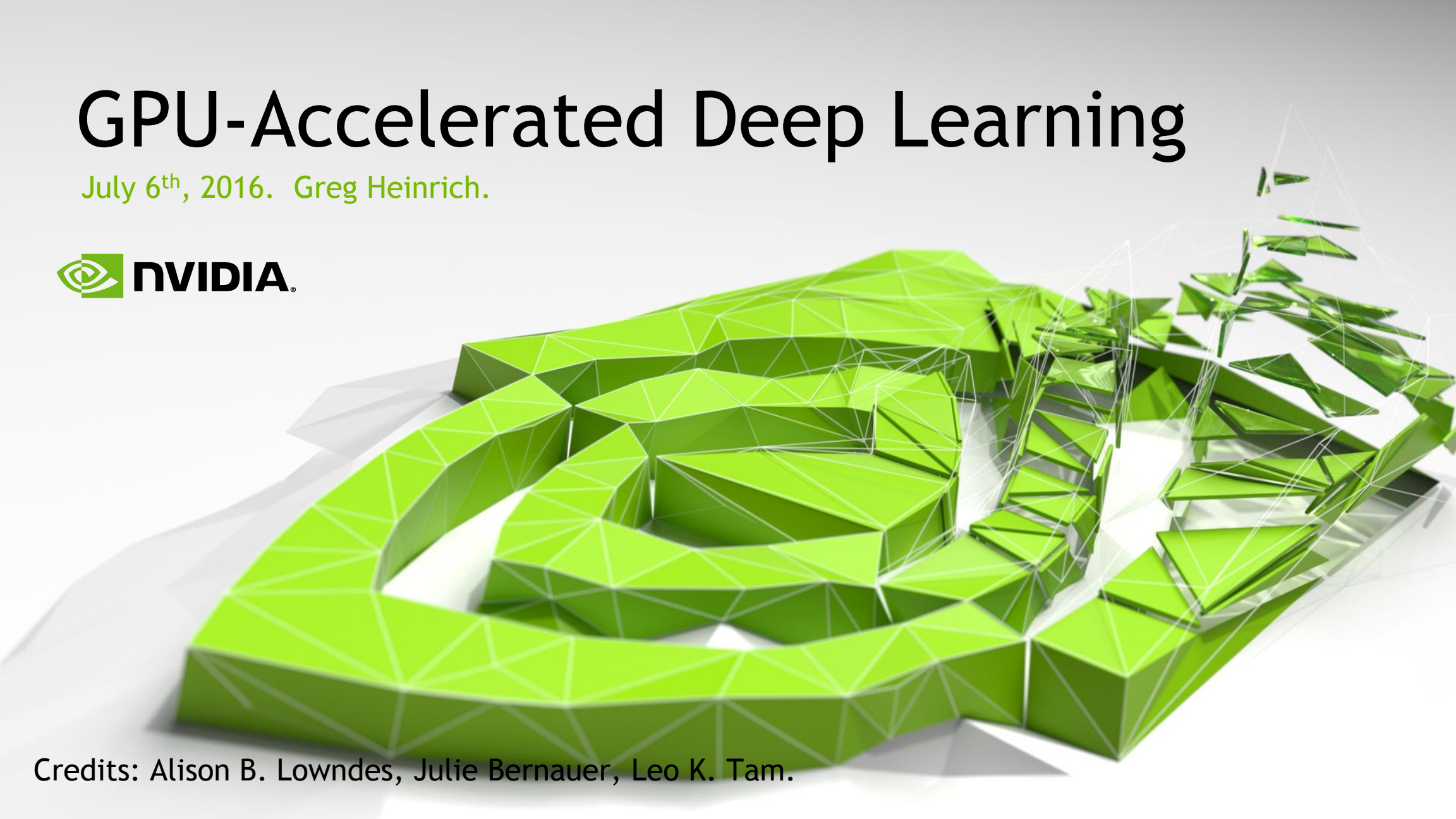
GPU-Accelerated Deep Learning

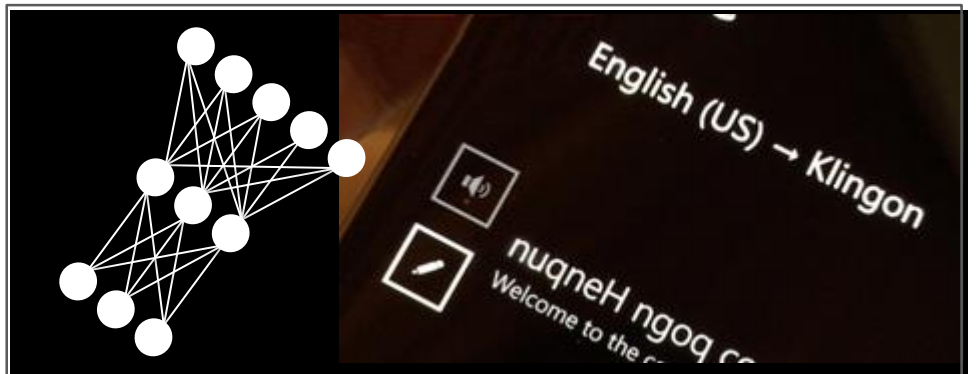
July 6th, 2016. Greg Heinrich.



