MATLAB EXPO 2018

Deploying Deep Learning Networks to Embedded GPUs and CPUs

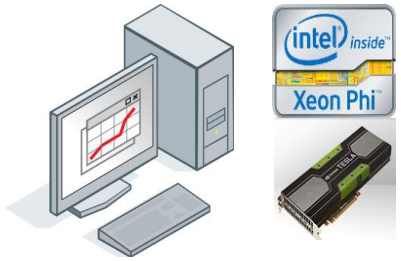
Rishu Gupta, PhD
Senior Application Engineer, Computer Vision

MATLAB Deep Learning Framework

- **Manage** large image sets
- **Automate** image labeling
- **Easy access** to models

- **Acceleration** with GPU's
- **Scale** to clusters

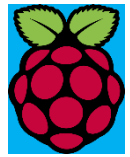
Multi-Platform Deep Learning Deployment



Desktop

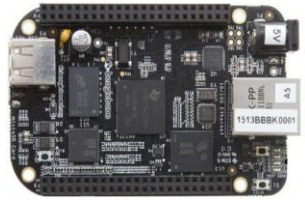
Data-center

Nvidia
TX1, TX2, TK1



Raspberry pi

Mobile



Beagle bone

. . .

Embedded

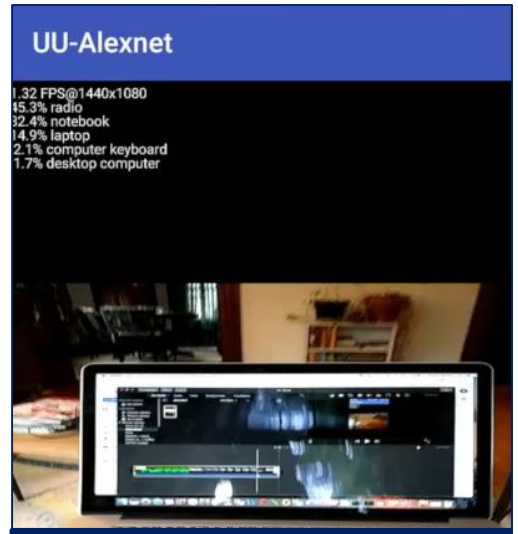
Multi-Platform Deep Learning Deployment

- Need code that takes advantage of:
  - NVIDIA® CUDA libraries, including cuDNN and TensorRT
  - Intel® Math Kernel Library for Deep Neural Networks (MKL-DNN) for Intel processors
  - ARM® Compute libraries for ARM processors

Multi-Platform Deep Learning Deployment

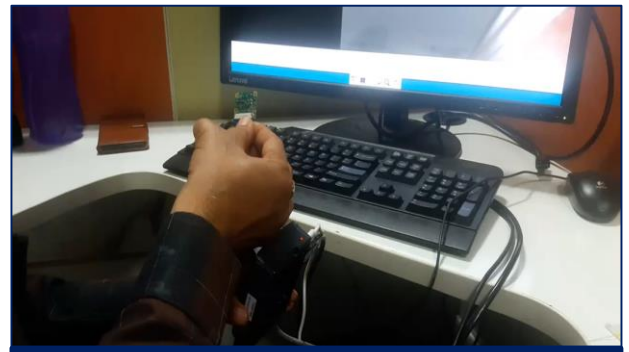
- Need code that takes advantage of:
  - NVIDIA® CUDA libraries, including cuDNN and TensorRT
  - Intel® Math Kernel Library for Deep Neural Networks (MKL-DNN) for Intel processors
  - ARM® Compute libraries for ARM processors

