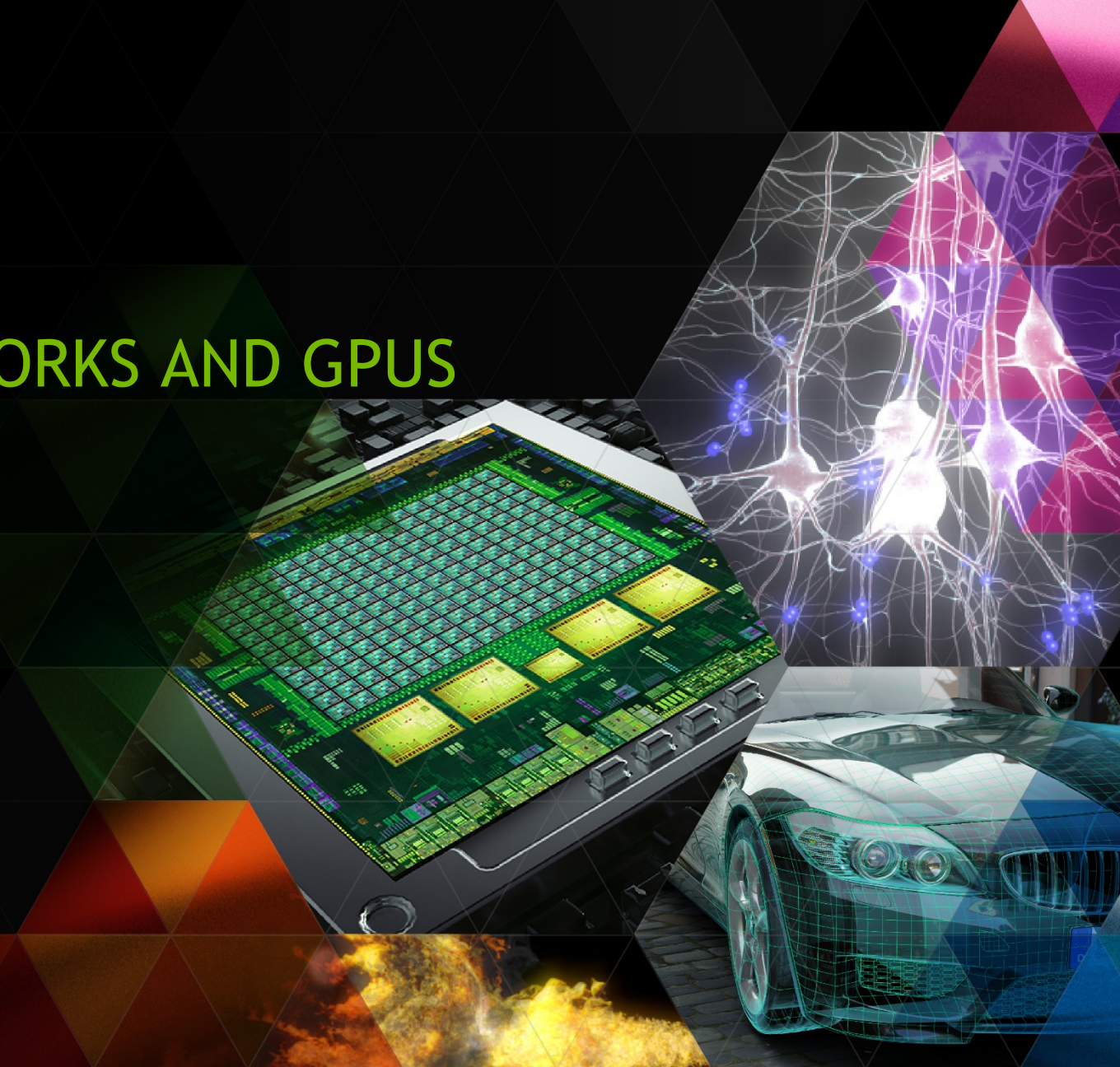
NVIDIA

DEEP NEURAL NETWORKS AND GPUS

Julien Demouth



GAMING

DESIGN

ENTERPRISE VIRTUALIZATION

HPC & CLOUD SERVICE PROVIDERS

AUTONOMOUS MACHINES

PC

DATA CENTER

MOBILE

THE WORLD LEADER IN VISUAL COMPUTING

GPU Computing

GPU COMPUTING

Run Computations on GPUs

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];
}
```

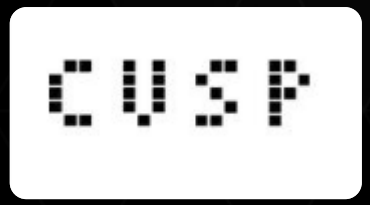
- GPU friendly version in CUDA

