

MatConvNet

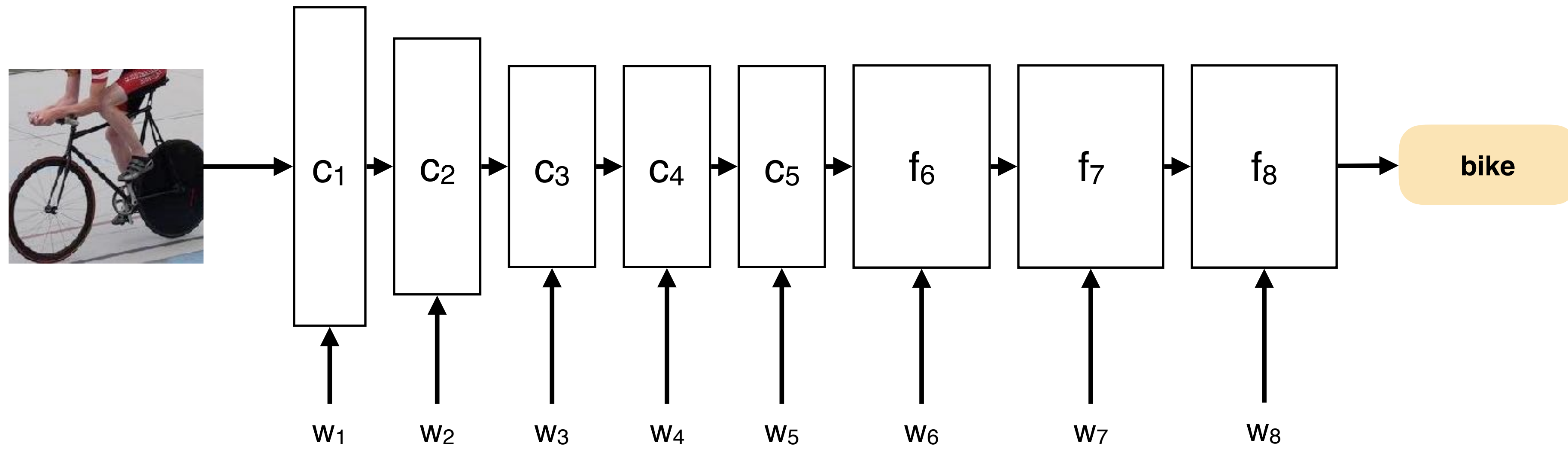
Deep learning research in MATLAB

Dr Andrea Vedaldi
University of Oxford

MATLAB Expo, October 2017

Deep learning: a magic box

Pixels & labels in, model parameters out





Confounding factors

Fonts

Distortions

Colors

Blur

Shadows

Borders

Textures

Sizes

...

Visual search

Fast retrieval, learn concepts on the fly

BBC NEWS SEARCH

Hillary Clinton

+ BBC News



Sign in



Objects/Scenes

Instances

People

Next >

Search results page 1 of 100 (5,000 results)

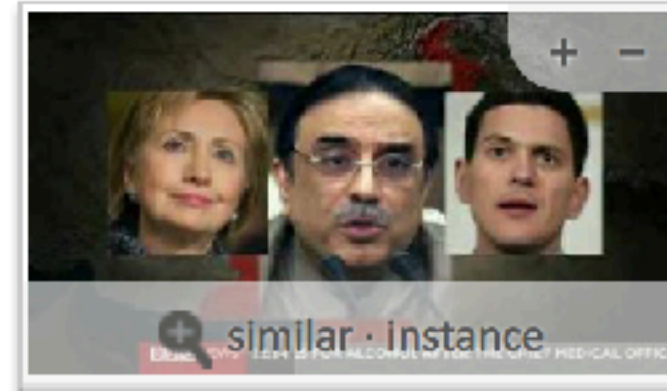
Ranked in 1.81s



BBC News at Ten



BBC News at Ten



BBC News at Ten



Newsnight



BBC Weekend News



BBC News at Ten



BBC News



The Record Europe



By the People: The...



