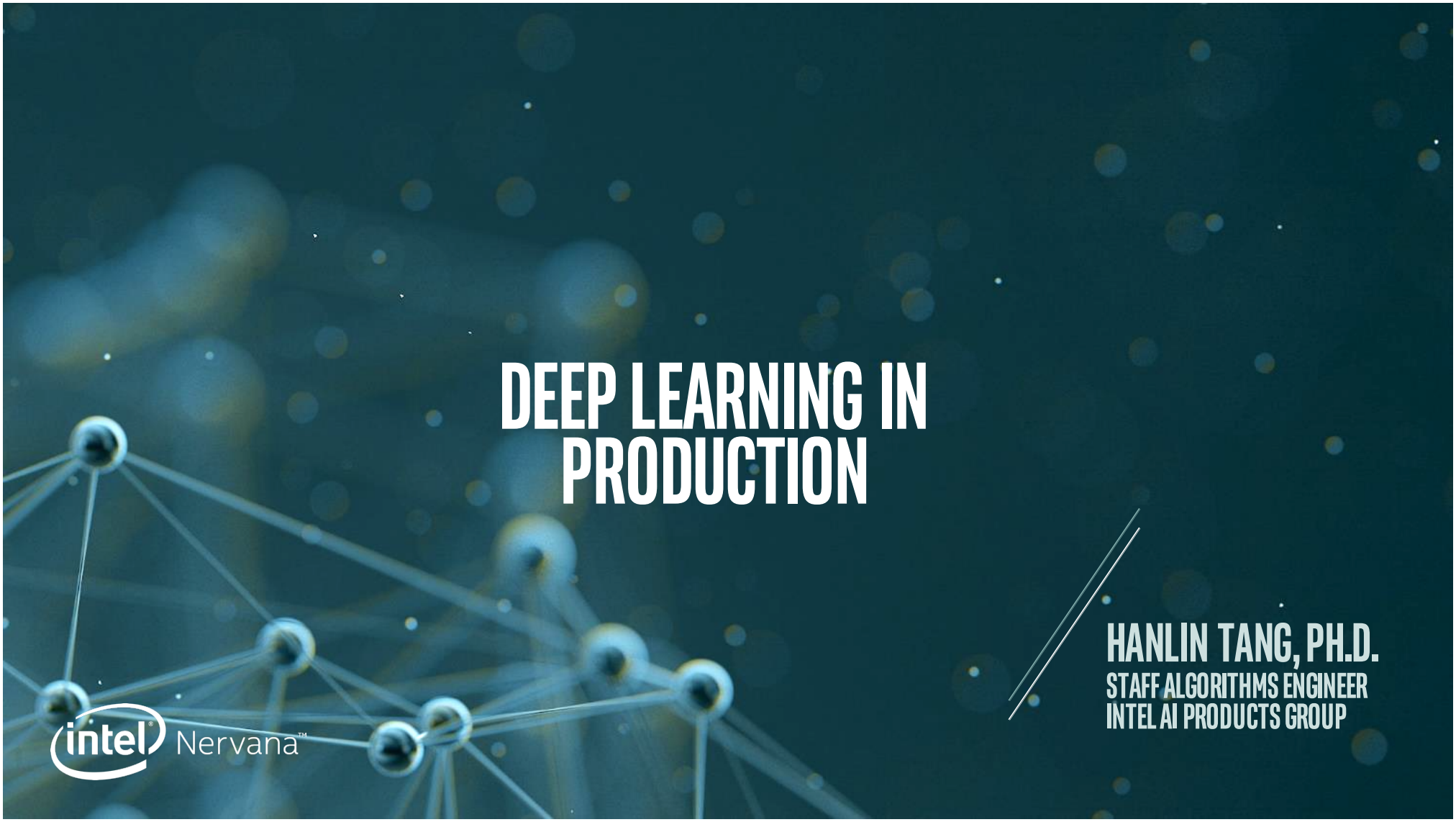

DEEP LEARNING IN PRODUCTION
intel Nervana™
HANLIN TANG, PH.D.
STAFF ALGORITHMS ENGINEER
INTEL AI PRODUCTS GROUP

# AI IS TRANSFORMATIVE








| CONSUMER | HEALTH | FINANCE | RETAIL | GOVERNMENT | ENERGY | TRANSPORT | INDUSTRIAL | OTHER |
|---|---|---|---|---|---|---|---|---|
| Smart Assistants | Enhanced Diagnostics | Algorithmic Trading | Support | Defense | Oil & Gas Exploration | Autonomous Cars | Factory Automation | Advertising |
| Chatbots | Drug Discovery | Fraud Detection | Experience | Data Insights | Smart Grid | Automated Trucking | Predictive Maintenance | Education |
| Search | Patient Care | Research | Marketing | Safety & Security | Operational Improvement | Aerospace | Precision Agriculture | Gaming |
| Personalization | Research | Personal Finance | Merchandising | Resident Engagement | Conservation | Shipping | Field Automation | Professional & IT Services |
| Augmented Reality | Sensory Aids | Risk Mitigation | Loyalty | Smarter Cities | | Search & Rescue | | Telco/Media |
| Robots | | | Supply Chain | | | | | Sports |
| | | | Security | | | | | |

*Source: Intel forecast*

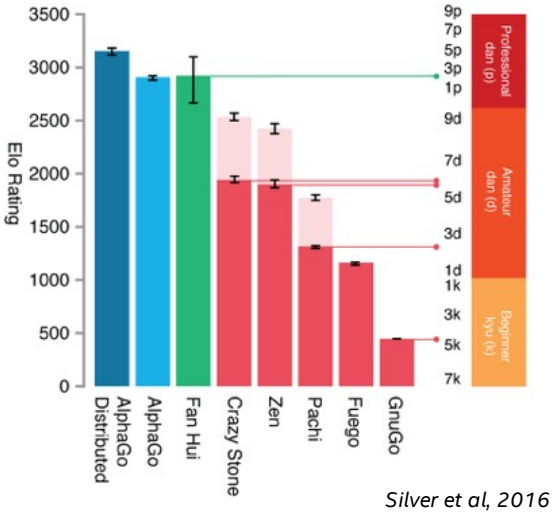
# DEEP LEARNING DRIVING AI APPLICATIONS

*Facebook DeepMask*

**How Google's AlphaGo Beat a Go World Champion**

Inside a man-versus-machine showdown

*The Atlantic, March 2016*




*Silver et al, 2016*


*"The error rate has been cut by a **factor of two** in all the languages, more than a factor of two in many cases. That's mostly due to deep learning and the way we have optimized it …"*

Alex Acero, Siri Senior Director, Apple
Article in Backhannel/WIRED, Aug 2016

*Source: Intel forecast*

# CLASSIC MACHINE LEARNING

$N \times N$

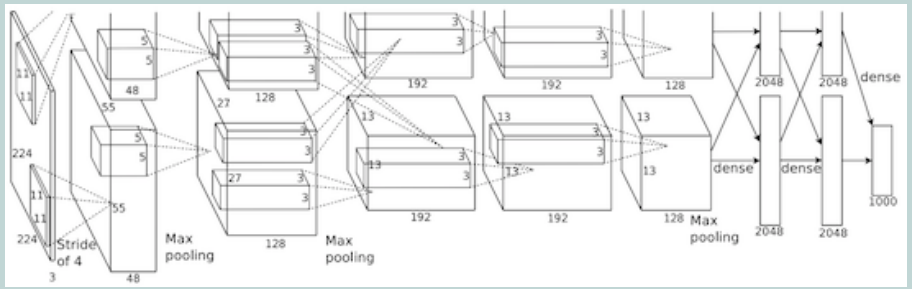
$(f_1, f_2, \ldots, f_K)$

**CLASSIFIER**
SVM
Random Forest
Naïve Bayes
Decision Trees
Logistic Regression
Ensemble methods

**Arjun**

# END-TO-END DEEP LEARNING

$N \times N$




~60 million parameters

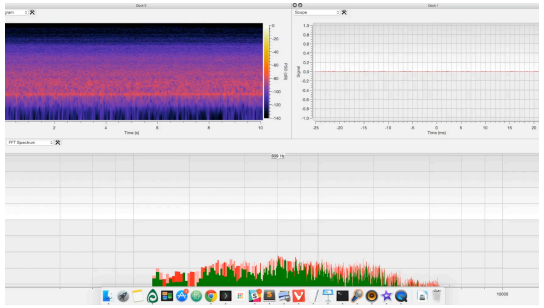
**Arjun**

But old practices apply:
Data Cleaning, Exploration, Data annotation, hyperparameters, etc.

