赛灵思加速人工智能应用

陈志勇
高级技术市场经理
安富利电子科技
2019年8月18日，厦门
About Avnet

- Founded in 1921
- Headquartered in Phoenix, Arizona
- AVT listed on the NYSE since 1960
- AVT listed on NASDAQ since 2018
- #128 on the FORTUNE 500 (U.S.) in 2018

Awards
- Top 10 for 2018 Gartner Supply Chain Top 25 - High-Tech industry category
- World’s Most Ethical Company by Ethisphere Institute from 2014-2018
This is the new Avnet

We design, make, supply and deliver technology solutions.
We work with customers of every size, in every corner of the world.

15,000+ Employees worldwide
2,500+ Engineers around the world
1M+ Engineering community members
2.1M Customers in 140+ countries
1,400+ Technology suppliers WW
122B Units shipped annually
125 Locations across the world

We guide today’s ideas into tomorrow’s technology.
The Avnet ecosystem
A Key Enabler for IoT Development

We simplify complexities by connecting reliable partners to solve your challenge.
Xilinx Machine Learning
Edge to Cloud
Deep Learning

Why now?
- New processors making DNN training feasible (Ops/$)
- Huge amounts of training data

Faster/Better Results
- Caused explosion in AI Research
- More Applications
- More Startups
- More innovation
- More Acquisitions . . .
Deep Learning: Training vs. Inference

**Training**: Process for machine to “learn” and optimize a model from data

**Inference**: Using trained model to predict/estimate outcomes from new observations
## Delivering Adaptable ML Compute Acceleration

<table>
<thead>
<tr>
<th></th>
<th>CPU (Sequential)</th>
<th>GPU (Parallel)</th>
<th>FPGA / SoC / ACAP</th>
<th>Custom ASIC</th>
</tr>
</thead>
<tbody>
<tr>
<td>SW Programmable</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
</tr>
<tr>
<td>HW Adaptable</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Workload Flexibility</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td></td>
</tr>
<tr>
<td>Throughput vs. Latency</td>
<td>▬</td>
<td></td>
<td>✓</td>
<td></td>
</tr>
<tr>
<td>Device / Power Efficiency</td>
<td>▬</td>
<td></td>
<td>✓</td>
<td></td>
</tr>
</tbody>
</table>

[Diagram showing comparisons between CPU, GPU, FPGA/SoC/ACAP, and Custom ASIC]
### Accelerate the Whole Application Around AI Inference

<table>
<thead>
<tr>
<th>Application</th>
<th>Typical AI Networks</th>
<th>Non-AI Workloads</th>
</tr>
</thead>
<tbody>
<tr>
<td>Smart Retail / Surveillance</td>
<td>CNN classification, detection, segmentation</td>
<td>multi-channel video decode</td>
</tr>
<tr>
<td>Speech Recognition</td>
<td>LSTM and BART</td>
<td>Speech clean-up, database lookup</td>
</tr>
<tr>
<td>Recommendation Engines</td>
<td>MLP</td>
<td>Keyword pre-post processing</td>
</tr>
<tr>
<td>Anomaly Detection</td>
<td>Random Forest</td>
<td>Smart NIC functions</td>
</tr>
<tr>
<td>Financial Tech</td>
<td>LSTM</td>
<td>Monte Carlo and other risk analysis models</td>
</tr>
</tbody>
</table>
Deep Learning Models

Multi-Layer Perceptron
- Classification
- Universal Function Approximator
- Autoencoder

Convolutional Neural Network
- Feature Extraction
- Object Detection
- Image Segmentation

Recurrent Neural Network
- Sequence and Temporal Data
- Speech to Text
- Language Translation

Classification
- “Dog”

Object Detection

Segmentation
Latest Inference Algorithmic Research

Optimize Compute with Reduced Precision CNNs & BNNs
- 8 bit solution loses no significant accuracy
- BNNs are improving rapidly
- Custom Floating point provide significantly more compute density

Model Pruning for higher effective performance: LSTM
- TIMIT 3-hour dataset – 20x
- LibreSpeech 100-hour dataset – 10x
- CustomerS1000-hour dataset – 20x
- CustomerS 3000-hour dataset – 5x

Not only weights, but also sparsity pattern encodes information
Xilinx Features for Implementing Efficient Inference Engines

Flexible Architecture for Any Precision

Flexible On-chip Memory for low latency
Xilinx ML Solution from Edge/Embedded to Cloud/DC

Deep Learning Applications

Featuring the Most Powerful FPGA in the Cloud

Cloud

Virtex Ultrascale+ VU9P

On Premises

Zynq Ultrascale+ MPSoC

Edge
### Xilinx ML Solution from Edge/Embedded to Cloud/DC

<table>
<thead>
<tr>
<th>Models</th>
<th>Edge/Embedded</th>
<th>Cloud/DC</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>20+</td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Software Stack</th>
<th>Edge/Embedded</th>
<th>Cloud/DC</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>DNNDK Pruning</td>
<td></td>
</tr>
<tr>
<td></td>
<td>DNNDK Quantizer</td>
<td>xFDNN Quantizer</td>
</tr>
<tr>
<td></td>
<td>DNNDK Compiler</td>
<td>xFDNN Compiler</td>
</tr>
<tr>
<td></td>
<td>DNNDK Runtime</td>
<td>xFDNN Runtime</td>
</tr>
<tr>
<td></td>
<td>SDSoc</td>
<td></td>
</tr>
<tr>
<td></td>
<td>SDAccel</td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>FPGA IP</th>
<th>Edge/Embedded</th>
<th>Cloud/DC</th>
</tr>
</thead>
<tbody>
<tr>
<td>DPU</td>
<td></td>
<td></td>
</tr>
<tr>
<td>LSTM</td>
<td></td>
<td>xDNN</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Platforms</th>
<th>Edge/Embedded</th>
<th>Cloud/DC</th>
</tr>
</thead>
<tbody>
<tr>
<td>Z7020 Board</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Z7020 SOM</td>
<td></td>
<td></td>
</tr>
<tr>
<td>ZCU102</td>
<td></td>
<td></td>
</tr>
<tr>
<td>ZCU104</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Ultra96</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>Xilinx U50, U200, U250, U280</td>
</tr>
</tbody>
</table>
Xilinx ML for Edge/Embedded
Xilinx Value Proposition for Edge/Embedded ML