```
__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];
}
```

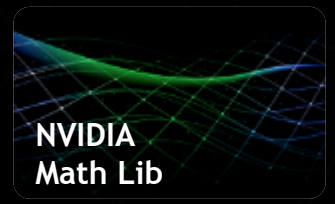
GPU ACCELERATED LIBRARIES

"Drop-in" Acceleration for Your Applications

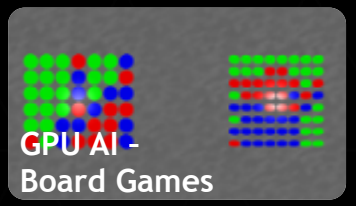
Linear Algebra
FFT, BLAS,
SPARSE, Matrix, cuSolver



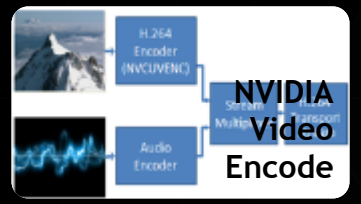
Numerical & Math
RAND, Statistics



Data Struct. & AI
Sort, Scan, Zero Sum



Visual Processing
Image & Video



Deep Neural Networks

ACCELERATING INSIGHTS

“Now You Can Build Google’s $1M Artificial Brain on the Cheap”

WIRED

GOOGLE DATACENTER

1,000 CPU Servers
2,000 CPUs • 16,000 cores

600 kWatts
$5,000,000

STANFORD AI LAB

3 GPU-Accelerated Servers
12 GPUs • 18,432 cores

4 kWatts
$33,000