BBC News at Ten



The Record Europe



BBC News at Ten



The Record Europe



The Record Europe



The Record Europe



BBC Weekend News



BBC News

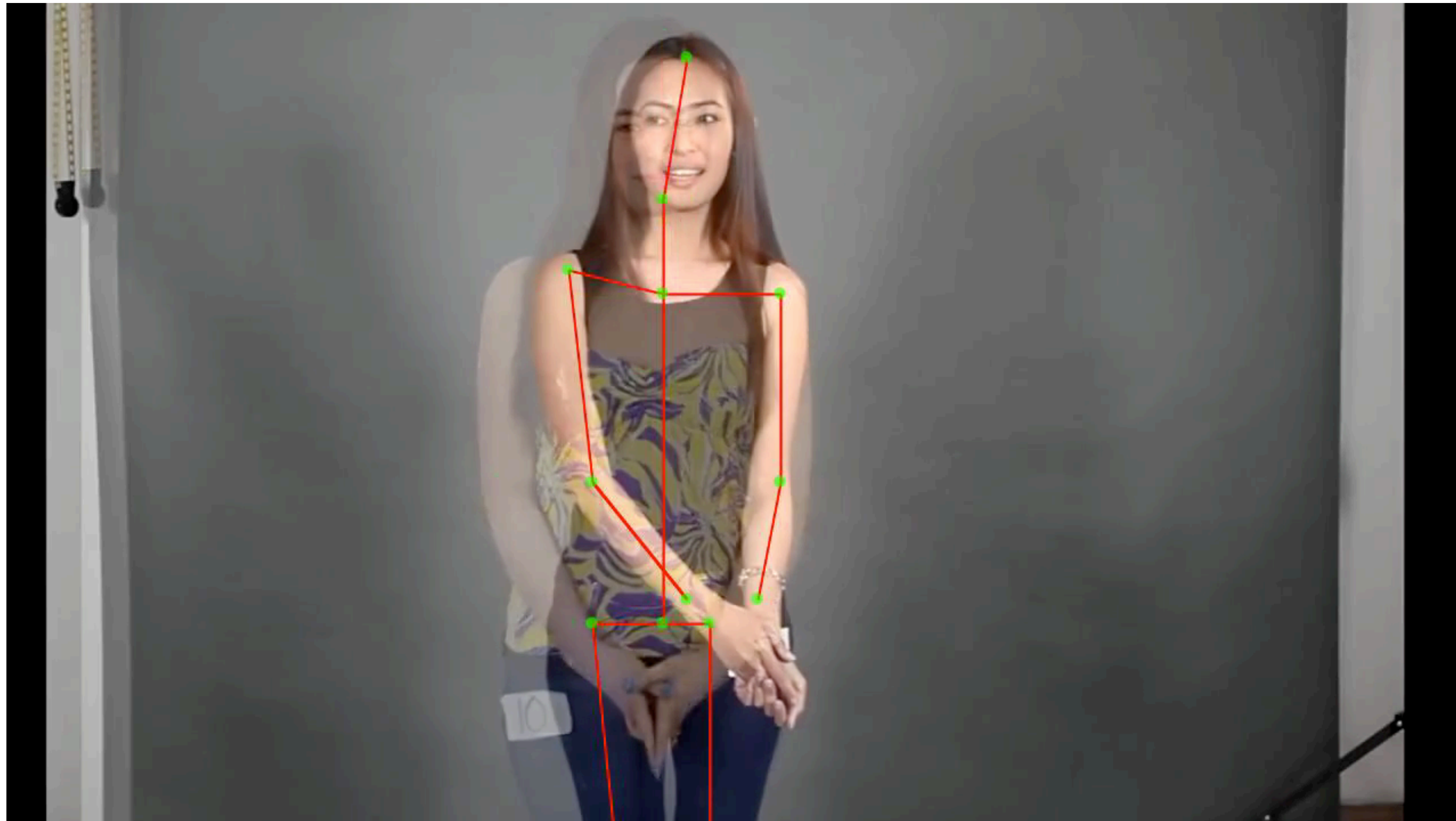


The Record Europe

Single shot (feed forward) detector



Dense part and keypoint labelling

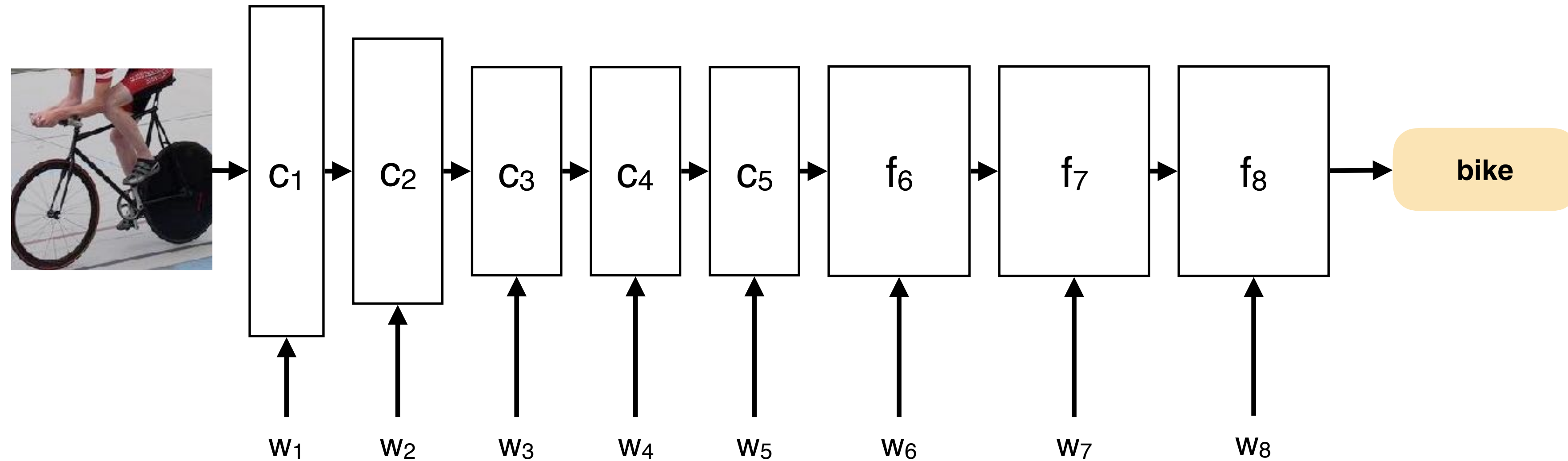


Real-time visual style transfer

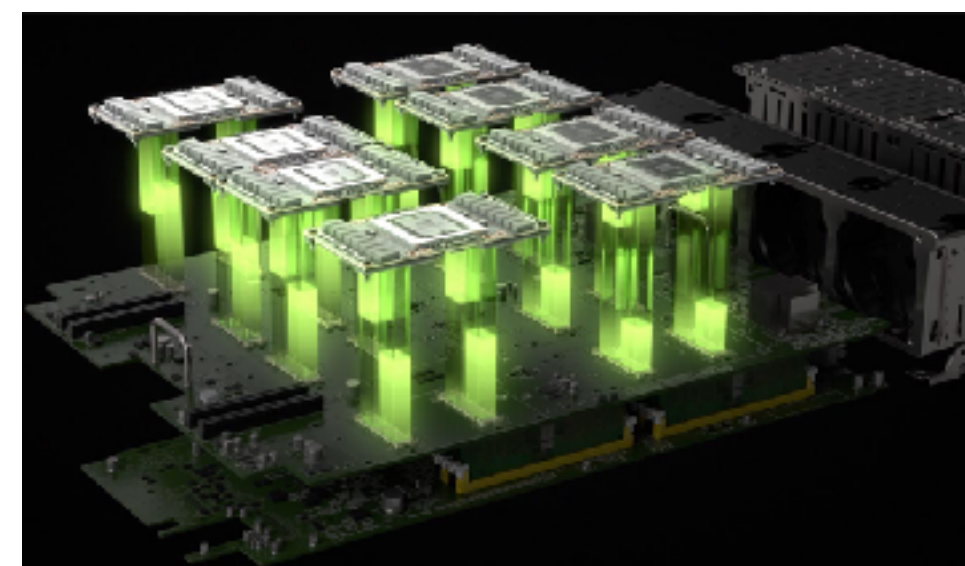


Demos

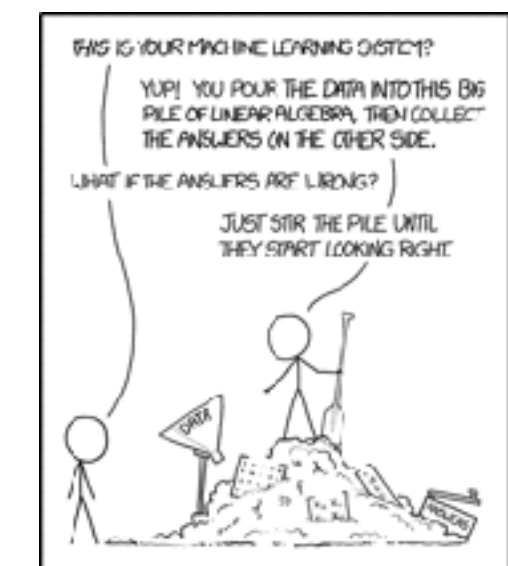
Big Data + GPU Compute + Optimisation



A few million
labelled images

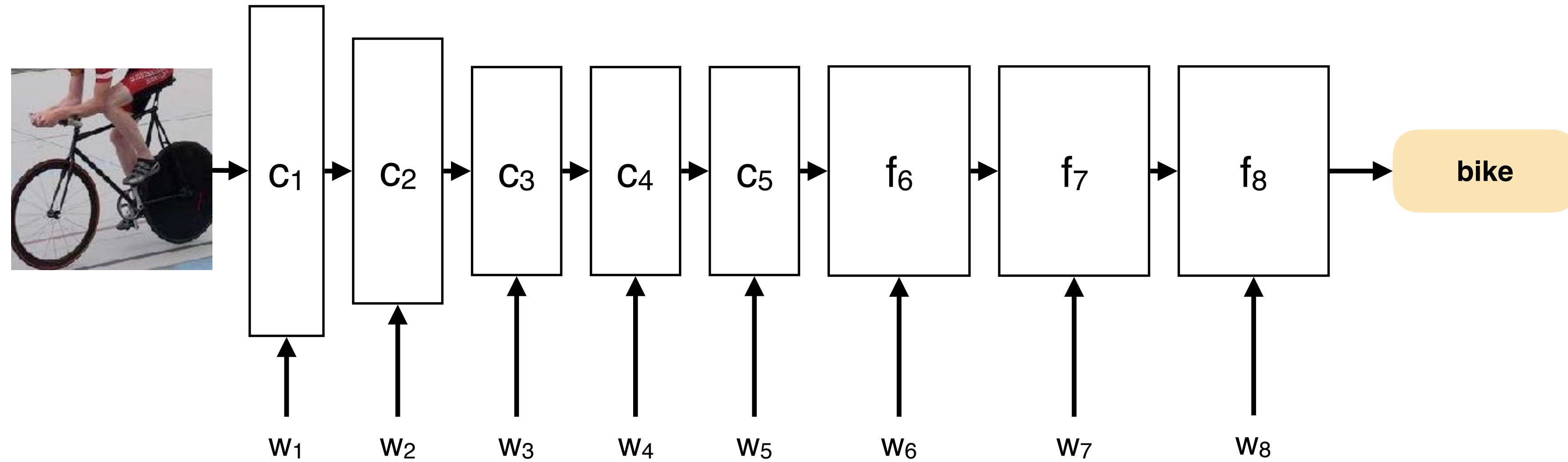


A few hundred
teraflops of compute
capability

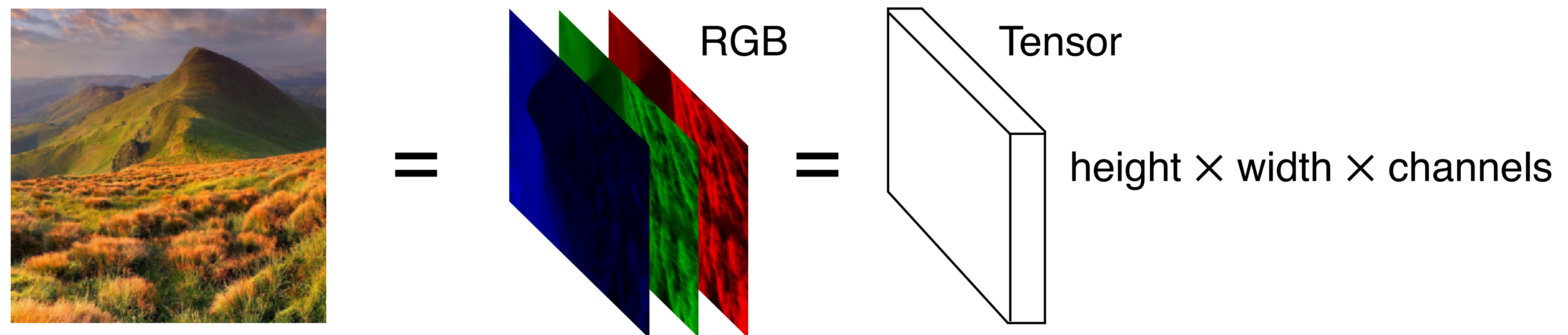


A few dozen
grad students

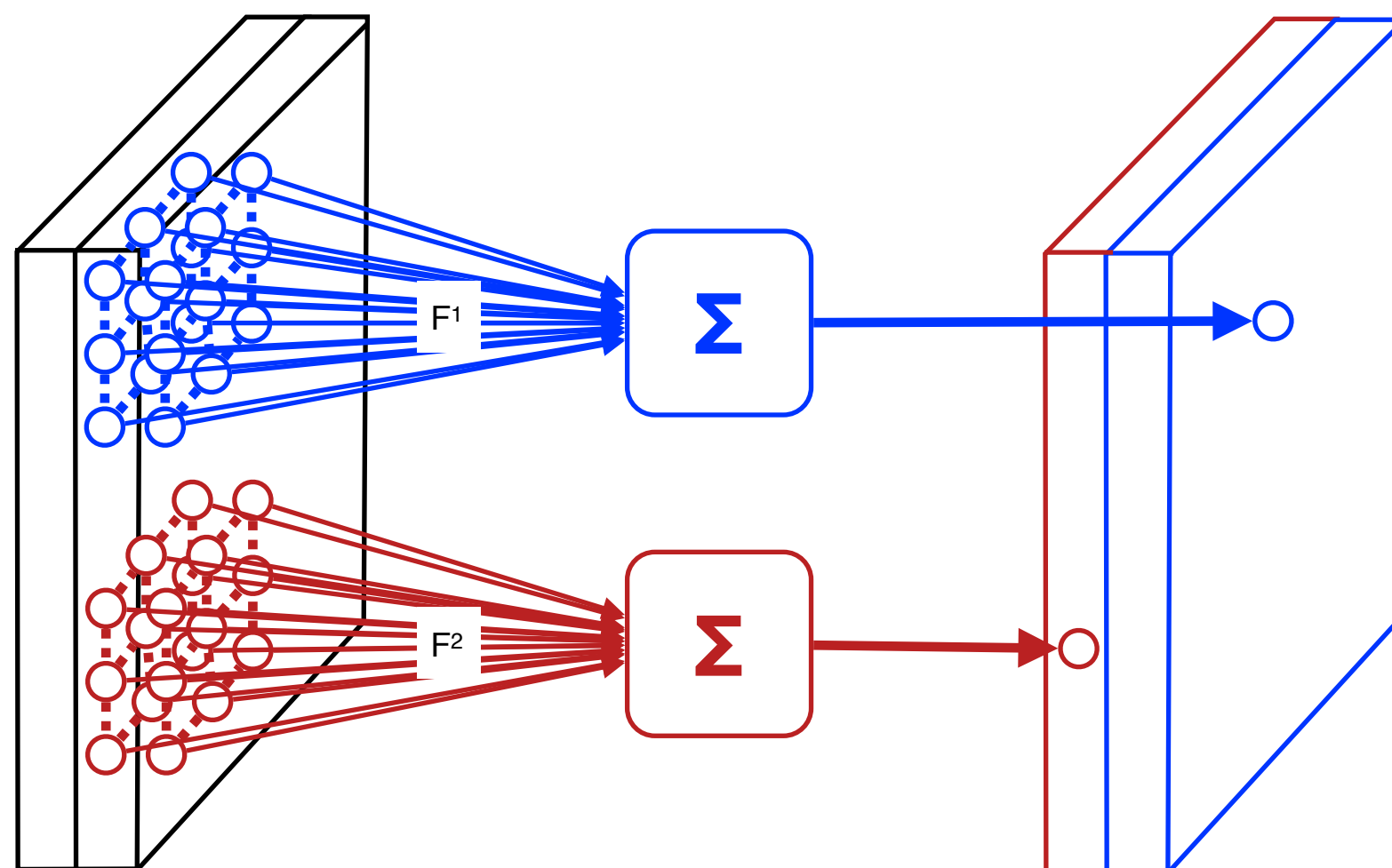
A composition of parametric linear and non-linear operators



Tensor data

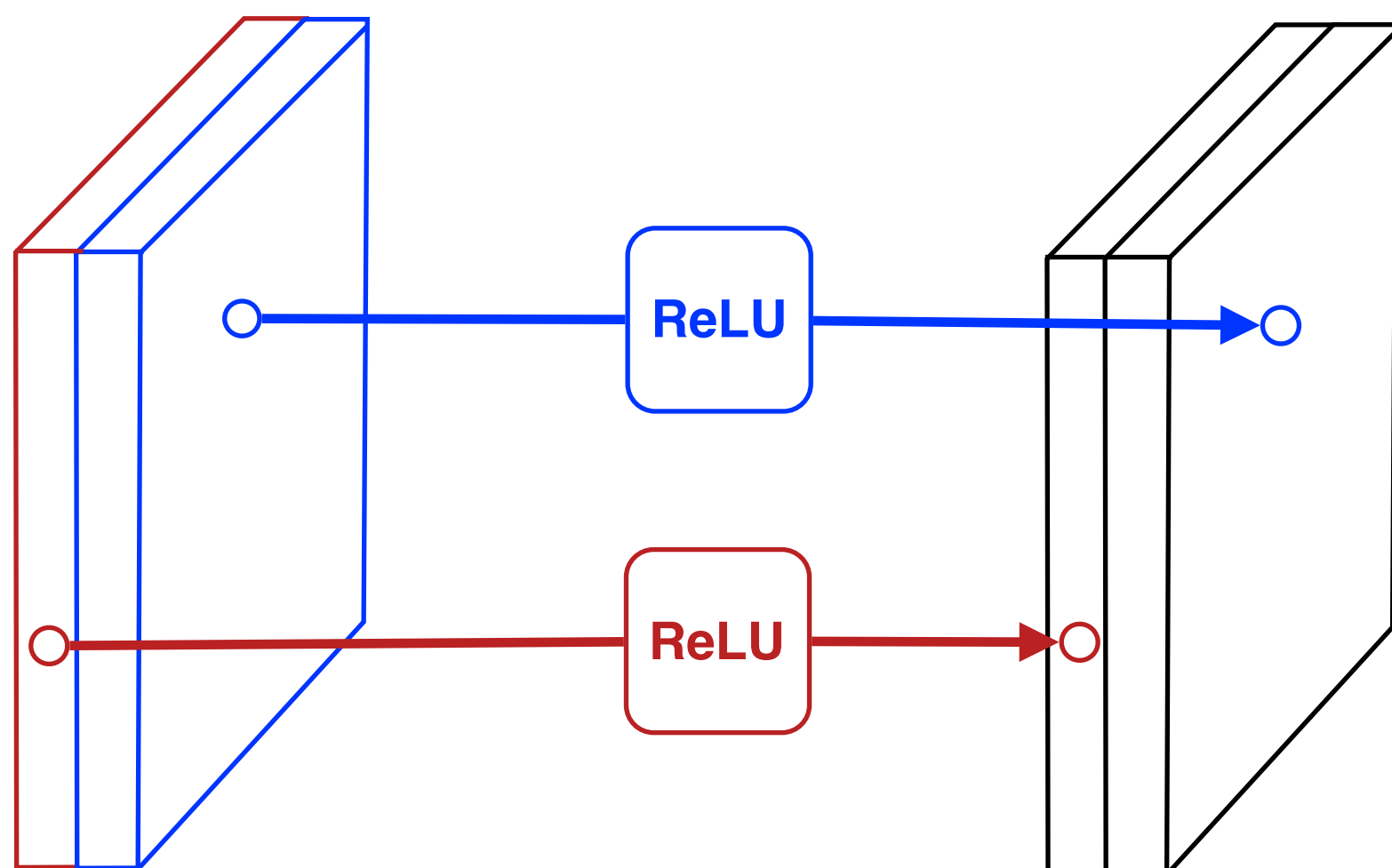


Linear convolution



- **Filter bank**
 - **several** filters
 - each generating an output channel
- **Tensor input-output**
 - **big** filters
 - multi-dimensional

