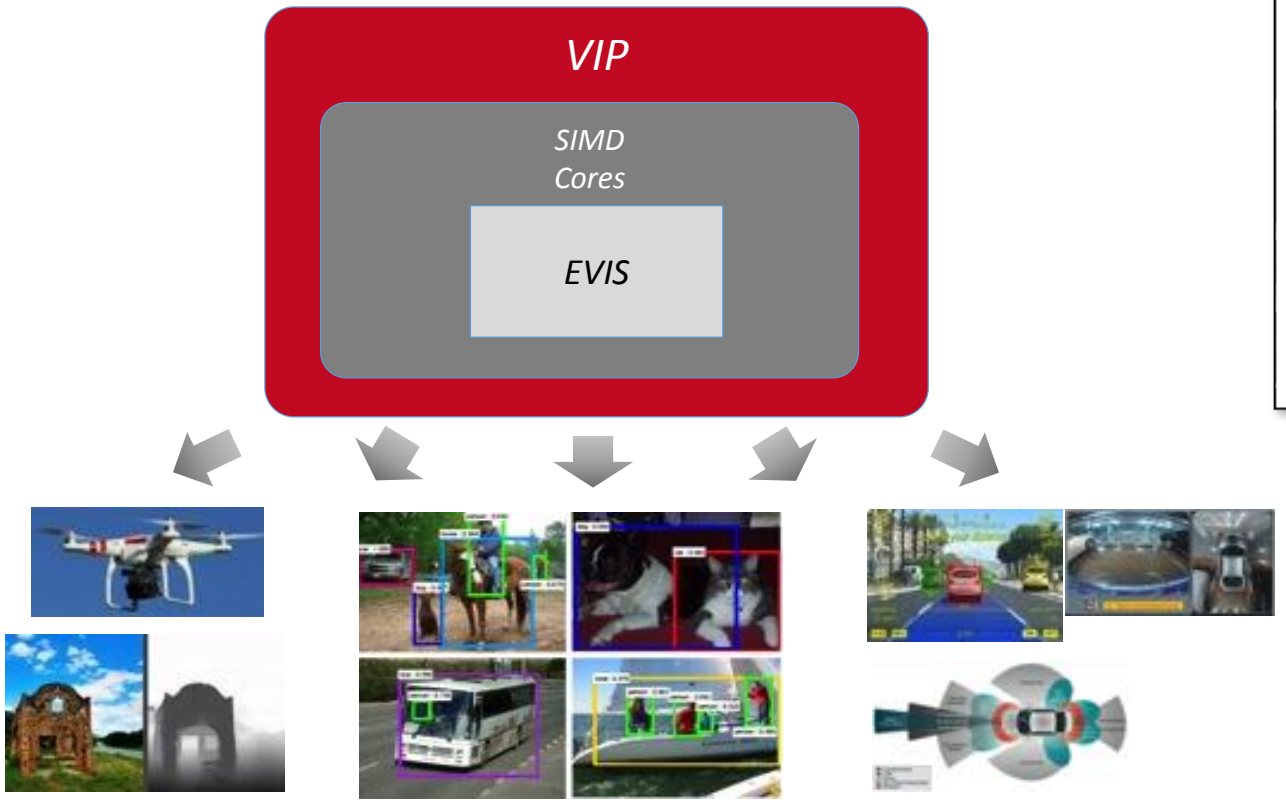
- ▶ Same program runs on single or multiple cores

VeriSilicon

Computer Vision Acceleration

▲Special HW acceleration for computer vision and machine learning

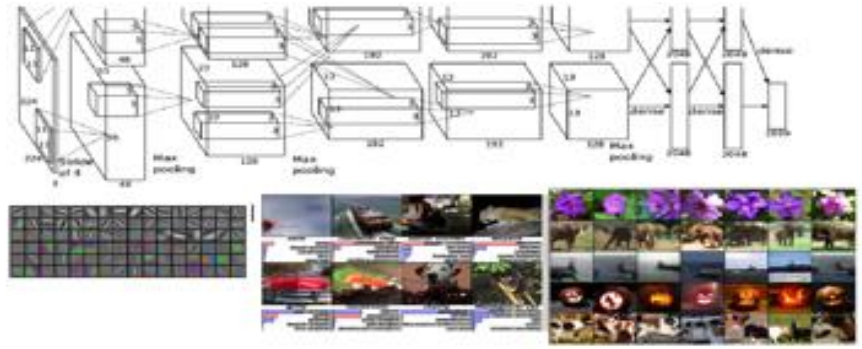
- ▶Easy programming with EVIS built-in
- ▶Significant improvement on PPA (Power, Performance, Area)