NVIDIA Jetson TX1 board



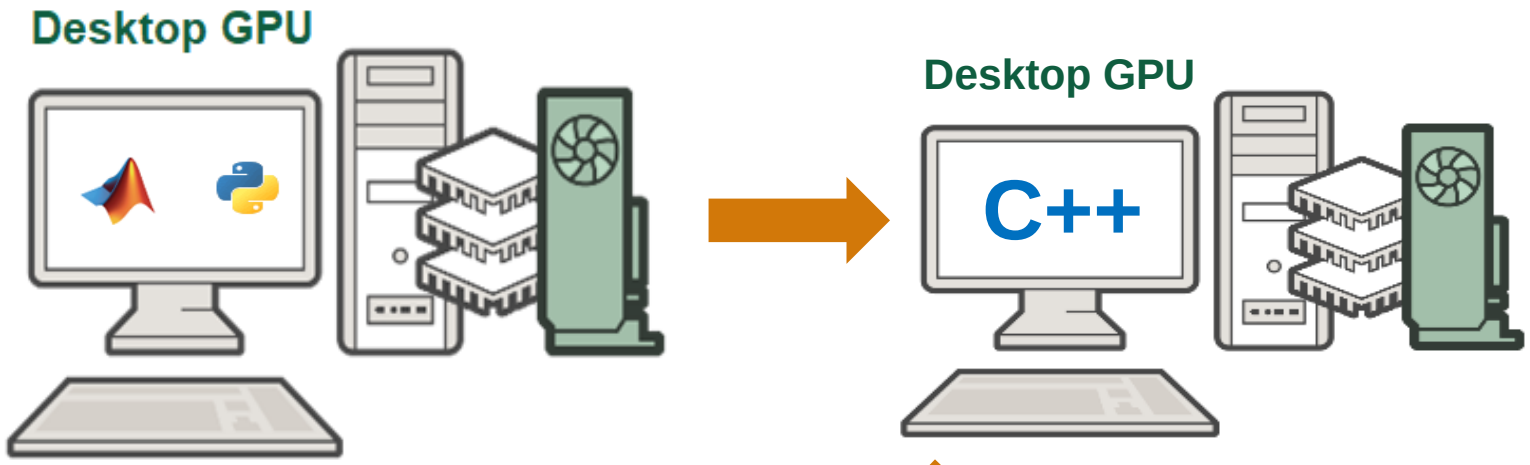
Android Phone

Intel Xeon Desktop PC



Raspberry Pi Board

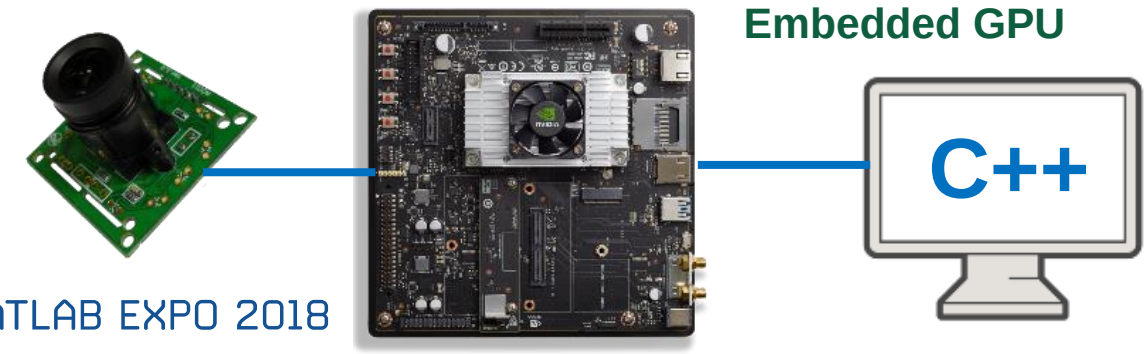
Algorithm Design to Embedded Deployment Workflow
Conventional Approach



Challenges
- Integrating multiple libraries and packages
- Verifying and maintaining multiple implementations
- Algorithm & vendor lock-in