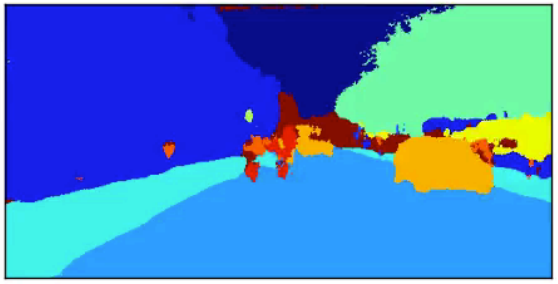
# GENERALITY


BabyCrawling 0.91


car 0.975
car 0.912
car 0.976
car 0.985

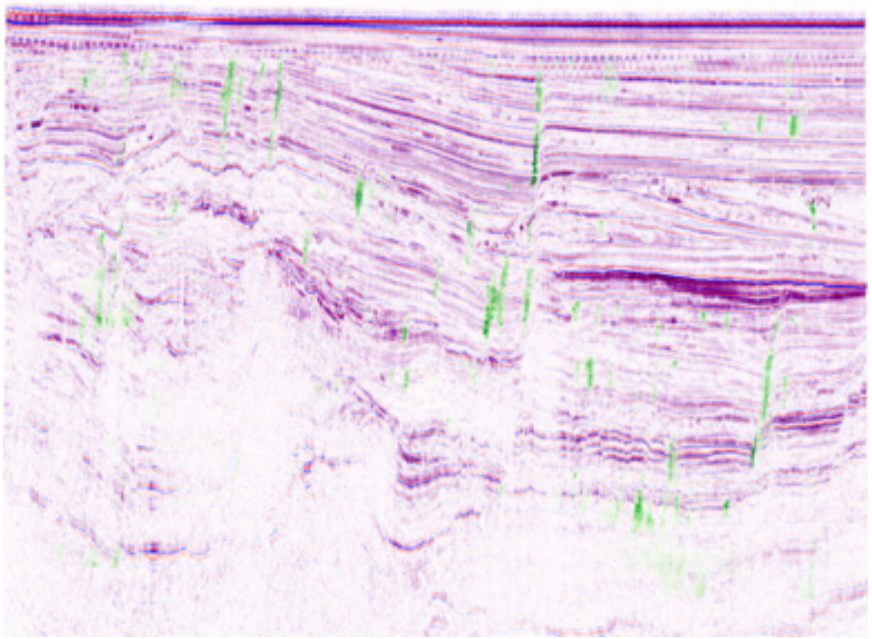



frog
cat
bird
truck
dog
automobile
deer
horse
ship
airplane

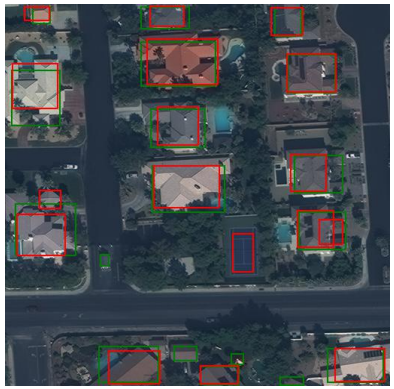


# APPLIED AI MODELS

## Oil & Gas: Fault Line Predictions



## Satellite Imagery

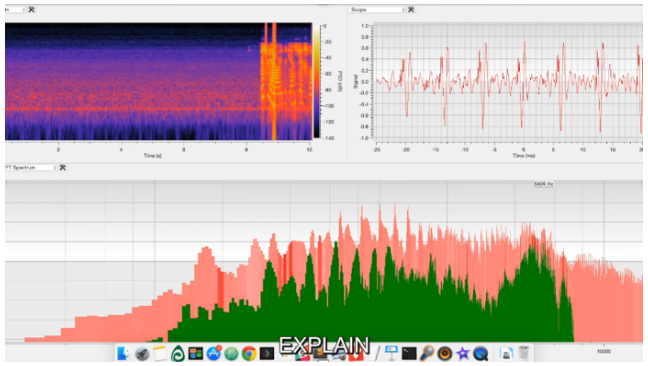


## Industry: Precision Agriculture

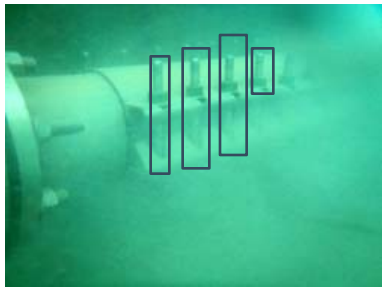
**Positive**

**Negative**

## Industry: Speech interfaces




## Industry: Underwater Robotics

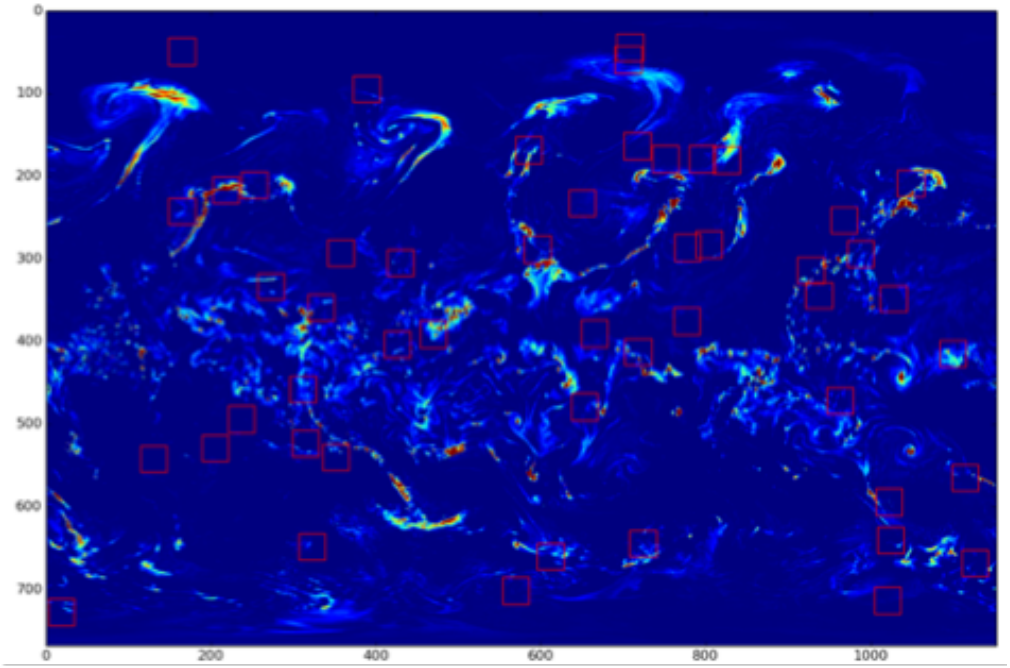


Infrastructure inspection

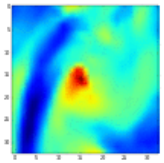
## Sports: Pose Estimation

## NERSC: Localize weather phenomenon

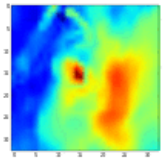
Water Vapor (TMQ) globally at one time point



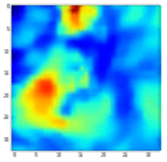

Measurements for a single localized hurricane



TMQ



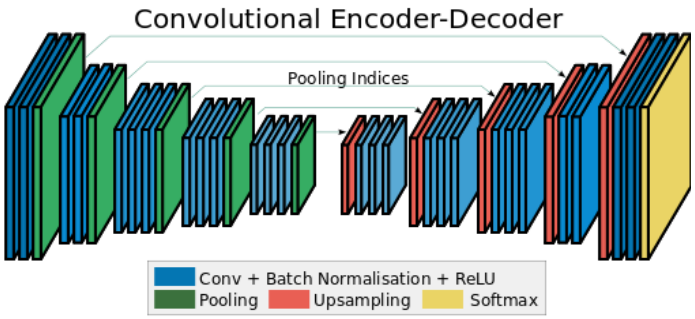
T@500



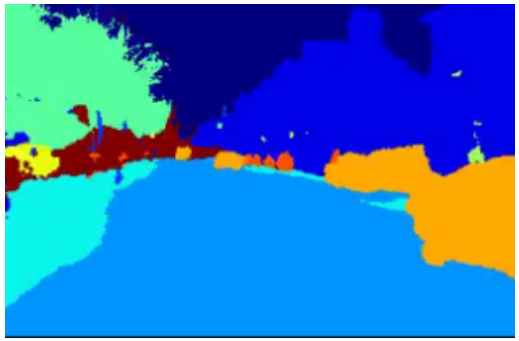
T@200

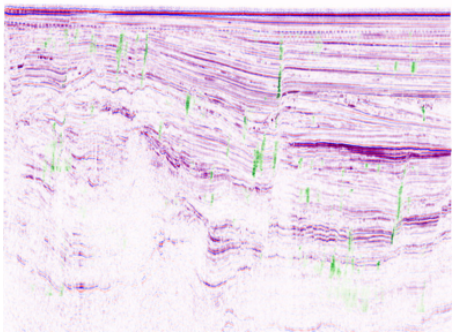
T@500 = Temperature at altitude of 200mb