Deep learning with COTS HPC systems, A. Coates, B. Huval, T. Wang, D. Wu, A. Ng, B. Catanzaro ICML 2013

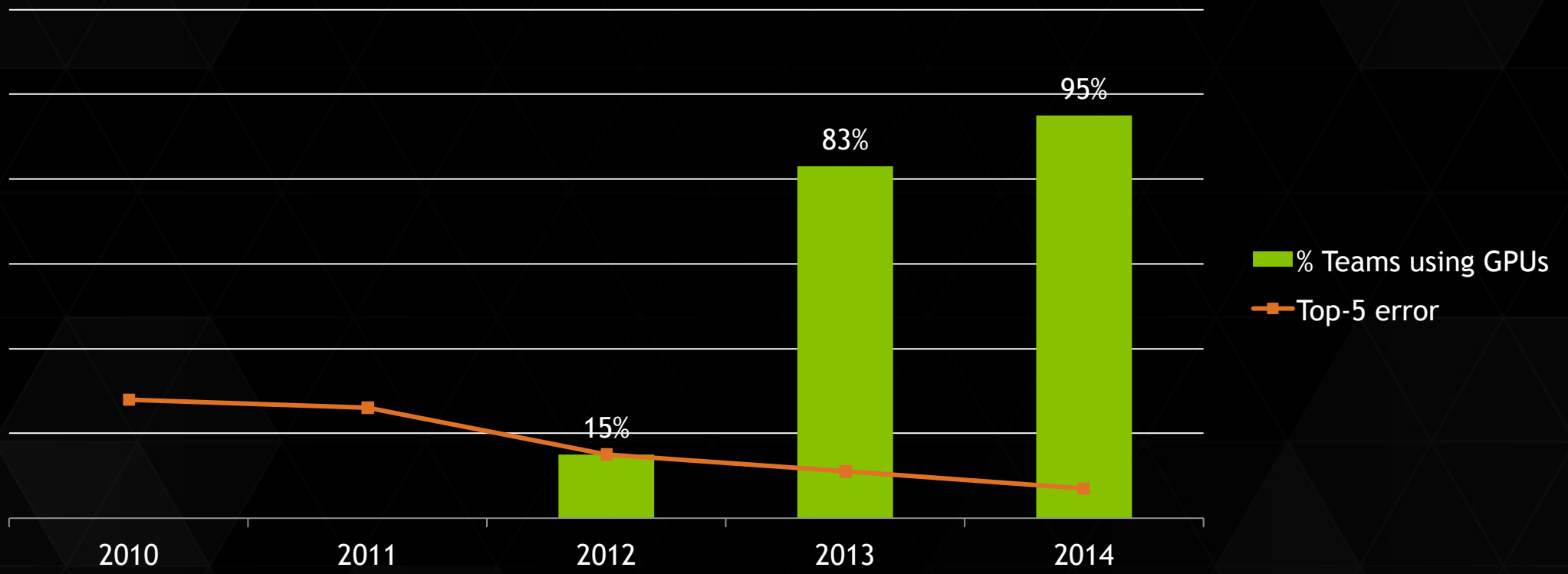
IMAGE CLASSIFICATION

Recognize not only Cats

“it is a baby”

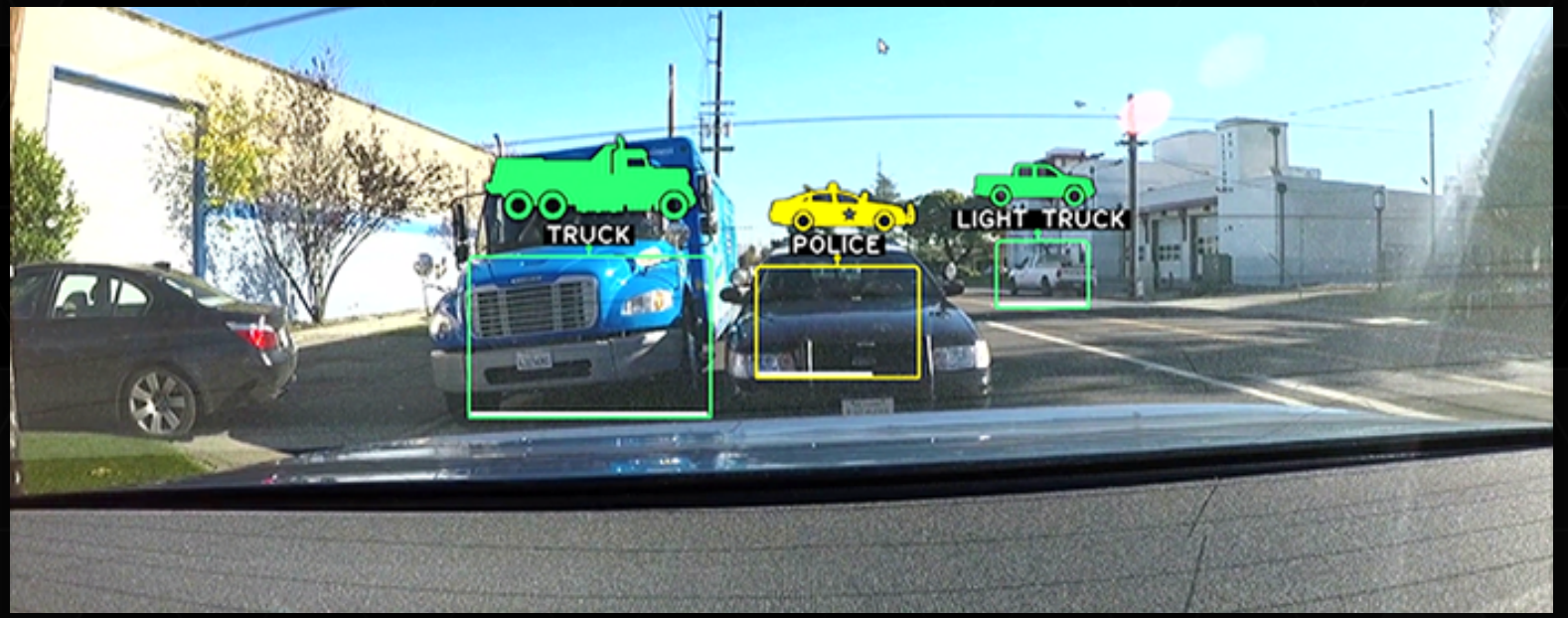
NATURAL IMAGE CLASSIFICATION

ImageNet: results for 2010-2014



ADVANCED DRIVER ASSISTANCE SYSTEMS

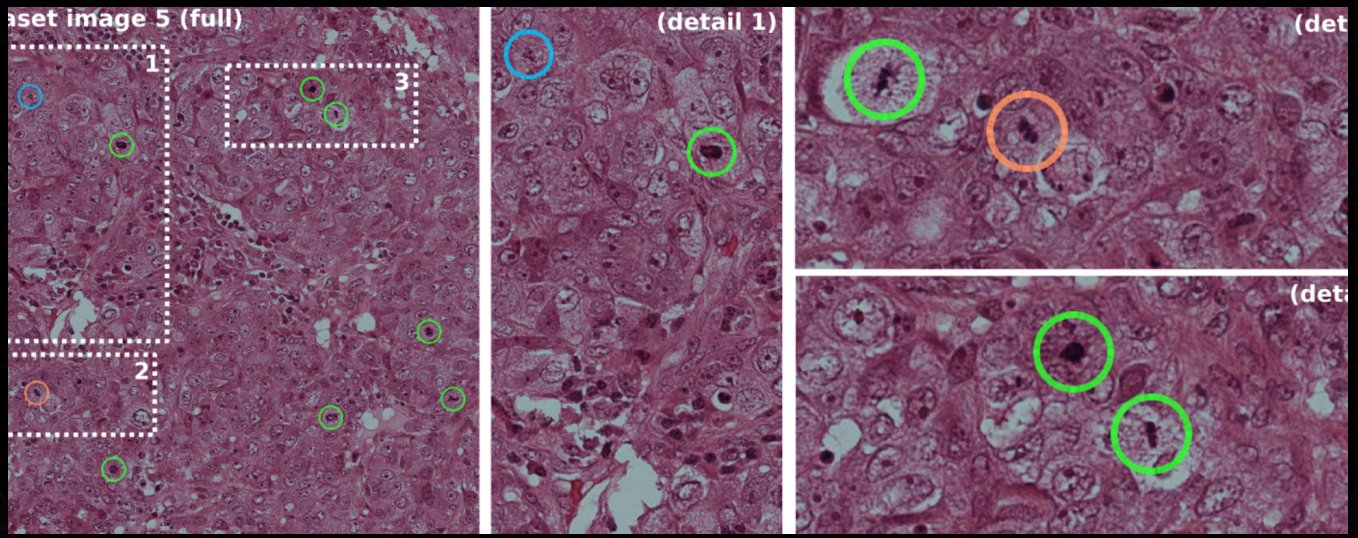
Localization and Classification



https://youtu.be/zsVsUvx8ieo

CANCER SCREENING

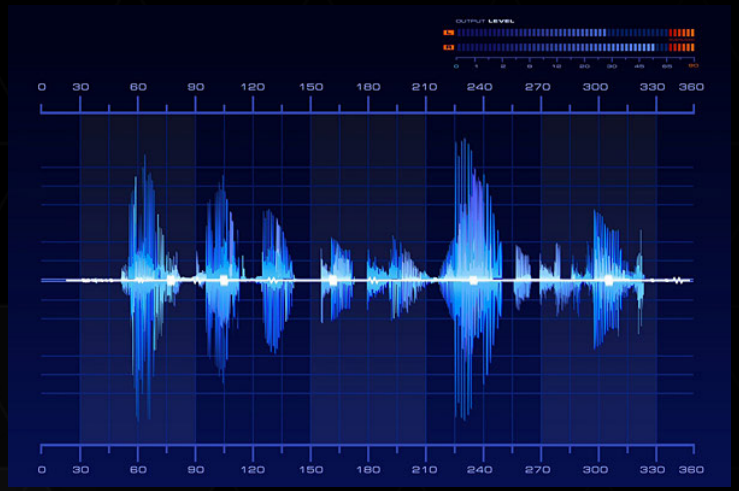
Mitosis Detecion



Ciresan et al. *Mitosis Detection in Breast Cancer Histology Images with Deep Neural Networks*, 2013

INSTANT SPEECH TRANSLATION

See Skype for Example



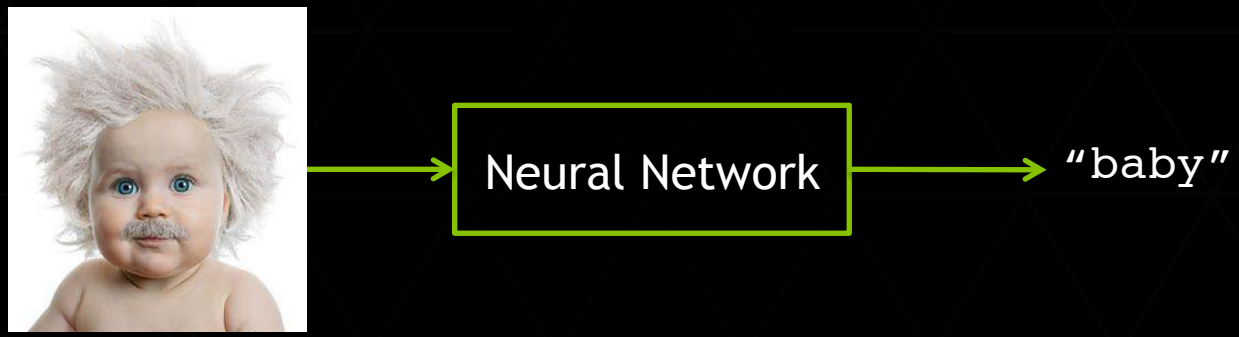
http://blogs.skype.com/2014/12/15/skype-translator-preview-an-exciting-journey-to-a-new-chapter-in-communication