Xilinx offers the optimal tradeoff among latency, power, cost, flexibility, scalability & time-to-market for Edge/Embedded ML.
Deep compression

Makes algorithm smaller and lighter

Highlight

Compression efficiency

Deep Compression Tool can achieve significant compression on CNN and RNN

Accuracy

Algorithm can be compressed 7 times without losing accuracy under SSD object detection framework
## DPU Utilization

### More DSP

<table>
<thead>
<tr>
<th>Arch</th>
<th>LUTs</th>
<th>Registers</th>
<th>BRAM*</th>
<th>DSP</th>
</tr>
</thead>
<tbody>
<tr>
<td>B512</td>
<td>17951</td>
<td>28280</td>
<td>69.5</td>
<td>97</td>
</tr>
<tr>
<td>B800</td>
<td>20617</td>
<td>35065</td>
<td>87</td>
<td>141</td>
</tr>
<tr>
<td>B1024</td>
<td>22327</td>
<td>39000</td>
<td>101.5</td>
<td>193</td>
</tr>
<tr>
<td>B1152</td>
<td>22796</td>
<td>40276</td>
<td>117.5</td>
<td>193</td>
</tr>
<tr>
<td>B1600</td>
<td>26270</td>
<td>50005</td>
<td>123</td>
<td>281</td>
</tr>
<tr>
<td>B2304</td>
<td>29592</td>
<td>57549</td>
<td>161.5</td>
<td>385</td>
</tr>
<tr>
<td>B3136</td>
<td>33266</td>
<td>69110</td>
<td>203.5</td>
<td>505</td>
</tr>
<tr>
<td>B4096</td>
<td>37495</td>
<td>84157</td>
<td>249.5</td>
<td>641</td>
</tr>
</tbody>
</table>

### More LUT

<table>
<thead>
<tr>
<th>Arch</th>
<th>LUTs</th>
<th>Registers</th>
<th>BRAM*</th>
<th>DSP</th>
</tr>
</thead>
<tbody>
<tr>
<td>B512</td>
<td>20759</td>
<td>33572</td>
<td>69.5</td>
<td>66</td>
</tr>
<tr>
<td>B1024</td>
<td>29155</td>
<td>49823</td>
<td>101.5</td>
<td>130</td>
</tr>
<tr>
<td>B1152</td>
<td>30043</td>
<td>49588</td>
<td>117.5</td>
<td>146</td>
</tr>
<tr>
<td>B1600</td>
<td>33130</td>
<td>60739</td>
<td>123</td>
<td>202</td>
</tr>
<tr>
<td>B2304</td>
<td>37055</td>
<td>72850</td>
<td>161.5</td>
<td>290</td>
</tr>
<tr>
<td>B3136</td>
<td>41714</td>
<td>86132</td>
<td>203.5</td>
<td>394</td>
</tr>
<tr>
<td>B4096</td>
<td>44583</td>
<td>99791</td>
<td>249.5</td>
<td>514</td>
</tr>
</tbody>
</table>

DPU provides flexible option depending on customer’s resources and continues to improve

*URAM also can be used by DPU if device supports, every URAM is roughly used as 3.7 BRAM
Perf Improvement with DPU

Performance Comparison (FPS)

- Current B4096*2 wo Prune

VGG-SSD: 12 FPS (end to end)
VGG16: 73 FPS (end to end)
VGG16 CONV part: 92 FPS (w/o FC layer)
ResNet50 CONV part: 118 FPS (w/o FC layer)
ResNet50: 179 FPS (end to end)
GoogLeNet: 445 FPS (end to end)

*The FPS of VGG-SSD of end to end performance
*The FPS of VGG16/ResNet50/GoogLeNet is of CONV part (w/o FC layer)
DPU Scalability

* With heterogenous DPUs

* B256/288/512/3136 work in progress
DNNDK Dev Flow

Five Steps with DNNDK

01 Model Compression
02 Model Compilation
03 Programming
04 Hybrid Compilation
05 Execution
第1步：使用Decent 进行模型压缩

decent – Deep compression Tool
decent_q – Quantization Tool
decent_p – Pruning Tool

- Consists of two separate tools
  - Quantization Tool
  - Pruning Tool

- Effects
  - Compress model size
    5x – 100x
  - Compress running time
    1.5x – 10x

- Platform
  - Caffe, Darknet
  - TensorFlow
    - Quantization Tool Beta version
    - Pruning Tool Internal version
第2步：使用 DNNC 进行模型编译

Programmable tensor-level DPU instruction set
- Compatible for Caffe/TensorFlow frameworks
- Flexible & scalable for various CNN layers
# Xilinx AI Developer Resource