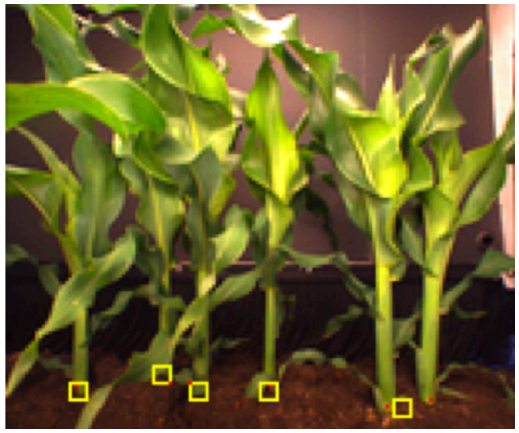
# SAME MODEL, DIFFERENT DATA

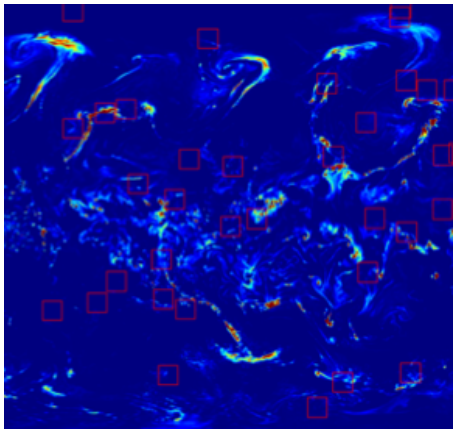



*Badrinarayanan et al, 2015*

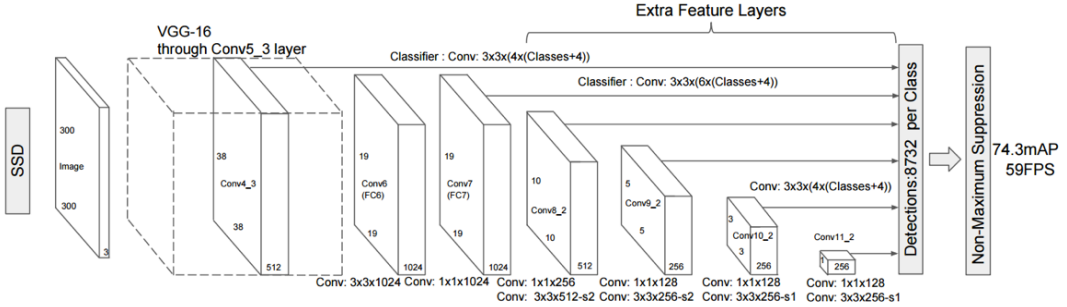




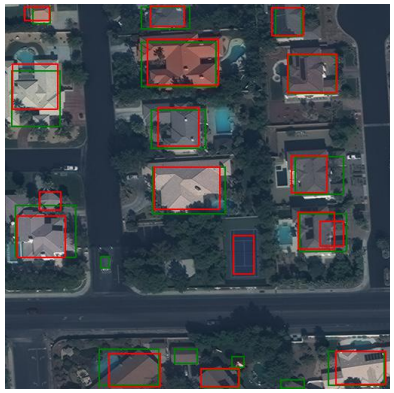


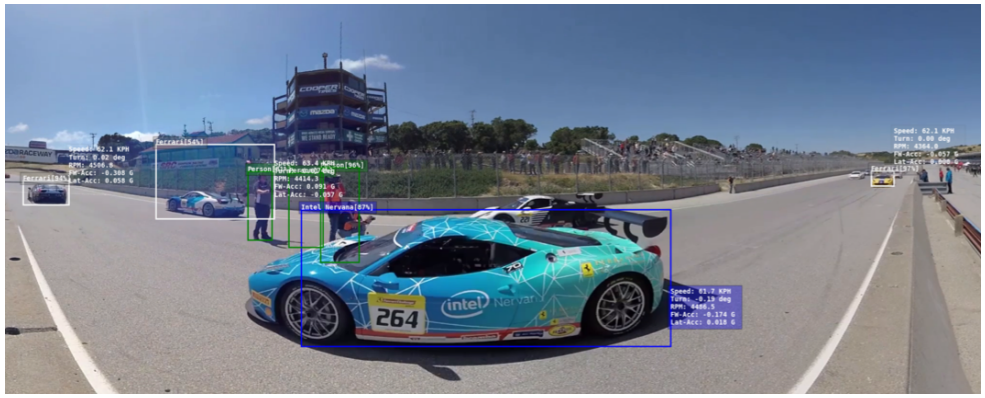


# SAME MODEL, DIFFERENT DATA

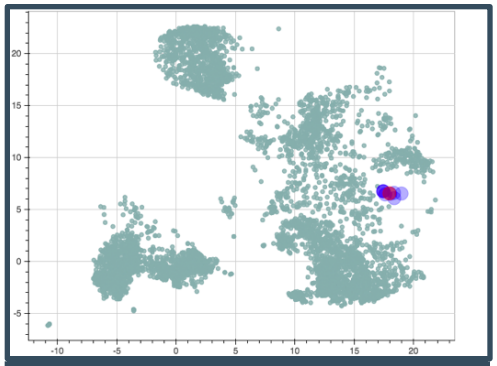



*Liu et al, 2016*



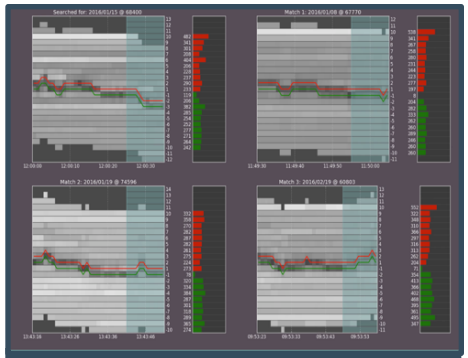



# SAME MODEL, DIFFERENT DATA



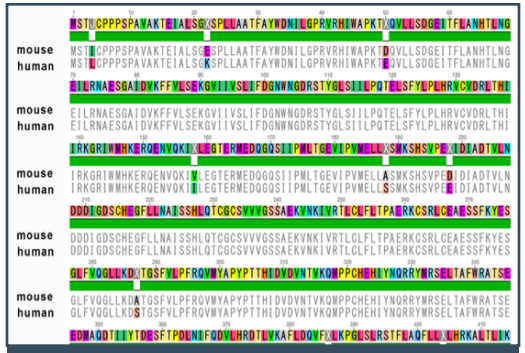
NATURAL LANGUAGE

- Document understanding
- Topic analysis



FINANCE

- Time-series analysis
- Anomaly detection

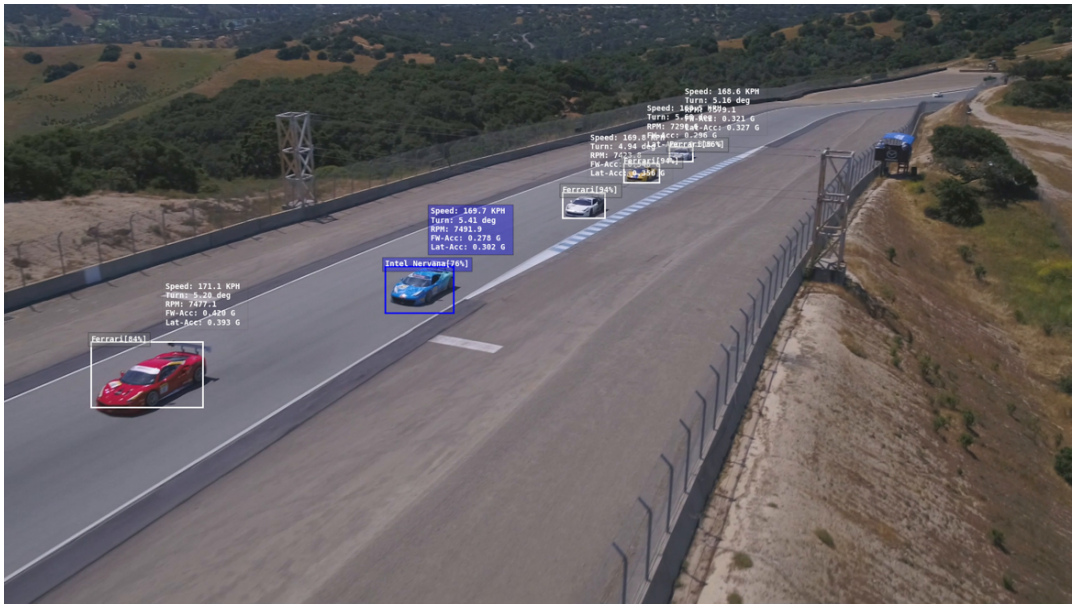



GENOMICS

- Sequence analysis

*#1. Same model has multiple applications. Consider tasks, not domains.*

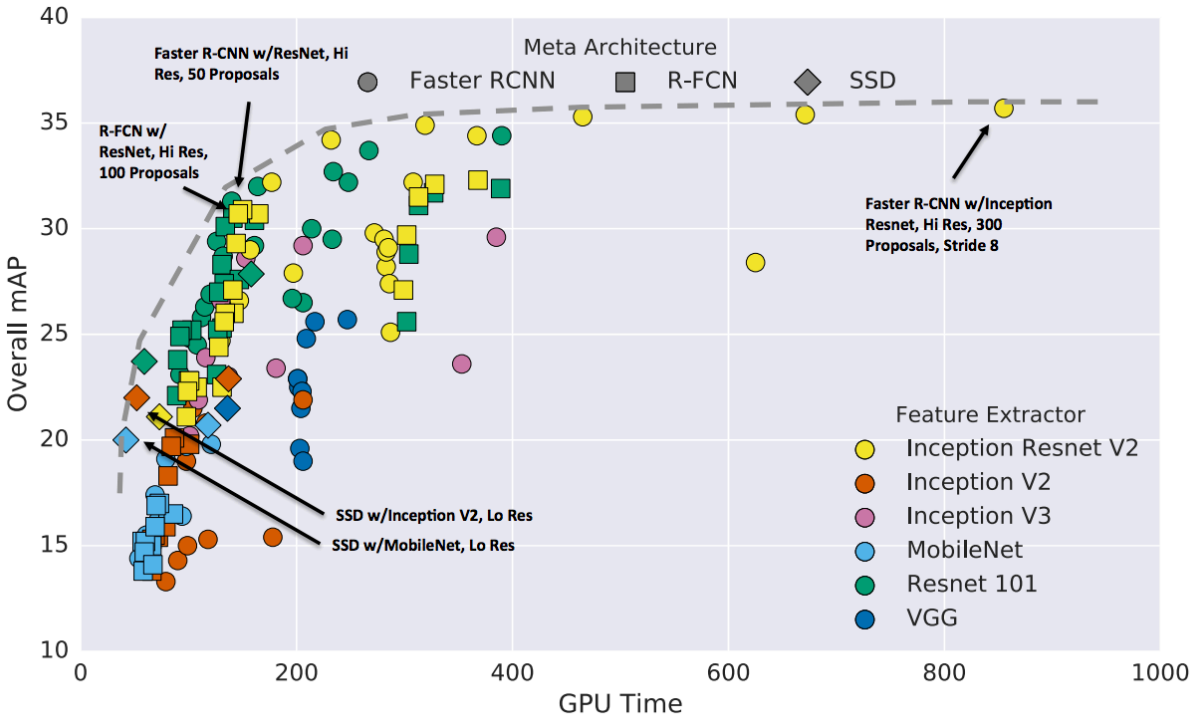
# MODEL SELECTION

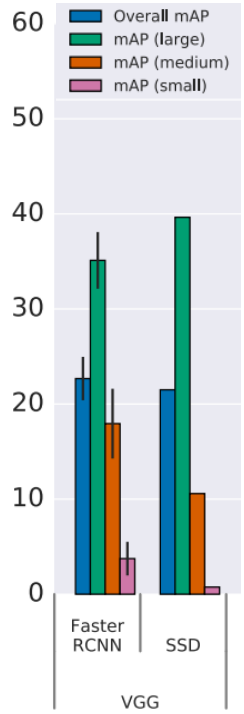


*Thomas Keller, Intel AI Products Group*

Many models: Faster-RCNN, Single Shot Detection, R-FCN, ...

# MODEL SELECTION

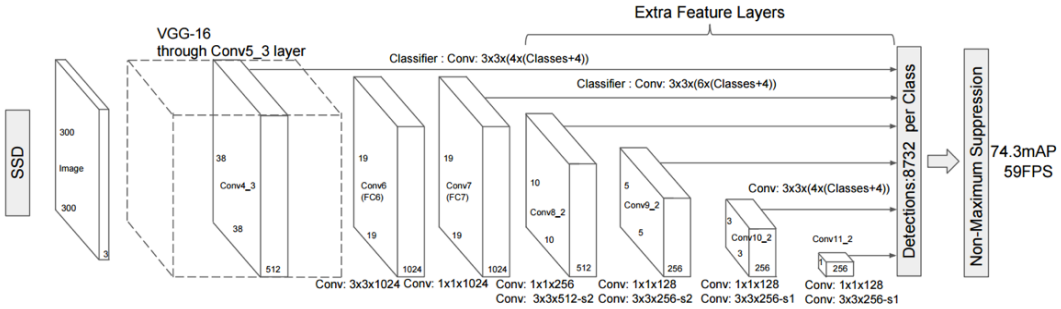



*#2. Latest model not always most relevant model*

Huang et al (2017), *Speed/accuracy trade-offs for modern convolutional object detectors*

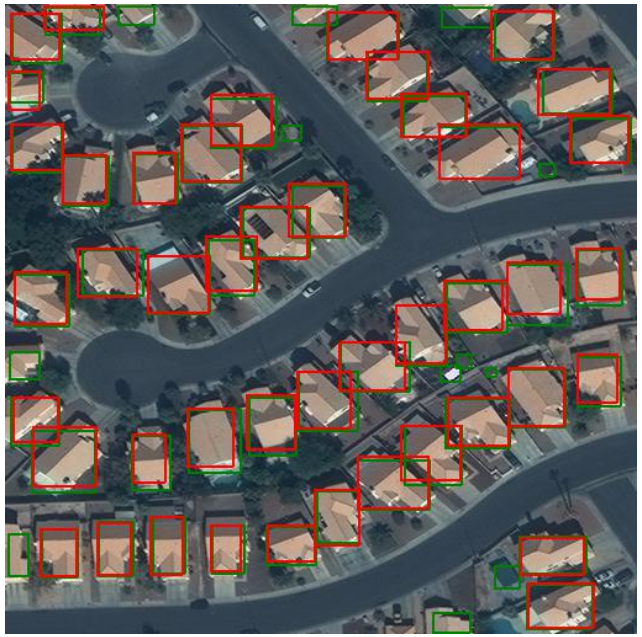
# KNOW YOUR MODEL PROVENANCE




**SSD model designed and optimized for PASCALVOC/MSCOCO dataset**

PASCALVOC vs. Satellite Imagery

- 20 objects vs. 200+ objects
- Rotation Symmetry
- Multi-spectral data
- **Rotated boxes**
- 10,000x10,000 pixels

# KNOW YOUR MODEL PROVENANCE

We are currently modifying topologies for satellite data:

- Predict angle of boxes
- Handle dense clusters
- Exploit spectral bands
- Better edge handling

*#3. Know your model provenance*

# INFERENCE: QUANTIZATION, PRUNING, AND DESIGN

| CNN architecture | Compression Approach | Data Type | Original → Compressed Model Size | Reduction in Model Size vs. AlexNet | Top-1 ImageNet Accuracy | Top-5 ImageNet Accuracy |
|---|---|---|---|---|---|---|
| AlexNet | None (baseline) | 32 bit | 240MB | 1x | 57.2% | 80.3% |
| AlexNet | SVD (Denton et al., 2014) | 32 bit | 240MB → 48MB | 5x | 56.0% | 79.4% |
| AlexNet | Network Pruning (Han et al., 2015b) | 32 bit | 240MB → 27MB | 9x | 57.2% | 80.3% |
| AlexNet | Deep Compression (Han et al., 2015a) | 5-8 bit | 240MB → 6.9MB | 35x | 57.2% | 80.3% |
| SqueezeNet (ours) | None | 32 bit | 4.8MB | **50x** | 57.5% | 80.3% |
| SqueezeNet (ours) | Deep Compression | 8 bit | 4.8MB → 0.66MB | **363x** | 57.5% | 80.3% |
| SqueezeNet (ours) | Deep Compression | 6 bit | 4.8MB → 0.47MB | **510x** | 57.5% | 80.3% |

***Iandola et al, 2016***

| Model | Computation cost (MAC) | | | | Running time | | mAP |
|---|---|---|---|---|---|---|---|
| | Shared CNN | RPN | Classifier | Total | ms | x(PVANET) | (%) |
| PVANET+ | 7.9 | 1.3 | 27.7 | 37.0 | 46 | 1.0 | 82.5 |
| Faster R-CNN + ResNet-101 | 80.5 | N/A | 219.6 | 300.1 | 2240 | 48.6 | 83.8 |
| Faster R-CNN + VGG-16 | 183.2 | 5.5 | 27.7 | 216.4 | 110 | 2.4 | 75.9 |
| R-FCN + ResNet-101 | 122.9 | 0 | 0 | 122.9 | 133 | 2.9 | 82.0 |