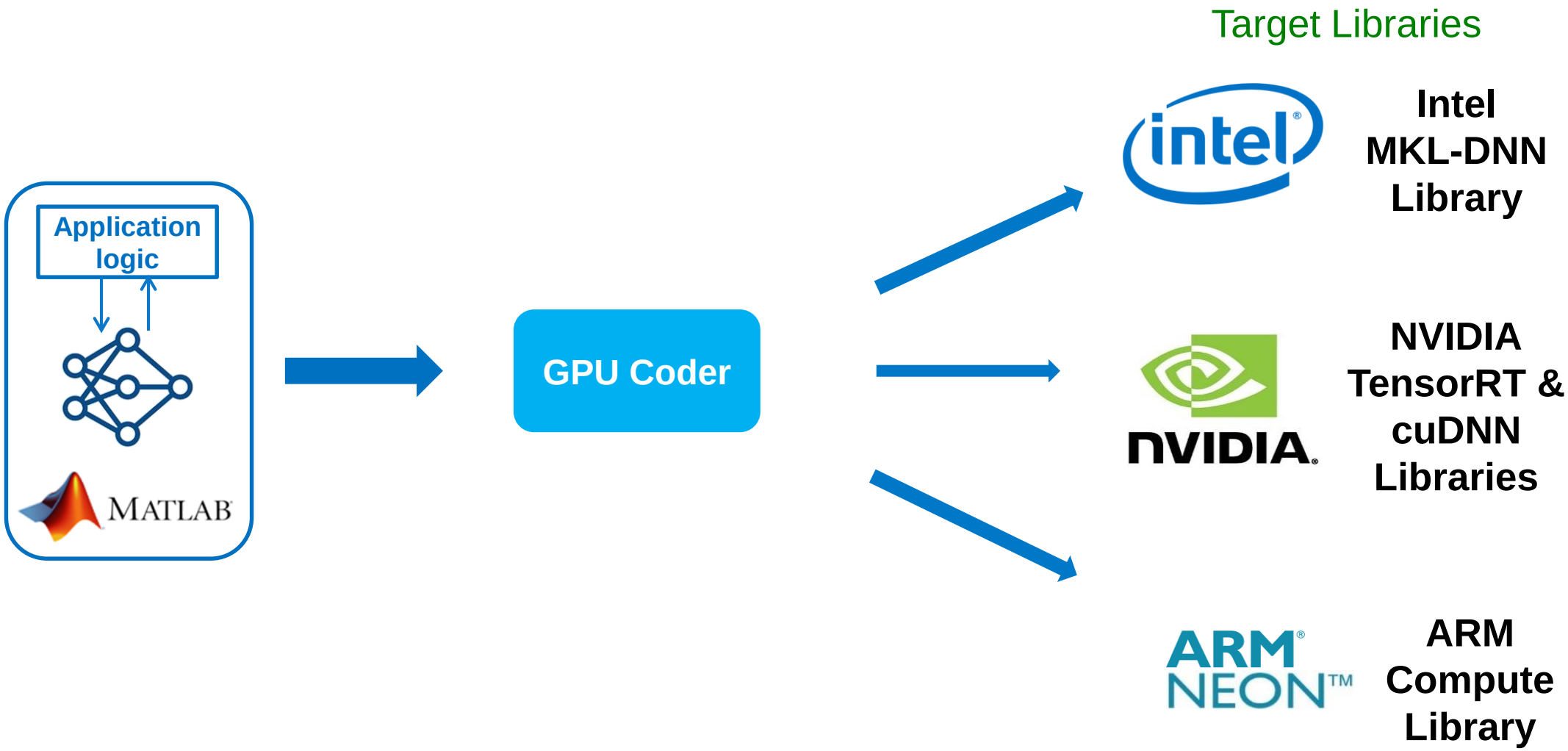
1 High-level language
Deep learning framework
Large, complex software stack

2 C/C++
Low-level APIs
Application-specific libraries



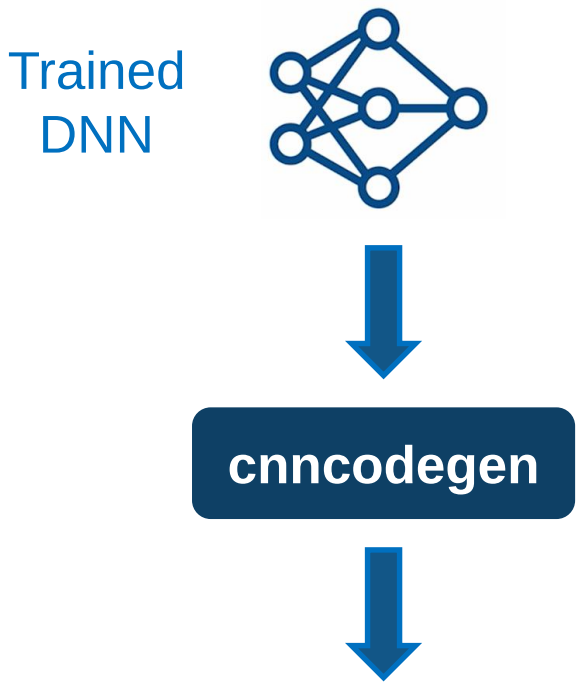
3 C/C++
Target-optimized libraries
Optimize for memory & speed