---

### Edge AI Tools

<table>
<thead>
<tr>
<th>Product</th>
<th>Documentation</th>
<th>Tool Download</th>
<th>File Size</th>
<th>MD5 Checksum</th>
</tr>
</thead>
<tbody>
<tr>
<td>DNNDK</td>
<td>DNNDK User Guide (UG1527)</td>
<td>xilinx_dnnlk_v0.8.10043.tar.gz</td>
<td>33.2 GB</td>
<td>561153e743c840f49bb34178800a934e54</td>
</tr>
<tr>
<td>DNNDK</td>
<td>Xilinx AI SDK User Guide (UG1504)</td>
<td>xilinx_sdk_v0.4.1.tar.gz</td>
<td>22.2 GB</td>
<td>981584e9767a892e94e72f743421537</td>
</tr>
<tr>
<td>DNNDK</td>
<td>Xilinx AI SDK Programming Guide (UG1235)</td>
<td>xilinx_sdk_v0.9.1.tar.gz</td>
<td>29.3 GB</td>
<td>85e584b53c2d0b7634e72f743421537</td>
</tr>
<tr>
<td>DNNDK for SDSoc</td>
<td>DNNDK User Guide for SDSoc (UG1531)</td>
<td>xilinx_dnnlk_v2.0.8_sodac_190214.tar.gz</td>
<td>657 MB</td>
<td>71165af5552487e4b4b9570773c49d1</td>
</tr>
</tbody>
</table>

### Edge AI Evaluation Boards

<table>
<thead>
<tr>
<th>Product</th>
<th>Documentation</th>
<th>Image Download</th>
<th>DNNDK Version</th>
<th>File Size</th>
<th>MD5 Checksum</th>
</tr>
</thead>
<tbody>
<tr>
<td>ZCU102 Kit</td>
<td>ZCU102 User Guide (UG1182)</td>
<td>xilinx_zu102_pro_v2019.3-desktop_buster-39750824_mg.zip</td>
<td>v3.0</td>
<td>689 MB</td>
<td>d4f11e3ad22e8e81f44f0a3d96e1e2553</td>
</tr>
<tr>
<td>ZCU104 Kit</td>
<td>ZCU104 User Guide (UG1267)</td>
<td>xilinx_zu104_pro_v2019.4-desktop_buster-39750824_mg.zip</td>
<td>v3.0</td>
<td>689 MB</td>
<td>d4f11e3ad22e8e81f44f0a3d96e1e2553</td>
</tr>
<tr>
<td>Artix Ultra 9S</td>
<td>Ultra 9S User Guide</td>
<td>xilinx_artix9锼p_v2019.2-desktop_buster-39750824_mg.zip</td>
<td>v2.0</td>
<td>689 MB</td>
<td>d4f11e3ad22e8e81f44f0a3d96e1e2553</td>
</tr>
</tbody>
</table>

### Edge AI Targeted Reference Designs (TRD)

<table>
<thead>
<tr>
<th>Product</th>
<th>Documentation</th>
<th>Image Download</th>
<th>File Size</th>
<th>MD5 Checksum</th>
</tr>
</thead>
<tbody>
<tr>
<td>DPUI TRD</td>
<td>DPUI Product Guide (P033)</td>
<td>zcu102_diagram_v2019.2-110002.zip</td>
<td>465 MB</td>
<td>21042953a513214e8856130276169077</td>
</tr>
</tbody>
</table>
Xilinx Edge AI Tutorial


<table>
<thead>
<tr>
<th>Tutorial</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>CIFAR10 Caffe Tutorial (UG1335)</td>
<td>Train, quantize and prune custom CNNs with the CIFAR10 dataset using Caffe and the Xilinx DNNDK tools.</td>
</tr>
<tr>
<td>Cats vs Dogs Caffe Tutorial (UG1336)</td>
<td>Train, quantize and prune a modified AlexNet CNN with the Kaggle Cats vs Dogs dataset using Caffe and the Xilinx DNNDK tools.</td>
</tr>
<tr>
<td>ML SSD PASCAL Caffe Tutorial (UG1340)</td>
<td>Train, quantize and compile SSD using PASCAL VOC 2007/2012 datasets with the Caffe framework and DNNDK tools, then deploy on a Xilinx ZCU102 target board.</td>
</tr>
<tr>
<td>DRPU Integration Lab (UG1350)</td>
<td>Build a custom system that utilizes the Xilinx Deep Learning Processor (DRPU) IP to accelerate machine learning algorithms.</td>
</tr>
<tr>
<td>Yolov3 Tutorial with Darknet to Caffe Converter and Xilinx DNNDK (UG1334)</td>
<td>Use the Yolov3 example, which converts the Darknet model to Caffe model, and uses the DNNDK tool chain for quantization, compilation, and deployment on the FPGA.</td>
</tr>
<tr>
<td>MNIST Classification with TensorFlow (UG1337)</td>
<td>Learn the DNNDK v0.9.0 TensorFlow design process for creating a compiled .so file that is ready for deployment on the Xilinx® DPU accelerator from a simple network model built using Python. This tutorial uses the MNIST test dataset.</td>
</tr>
<tr>
<td>CIFAR10 Classification with TensorFlow (UG1338)</td>
<td>Learn the DNNDK v0.9.0 TensorFlow design process for creating a compiled .so file that is ready for deployment on the Xilinx® DPU accelerator from a simple network model built using Python. This tutorial uses the CIFAR-10 test dataset.</td>
</tr>
</tbody>
</table>
# Xilinx AI Model Zoo

## Model Zoo Collection

The Xilinx AI Model Zoo is a collection of pre-trained machine learning models designed for various applications. These models are optimized for deployment on Xilinx FPGAs, offering high performance and flexibility. The models are available for download directly from the Xilinx website, with links provided in the table below.