***Intel Labs*** *(Yu et al, CVPR 2017)*

*#4. Quantize, prune, and compress*

# THE COMING FLOOD OF DATA

## BY 2020...

The average internet user will generate
**~1.5 GB OF TRAFFIC PER DAY**

Smart hospitals will generate over
**3,000 GB PER DAY**

Self driving cars will generate over
**4,000 GB PER DAY... EACH**

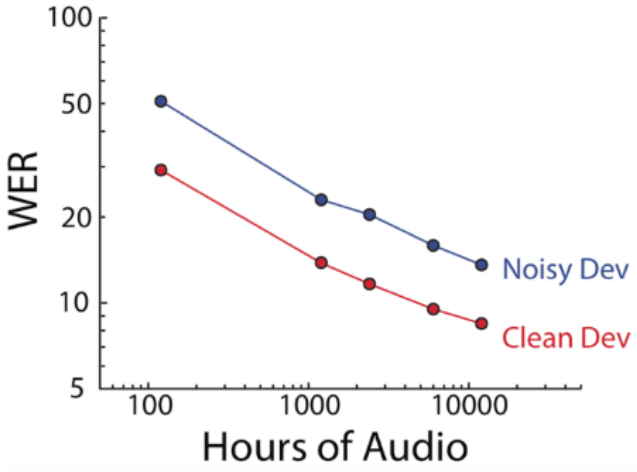
A connected plane will generate over
**40,000 GB PER DAY**

A connected factory will generate over
**1,000,000 GB PER DAY**

Manual Annotations → **Training Data**

*All numbers are approximated*
*http://www.cisco.com/c/en/us/solutions/service-provider/vni-network-traffic-forecast/infographic.html*
*http://www.cisco.com/c/en/us/solutions/collateral/service-provider/global-cloud-index-gci/Cloud_Index_White_Paper.html*
*https://datafloq.com/read/self-driving-cars-create-2-petabytes-data-annually/172*
*http://www.cisco.com/c/en/us/solutions/collateral/service-provider/global-cloud-index-gci/Cloud_Index_White_Paper.html*
*http://www.cisco.com/c/en/us/solutions/collateral/service-provider/global-cloud-index-gci/Cloud_Index_White_Paper.html*

# DATA IS THE NEW OIL




*Baidu DS2*

| 8 Million | 0.5 Million | 1.9 Billion | 4800 | 1.8 |
|---|---|---|---|---|
| Video URLs | Hours of Video | Frame Features | Classes | Avg. Labels / Video |

*Google's YouTube-8M*

*Spacenet*


*Imagenet*

*#5. Invest in data, even for early POCs*

# OPERATING IN A DATA LIMITED WORLD

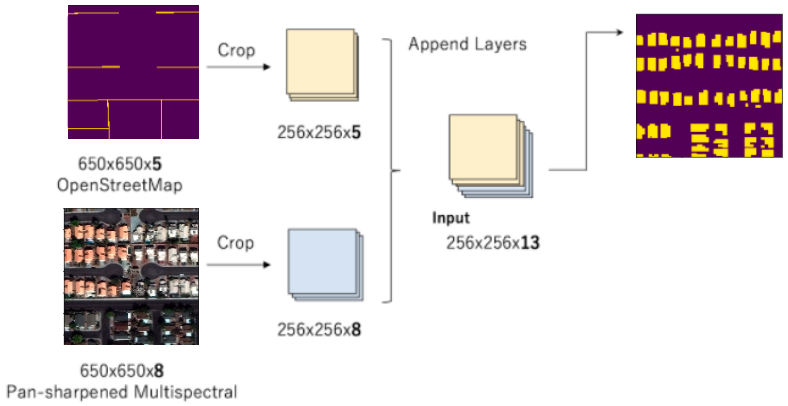
- Data Augmentation is key
  - Vision: Rotate, flip, color noise, crop, etc.
  - NLP: thesaurus replacement.
  - Speech: overlay noise sources relevant to your application (e.g. car backgrounds).

- Hidden data sources:
  - **Example**: building detection in satellite imagery.
  - **Example**: multi-modal emotion detection (image, sentiment analysis on transcript, speech analysis, character consistency, etc.)

*#6: Augment, augment, augment*

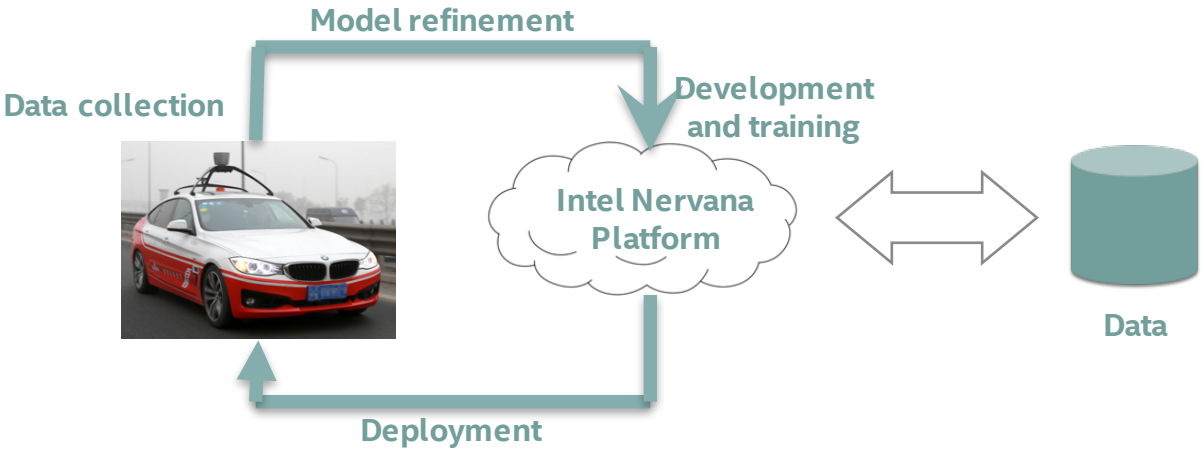
| | SSD300 | | | | |
|---|---|---|---|---|---|
| more data augmentation? | | ✔ | ✔ | ✔ | ✔ |
| include $\{\frac{1}{2}, 2\}$ box? | ✔ | | ✔ | ✔ | ✔ |
| include $\{\frac{1}{3}, 3\}$ box? | ✔ | | | ✔ | ✔ |
| use atrous? | ✔ | ✔ | ✔ | | ✔ |
| VOC2007 test mAP | 65.5 | 71.6 | 73.7 | 74.2 | **74.3** |

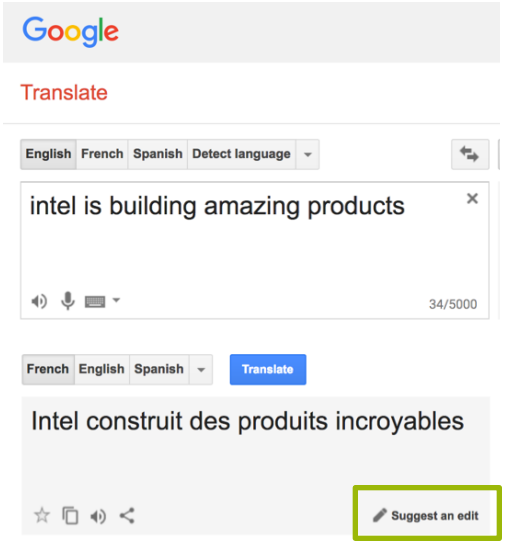
*Liu et al, 2016*




Kohei Ozaki, *Winning solution to Spacenet challenge (2017)*

# CLOSE THE LOOP



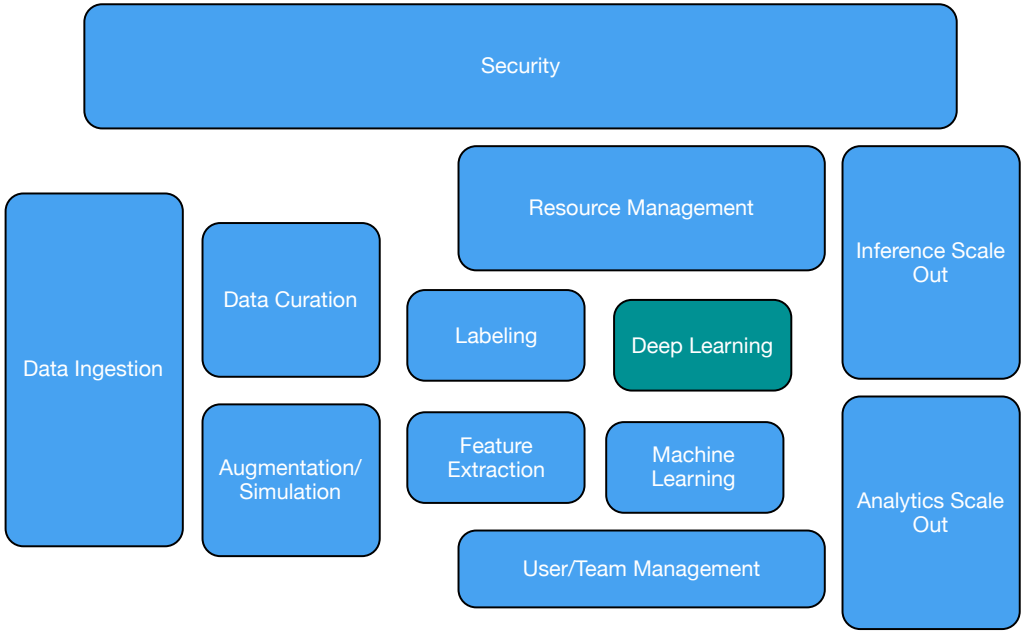




translate.google.com

*#7. Close the loop. Annotate on the edge.*

*#8. Monitor in Production*

# DEEP LEARNING IS PART OF A SYSTEM




**Key components:**

- Instrumentation and monitoring of model performance
- Edge annotation and accretion training
- Simulation and validation before model update
- Adjustable model sensitivity by location, time of day, etc.
- Inference optimizations (quantization, pruning, sparsity)