Non-linear activation



- Simple non-linear functions
 - $\max\{0, \mathbf{x}\}$

AlexNet (2012)

5 convolutional layers

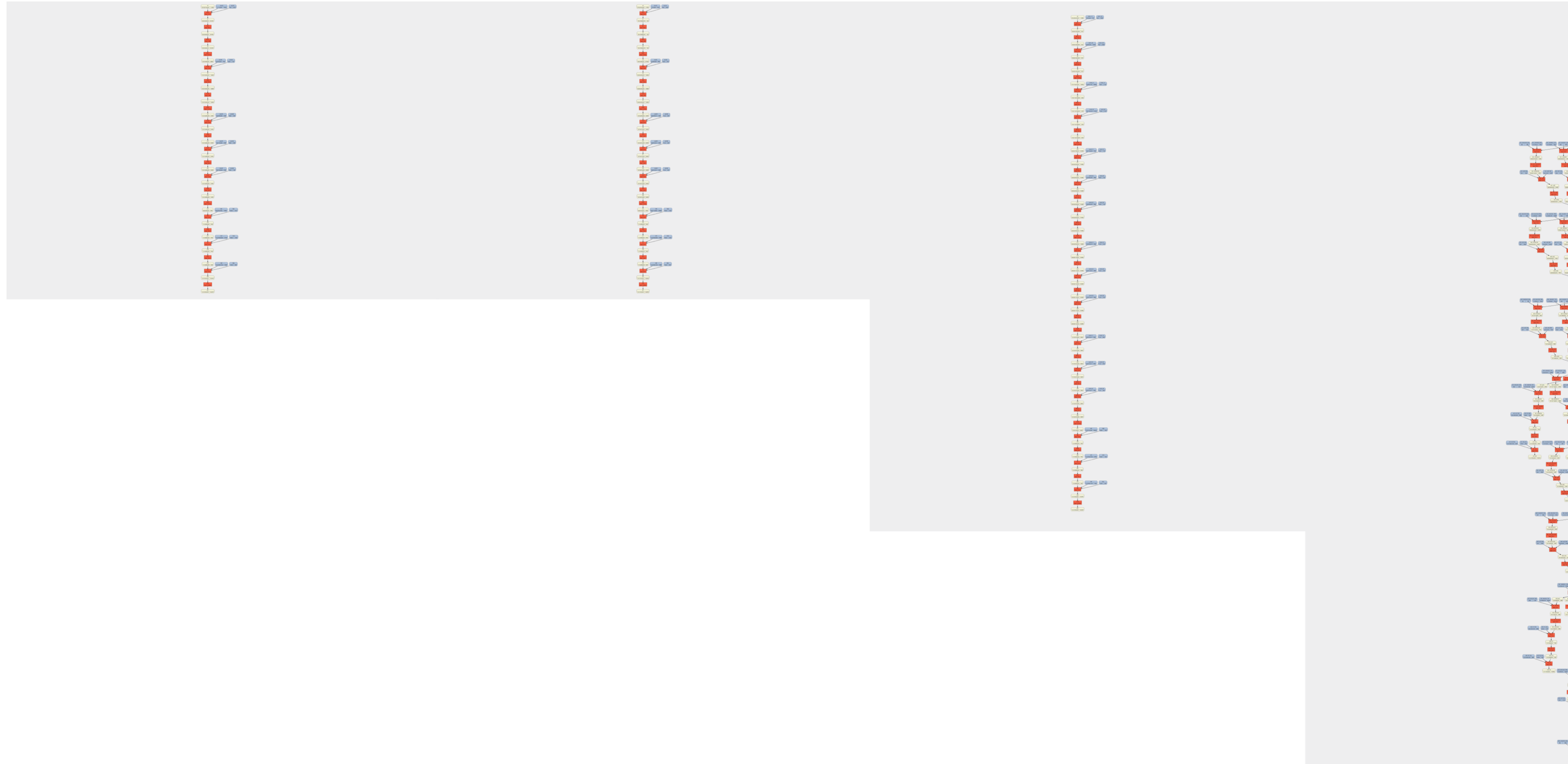
3 fully-connected layers



AlexNet (2012)

VGG-M (2013)

VGG-VD-16 (2014)



How deep is deep enough?

AlexNet (2012)

VGG-M (2013)

VGG-VD-16 (2014)

GoogLeNet (2014)



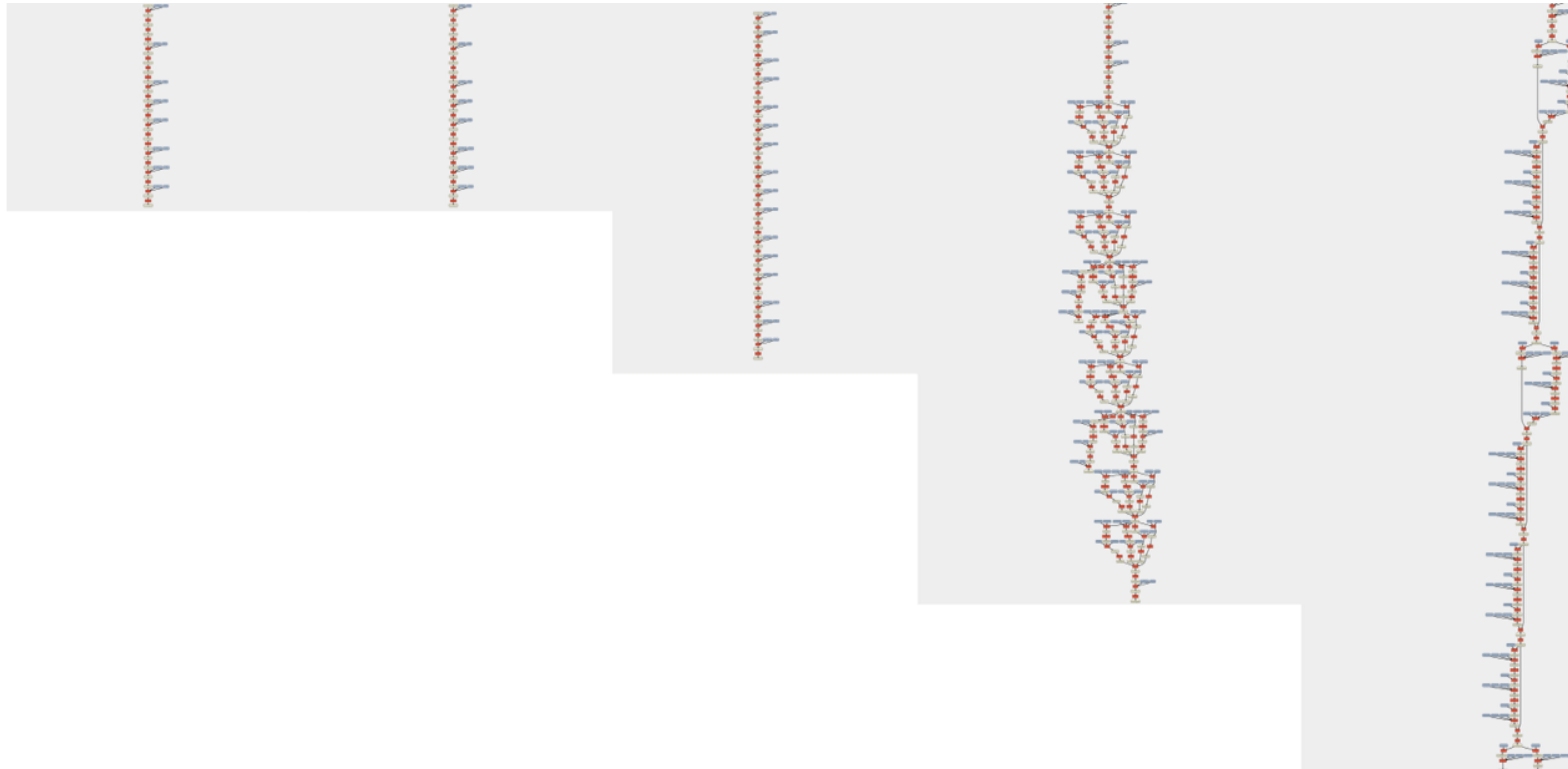
How deep is deep enough?

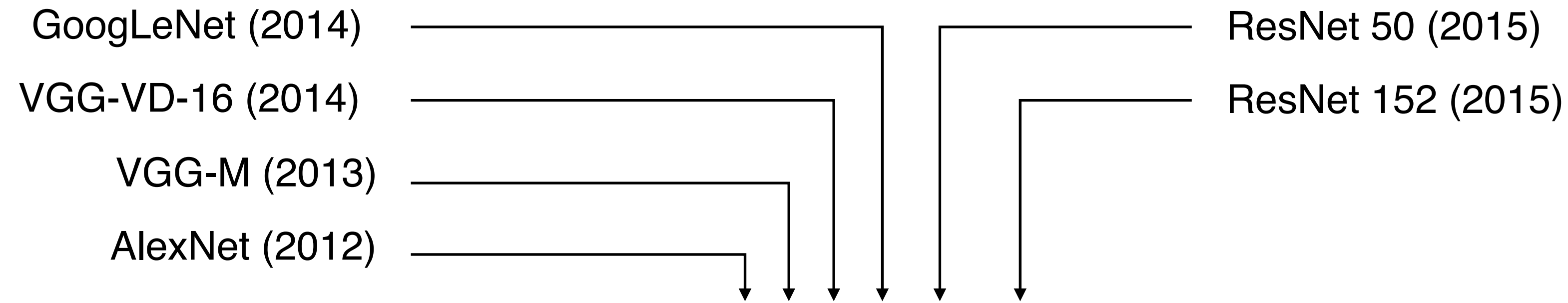
AlexNet (2012)

VGG-M (2013)

VGG-VD-16 (2014)

GoogLeNet (2014)