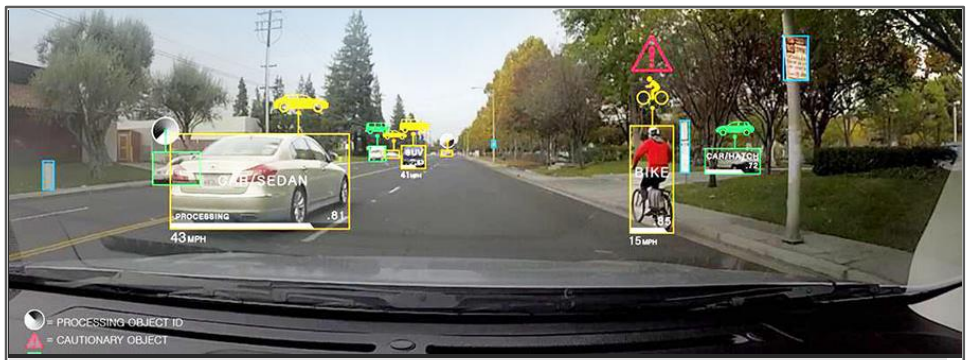
Credits: Alison B. Lowndes, Julie Bernauer, Leo K. Tam.

PRACTICAL DEEP LEARNING EXAMPLES

Image Classification, Object Detection, Localization,
Action Recognition, Scene Understanding



Speech Recognition, Speech Translation,
Natural Language Processing



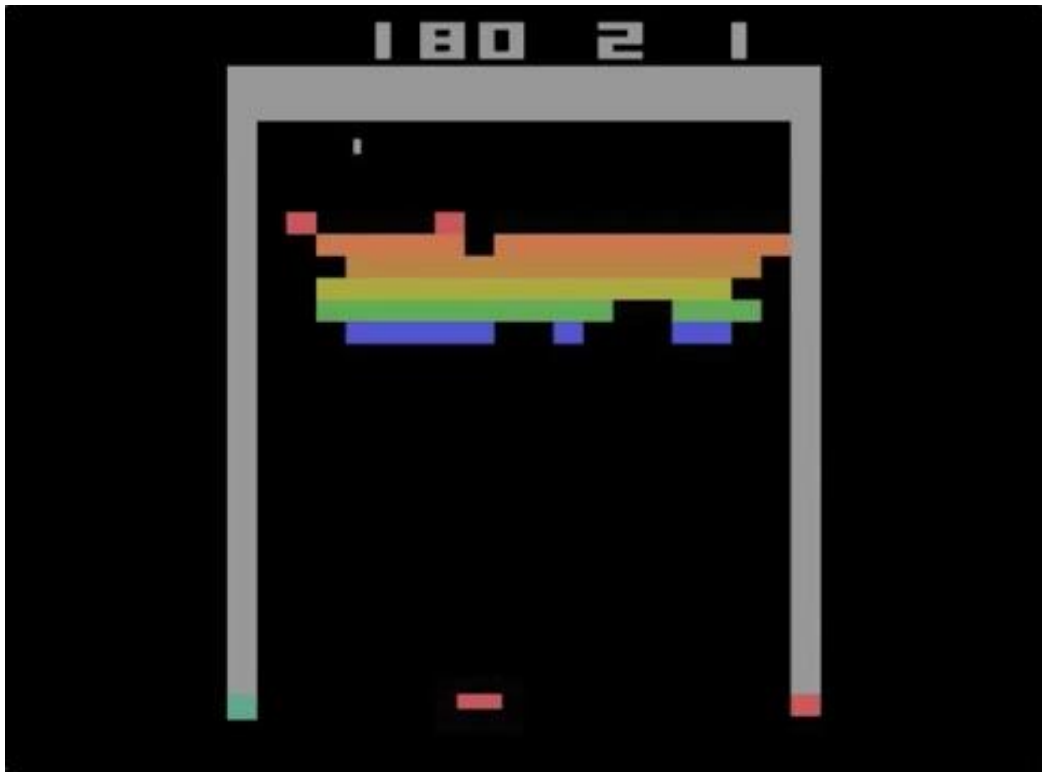
Pedestrian Detection, Traffic Sign Recognition