## Table of Models

<table>
<thead>
<tr>
<th>Application</th>
<th>Model</th>
<th>Model Download Link</th>
<th>File Size</th>
<th>MD5 Checksum</th>
<th>ModelName</th>
<th>Input Size</th>
<th>FPS per Image</th>
<th>Parameters</th>
<th>Framework</th>
<th>Training Set</th>
</tr>
</thead>
<tbody>
<tr>
<td>Image</td>
<td>Classification</td>
<td>pretrained-50</td>
<td><a href="https://www.xilinx.com/bin/public/openDownload?filename=all_models_20190528.zip">https://www.xilinx.com/bin/public/openDownload?filename=all_models_20190528.zip</a></td>
<td>203.9 MB</td>
<td>9e5c32e563b13c05f3d1d36741c66fda</td>
<td>pretrained-50</td>
<td>224x224</td>
<td>7.76</td>
<td>25.6 MB</td>
<td>caffe_caffe</td>
</tr>
<tr>
<td>Image</td>
<td>Classification</td>
<td>inception_v1</td>
<td><a href="https://www.xilinx.com/bin/public/openDownload?filename=all_models_20190528.zip">https://www.xilinx.com/bin/public/openDownload?filename=all_models_20190528.zip</a></td>
<td>35.8 MB</td>
<td>5a8f65dc7e1d04d7f53ac06e8</td>
<td>inception_v1</td>
<td>224x224</td>
<td>3.46</td>
<td>3.0 MB</td>
<td>caffe_caffe</td>
</tr>
<tr>
<td>Image</td>
<td>Classification</td>
<td>inception_v2</td>
<td><a href="https://www.xilinx.com/bin/public/openDownload?filename=all_models_20190528.zip">https://www.xilinx.com/bin/public/openDownload?filename=all_models_20190528.zip</a></td>
<td>528.7 MB</td>
<td>3c4b7b6c79f50f7c7115e3c9e8b9f8e</td>
<td>inception_v2</td>
<td>224x224</td>
<td>8.03</td>
<td>30.0 MB</td>
<td>caffe_caffe</td>
</tr>
<tr>
<td>Image</td>
<td>Classification</td>
<td>inception_v3</td>
<td><a href="https://www.xilinx.com/bin/public/openDownload?filename=all_models_20190528.zip">https://www.xilinx.com/bin/public/openDownload?filename=all_models_20190528.zip</a></td>
<td>150.1 MB</td>
<td>45d4b68936d234be706f15d7</td>
<td>inception_v3</td>
<td>200x200</td>
<td>11.16</td>
<td>23.0 MB</td>
<td>caffe_caffe</td>
</tr>
<tr>
<td>Image</td>
<td>Classification</td>
<td>mobilenet_v2</td>
<td><a href="https://www.xilinx.com/bin/public/openDownload?filename=all_models_20190528.zip">https://www.xilinx.com/bin/public/openDownload?filename=all_models_20190528.zip</a></td>
<td>31.6 MB</td>
<td>3a1c8287c272af7fd6d6184e4a8</td>
<td>mobilenet_v2</td>
<td>224x224</td>
<td>92.0</td>
<td>8.4 MB</td>
<td>caffe_caffe</td>
</tr>
<tr>
<td>Image</td>
<td>Classification</td>
<td>tf_resnet50</td>
<td><a href="https://www.xilinx.com/bin/public/openDownload?filename=all_models_20190528.zip">https://www.xilinx.com/bin/public/openDownload?filename=all_models_20190528.zip</a></td>
<td>244.4 MB</td>
<td>09934f750050b082be88b689</td>
<td>tf_resnet50</td>
<td>224x224</td>
<td>0.93</td>
<td>25.0 MB</td>
<td>tensorflow</td>
</tr>
<tr>
<td>Image</td>
<td>Classification</td>
<td>tf_inception_v1</td>
<td><a href="https://www.xilinx.com/bin/public/openDownload?filename=all_models_20190528.zip">https://www.xilinx.com/bin/public/openDownload?filename=all_models_20190528.zip</a></td>
<td>3.4 MB</td>
<td>da3064c3a4f3a4b2825a4e15e2a</td>
<td>inception_v1</td>
<td>224x224</td>
<td>5.85</td>
<td>6.5 MB</td>
<td>tensorflow</td>
</tr>
<tr>
<td>Image</td>
<td>Classification</td>
<td>tf_mobilnet_v2</td>
<td><a href="https://www.xilinx.com/bin/public/openDownload?filename=all_models_20190528.zip">https://www.xilinx.com/bin/public/openDownload?filename=all_models_20190528.zip</a></td>
<td>1.6 MB</td>
<td>ac073255b5f7527c5f14a61b3b</td>
<td>mobilenet_v2</td>
<td>224x224</td>
<td>1.17</td>
<td>6.3 MB</td>
<td>tensorflow</td>
</tr>
<tr>
<td>Aviv Netface Detection</td>
<td>yolov3-tiny</td>
<td><a href="https://www.xilinx.com/bin/public/openDownload?filename=all_models_20190528.zip">https://www.xilinx.com/bin/public/openDownload?filename=all_models_20190528.zip</a></td>
<td>2.8 MB</td>
<td>20f4d881f9c5b56fe406f6a66</td>
<td>yolov3-tiny</td>
<td>416x416</td>
<td>9.35</td>
<td>0.72 MB</td>
<td>caffe_caffe</td>
<td><a href="https://github.com/Aviv308">Aviv308</a></td>
</tr>
<tr>
<td>Aviv Pedestrian Detection</td>
<td>yolov3-tiny</td>
<td><a href="https://www.xilinx.com/bin/public/openDownload?filename=all_models_20190528.zip">https://www.xilinx.com/bin/public/openDownload?filename=all_models_20190528.zip</a></td>
<td>2.6 MB</td>
<td>8fb302d772376ba497d572be21b6c</td>
<td>yolov3-tiny</td>
<td>416x416</td>
<td>19.85</td>
<td>0.47 MB</td>
<td>tensorflow</td>
<td><a href="https://github.com/Aviv308">Aviv308</a></td>
</tr>
</tbody>
</table>

---

**Note:** The FPS (frames per second) values represent the performance of the models on average images. Actual performance may vary depending on the hardware and image characteristics.
Xilinx AI Demo Zoo

- AI SDK Demo
- 8-ch RTSP stream VCU+DPU demo
- Multi-model multi-camera Demo
- Demo Guide
Xilinx AI Forums


Announcements

Welcome to the Deephi DNNDK Community Forum. This community should serve as a resource to ask and learn about using Deephi DNNDK on all supported platforms, new feature announcements and troubleshooting AI applications.