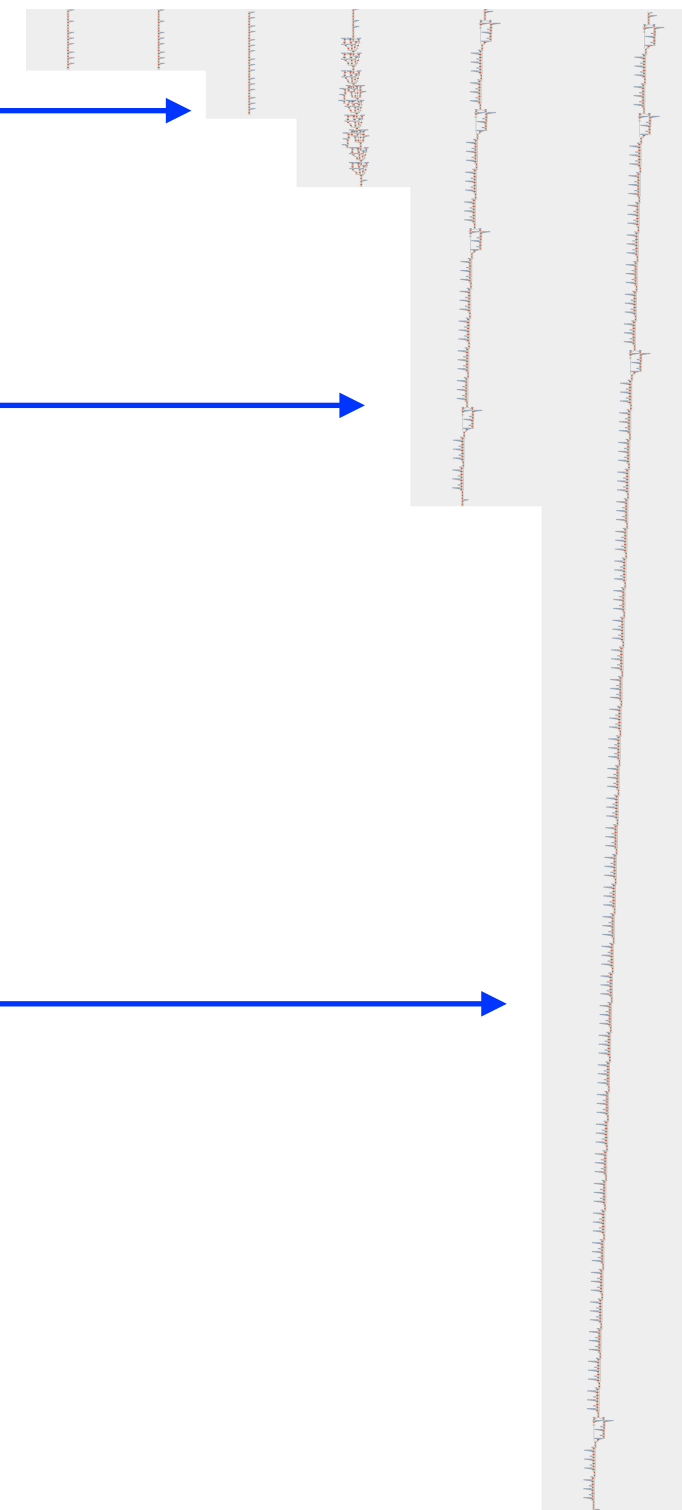




16 convolutional layers

50 convolutional layers

152 convolutional layers



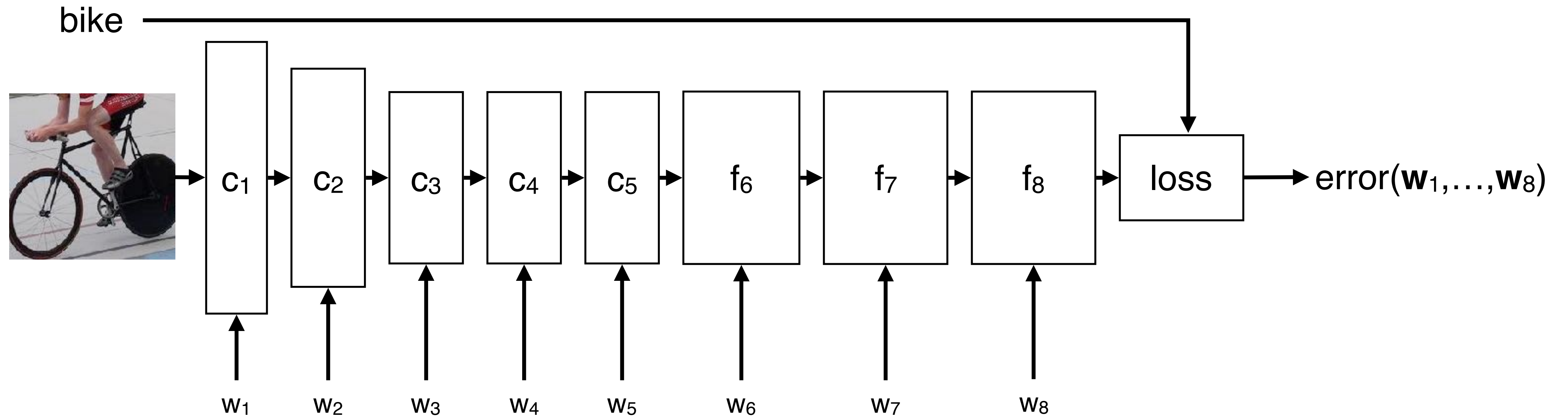
Krizhevsky, I. Sutskever, and G. E. Hinton. *ImageNet classification with deep convolutional neural networks*. In Proc. NIPS, 2012.

C. Szegedy, W. Liu, Y. Jia, P. Sermanet, S. Reed, D. Anguelov, D. Erhan, V. Vanhoucke, and A. Rabinovich. *Going deeper with convolutions*. In Proc. CVPR, 2015.

K. Simonyan and A. Zisserman. *Very deep convolutional networks for large-scale image recognition*. In Proc. ICLR, 2015.

K. He, X. Zhang, S. Ren, and J. Sun. *Deep residual learning for image recognition*. In Proc. CVPR, 2016.

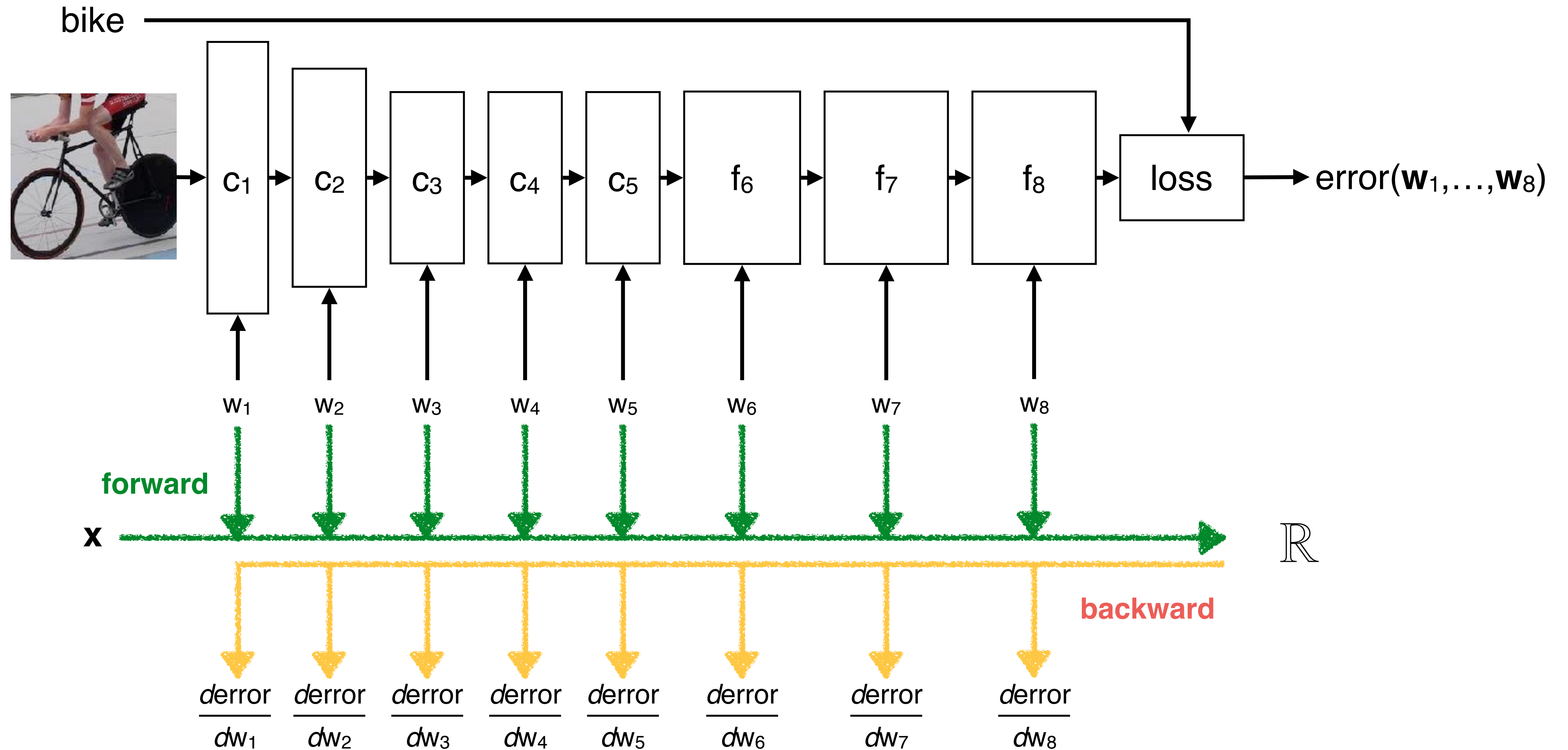
Learning = optimise the parameters w to minimise a fitting error



The error function is optimised using (stochastic) **gradient descent**

We require the **error function derivatives**

Efficient computation of the gradient



Requirements

Flexible and usable API

- Concise & powerful
- Automatic differentiation

Extensible

- Keep up with research
- Test new ideas

Efficient

- GPU
- Optimised compute graph

MATLAB

- Simple yet powerful language
- Historically, widely adopted in computer vision and robotics
- Great GPU support
- Great documentation
- Recently, native support for deep nets...

The first modern deep learning toolbox in MATLAB

The screenshot shows the MatConvNet website homepage. The navigation bar includes 'MatConvNet', 'Home', 'Getting Started', 'Functions', and 'Contributing'. A sidebar on the left lists: 'MatConvNet: CNNs for MATLAB', 'Obtaining MatConvNet', 'Documentation', 'Extensions', 'Getting started', 'Use cases', and 'Other information'. The main content area features the title 'MatConvNet: CNNs for MATLAB' with links for 'Download', 'Code & issues', 'Pre-trained models', and 'Discussion forum'. A paragraph describes MatConvNet as a MATLAB toolbox for CNNs. A news section lists updates: 'New: 1.0-beta25' (new modular system `vl_contrib`), 'New: 1.0-beta24' (bugfixes, new examples), 'New: 1.0-beta23' (new `vl_nmrpool` and Fast-RCNN demo), and 'New: 1.0-beta22' (bugfixes). Below are sections for 'Obtaining MatConvNet' (tarball, GIT, citation), 'Documentation' (manual, MATLAB functions, FAQ, discussion group), and 'Extensions' (third-party contributions).

Why?

- Fully MATLAB-hackable
- As efficient as other tools (Caffe, TensorFlow, Torch, ...)

Real-world state-of-the-art applications

- See demos
- Many more

Cutting-edge research

- 900+ citations in academic papers

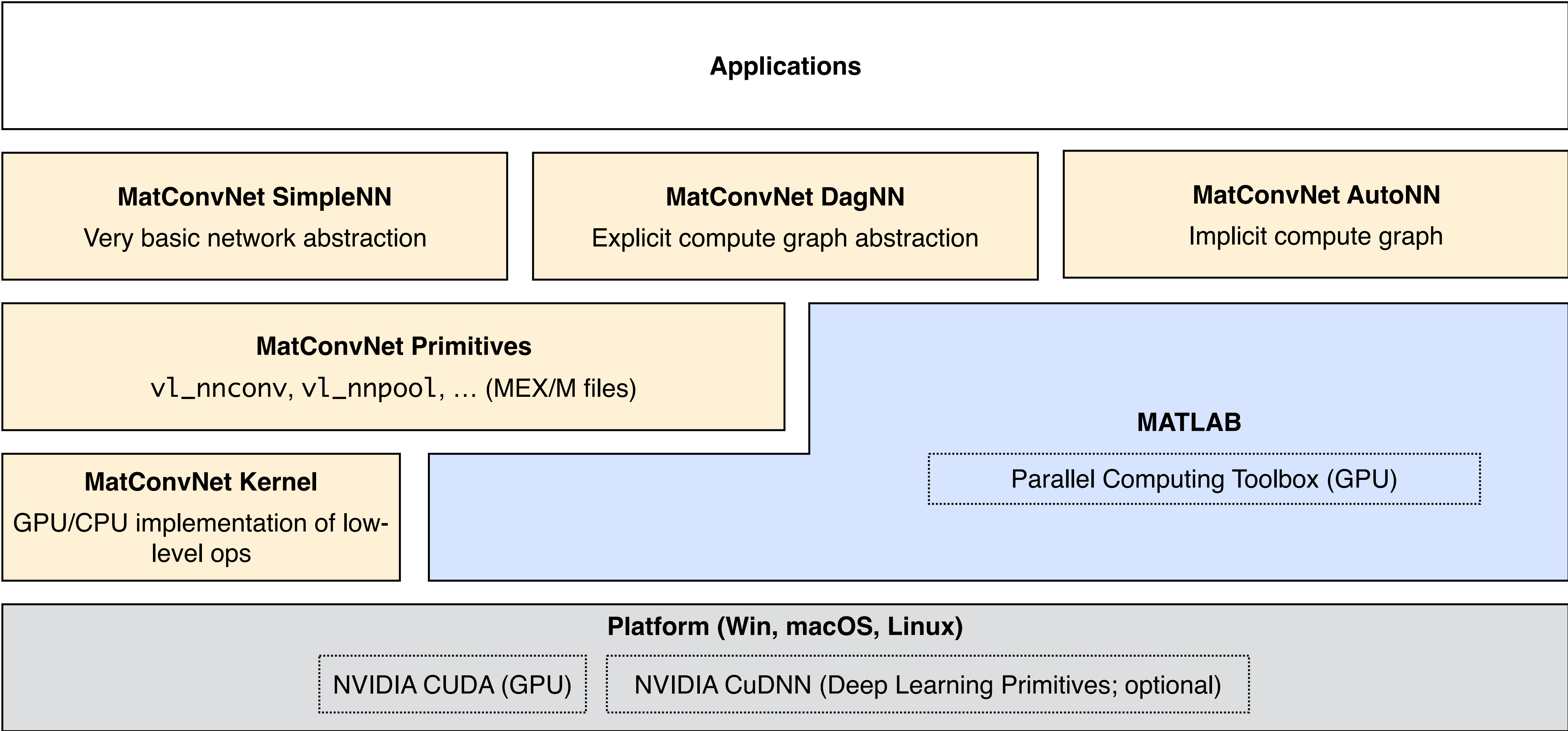
Education

- Several international courses use it

Pedigree

- Spawn of VLFeat (Mark Everingham Award)
- Has been around since the “beginning” (~2012)

Deep learning sandwich



Pure MATLAB code

What can you do with it

Use a pre-trained model

- VGG-VD, ResNet, ResNext, SSD, R-CNN, ...