More Details

NEURAL NETWORK

Inference

- A neural network takes an input and computes a result from it



- It is the *inference*

NEURAL NETWORK

Training

- The network is parameterized by weights

- The weights have to be *learned*. It is the *training*

"baby"

"baby"

"baby"

NEURAL NETWORK

Loss

- Labeled training set (Supervised learning)
  - For each sample, we know the ground truth

, “baby”

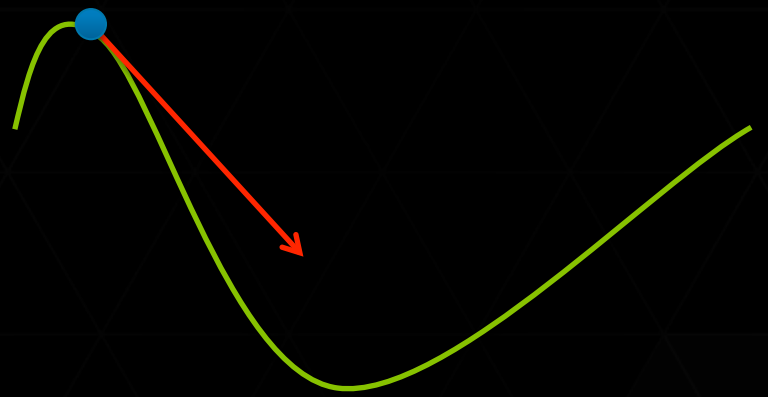
- *Loss* function
  - Measures how badly the network infers (i.e. high if the network does poorly)
  - Example of loss function: Mean of square errors (MSE)

$$\frac{1}{m}\sum_{i=1}^{m}\|y_i - \mathrm{net}(x_i)\|^2$$

NEURAL NETWORK

Loss Minimization

- Minimize the loss using a gradient descent