Most Recent Threads
Before you post, please read our Community Forums Guidelines or to get started see our Community Forum Help.

Discussions

XILINX_SDK_AI package versions varies for differ... by yashaswinil.shankar on 08-09-2019 04:55 AM • Latest post on
Out-of-box Supported Boards

- ZCU102
- ZCU104
- Avnet Ultra96
Video Surveillance ML Solutions

Intelligent IP Camera Solution

Face recognition camera with Zynq7020

Video Analytics Acceleration Solution

12-channel 1080P Video Analytics with ZU9EG
Video Surveillance ML Ref Design

- Detection & Tracking
- Person Attributes
  - Gender: Female
  - Upper color: Yellow
  - Lower color: White
  - Hat: No
  - Backpack: No
  - Handbag: No
  - Other bag: No
- Gender: Male
  - Upper color: Black
  - Lower color: Black
  - Hat: No
  - Backpack: No
  - Handbag: No
  - Other bag: No

- Detection & Tracking
- Person Attributes
- Car Attributes
- License Recognition
  - Color: White
  - Type: BUICK
  - Number: CLC689
  - Color: Blue
  - Number: 渝C LC689
ADAS/AD ML Reference Design

2D/3D Object Detection

Lane Detection

Pedestrian Detection

Segmentation

Pose Estimation

Segmentation + Detection
8CH Detection Demo

- Xilinx device
  - ZU9EG

- Network
  - SSD compact version

- Input image size to DPU
  - 480 * 360

- Operations per frame
  - 4.9G

- Performance
  - 30fps per channel

*Removed Video
4-ch Segmentation + Detection Demo

> Xilinx device
  » ZU9EG

> Network
  » FPN compact version
  » SSD compact version

> Input image size to DPU
  » FPN – 512 * 256
  » SSD – 480 * 360

> Operations per frame
  » FPN – 9G
  » SSD – 4.9G

> Performance
  » 15fps per channel

*Removed Video*
## Supported DNN (Deep Neural Network) by Applications

<table>
<thead>
<tr>
<th>Application</th>
<th>Function</th>
<th>Algorithm</th>
<th>Developed</th>
<th>Pruned</th>
<th>Deployed</th>
</tr>
</thead>
<tbody>
<tr>
<td>Face</td>
<td>Face detection</td>
<td>SSD, Densebox</td>
<td>✔</td>
<td>✔</td>
<td>✔</td>
</tr>
<tr>
<td></td>
<td>Landmark Localization</td>
<td>Coordinates Regression</td>
<td>✔</td>
<td>N / A</td>
<td>✔</td>
</tr>
<tr>
<td></td>
<td>Face recognition</td>
<td>ResNet + Triplet / A-softmax Loss</td>
<td>✔</td>
<td>✔</td>
<td>✔</td>
</tr>
<tr>
<td></td>
<td>Face attributes recognition</td>
<td>Classification and regression</td>
<td>✔</td>
<td>N / A</td>
<td>✔</td>
</tr>
<tr>
<td>Pedestrian</td>
<td>Pedestrian Detection</td>
<td>SSD</td>
<td>✔</td>
<td>✔</td>
<td>✔</td>
</tr>
<tr>
<td></td>
<td>Pose Estimation</td>
<td>Coordinates Regression</td>
<td>✔</td>
<td>✔</td>
<td>✔</td>
</tr>
<tr>
<td></td>
<td>Person Re-identification</td>
<td>ResNet + Loss Fusion</td>
<td>✔</td>
<td>✔</td>
<td>✔</td>
</tr>
<tr>
<td>Video Analytics</td>
<td>Object detection</td>
<td>SSD, RefineDet</td>
<td>✔</td>
<td>✔</td>
<td>✔</td>
</tr>
<tr>
<td></td>
<td>Pedestrian Attributes Recognition</td>
<td>GoogleNet</td>
<td>✔</td>
<td>✔</td>
<td>✔</td>
</tr>
<tr>
<td></td>
<td>Car Attributes Recognition</td>
<td>GoogleNet</td>
<td>✔</td>
<td>✔</td>
<td>✔</td>
</tr>
<tr>
<td></td>
<td>Car Logo Detection</td>
<td>DenseBox</td>
<td>✔</td>
<td>✔</td>
<td>✔</td>
</tr>
<tr>
<td></td>
<td>Car Logo Recognition</td>
<td>GoogleNet + Loss Fusion</td>
<td>✔</td>
<td>✔</td>
<td>✔</td>
</tr>
<tr>
<td></td>
<td>License Plate Detection</td>
<td>Modified DenseBox</td>
<td>✔</td>
<td>✔</td>
<td>✔</td>
</tr>
<tr>
<td></td>
<td>License Plate Recognition</td>
<td>GoogleNet + Multi-task Learning</td>
<td>✔</td>
<td>✔</td>
<td>✔</td>
</tr>
<tr>
<td>ADAS/AD</td>
<td>Object Detection</td>
<td>SSD, YOLOv2, YOLOv3</td>
<td>✔</td>
<td>✔</td>
<td>✔</td>
</tr>
<tr>
<td></td>
<td>3D Car Detection</td>
<td>F-PointNet, AVOD-FPN</td>
<td>✔</td>
<td>✔</td>
<td>✔</td>
</tr>
<tr>
<td></td>
<td>Lane Detection</td>
<td>VPGNet</td>
<td>✔</td>
<td>✔</td>
<td>✔</td>
</tr>
<tr>
<td></td>
<td>Traffic Sign Detection</td>
<td>Modified SSD</td>
<td>✔</td>
<td>✔</td>
<td>✔</td>
</tr>
<tr>
<td></td>
<td>Semantic Segmentation</td>
<td>FPN</td>
<td>✔</td>
<td>✔</td>
<td>✔</td>
</tr>
<tr>
<td></td>
<td>Drivable Space Detection</td>
<td>MobilenetV2-FPN</td>
<td>✔</td>
<td>✔</td>
<td>✔</td>
</tr>
<tr>
<td></td>
<td>Multi-task (Detection+Segmentation)</td>
<td>Deephi</td>
<td>✔</td>
<td>✔</td>
<td>✔</td>
</tr>
</tbody>
</table>
Supported Operators