Learn a new model

- Arbitrary compute graphs
- SGD on multi GPUs

Create new layer types

- Native MATLAB (gpuArrays)

Hack the compute graph

- Visualisation, debugging, optimisations

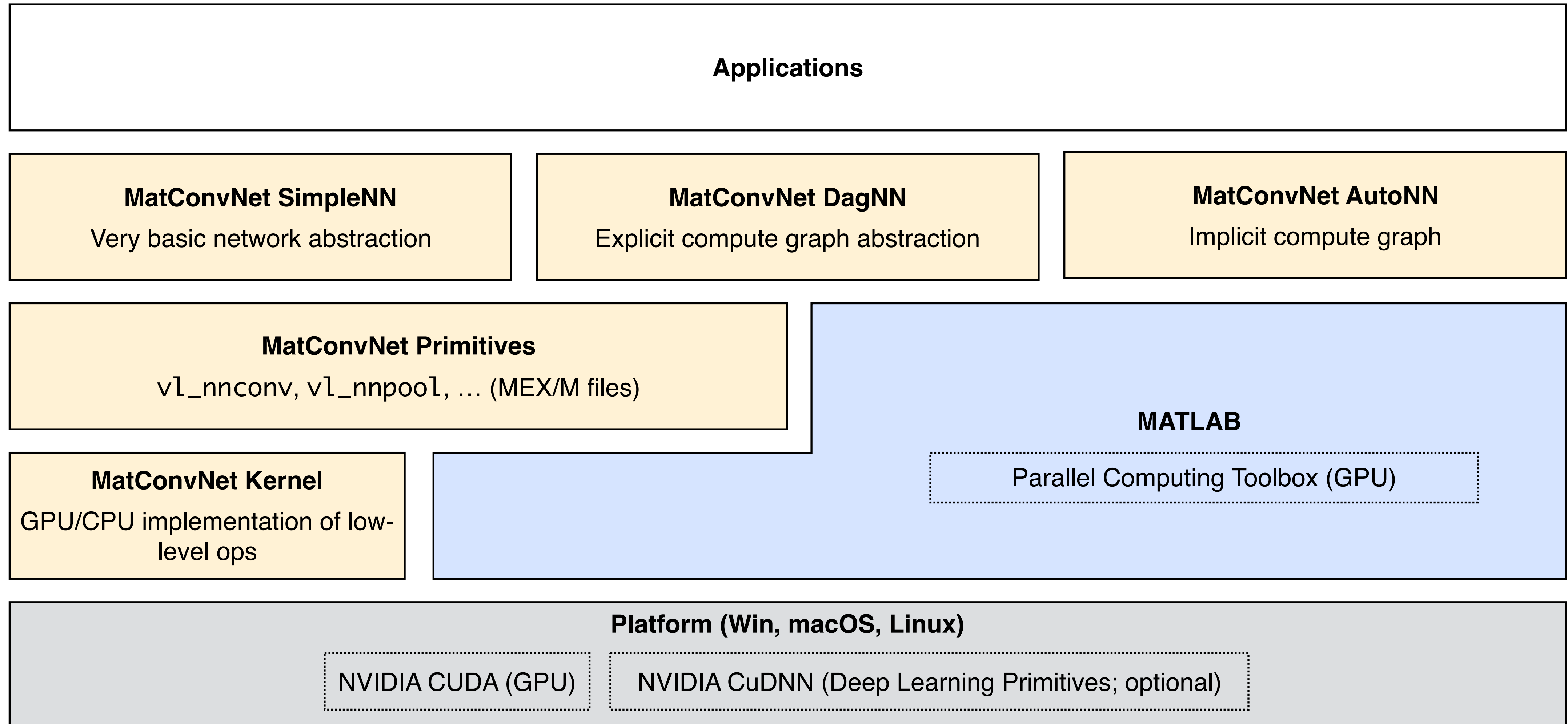
Hack autodiff

- Define a new API

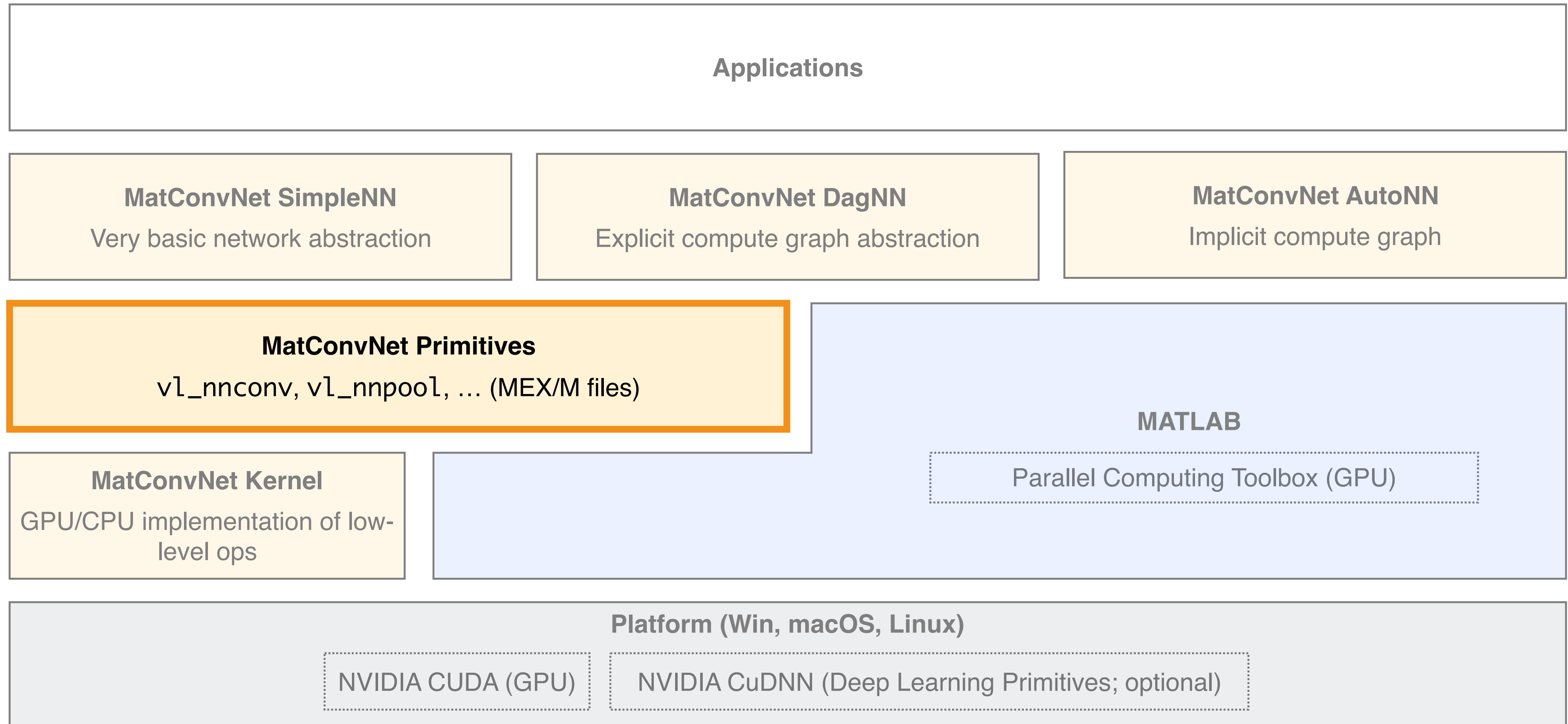
Hack everything

- Everything is open

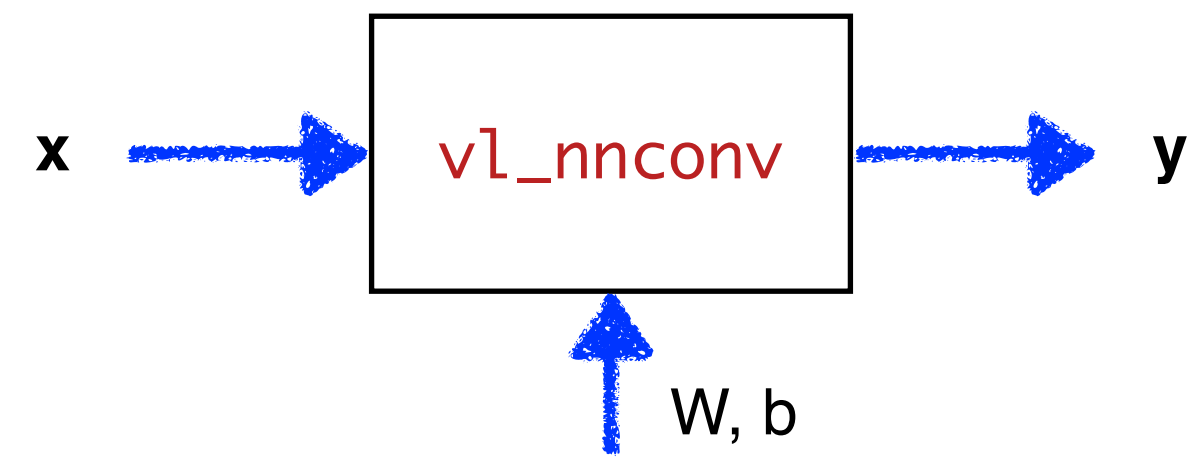
Deep learning sandwich



Deep learning sandwich

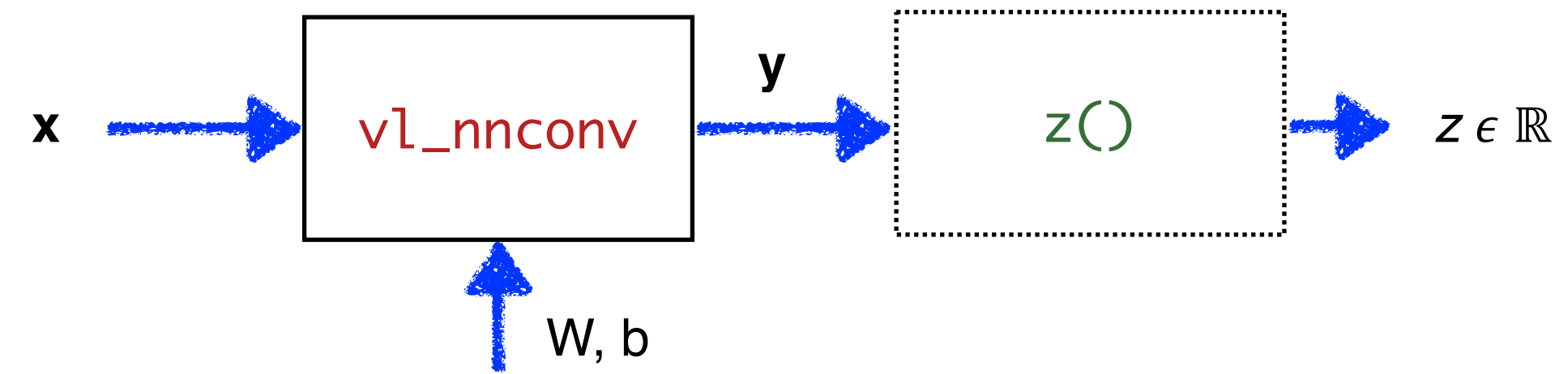


forward (eval)