  - For example, stochastic gradient descent with momentum and learning rate

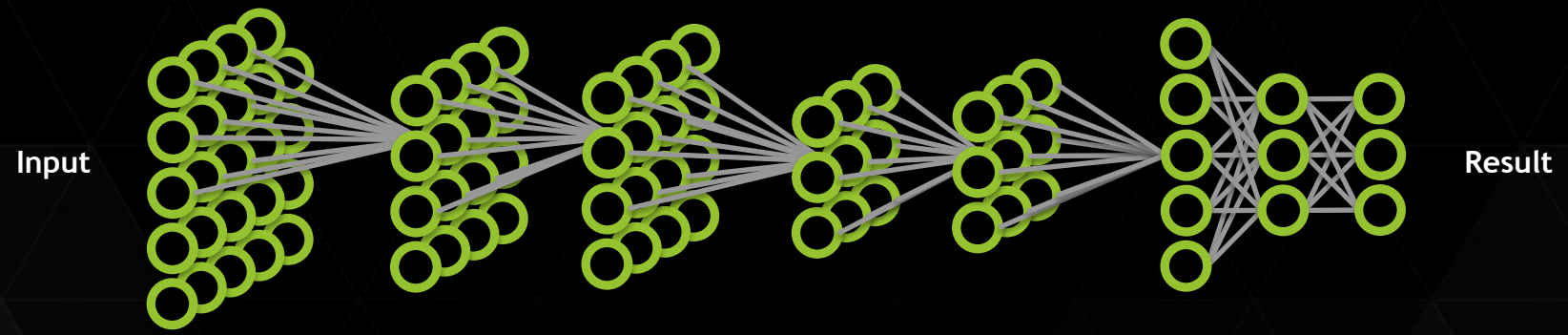


- Gradient computed with respect to the weights of the network

DEEP NEURAL NETWORK

Architecture

- Neurons are grouped into layers (deep = several hidden layers)
  - Different types of layers: Fully connected, convolutional, …

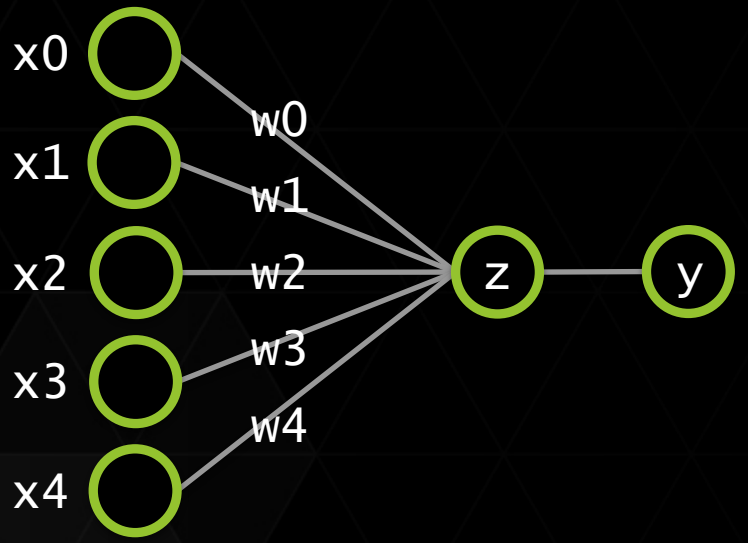


- The weights are on the connections between the neurons

NEURAL NETWORK

Simple Neuron

- A neuron is connected to other neurons on the “previous” layer




```
z = w0*x0 + w1*x1 + w2*x2 + w3*x3 + w4*x4

y = h(z)

h(x) = max(x, 0) // Relu
```

- Activation function (h): Tanh, sigmoid, ReLU, maxout

GPU Implementation

cuDNN

Building Blocks

- A kind of “BLAS” for deep neural networks
  - Core routines to implement a deep neural network
- Developed by NVIDIA
  - See https://developer.nvidia.com/cuDNN
- Integrated into several frameworks (Theano, Torch, Caffe, yours?)

cuDNN

GEMM

- Many operations in deep neural networks expressed as GEMM
  - GEMM: Dense matrix product – Extremely efficient on a GPU
- Writing an extremely fast GEMM is hard
  - See: https://github.com/NervanaSystems/maxas
  - Fast GEMM is written directly in assembly language (using a generator)

RESOURCES

Where to Learn

- Andrej Kaparthy's class on Deep Neural Networks
  - http://cs231n.github.io
- Sander Dieleman's blog posts
  - http://benanne.github.io/posts/
- The research papers on ArXiV (or elsewhere): http://arxiv.org