Only a small part of “Deep Learning” deployment is actually Deep Learning

# DEEP LEARNING IS PART OF A SYSTEM

## Automated Driving

5G

### VEHICLE

Autonomous Driving Functions
Trajectory Enumeration, Path Planning, Selection & Maneuvering
Driving Policy, Path Selection

In-vehicle inference HW

Sensor Processing & Fusion
Object ID & Classification

Anomaly Detection

### NETWORK


Data Storage

Dataset Management & Traceability

*Compressed Models*

*OTA SW /FW Updates*

*Data Formatting*

### PLATFORM

Neural Network Design for Target Hardware, & usage
(Vision, Data Driven, etc.)

Model Training
Single & Multi-node optimized Frameworks

Model Inference >than Real Time
Model Simulation & Verification

# INTEL® NERVANA™ PORTFOLIO

**EXPERIENCES**

**PLATFORMS**

Intel® Nervana™ Cloud & Appliance

Intel® Nervana™ DL Studio

Intel® Computer Vision SDK

Movidius Fathom

intel Saffron™

**FRAMEWORKS**

Spark MLLIB BIGDL

neon

TensorFlow

Microsoft CNTK

mxnet

torch

Caffe

Caffe2

Chainer

theano

**LIBRARIES**

python Intel Python Distribution

Intel® Data Analytics Acceleration Library (DAAL)

Intel® Nervana™ Graph*

Intel® Math Kernel Library (MKL, MKL-DNN)

**HARDWARE**


Compute

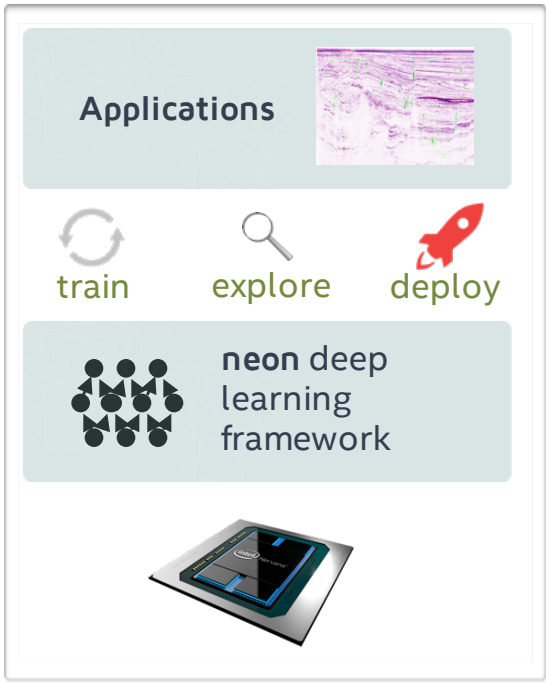
Memory & Storage

Networking

INSIDE AI

*Future

# DEEP LEARNING STACK

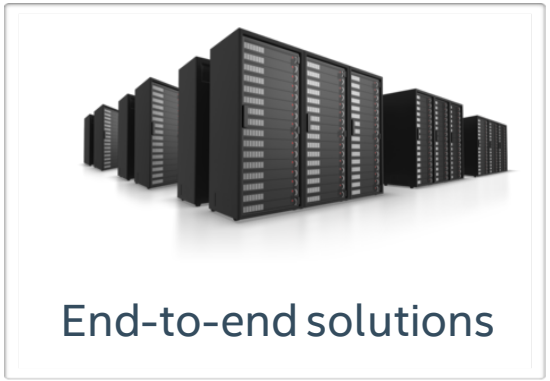



Algorithm expertise

Systems at scale

Framework innovations

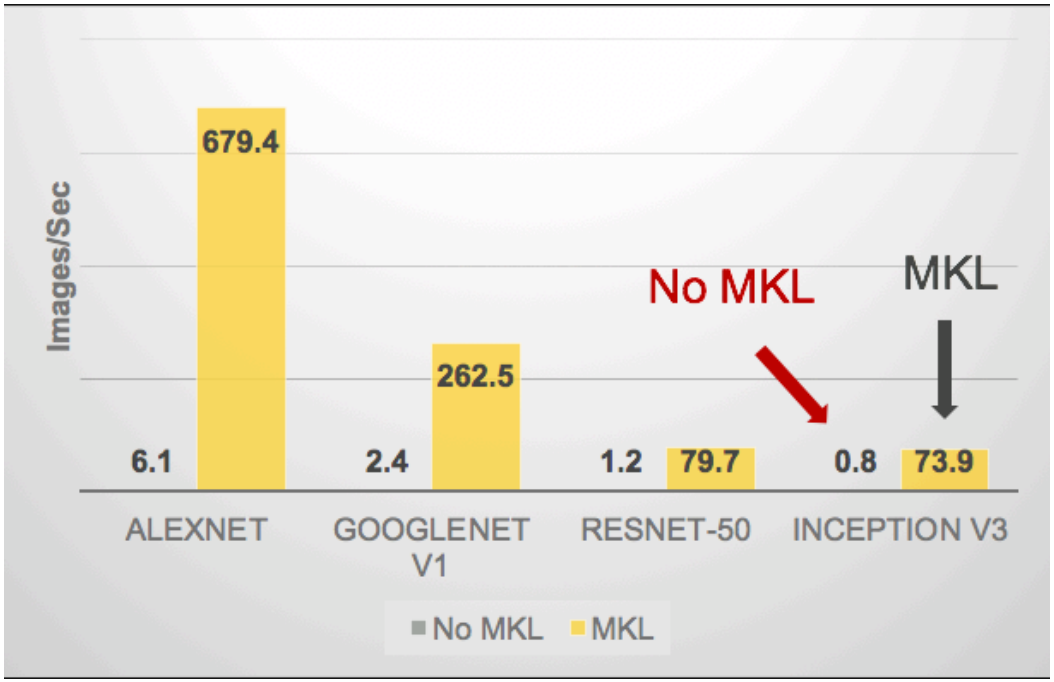
AI-specific Silicon



End-to-end solutions

*#9. Use Intel Nervana portfolio*

# USE MATH KERNEL LIBRARY FOR SPEED-UP ON CPUS




*#10. Enable MKL on CPUs*

mxnet

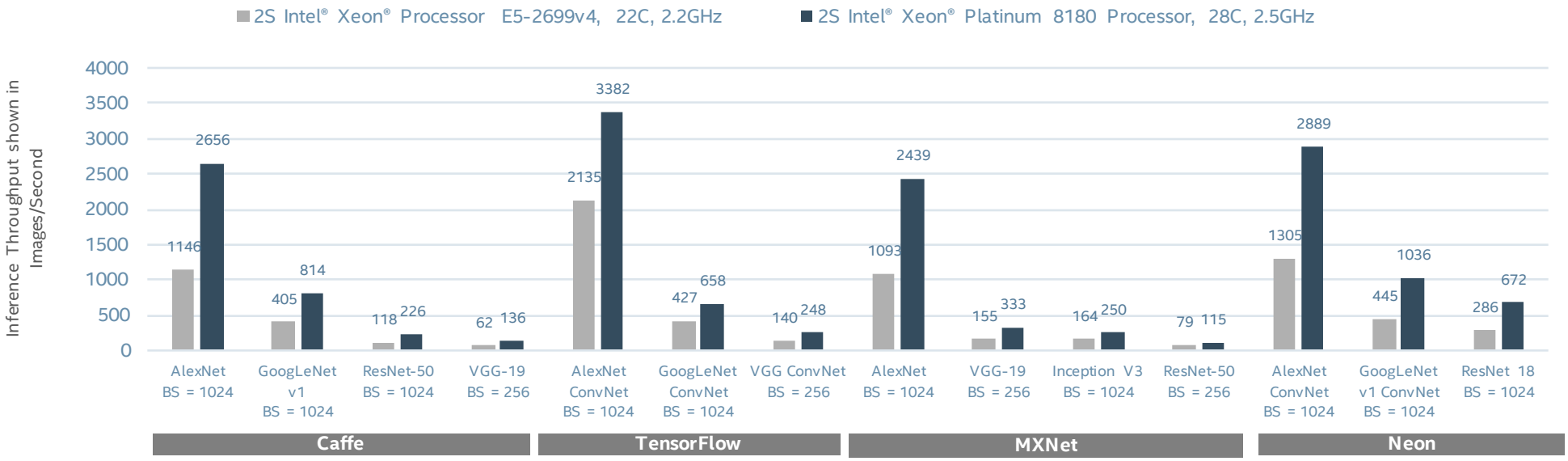
Caffe

TensorFlow

*OS: Linux version 3.13.0-86-generic (buildd@lgw01-51) (gcc version 4.8.2 (Ubuntu 4.8.2-19ubuntu1) ) #131-Ubuntu SMP Thu May 12 23:33:13 UTC 2016. MxNet Tip of tree: commit de41c736422d730e7cfad72dd6afc229ce08cf90, Tue Nov 1 11:43:04 2016 +0800. MKL 2017 Gold update 1*