OpenCL code for CNN

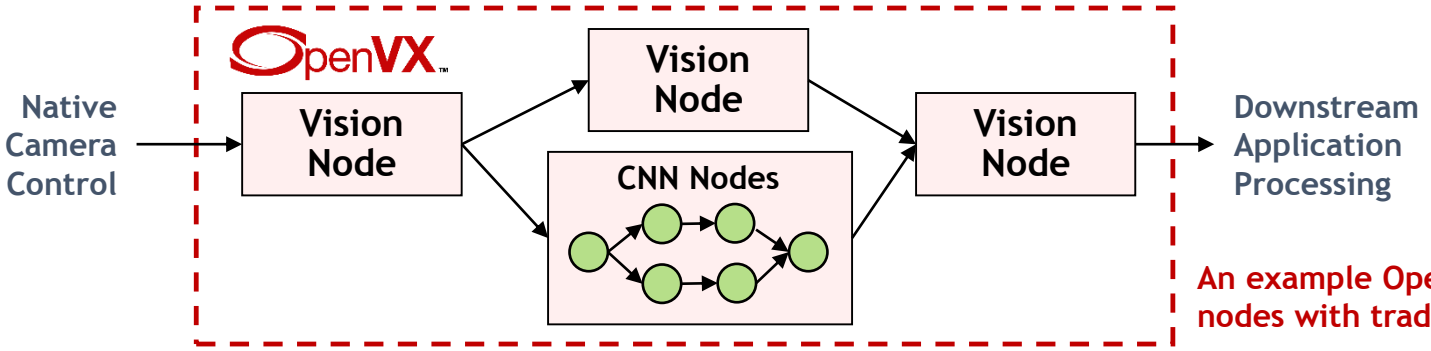
```
__kernel void cnn_layer(
  __read_only void* network_descriptor,
  __read_only image2d_array_t input,
  __write_only image2d_array_t output){
  VXC_NeuralNet(network_descriptor, input, output);
}
…
void app_cnn(…){
  …
  cnn_kernel = clCreateKernel(…"cnn_layer",…);
  … // set up cnn_layer 1 and run
  clEnqueueNDRangeKernel(cmd_queue, cnn_kernel, …, event1);
  … // set up cnn_layer n and run
  clEnqueueNDRangeKernel(cmd_queue, cnn_kernel, …, eventn);

  clWaitForEvents(1, &eventn);
// CNN done, do something else
  …
}
```