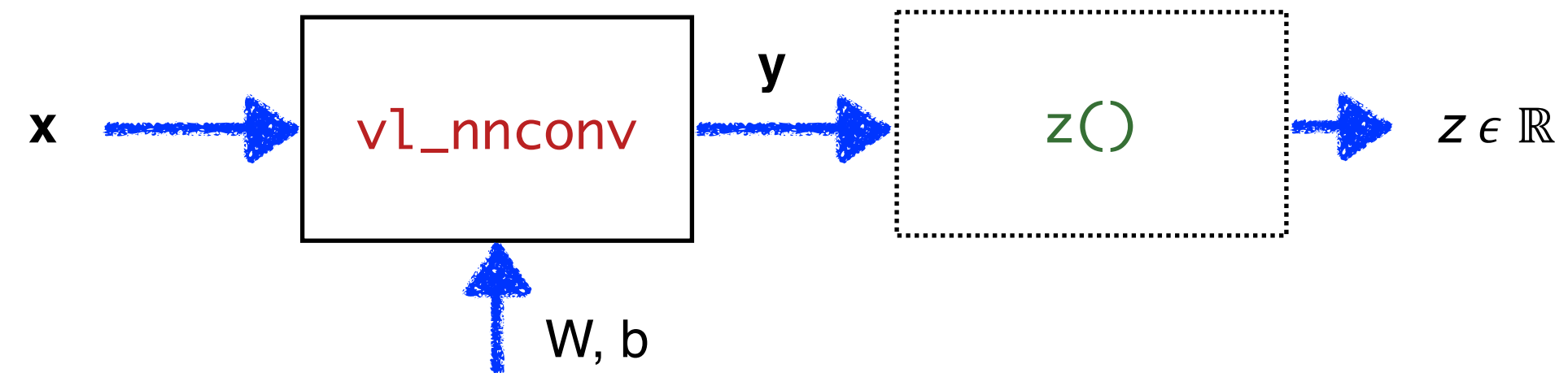
$$y = \text{vl_nnconv}(x, W, b)$$

forward (eval)



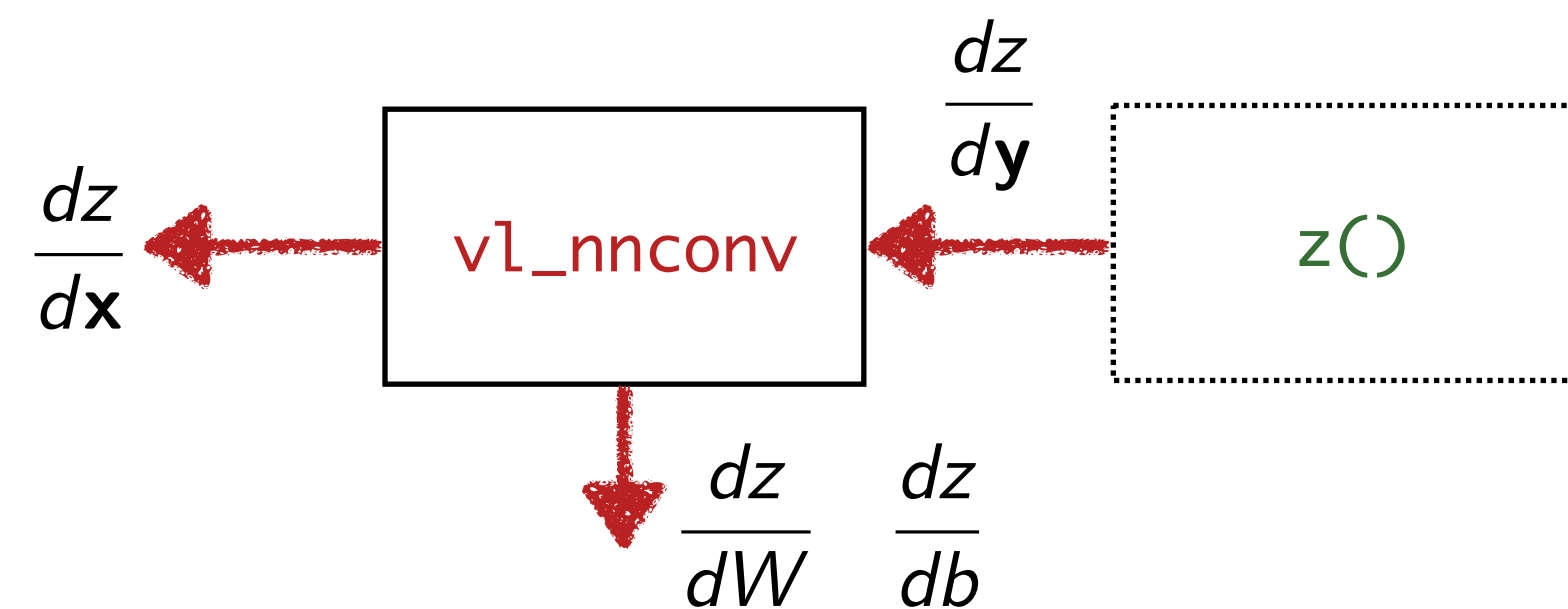
$$y = \text{vl_nnconv}(x, W, b)$$

forward (eval)



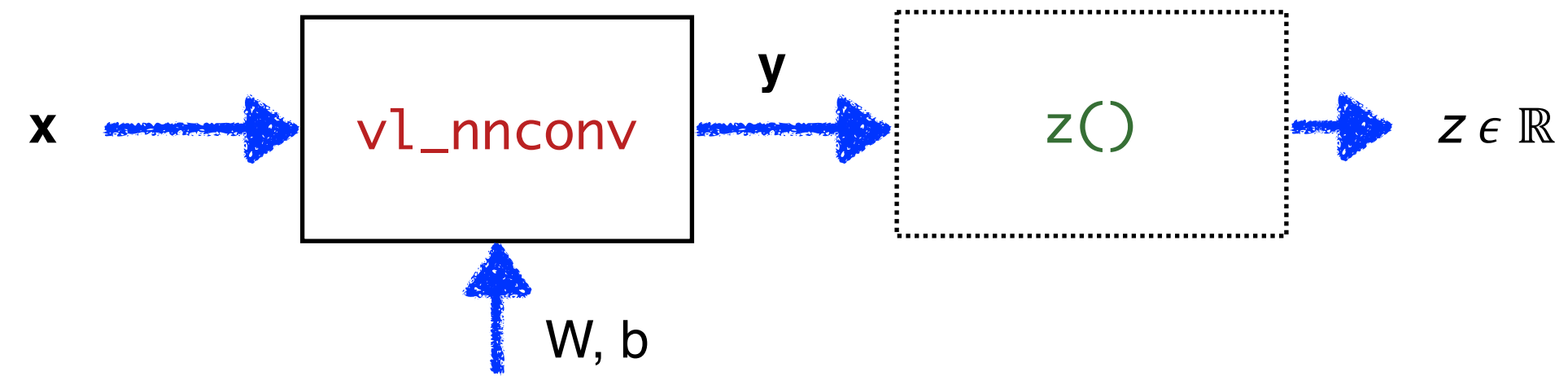
$$y = \text{v1_nnconv}(x, W, b)$$

backward (backprop)



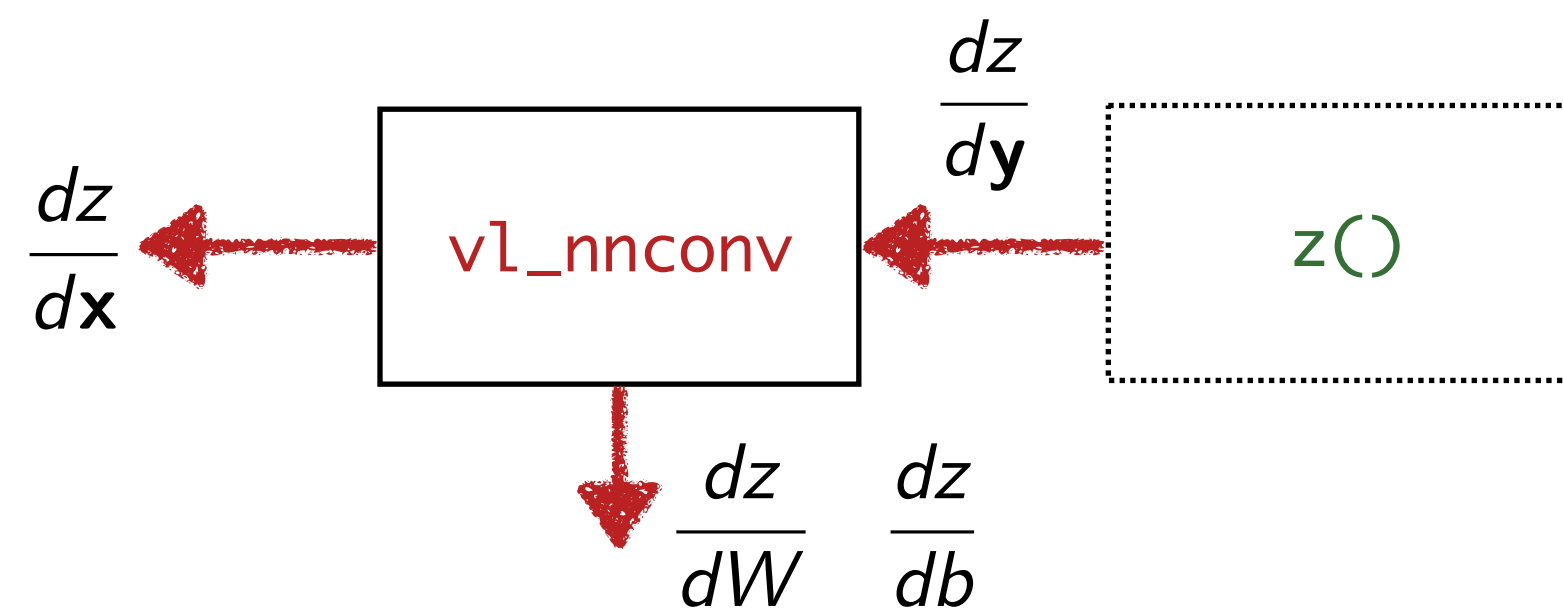
$$\text{dzdx} = \text{v1_nnconv}(x, W, b, \text{dzdy})$$

forward (eval)



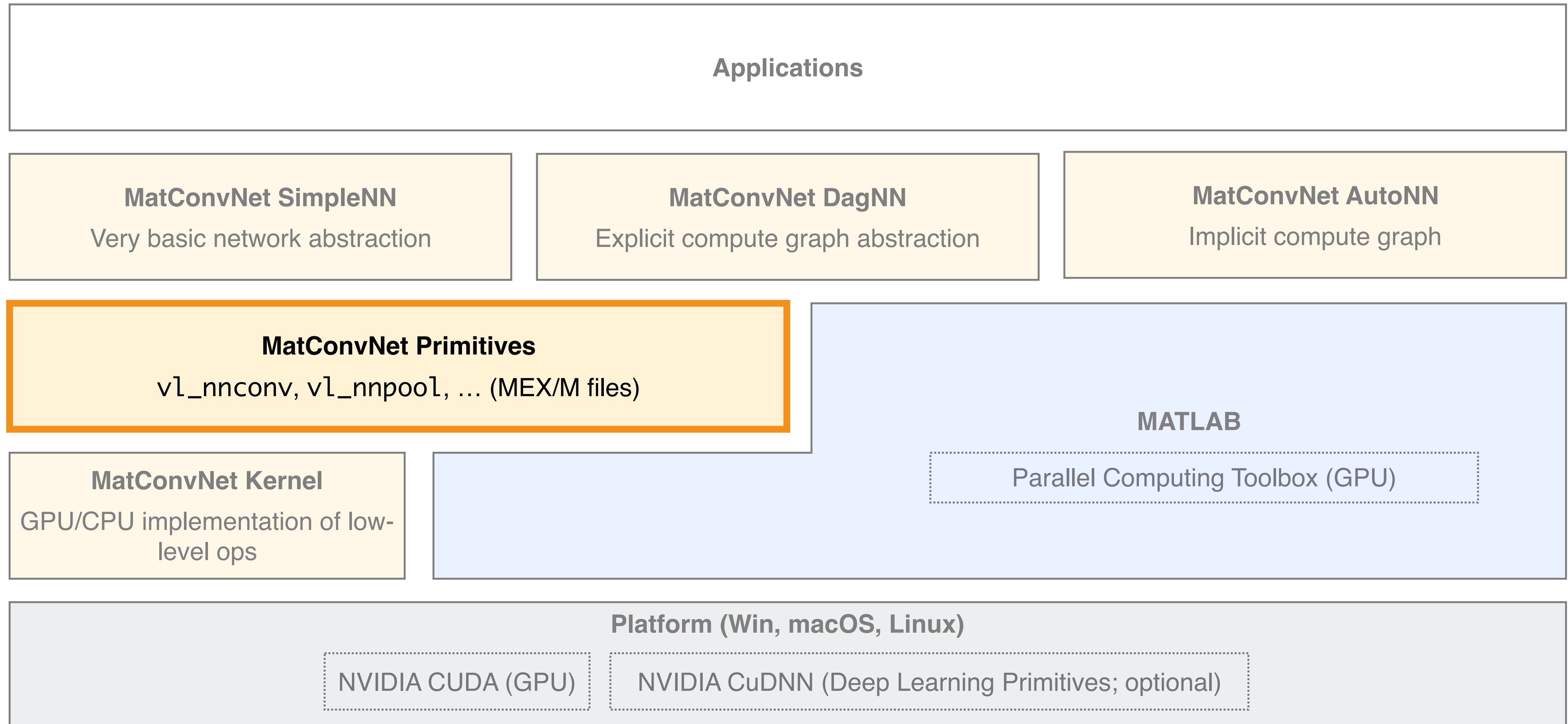
$$y = \text{vl_nnconv}(x, W, b)$$

backward (backprop)