# UP TO 2.4X HIGHER INFERENCE THROUGHPUT

## ON INTEL® XEON® PLATINUM 8180 PROCESSOR




Inference throughput measured with FP32 instructions. Inference with INT8 will be higher. Additional optimizations may further improve performance.

**Intel® Xeon® Platinum Processor delivers Inference throughput performance across different frameworks**

# 10 TIPS FOR BUILDING AI FOR ENTERPRISE

1. Same model has multiple applications. Think task not domains.
2. Latest model is not always the most relevant model.
3. Know your model provenance.
4. Inference: quantize, prune, and compress.
5. Invest in labeling data, even for early POCs.
6. Augment, augment, augment.
7. Close the loop. Annotate on the edge.
8. Monitor in production.
9. Use Intel Nervana technology (neon, Crest, Xeon, DL Platform, etc.)
10. Enable Math Kernel Library acceleration in your framework of choice


intel®
Nervana™

# LEGAL NOTICES & DISCLAIMERS

This document contains information on products, services and/or processes in development. All information provided here is subject to change without notice. Contact your Intel representative to obtain the latest forecast, schedule, specifications and roadmaps.

Intel technologies' features and benefits depend on system configuration and may require enabled hardware, software or service activation. Learn more at intel.com, or from the OEM or retailer. No computer system can be absolutely secure.

Tests document performance of components on a particular test, in specific systems. Differences in hardware, software, or configuration will affect actual performance. Consult other sources of information to evaluate performance as you consider your purchase. For more complete information about performance and benchmark results, visit http://www.intel.com/performance.

Cost reduction scenarios described are intended as examples of how a given Intel-based product, in the specified circumstances and configurations, may affect future costs and provide cost savings. Circumstances will vary. Intel does not guarantee any costs or cost reduction.

Statements in this document that refer to Intel's plans and expectations for the quarter, the year, and the future, are forward-looking statements that involve a number of risks and uncertainties. A detailed discussion of the factors that could affect Intel's results and plans is included in Intel's SEC filings, including the annual report on Form 10-K.

The products described may contain design defects or errors known as errata which may cause the product to deviate from published specifications. Current characterized errata are available on request.

No license (express or implied, by estoppel or otherwise) to any intellectual property rights is granted by this document.

Intel does not control or audit third-party benchmark data or the web sites referenced in this document. You should visit the referenced web site and confirm whether referenced data are accurate.

Intel, the Intel logo, Pentium, Celeron, Atom, Core, Xeon and others are trademarks of Intel Corporation in the U.S. and/or other countries.
*Other names and brands may be claimed as the property of others.

# CONFIGURATION DETAILS

**2S Intel® Xeon® processor E5-2697A v4 on Apache Spark™ with MKL2017 up to 18x performance increase compared to 2S E5-2697 v2 + F2JBLAS machine learning training**
BASELINE: Intel® Xeon® Processor E5-2697 v2 (12 Cores, 2.7 GHz), 256GB memory, CentOS 6.6*, F2JBLAS: https://github.com/fommil/netlib-java, Relative performance 1.0

Intel® Xeon® processor E5-2697 v2 Apache* Spark* Cluster: 1-Master + 8-Workers, 10Gbit/sec Ethernet fabric, Each system with 2 Processors, Intel® Xeon® processor E5-2697 v2 (12 Cores, 2.7 GHz), Hyper-Threading Enabled, 256GB RAM per System, 1-240GB SSD OS Drive, 12-3TB HDDs Data Drives Per System, CentOS* 6.6, Linux 2.6.32-642.1.1.el6.x86_64, Intel® Intel® MKL 2017 build U1_20160808, Cloudera Distribution for Hadoop (CDH) 5.7, Apache* Spark* 1.6.1 standalone, OMP_NUM_THREADS=1 set in CDH*, Total Java Heap Size of 200GB for Spark* Master and Workers, Relative performance up to 3.4x

Intel® Xeon® processor E5-2699 v3 Apache* Spark* Cluster: 1-Master + 8-Workers, 10Gbit/sec Ethernet fabric, Each system with 2 Processors, Intel® Xeon® processor E5-2699 v3 (18 Cores, 2.3 GHz), Hyper-Threading Enabled, 256GB RAM per System, 1-480GB SSD OS Drive, 12-4TB HDDs Data Drives Per System, CentOS* 7.0, Linux 3.10.0-229.el7.x86_64, Intel® Intel® MKL 2017 build U1_20160808, Cloudera Distribution for Hadoop (CDH) 5.7, Apache* Spark* 1.6.1 standalone, OMP_NUM_THREADS=1 set in CDH*, Total Java Heap Size of 200GB for Spark* Master and Workers, Relative performance up to 8.8x

Intel® Xeon® processor E5-2697A v4 Apache* Spark* Cluster: 1-Master + 8-Workers, 10Gbit Ethernet/sec fabric, Each system with 2 Processors, Intel® Xeon® processor E5-2697A v4 (16 Cores, 2.6 GHz), Hyper-Threading Enabled, 256GB RAM per System, 1-800GB SSD OS Drive, 10-240GB SSDs Data Drives Per System, CentOS* 6.7, Linux 2.6.32-573.12.1.el6.x86_64, Intel® MKL 2017 build U1_20160808, Cloudera Distribution for Hadoop (CDH) 5.7, Apache* Spark* 1.6.1 standalone, OMP_NUM_THREADS=1 set in CDH*, Total Java Heap Size of 200GB for Spark* Master and Workers, Relative performance up to 18x

Machine learning algorithm used for all configurations : Alternating Least Squares ALS Machine Learning Algorithm https://github.com/databricks/spark-perf

**Intel® Xeon Phi™ Processor 7250 GoogleNet V1 Time-To-Train Scaling Efficiency up to 97% on 32 nodes**
32 nodes of Intel® Xeon Phi™ processor 7250 (68 Cores, 1.4 GHz, 16GB MCDRAM: flat mode), 96GB DDR4 memory, Red Hat* Enterprise Linux 6.7, export OMP_NUM_THREADS=64 (the remaining 4 cores are used for driving communication) MKL 2017 Update 1, MPI: 2017.1.132, Endeavor KNL bin1 nodes, export I_MPI_FABRICS=tmi, export I_MPI_TMI_PROVIDER=psm2, Throughput is measured using "train" command. Data pre-partitioned across all nodes in the cluster before training. There is no data transferred over the fabric while training. Scaling efficiency computed as: (Single node performance / (N * Performance measured with N nodes))*100, where N = Number of nodes

Intel® Caffe: Intel internal version of Caffe

GoogLeNetV1: http://static.googleusercontent.com/media/research.google.com/en//pubs/archive/43022.pdf, batch size 1536

**Intel® Xeon Phi™ processor 7250 up to 400x performance increase with Intel Optimized Frameworks compared to baseline out of box performance**
BASELINE: Caffe Out Of the Box, Intel® Xeon Phi™ processor 7250 (68 Cores, 1.4 GHz, 16GB MCDRAM: cache mode), 96GB memory, Centos 7.2 based on Red Hat* Enterprise Linux 7.2, BVLC-Caffe: https://github.com/BVLC/caffe, with OpenBLAS, Relative performance 1.0

NEW: Caffe: Intel® Xeon Phi™ processor 7250 (68 Cores, 1.4 GHz, 16GB MCDRAM: cache mode), 96GB memory, Centos 7.2 based on Red Hat* Enterprise Linux 7.2, Intel® Caffe: : https://github.com/intel/caffe based on BVLC Caffe as of Jul 16, 2016, MKL GOLD UPDATE1, Relative performance up to 400x

AlexNet used for both configuration as per https://papers.nips.cc/paper/4824-large-image-database-classification-with-deep-convolutional-neural-networks.pdf, Batch Size: 256

**Intel® Xeon Phi™ Processor 7250, 32 node cluster with Intel® Omni Path Fabric up to 97% GoogleNetV1 Time-To-Train Scaling Efficiency**
Caffe: Intel® Xeon Phi™ processor 7250 (68 Cores, 1.4 GHz, 16GB MCDRAM: flat mode), 96GB DDR4 memory, Red Hat* Enterprise Linux 6.7, Intel® Caffe: https://github.com/intel/caffe, not publically available yet

export OMP_NUM_THREADS=64 (the remaining 4 cores are used for driving communication)

MKL 2017 Update 1, MPI: 2017.1.132, Endeavor KNL bin1 nodes, export I_MPI_FABRICS=tmi, export I_MPI_TMI_PROVIDER=psm2, Throughput is measured using "train" command. Split the images across nodes and copied locally on each node at the beginning of training. No IO happens over fabric while training.