Breast Cancer Cell Mitosis Detection,
Volumetric Brain Image Segmentation

DEEP LEARNING IS FUN

DQN: deep Q-learning network

Mastering Breakout



Video:
https://www.youtube.com/watch?v=TmPfTpjtdgg

DEEP LEARNING IS FUN

Character-level text generator

Have a neural network read Linux kernel source code and see what beautiful code it generates:

http://karpathy.github.io/2015/05/21/rnn-effectiveness/

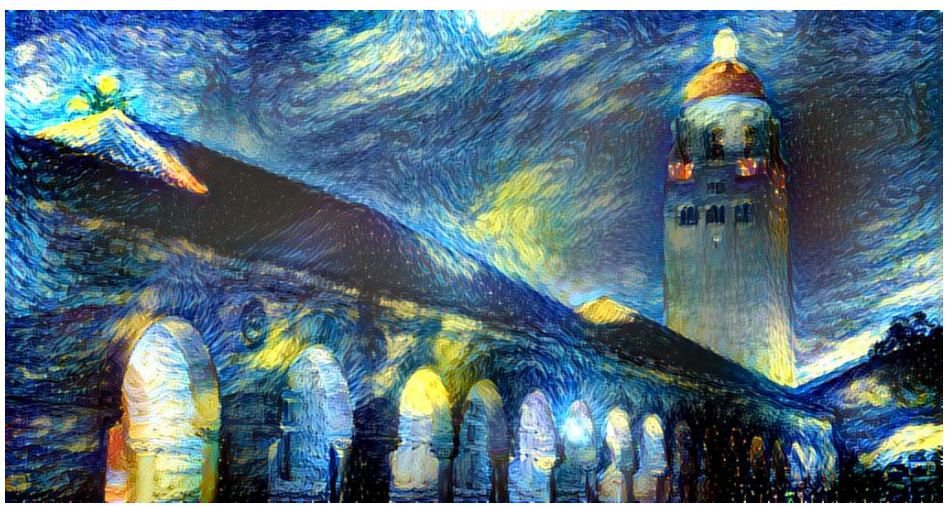
```
/*
 * Increment the size file of the new incorrect UI_FILTER group information
 * of the size generatively.
 */
static int indicate_policy(void)
{
  int error;
  if (fd == MARN_EPT) {
    /*
     * The kernel blank will coeld it to userspace.
     */
    if (ss->segment < mem_total)
      unblock_graph_and_set_blocked();
    else
      ret = 1;
    goto bail;
  }
  segaddr = in_SB(in.addr);
  selector = seg / 16;
  setup_works = true;
  for (i = 0; i < blocks; i++) {
    seq = buf[i++];
    bpf = bd->bd.next + i * search;
    if (fd) {
      current = blocked;
    }
  }
  rw->name = "Getjbbregs";
  bprm_self_clearl(&iv->version);
  regs->new = blocks[(BPF_STATS << info->historidac)] | PFMR_CLOBATHINC_SECONDS << 12;
  return segtable;
}
```