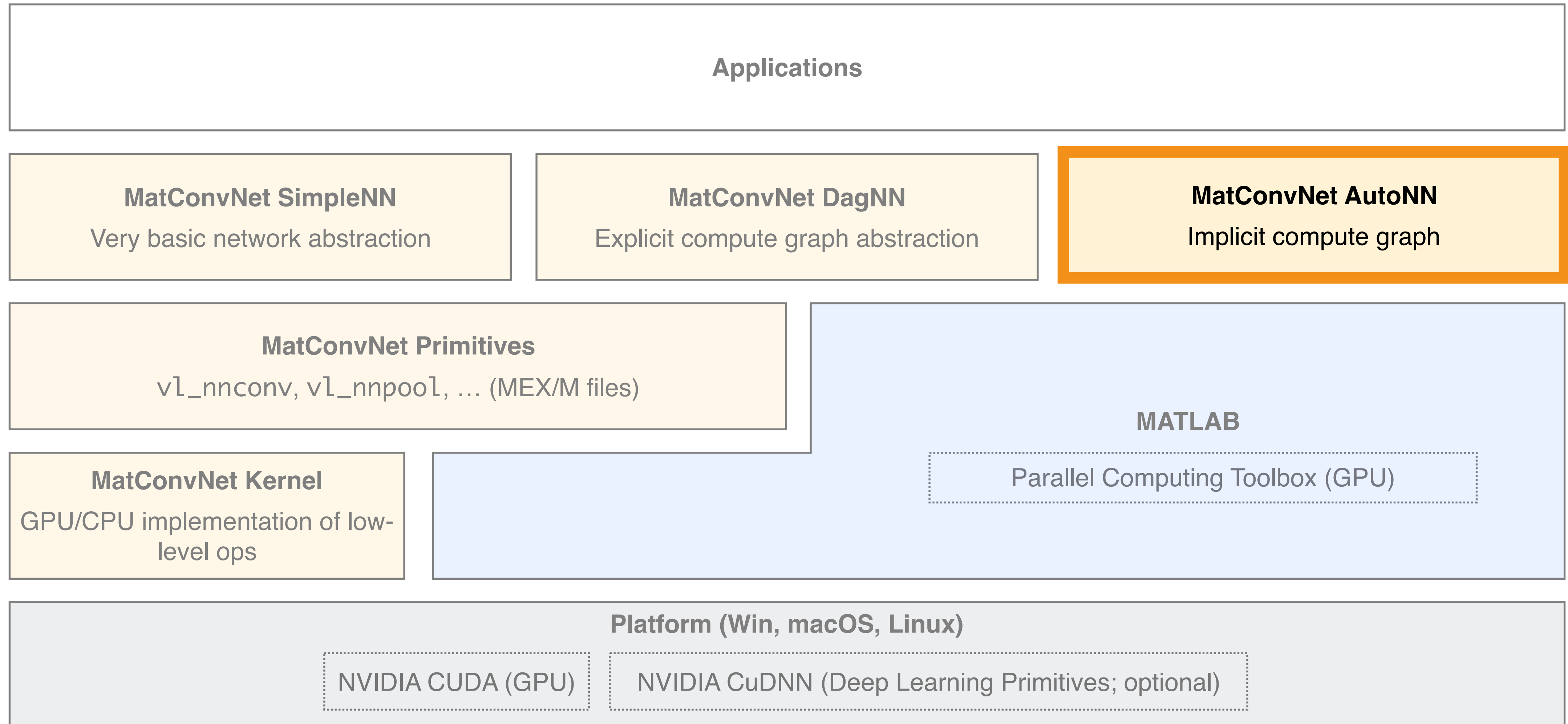


$$\frac{dz}{dx} = \text{vl_nnconv}(x, W, b, \frac{dz}{dy})$$

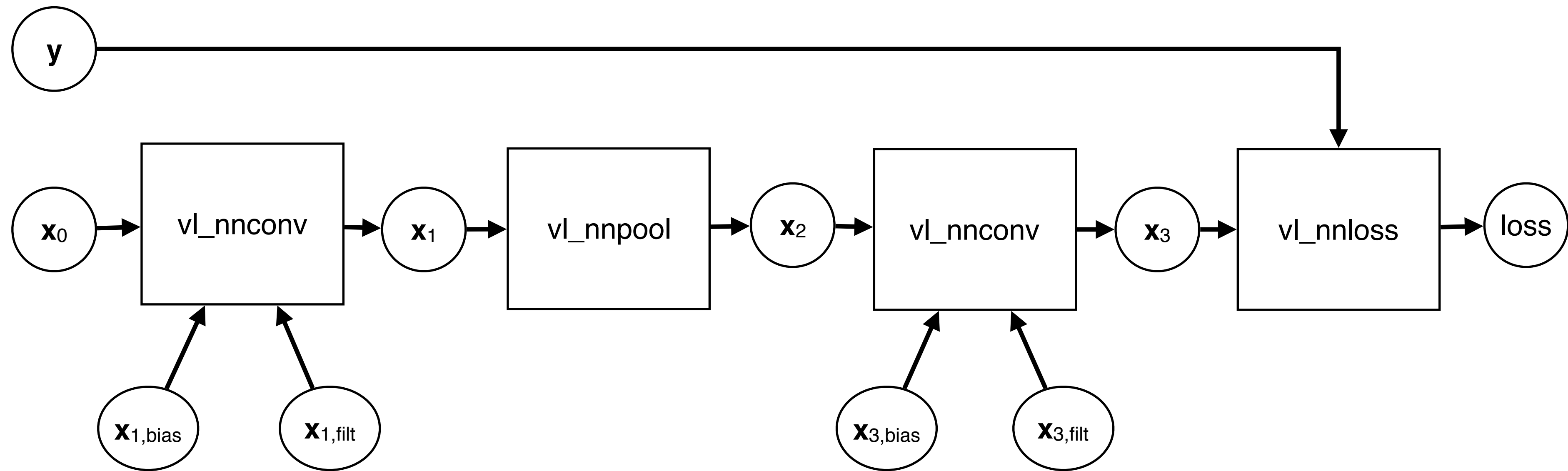
Deep learning sandwich



Deep learning sandwich



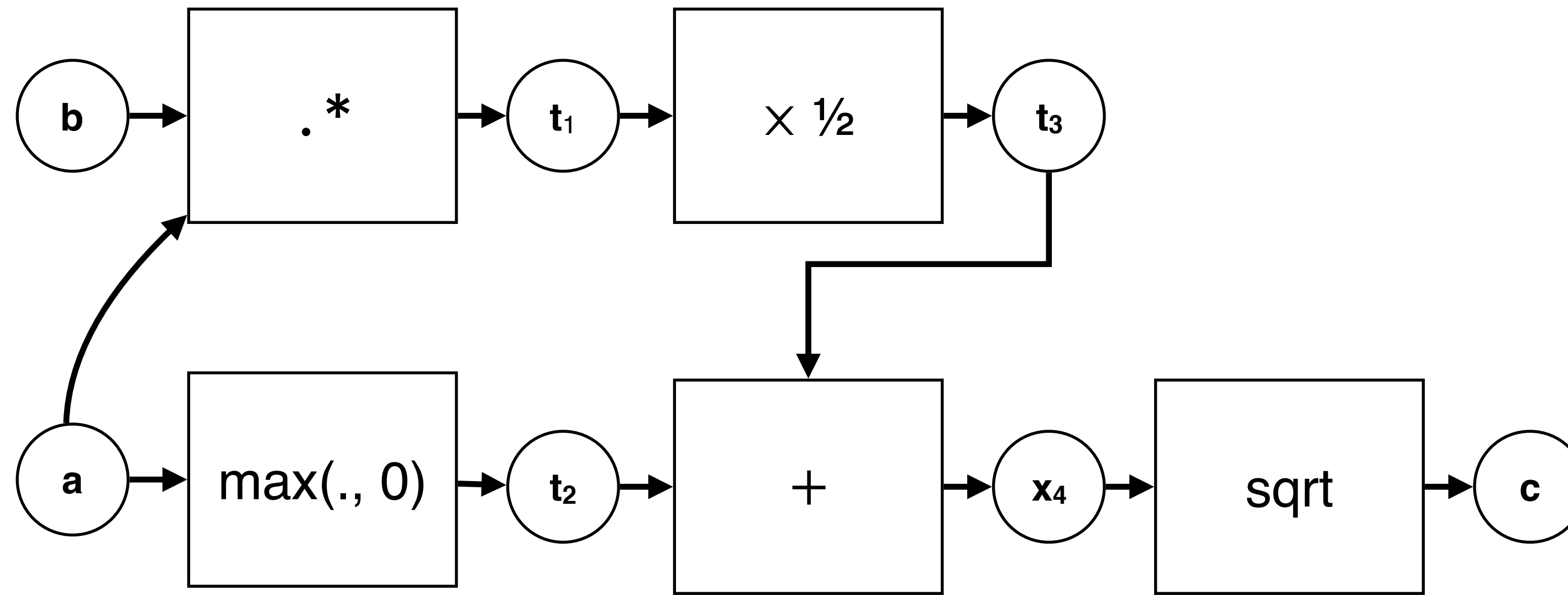
Defining and evaluating a deep network



```

% Define network & loss
x0 = Input() ;
y = Input();
x1 = vl_nnconv(x0, 'size', [5, 5, 1, 20]) ;
x2 = vl_nnpool(x1, 2, 'stride', 2) ;
x3 = vl_nnconv(x2, 'size', [5, 5, 20, 10]) ;
loss = vl_nnloss(x3, y);
    
```

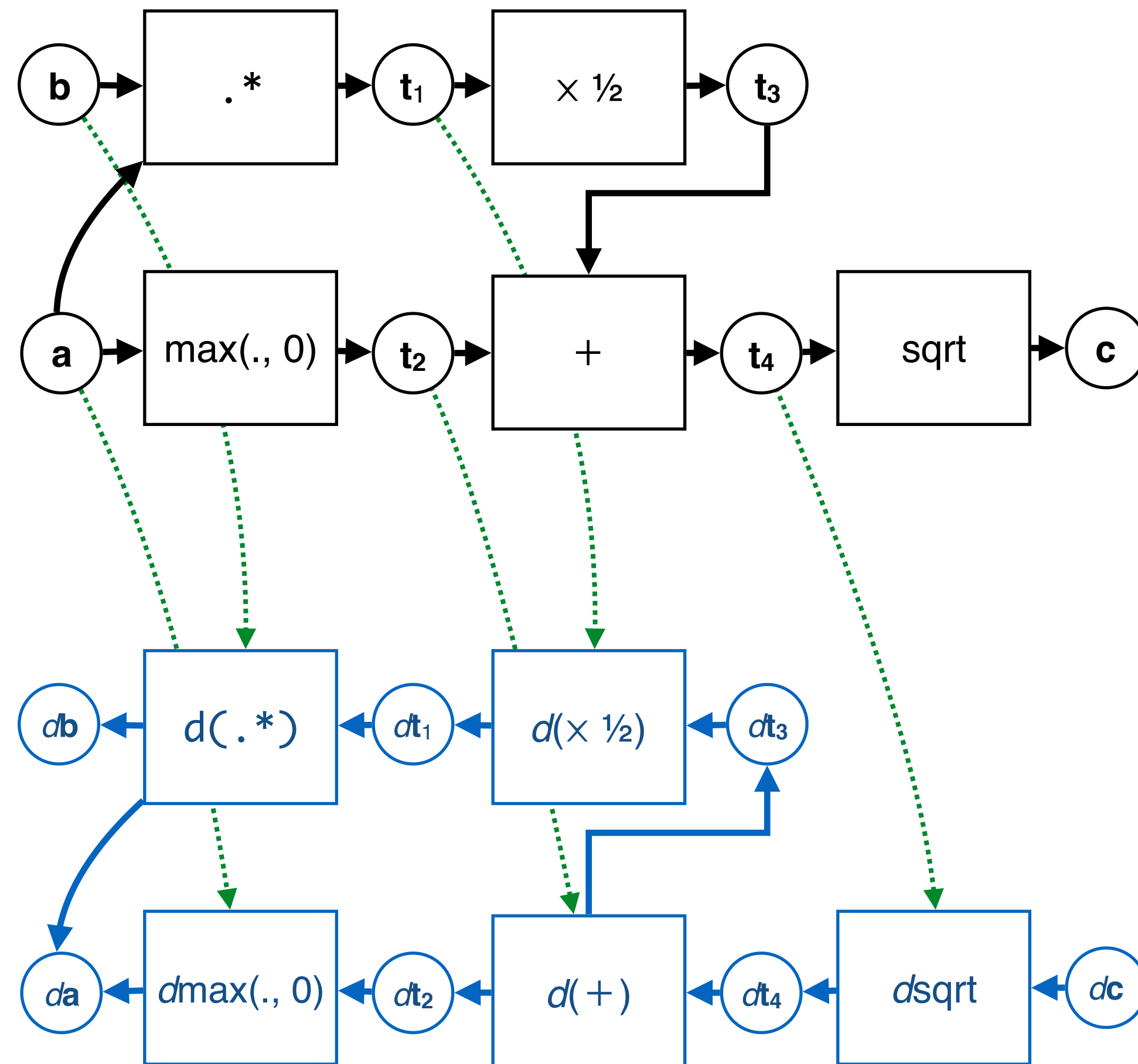
Defining and evaluating a deep network



```

% Define compute graph
a = Input() ;
b = Input() ;
c = sqrt(max(a,0) + a.*b/2) ;
  