GoogLeNetV1: http://static.googleusercontent.com/media/research.google.com/en//pubs/archive/43022.pdf, batch size 1536

**Intel® Xeon Phi™ processor Knights Mill up to 4x estimated performance improvement over Intel® Xeon Phi™ processor 7290**
BASELINE: Intel® Xeon Phi™ Processor 7290 (16GB, 1.50 GHz, 72 core) with 192 GB Total Memory on Red Hat Enterprise Linux* 6.7 kernel 2.6.32-573 using MKL 11.3 Update 4, Relative performance 1.0

NEW: Intel® Xeon phi™ processor family – Knights Mill, Relative performance up to 4x

**Intel® Arria 10 – 1150 FPGA energy efficiency on Caffe/AlexNet up to 25 img/s/w with FP16 at 297MHz**
Vanilla AlexNet Classification Implementation as specified by http://www.cs.toronto.edu/~fritz/absps/imagenet.pdf, Training Parameters taken from Caffe open-source Framework are 224x224x3 Input, 1000x1 Output, FP16 with Shared Block-Exponents, All compute layers (incl. Fully Connected) done on the FPGA except for Softmax, Arria 10-1150 FPGA, -1 Speed Grade on Altera PCIe DevKit with x72 DDR4 @ 1333 MHz, Power measured through on-board power monitor (FPGA POWER ONLY), ACDS 16.1 Internal Builds + OpenCL SDK 16.1 Internal Build, Compute machine is an HP Z620 Workstation, Xeon E5-1660 at 3.3 GHz with 32GB RAM. The Xeon is not used for compute.

# CONFIGURATION DETAILS (CONT'D)

Optimization Notice: Intel's compilers may or may not optimize to the same degree for non-Intel microprocessors for optimizations that are not unique to Intel microprocessors. These optimizations include SSE2, SSE3, and SSSE3 instruction sets and other optimizations. Intel does not guarantee the availability, functionality, or effectiveness of any optimization on microprocessors not manufactured by Intel. Microprocessor-dependent optimizations in this product are intended for use with Intel microprocessors. Certain optimizations not specific to Intel microarchitecture are reserved for Intel microprocessors. Please refer to the applicable product User and Reference Guides for more information regarding the specific instruction sets covered by this notice. Notice Revision #20110804.

Results have been estimated or simulated using internal Intel analysis or architecture simulation or modeling, and provided to you for informational purposes. Any differences in your system hardware, software or configuration may affect your actual performance.

Software and workloads used in performance tests may have been optimized for performance only on Intel microprocessors.
Performance tests, such as SYSmark and MobileMark, are measured using specific computer systems, components, software, operations and functions. Any change to any of those factors may cause the results to vary. You should consult other information and performance tests to assist you in fully evaluating your contemplated purchases, including the performance of that product when combined with other products. For more complete information visit http://www.intel.com/performance/datacenter. Tested by Intel as of 14 June 2016. Configurations:

Faster and more scalable than GPU claim based on Intel analysis and testing

Up to 2.3x faster training per system claim based on AlexNet* topology workload (batch size = 1024) using a large image database running 4-nodes Intel Xeon Phi processor 7250 (16 GB MCDRAM, 1.4 GHz, 68 Cores) in Intel® Server System LADMP2312KXXX41, 96GB DDR4-2400 MHz, quad cluster mode, MCDRAM flat memory mode, Red Hat Enterprise Linux* 6.7 (Santiago), 1.0 TB SATA drive WD1003FZEX-00MK2A0 System Disk, running Intel® Optimized DNN Framework (internal development version) training 1.33 million images in 10.5 hours compared to 1-node host with four NVIDIA "Maxwell" GPUs training 1.33 million images in 25 hours (source: http://www.slideshare.net/NVIDIA/gtc-2016-opening-keynote slide 32).

Up to 38% better scaling efficiency at 32-nodes claim based on GoogLeNet deep learning image classification training topology using a large image database comparing one node Intel Xeon Phi processor 7250 (16 GB MCDRAM, 1.4 GHz, 68 Cores) in Intel® Server System LADMP2312KXXX41, DDR4 96GB DDR4-2400 MHz, quad cluster mode, MCDRAM flat memory mode, Red Hat* Enterprise Linux 6.7, Intel® Optimized DNN Framework with 87% efficiency to unknown hosts running 32 each NVIDIA Tesla* K20 GPUs with a 62% efficiency (Source: http://arxiv.org/pdf/1511.00175v2.pdf showing FireCaffe* with 32 NVIDIA Tesla* K20s (Titan Supercomputer*) running GoogLeNet* at 20x speedup over Caffe* with 1 K20).

Up to 6 SP TFLOPS based on the Intel Xeon Phi processor peak theoretical single-precision performance is preliminary and based on current expectations of cores, clock frequency and floating point operations per cycle. FLOPS = cores x clock frequency x floating-point operations per second per cycle

Up to 3x faster single-threaded performance claim based on Intel estimates of Intel Xeon Phi processor 7290 vs. coprocessor 7120 running XYZ workload.

Up to 2.3x faster training per system claim based on AlexNet* topology workload (batch size = 1024) using a large image database running 4-nodes Intel Xeon Phi processor 7250 (16 GB MCDRAM, 1.4 GHz, 68 Cores) in Intel® Server System LADMP2312KXXX41, 96GB DDR4-2400 MHz, quad cluster mode, MCDRAM flat memory mode, Red Hat Enterprise Linux* 6.7 (Santiago), 1.0 TB SATA drive WD1003FZEX-00MK2A0 System Disk, running Intel® Optimized DNN Framework, Intel® Optimized Caffe (internal development version) training 1.33 billion images in 10.5 hours compared to 1-node host with four NVIDIA "Maxwell" GPUs training 1.33 billion images in 25 hours (source: http://www.slideshare.net/NVIDIA/gtc-2016-opening-keynote slide 32).

Up to 38% better scaling efficiency at 32-nodes claim based on GoogLeNet deep learning image classification training topology using a large image database comparing one node Intel Xeon Phi processor 7250 (16 GB MCDRAM, 1.4 GHz, 68 Cores) in Intel® Server System LADMP2312KXXX41, DDR4 96GB DDR4-2400 MHz, quad cluster mode, MCDRAM flat memory mode, Red Hat* Enterprise Linux 6.7, Intel® Optimized DNN Framework with 87% efficiency to unknown hosts running 32 each NVIDIA Tesla* K20 GPUs with a 62% efficiency (Source: http://arxiv.org/pdf/1511.00175v2.pdf showing FireCaffe* with 32 NVIDIA Tesla* K20s (Titan Supercomputer*) running GoogLeNet* at 20x speedup over Caffe* with 1 K20).

Up to 50x faster training on 128-node as compared to single-node based on AlexNet* topology workload (batch size = 1024) training time using a large image database running one node Intel Xeon Phi processor 7250 (16 GB MCDRAM, 1.4 GHz, 68 Cores) in Intel® Server System LADMP2312KXXX41, 96GB DDR4-2400 MHz, quad cluster mode, MCDRAM flat memory mode, Red Hat Enterprise Linux* 6.7 (Santiago), 1.0 TB SATA drive WD1003FZEX-00MK2A0 System Disk, running Intel® Optimized DNN Framework, training in 39.17 hours compared to 128-node identically configured with Intel® Omni-Path Host Fabric Interface Adapter 100 Series 1 Port PCIe x16 connectors training in 0.75 hours. Contact your Intel representative for more information on how to obtain the binary. For information on workload, see https://papers.nips.cc/paper/4824-large image_database-classification-with-deep-convolutional-neural-networks.pdf.

Up to 30x software optimization improvement claim based on customer CNN training workload running 2S Intel® Xeon® processor E5-2680 v3 running Berkeley Vision and Learning Center* (BVLC) Caffe + OpenBlas* library and then run tuned on the Intel® Optimized Caffe (internal development version) + Intel® Math Kernel Library (Intel® MKL).