DEEP LEARNING IS FUN

Neural style

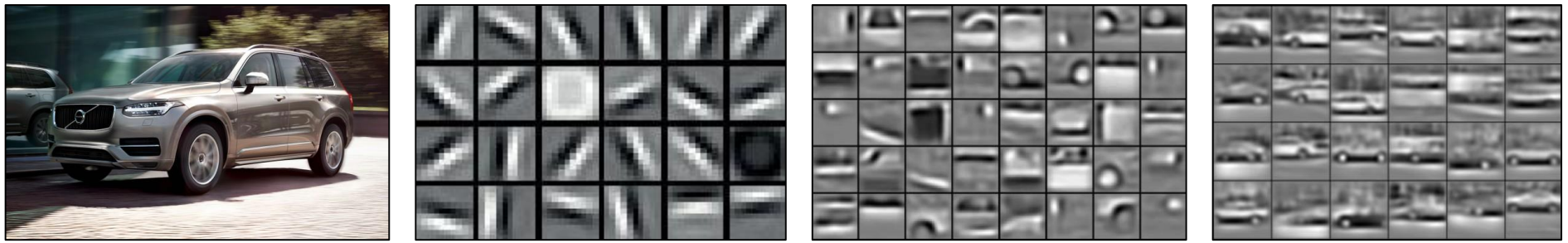
+

=

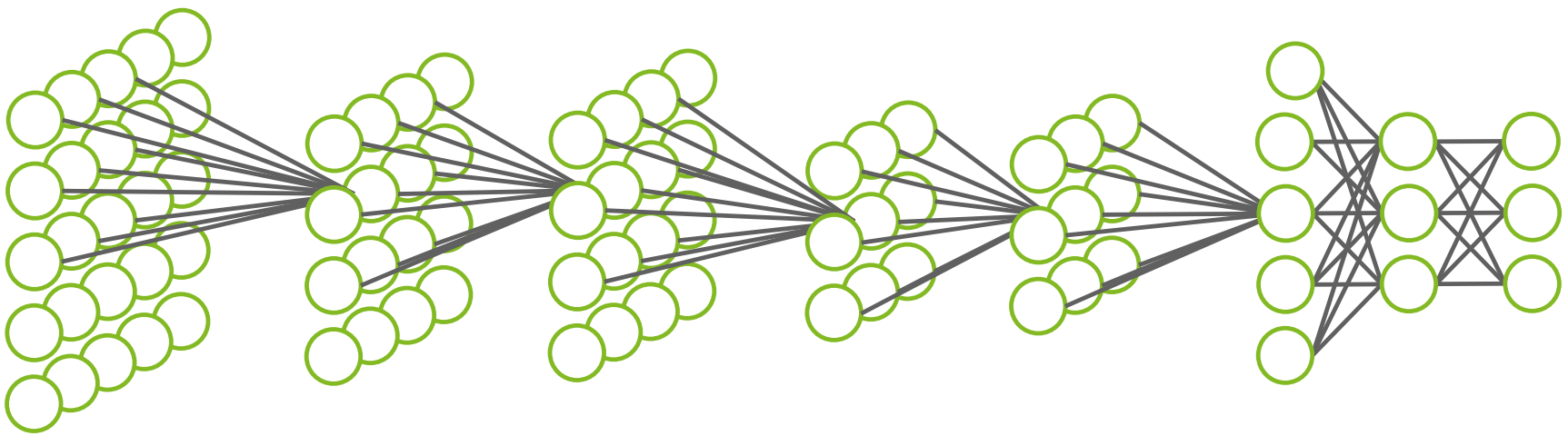


https://github.com/jcjohnson/neural-style/

WHAT IS DEEP LEARNING?



Image



"Volvo XC90"

Typical Network (VGG-16): 138M parameters, 15B connections, training requires billions of GFLOP

Image source: "Unsupervised Learning of Hierarchical Representations with Convolutional Deep Belief Networks" ICML 2009 & Comm. ACM 2011. Honglak Lee, Roger Grosse, Rajesh Ranganath, and Andrew Ng.

THE NEED TO GO PARALLEL

We can increase the number of transistors but we cannot increase frequency

The mother of all equations:

$$P_{dynamic} \propto C.{V_{dd}}^2.f$$

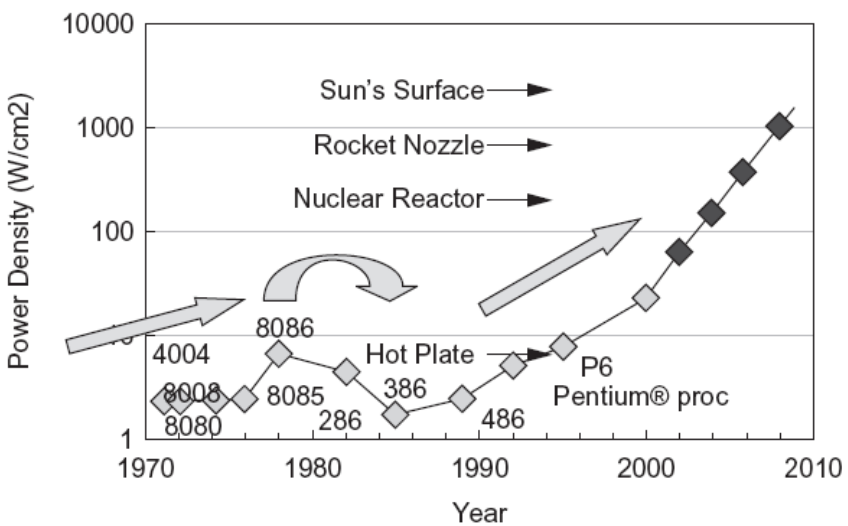
$$f_{max} \propto \frac{(V_{dd}-V_{th})^{\alpha}}{V_{dd}},\ \alpha<2$$

Problem: power grows exponentially with frequency

Solution: parallelism!

Intel VP Patrick Gelsinger (ISSCC 2001)

"If scaling continues at present pace, by 2005, high speed processors would have power density of nuclear reactor, by 2010, a rocket nozzle, and by 2015, surface of sun



GPU PARALLELISM

Single Instruction Multiple Threads

Single instruction flow:

save transistors and power

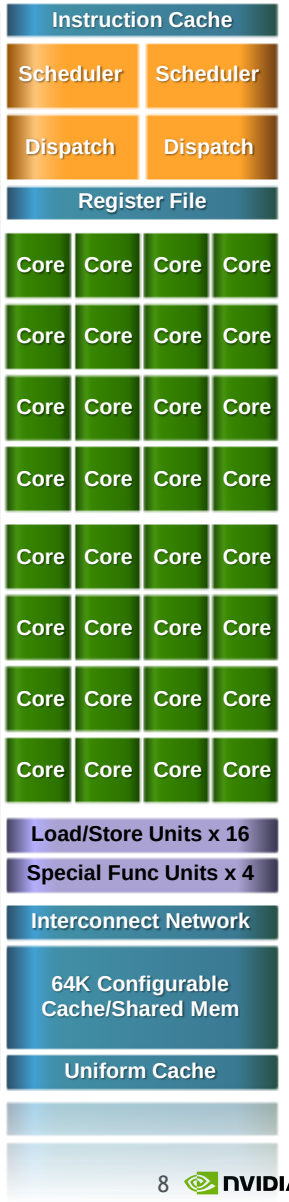
Multiple threads:

operate on multiple data concurrently

split workload, keep frequency low

Latency hiding:

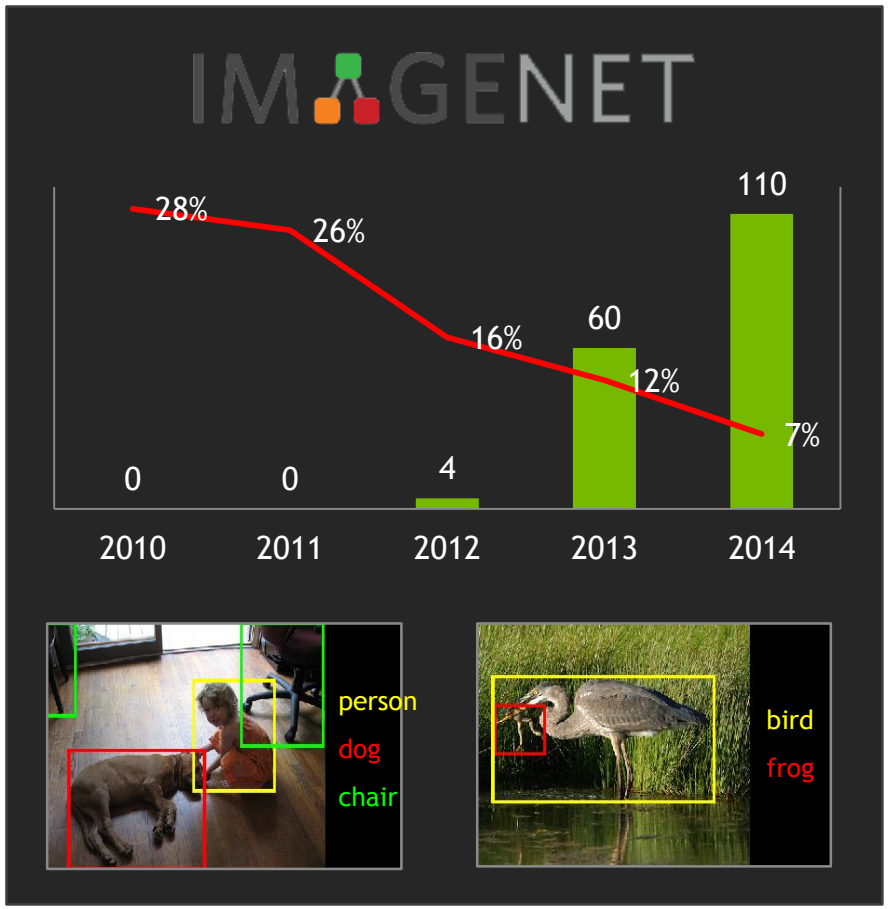
oversubscription hides memory latency, delivers high throughput