```


Why this instead of Maple / Symbolic Toolbox



Autodiff **is not** symbolic differentiation

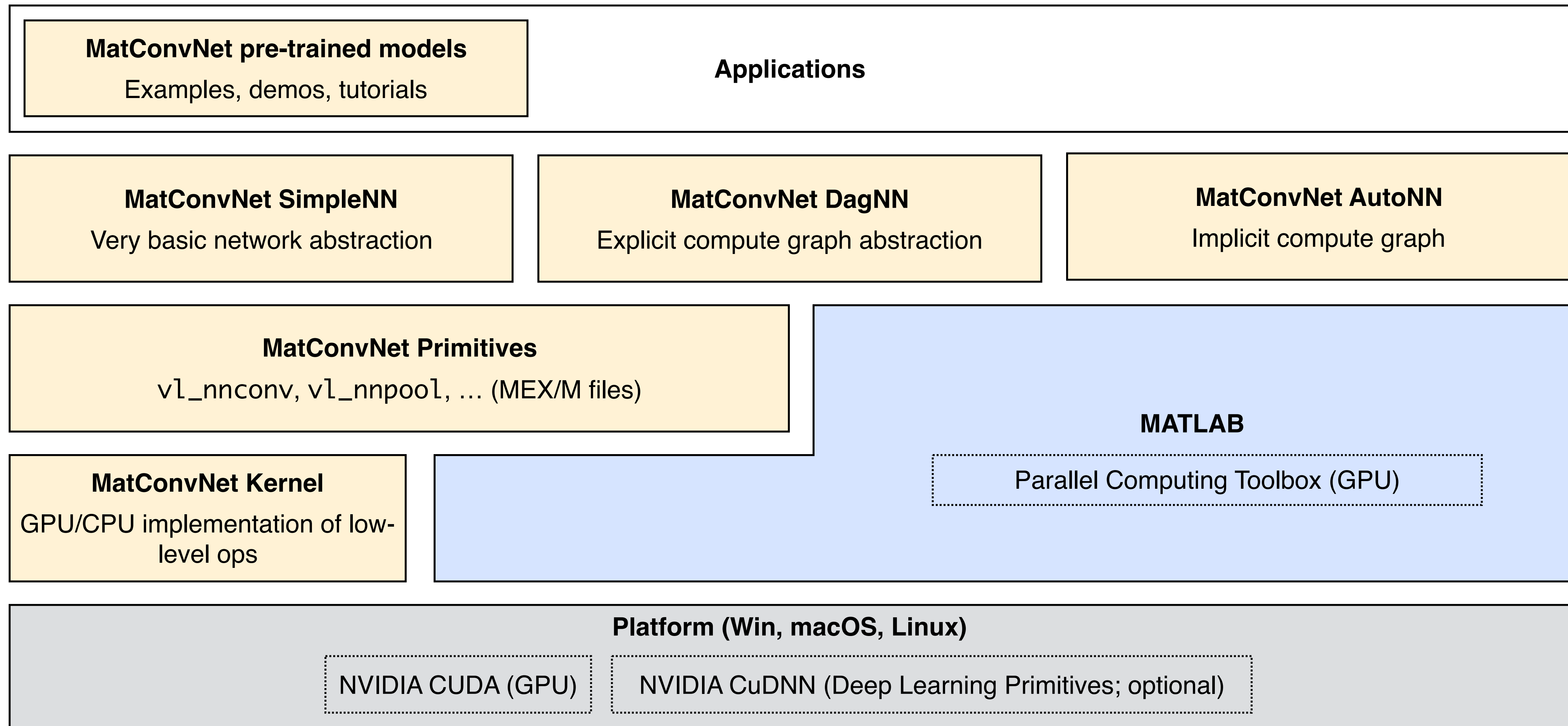
Autodiff

- computes derivatives numerically
- as efficiently as possible

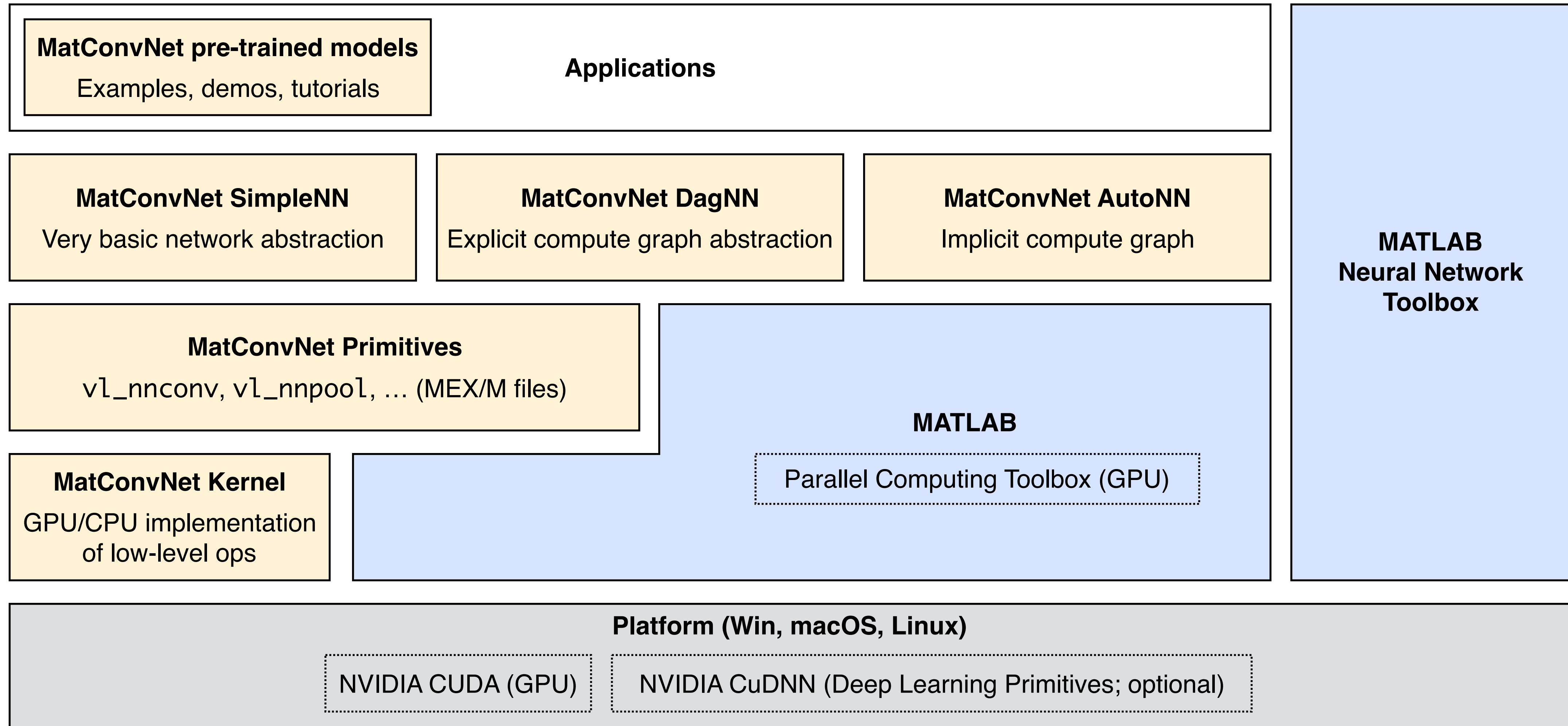
Under the hood

- Autodiff appends a backward extension to the graph
- executing the graph computes both function and its derivative

An increasingly powerful alternative

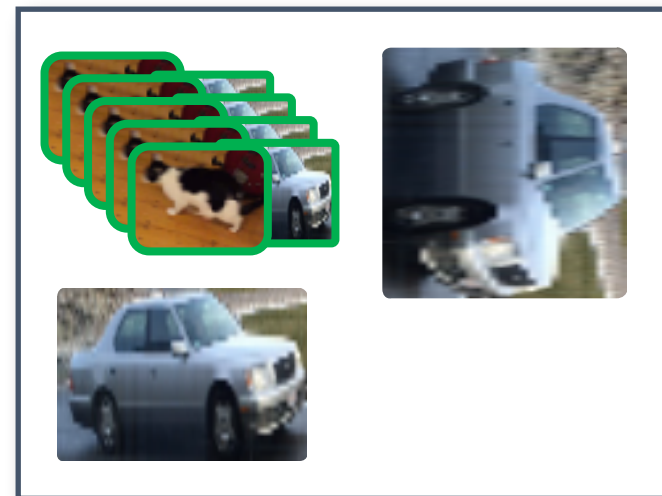


An increasingly powerful alternative

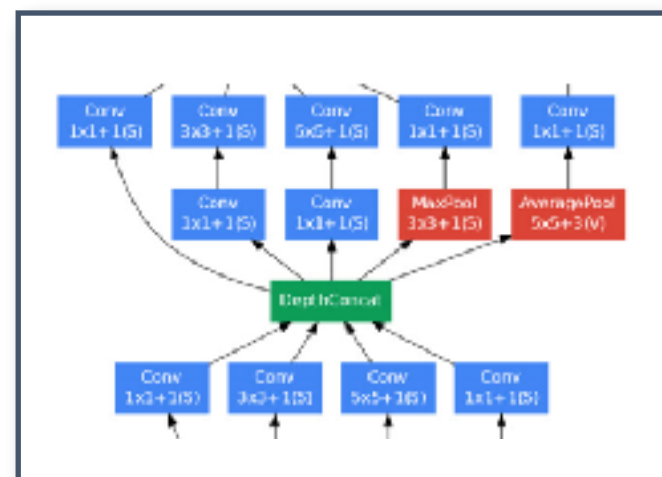


R2017a

R2017b



Data Access



Networks



Train



Deploy / Share

<ul style="list-style-type: none"> • App for Ground Truth labeling • Alexnet, VGG-16, VGG-19 • Caffe model importer 	<ul style="list-style-type: none"> • TensorFlow-Keras importer • GoogLeNet model • Label for semantic segmentation • Resize & augment images
<ul style="list-style-type: none"> • CNN Regression • Object detection using Fast R-CNN and R-CNN • Object detector evaluation 	<ul style="list-style-type: none"> • LSTM (time series, text) • DAG Networks • Create new layers
<ul style="list-style-type: none"> • Multi-GPUs in parallel • Visual features using activations 	<ul style="list-style-type: none"> • Validation • Training plots • Hyper-parameter optimization
<div data-bbox="1749 1369 2215 1585" data-label="Text"> <p>New Product</p> </div>	
<ul style="list-style-type: none"> • GPU Coder: convert MATLAB models to NVIDIA CUDA code 	



<http://vlfeat.org/matconvnet/>



<https://github.com/vlfeat/matconvnet>



Karel Lenc



Sam Albanie



Joao Henriques