OpenVX Neural Net Extension

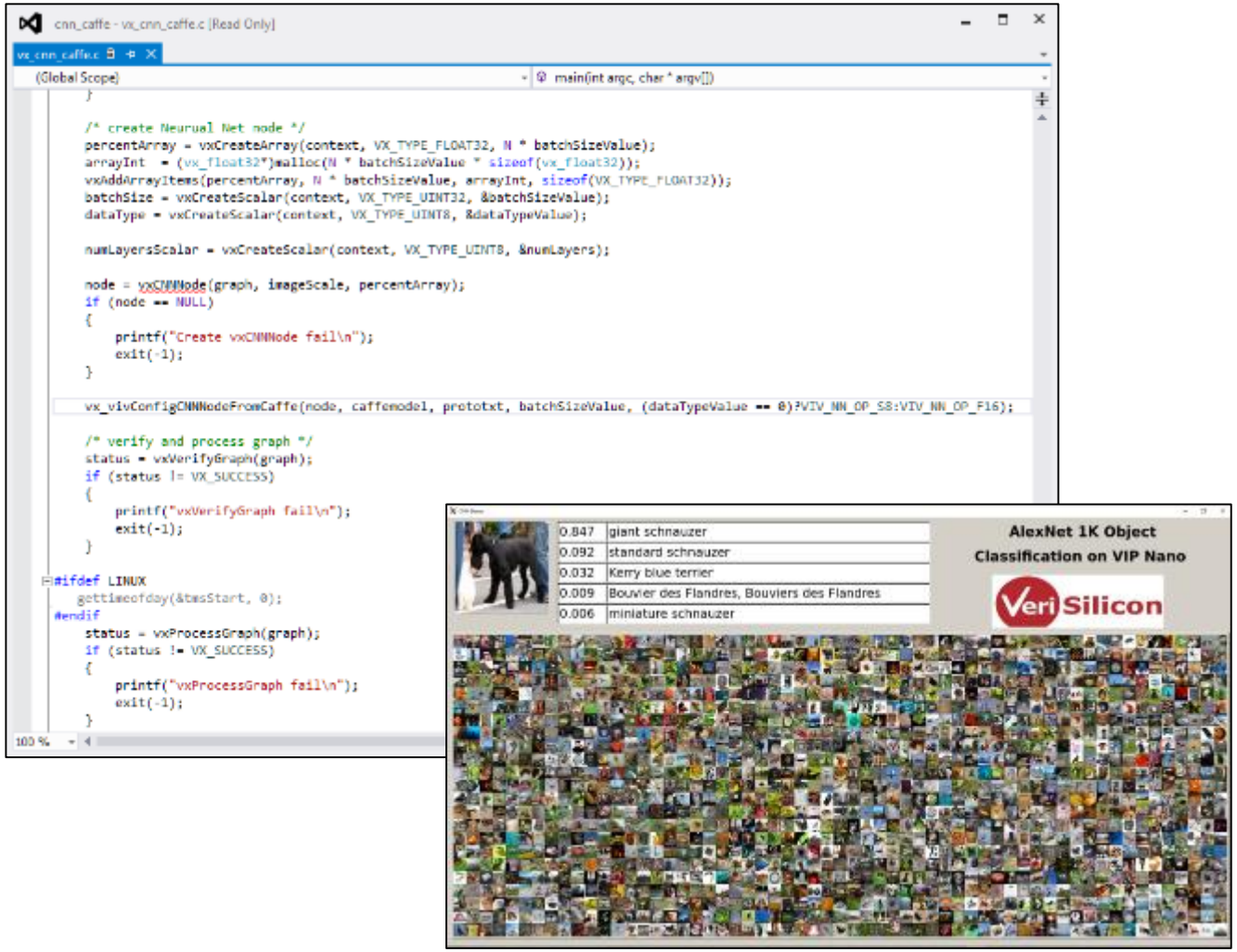
- **Convolutional Neural Network topologies can be represented as OpenVX graphs**
  - Layers are represented as OpenVX nodes
  - Layers connected by multi-dimensional tensors objects
  - Layers include convolution, normalization, pooling, fully-connected, soft-max
  - Also activation layers – with nine different activation functions
  - CNN nodes can be mixed with traditional vision nodes
- **Import/Export Extension**
  - Efficient handling of network weights/biases or complete networks
- **The specification is provisional**
  - Seeking feedback from the deep learning community