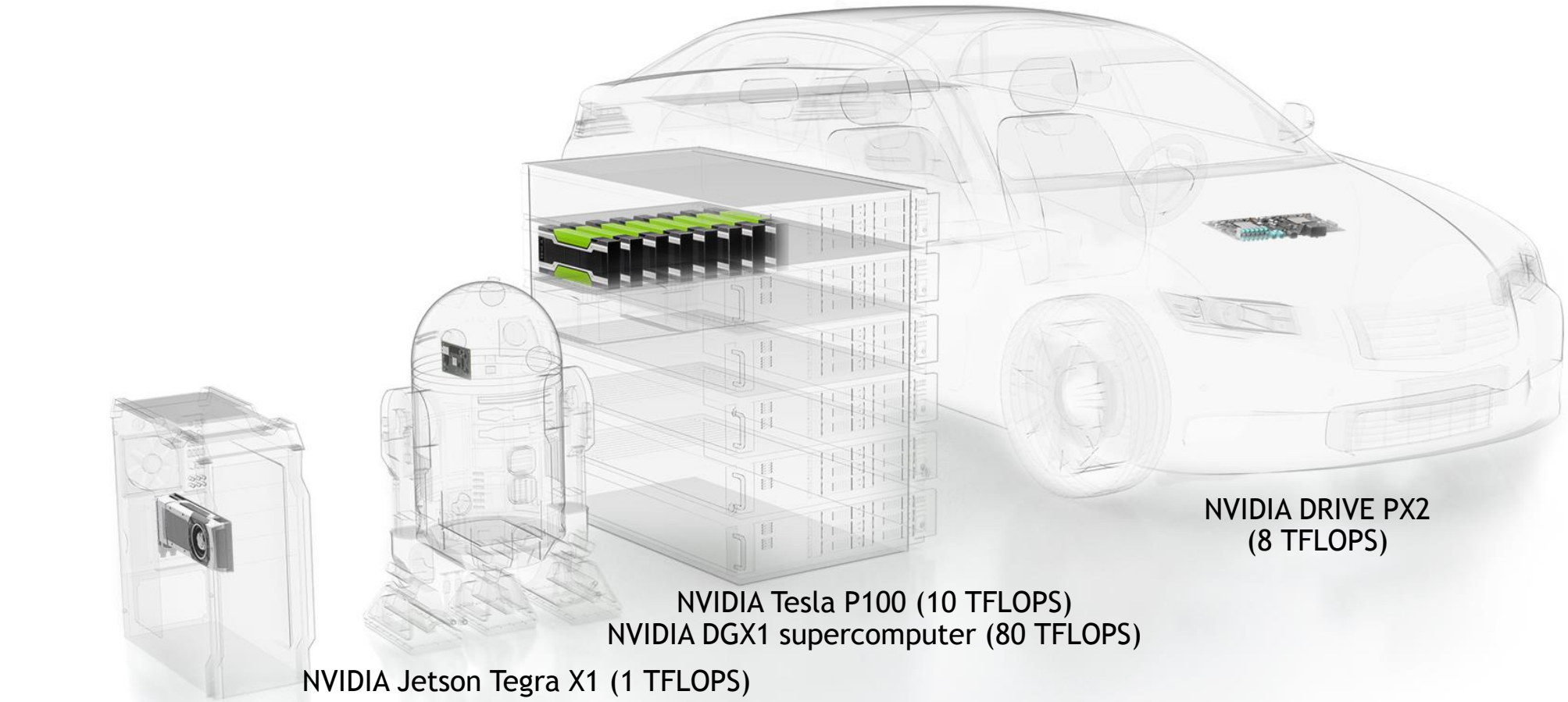
WHY ARE GPUS GOOD FOR DEEP LEARNING?

	Neural Networks	GPUs
Inherently Parallel	✓	✓
Matrix Operations	✓	✓
FLOPS	✓	✓

- GPUs deliver --
  - same *or better* prediction accuracy
  - faster results
  - smaller footprint
  - lower power



DEEP LEARNING EVERYWHERE



NVIDIA DRIVE PX2
(8 TFLOPS)

NVIDIA Tesla P100 (10 TFLOPS)
NVIDIA DGX1 supercomputer (80 TFLOPS)

NVIDIA Jetson Tegra X1 (1 TFLOPS)

NVIDIA GeForce Titan X (7 TFLOPS)

CUDA

Framework to Program NVIDIA GPUs

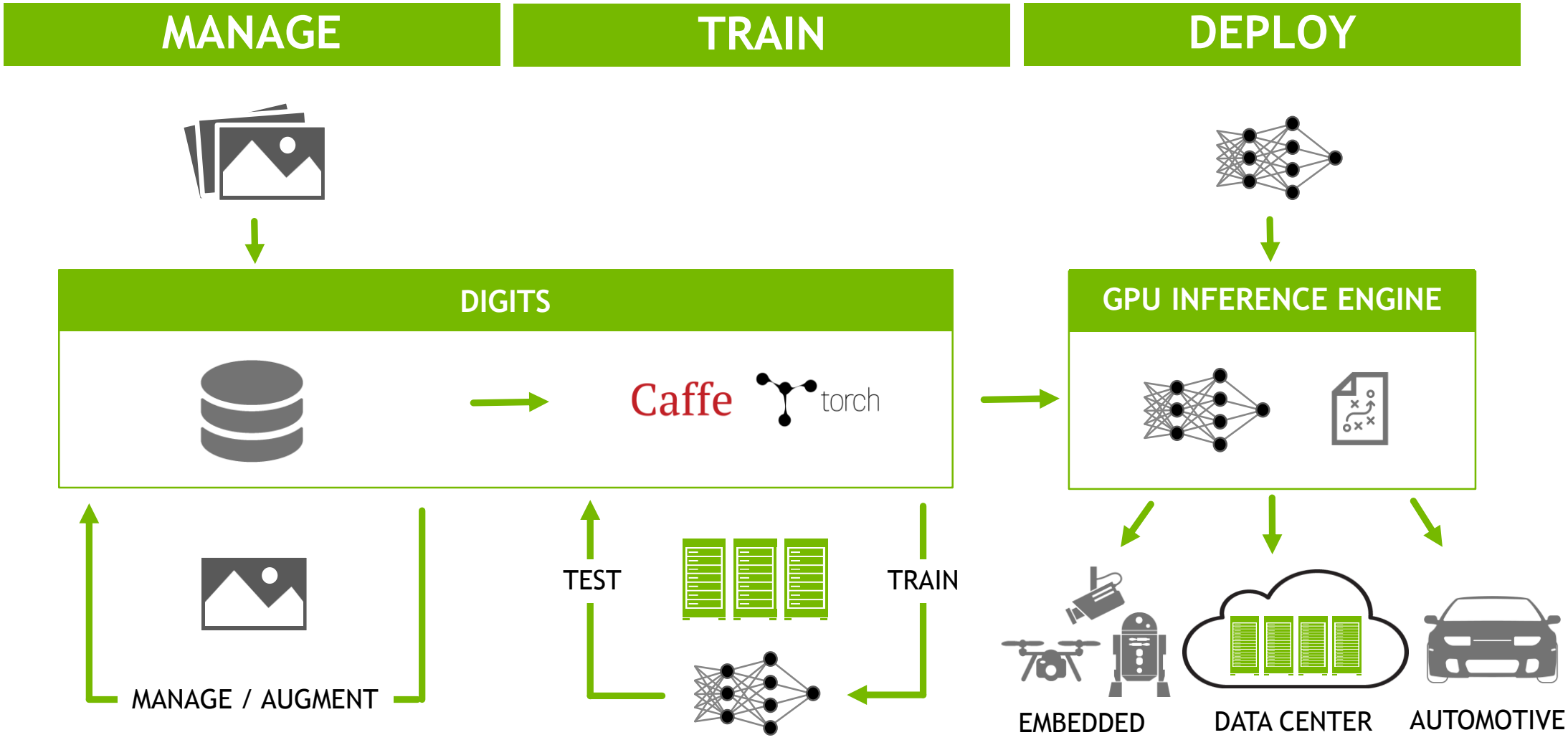
A simple sum of two vectors (arrays) in C

```
void vector_add(int n, const float *a, const float *b, float *c)
{
  for( int idx = 0 ; idx < n ; ++idx )
    c[idx] = a[idx] + b[idx];
}
```