- Conv
  - Dilation
- Pooling
  - Max
  - Average
- ReLU / Leaky Relu / ReLu6
- Full Connected (FC)
- Batch Normalization
- Concat
- Elementwise

- Deconv
- Depthwise conv
- Mean scale
- Upsampling
- Split
- Reorg
- Resize (Optional)
- Softmax (Optional)
- Sigmoid (Optional)
Basic and Professional Editions

> **Public Access Timeframe**
  >> Basic: Now
  >> Basic with Tensorflow: Apr 2019
  >> Professional: May 2019

> **Basic in AWS Cloud – Apr 2019**

> **Add-on design service – SoW**

Everything you need to do it yourself

Free

**Basic**
- Compiler
- Quantizer
- Pruned Models
- Unlimited Deployment

**Professional**
- 3-day On-site Training
- Pruning Tools
- Compiler
- Quantizer
- Pruned Models
- Unlimited Deployment

Access Pruning Technology &
3-day on-site training by a top-notch ML expert &
30-day evaluation with encrypted pruning output

For Professional Edition pricing, please inquiry Xilinx AI marketing
Availability

> DNNDK & DPU
  > DNNDK basic edition - Download from Xilinx.com
  > Pruning tool, separate upon request
  > DPU available for evaluation & system integration upon request

> Demos & Ref Designs
  > General: Resnet50, Googlenet, VGG16, SSD, Yolo v2/v3, Tiny Yolo v2/v3, Mobilenet v1/v2 etc..
  > Video surveillance: face detection & traffic structure
  > ADAS/AD: multi-channel detection & segmentation
  > DPU TRD (Work in progress)

> Documentation
  > DNNDK user guide – UG1327
  > DNNDK for SDSoC user guide – UG1331
  > Edge AI tutorials - https://github.com/Xilinx/Edge-AI-Platform-Tutorials
  > DPU product guide & tutorial (Work in progress)

> Request or Inquiry
  > Please contact Andy Luo, andy.luo@xilinx.com
Customizable Performance

- Highest perf/watt
- Lower precision optimized
- Optimizations for Throughput and Latency

Flexibility

- Well suited for evolving Deep Learning Field
- Support for all types – CNN, DNN, LSTM
- Integrate with custom application in FPGA
- Full Software Stack for Applications

Scalable

- Configurable from 1 to many xDNN Engines
- Pool Xilinx cards for higher performance
- Deployed today on edge or cloud
ML Suite Features

Based on xDNN v2

Supported Frameworks:
• Caffe
• MxNet
• TensorFlow

Examples
• DeepDetect REST Tutorial
• DeepDetect Webcam
• Image Classification
• x8 FPGA Pooling GoogLeNet v1 Demo on AWS F1 instance

xfDNN Tools
• Compiler
• Quantizer

Easy to use Python Interface

Precompiled Models
• 8/16-bit GoogLeNet v1
• 8/16-bit ResNet50
• 8/16-bit Flowers-102
• 8/16-bit GoogleNet v3
• Face Detection
• Yolo-v2 for object detection

Supported Server Platforms
• Intel x86
• FaaS
• AWS F1
• Xilinx SDx boards
  • U200, U250, U280
Seamless Deployment with Open Source Software

From Community

From Xilinx

xDNN CNN Processing Engine

xFDNN Middleware, Tools and Runtime

Caffe

RESTful API

Deploy

TensorFlow Q4 2017
xfDNN Flow

xfDNN flow

Tensorflow  →  MxNet  →  Caffe

Framework Tensor Graph to Xilinx Tensor Graph

xfDNN Tensor Graph Optimization

ONNX

CNTK  →  Caffe2  →  PyTorch

Model Weights
Calibration Set

Image

xfDNN Compiler  →  xfDNN Compression

xCNN Runtime

(pyton API)

CPU Layers  →  FPGA Layers

https://github.com/Xilinx/ml-suite
ML Suite Overlays with xDNN Processing Engines

Adaptable
- AI algorithms are changing rapidly
- Adjacent acceleration opportunities

Realtime
- 10x Low latency than CPU and GPU
- Data flow processing

Efficient
- Performance/watt
- Low Power
**xfDNN Inference Toolbox**

<table>
<thead>
<tr>
<th>Graph Compiler</th>
<th>Network Optimization</th>
<th>xfDNN Quantizer</th>
</tr>
</thead>
<tbody>
<tr>
<td>• Build network graphs from Frameworks</td>
<td>• Fused Layer Optimization</td>
<td>• Easily deploy pre-trained floating point models on 8 bit.</td>
</tr>
<tr>
<td>• Optimizes for Inference</td>
<td>• On-Chip Memory enables Streaming</td>
<td>• Maintains accuracy without needing lengthy retraining</td>
</tr>
<tr>
<td>• Generate code for xDNN IP</td>
<td>• “One Shot” Inference eliminates CPU calls</td>
<td>• Easy and Fast</td>
</tr>
<tr>
<td>• HW/SW partition</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

- *Graph Compiler:* Structured diagrams illustrating how to build network graphs from frameworks, optimizing for inference, and generating code for xDNN IP, along with HW/SW partitioning.

- *Network Optimization:* Diagrams showing fused layer optimization and on-chip memory enabling streaming, including the elimination of CPU calls through "One Shot" inference.