An example OpenVX graph mixing CNN nodes with traditional vision nodes

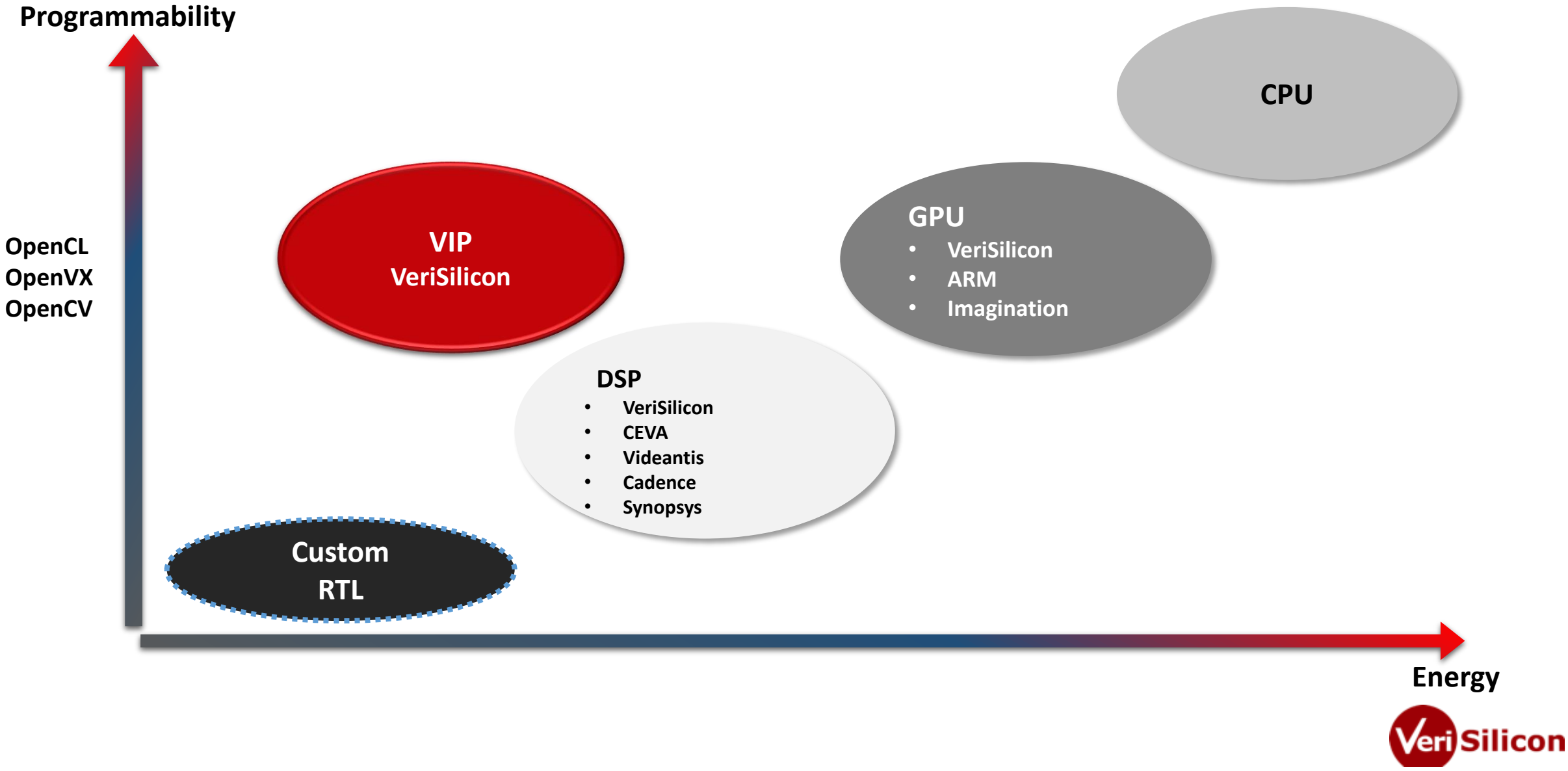
Running Caffe CNN in OpenVX

- VeriSilicon VIP8000 offers a unified SW framework to run OpenVX and CNN
- Simply create a CNN node in a OpenVX graph, and configure the node with the caffemodel and prototxt files from Caffe



VeriSilicon

Processor Architectures



What is ahead of us

- **Deep neural net has come to embedded space for visual recognition**
- **More sophisticated recognition demands are just around the corner**
  - Vision + natural language processing
  - Action recognition
  - Multi-sensor, high-dimensional data input
- **A scalable, extendible processor architecture with unified programming model brings “natural user interface” to embedded systems**

VeriSilicon

www.verisilicon.com

VeriSilicon

Empowering Product Creators to Harness Embedded Vision

The Embedded Vision Alliance (www.Embedded-Vision.com) is a partnership of 50+ leading embedded vision technology and services suppliers

Mission: Inspire and empower product creators to incorporate visual intelligence into their products

The Alliance provides low-cost, high-quality technical educational resources for product developers

Register for updates at www.Embedded-Vision.com

The Alliance enables vision technology providers to grow their businesses through leads, ecosystem partnerships, and insights

For membership, email us: membership@Embedded-Vision.com

Join us at the Embedded Vision Summit
May 1-3, 2017—Santa Clara, California

The only industry event focused on enabling product creators to create "machines that see"

- *"Awesome! I was very inspired!"*
- *"Fantastic. Learned a lot and met great people."*
- *"Wonderful speakers and informative exhibits!"*

Embedded Vision Summit 2017 highlights:

- **Inspiring keynotes** by leading innovators
- High-quality, practical **technical, business and product talks**
- Exciting **demos** of the latest apps and technologies

Visit www.EmbeddedVisionSummit.com to sign up for updates