GPU friendly version in CUDA

```
__global__ void vector_add(int n, const float *a, const float *b, float *c)
{
  int idx = blockIdx.x*blockDim.x + threadIdx.x;
  if( idx < n )
    c[idx] = a[idx] + b[idx];
}
```

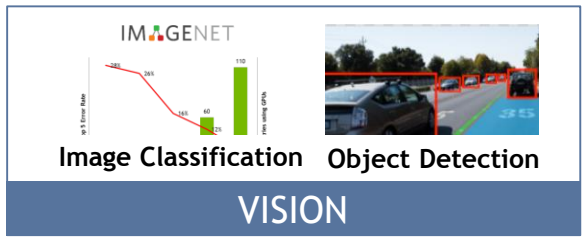
A COMPLETE COMPUTE PLATFORM



NVIDIA DEEP LEARNING SDK

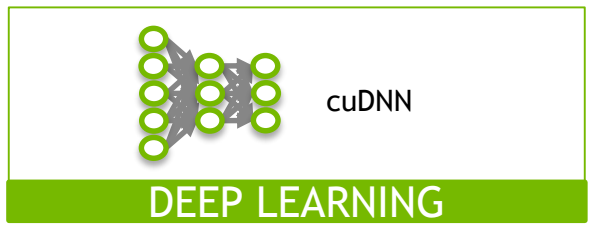
High Performance GPU-Acceleration for Deep Learning

APPLICATIONS



FRAMEWORKS

DEEP LEARNING SDK

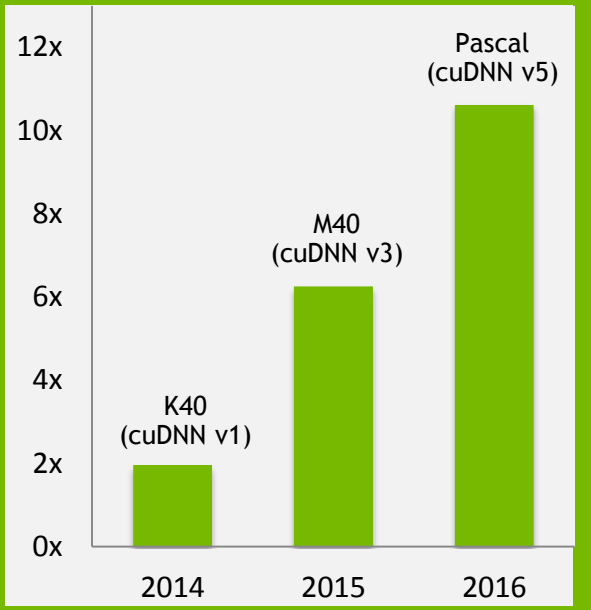


NVIDIA CUDNN

Building blocks for accelerating deep neural networks on GPUs

- High performance deep neural network training and inference
- Accelerates Caffe, CNTK, Tensorflow, Theano, Torch
- Performance continues to improve over time

developer.nvidia.com/*cudnn*



AlexNet training throughput based on 20 iterations, CPU: 1x E5-2680v3 12 Core 2.5GHz.