- *xfDNN Quantizer:* Description highlighting the easy deployment of pre-trained floating point models on 8-bit, maintaining accuracy without lengthy retraining, and being easy and fast.
xfDNN Quantizer: Fast and Easy

1) Provide FP32 network and model
   - E.g., prototxt and caffemodel

2) Provide a small sample set, no labels required
   - 16 to 512 images

3) Specify desired precision
   - Quantizes to <8 bits to match Xilinx’s DSP
Xilinx ML Processing Engine - xDNN

<table>
<thead>
<tr>
<th>Features</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Strides: W: 1,2,4,8; H: 1,2,4,8</td>
</tr>
<tr>
<td></td>
<td>Padding: Same, Valid</td>
</tr>
<tr>
<td></td>
<td>Dilation: Factor: 1,2,4</td>
</tr>
<tr>
<td></td>
<td>Activation: ReLU</td>
</tr>
<tr>
<td></td>
<td>Bias: Value Per Channel</td>
</tr>
<tr>
<td></td>
<td>Scaling: Scale &amp; Shift Value Per Channel</td>
</tr>
<tr>
<td>Supported Operations</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Strides: W: 1,2,4,8; H: 1,2,4,8</td>
</tr>
<tr>
<td></td>
<td>Padding: Same, Valid</td>
</tr>
<tr>
<td></td>
<td>Strides: W: 1,2,4,8; H: 1,2,4,8</td>
</tr>
<tr>
<td></td>
<td>Padding: Same, Valid</td>
</tr>
<tr>
<td>Element-wise Add</td>
<td>Width &amp; Height must match; Depth can mismatch.</td>
</tr>
<tr>
<td>Memory Support</td>
<td>On-Chip Buffering, DDR Caching</td>
</tr>
<tr>
<td>Expanded set of image sizes</td>
<td>Square, Rectangular</td>
</tr>
<tr>
<td>Upsampling</td>
<td>Strides: Factor: 2,4,8,16</td>
</tr>
<tr>
<td>Miscellaneous</td>
<td>Data width: 16-bit or 8-bit</td>
</tr>
</tbody>
</table>

- Programmable Feature-set
- Tensor level instructions
- 700M+ DSP Freq (VU9P)
- Customer network acceleration
Computer Vision On Xilinx FPGA
OpenCV for Xilinx 介绍

- Xilinx并没有自己的机器视觉算法，HLS中所有的算法来源都是OpenCV。
- 目前HLS提供的机器视觉算法函数，都只是opencv原版函数的一个重构，功能以及接口参数基本上同原opencv函数保持，适合于HLS综合成hdl代码硬件实现。
- 客户可以直接调用这些函数，也可以参考它们的实现，针对自己的算法做修改。
- 毕竟opencv几千个函数，不可能所有的都提供HLS重构实现，Avnet可以协助客户做的：
  - 在客户已有自己的算法的前提下，Avnet 帮助客户评估在PS/PL实现的优劣（注重于从性能以及资源代价等方面去帮客户评估）。
  - 如果在PS实现，那么客户并不需要做太多工作，客户的代码可以比较容易的就移植到Zynq的ARM上。
  - 某些性能要求苛刻的场合，需要PL逻辑加速的，Avnet 会帮助客户用HLS把已有的软件算法转换成PL实现。
- Avnet 可以提供基于OpenCV 开源算法的Zynq平台实现方法培训
# xFopencv: HW Accelerated OpenCV Functions

<table>
<thead>
<tr>
<th>Level 1</th>
<th>Level 2</th>
<th>Level 3</th>
</tr>
</thead>
<tbody>
<tr>
<td>Absolute difference</td>
<td>Channel combine</td>
<td>Box</td>
</tr>
<tr>
<td>Accumulate</td>
<td>Channel extract</td>
<td>Gaussian</td>
</tr>
<tr>
<td>Accumulate squared</td>
<td>Color convert</td>
<td>Median</td>
</tr>
<tr>
<td>Accumulate weighted</td>
<td>Convert bit depth</td>
<td>Sobel</td>
</tr>
<tr>
<td>Arithmetic addition</td>
<td>Table lookup</td>
<td>Custom convolution</td>
</tr>
<tr>
<td>Arithmetic subtraction</td>
<td>Histogram</td>
<td>Fast corner</td>
</tr>
<tr>
<td>Bitwise: AND, OR, XOR, NOT</td>
<td>Gradient Phase</td>
<td>Harris corner</td>
</tr>
<tr>
<td>Pixel-wise multiplication</td>
<td>Min/Max Location</td>
<td>Erode</td>
</tr>
<tr>
<td>Integral image</td>
<td>Mean &amp; Standard Deviation</td>
<td>Remap</td>
</tr>
<tr>
<td>Gradient Magnitude</td>
<td>Thresholding</td>
<td>Equalize Histogram</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Color Detection</td>
</tr>
<tr>
<td></td>
<td></td>
<td>StereoLBM</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Histogram of Oriented Gradients (HOG)</td>
</tr>
<tr>
<td></td>
<td></td>
<td>ORB</td>
</tr>
<tr>
<td></td>
<td></td>
<td>SVM (binary)</td>
</tr>
<tr>
<td></td>
<td></td>
<td>OTSU Thresholding</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Mean Shift Tracking (MST)</td>
</tr>
<tr>
<td></td>
<td></td>
<td>LK Dense Optical Flow</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Canny edge detection</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Image pyramid</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Color Detection</td>
</tr>
</tbody>
</table>
赛灵思高层次综合工具（Vivado HLS）

- 是一种从 C -> RTL 语言的转换工具
- 全面覆盖 C、C++、OpenCL，能够进行浮点运算和任意精度浮点运算
- 可以输出 Verilog和VHDL代码
- 可以通过制定约束（Directive）来提高运算性能和优化资源利用率
- 从算法验证到硬件实现的自动化工作流程
- 可以集成到嵌入式、System Generator 和 IP Integrator 中
- 适合C算法的工程师进行快速硬件验证
- 赛灵思提供基于OpenCV的库函数，适用于图像处理
Computer Vision / Sensor Fusion with the PicoZed Embedded Vision Kit

```c
main()
    capture (in);
    filter(in,out);
    display(out);
```

Avnet SDSoc Platform (PZ-EMBV)

SDSoC Environment

- SDSoc Generated Platform DMA AXI-S
- AVNET
- HDMI
- AXI
- PS
- PL
- Camera
- Thermal
- HDMI
- Sobel
- auResiz e
- Combin e
- Application
  - Libraries
  - Stub
  - Drivers
- ZYNQ
Computer Vision / Sensor Fusion with the PicoZed Embedded Vision Kit

Visible image

Sobel Edges

Image Fusion

Thermal image

Warped

Resized

SOBEL

COMBINE

WARP

RESIZE
Computer Vision Design Example: 4K60 Dense Optical Flow

```c
main()
{
    imread(A);
    imread(B);
    denseOpticalFlowPyrltr(A, B, out);
    imshow(out);
}
```
Computer Vision Design Example: Stereo Disparity Map

```c
main()
{
    imread(A);
    stereoRectify(A, B, C, D);
    stereoLBM(C, D, out);
    imshow(out);
}
```
Avnet 开发板集锦 (http://ultrazed.org/)

- MicroZed: 基于Zynq 平台的袖珍开发板
- PicoZed: 基于Zynq 平台可用于生产的核心板
- UltraZed: 基于 ZU3EG的学习板和核心板
- Ultra96: 用于96 社区的 ZU3EG 开发板，也可用于生产。
- Zedboard: 基于Zynq 平台的开发版，拥有最多客户群体的开发板。
- MiniZed: 基于7Z007 单核平台的开发板
- Mini-ITX: 基于Zynq 平台的工业 Mini-ITX 开发板
- FPGA 系列：6SLX9, 7A35T/50T, KU040
Avnet 开发板 – Ultra96

- The Ultra96-V2 updates and refreshes the Ultra96 product that was released in 2018.
- Like Ultra96, the Ultra96-V2 is an Arm-based, Xilinx Zynq UltraScale+™ MPSoC development board based on the Linaro 96Boards Consumer Edition (CE) specification.
- Ultra96-V2 has been designed with a certified radio module from Microchip.
- Additionally, all components are updated to allow industrial temperature grade options. Additional power control and monitoring will be possible with the included Infineon Pmics.
- Support DNNDK, PYNQ.

Features
- Xilinx Zynq UltraScale+ MPSoC ZU3EG A484
- Micron 2 GB (512M x32) LPDDR4 Memory
- Delkin 16 GB microSD card + adapter
- PetaLinux environment available for download
- Microchip Wi-Fi / Bluetooth
- Mini DisplayPort (MiniDP or mDP
- 85mm x 54mm form factor
- Linaro 96Boards Consumer Edition compatible
Xilinx DPU IP Deployment in Ultra96 (XCZU3EG)

The tutorial is here:

Introduction

This tutorial demonstrates how to build a custom system that utilizes the 1.2.0 version of Xilinx® Deep Learning Processor (DPU) IP to accelerate machine learning algorithms using the following development flow:

1. Build the hardware platform in the Vivado® Design Suite.
2. Generate the Linux platform in PetaLinux.
3. Use Xilinx SDK to build two machine learning applications that take advantage of the DPU.

Note: The Ultra96 will be the targeted hardware platform.
自动车牌识别参考设计

特性
- 自定义深度学习网络
- Tensorflow 框架
- MMdnn 框架转换
- Xilinx DeePhi DNNDK 2.08
- Xilinx DeePhi DPU

- 精确度 92%
- 性能 37.6 plate/s

主要部件
- Avnet Ultra96 开发板
- Xilinx Zynq UltraScale+ MPSoC ZU3EG
- 2GB LPDDR4
- Wi-Fi
- USB 摄像机

目标应用
- 停车场自动化
- 公路计费

For further enquiries, please contact us at Asia-GDS-Contact@Avnet.com
Avnet 产品 – 人脸识别相机

1080P 人脸识别一体摄像机

专用深度学习处理器，最大支持3万张人脸库
• 大规模神经网络算法，首创嵌入式端侧人脸识别比对一体；
• 专业IP Camera SOC, 4K uHD超高清H.264/H.265视频编码；
• 多路干接点输出可直接控制本地联动设备；
• 内置超级电容，断电可持续工作，有效保护TF卡内关键数据

主要应用领域：
• 智慧社区
• 超市及无人零售
• 门禁及道闸
• 交通站点
• 通行卡口
Avnet Embedded Vision Campaign

https://www.avnet.com/wps/portal/apac/products/c/embedded-vision/

From complexity to clarity

For most humans, sight is intuitive. For machines, sight is an incredibly complex task. Embedded vision technology can help machines "see" by quickly extracting intelligence from images in real time and under various lighting conditions. In the automotive space, this can enable autonomous cars to avoid that pedestrian in the crosswalk or roadside collision faster and more efficiently than ever before.

From autonomous driving to surgical robots and automated factories, the latest innovations depend on sophisticated embedded vision solutions that turn daunting new technological complexity into clarity.
Avnet FPGA Solution Guide

- Reference Design
- IP Core
- Development Kits

A New Chapter in Embedded Designs
Avnet AI Training and Solutions

- Training Event: AI workshop
  - Propose AI workshop to present Xilinx AI solution on Edge (DNNDK) and cloud (ML suite)

- Reference design:
  - Automatic Number Plate Recognition by ADS
  - DNNDK tool chain
  - Based on Ultra96

- Product:
  - Face recognition camera
  - Based on Deephi XC7Z020 module
Thank you