"NVIDIA has improved the speed of cuDNN with each release while extending the interface to more operations and devices at the same time."

— Evan Shelhamer, Lead Caffe Developer, UC Berkeley

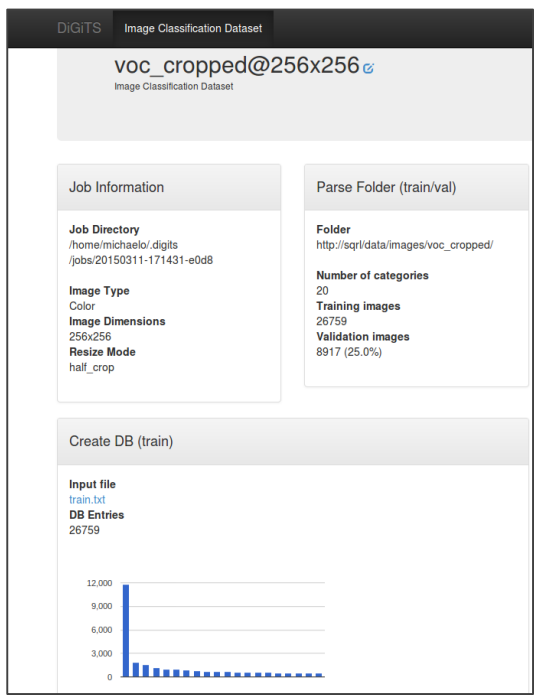
GPU-ACCELERATED DEEP LEARNING FRAMEWORKS

	CAFFE	TORCH	THEANO	TENSORFLOW	CHAINER
	Deep Learning Framework	Scientific Computing Framework	Math Expression Compiler	Deep Learning Framework	Deep Learning Framework
cuDNN	5	5	4	5	5
Multi-GPU	✓	✓	✓	✓	✓
Multi-Node	~	✓	✗	✓	✓
License	BSD-2	BSD	BSD	Apache 2.0	MIT
Interface(s)	Text-based definition files, Python, MATLAB	Python, Lua, MATLAB	Python	Python, C++	Python
Embedded	✓	✓	✗	✓	✗

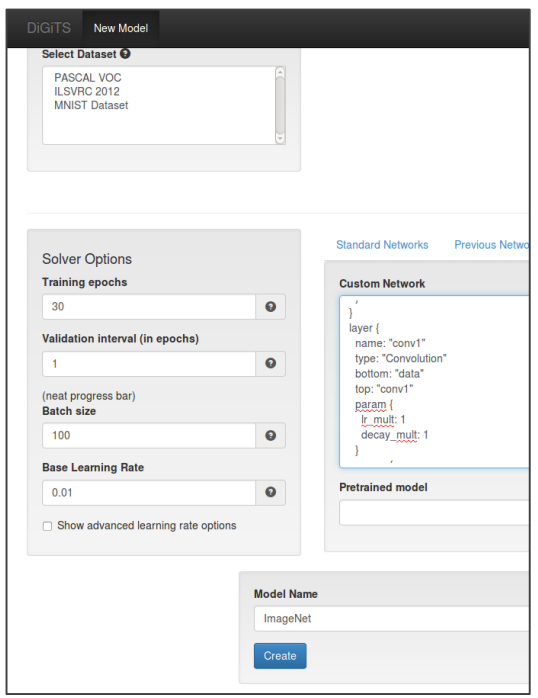
NVIDIA DIGITS

Interactive Deep Learning GPU Training System

Process Data

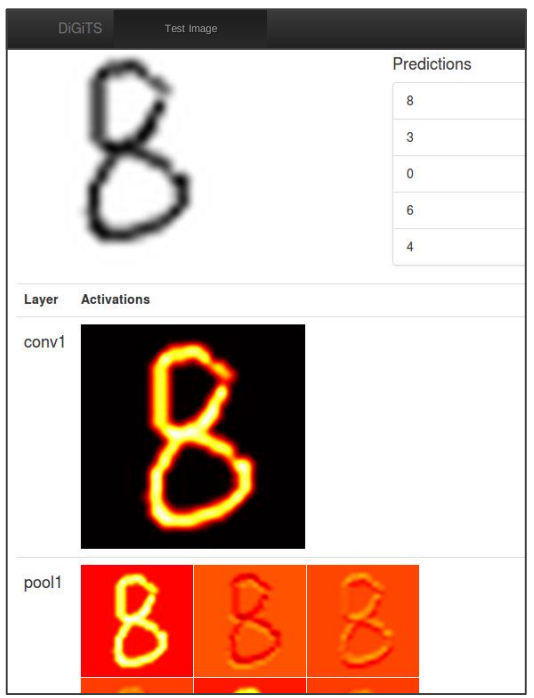


Configure DNN



Monitor Progress

Visualize Layers



developer.nvidia.com/digits