Solution- GPU Coder for Deep Learning Deployment

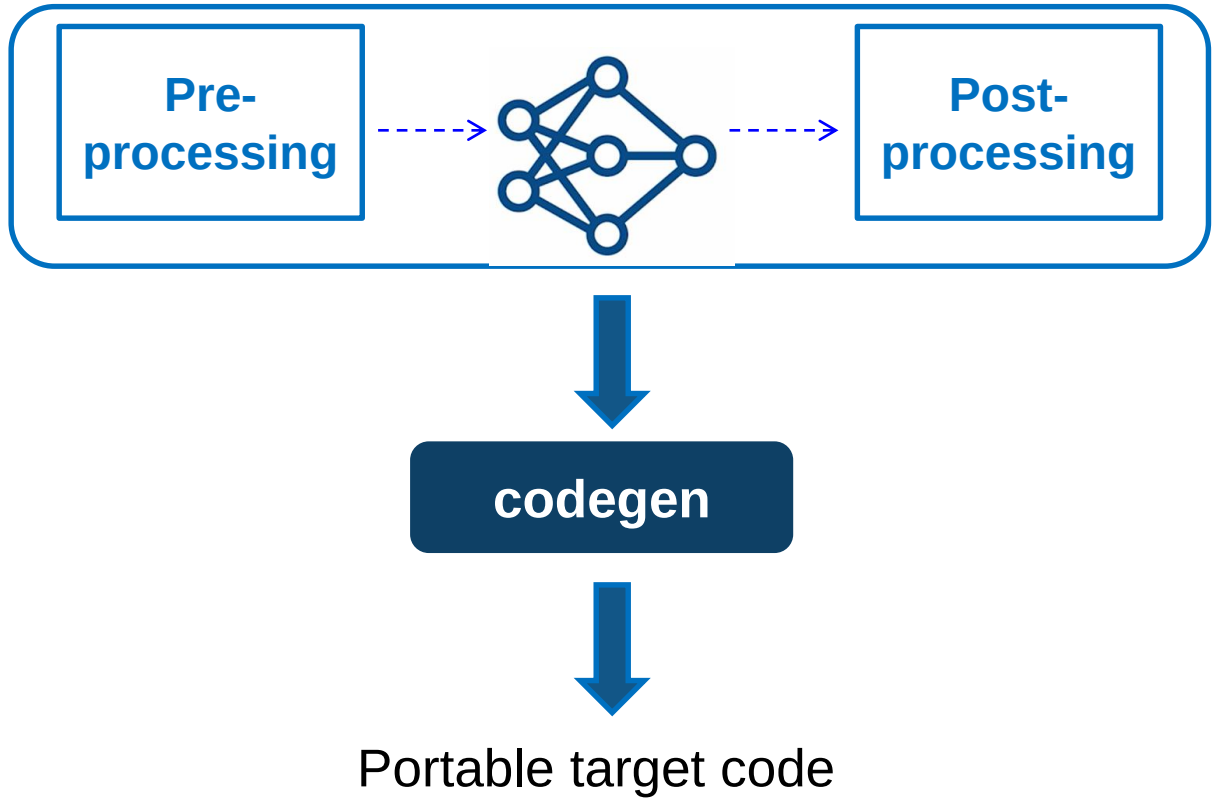


Deep Learning Deployment Workflows

INFERENCE ENGINE DEPLOYMENT

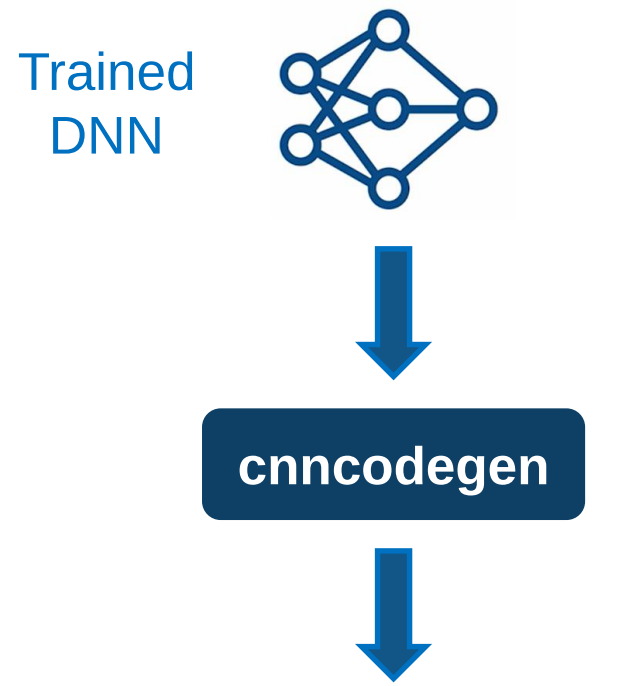


INTEGRATED APPLICATION DEPLOYMENT



Workflow for Inference Engine Deployment

INFERENCE ENGINE DEPLOYMENT



Steps for inference engine deployment

1. Generate the code for trained model

```
>> cnncodegen(net, 'targetlib', 'cudnn')
```

2. Copy the generated code onto target board

3. Build the code for the inference engine

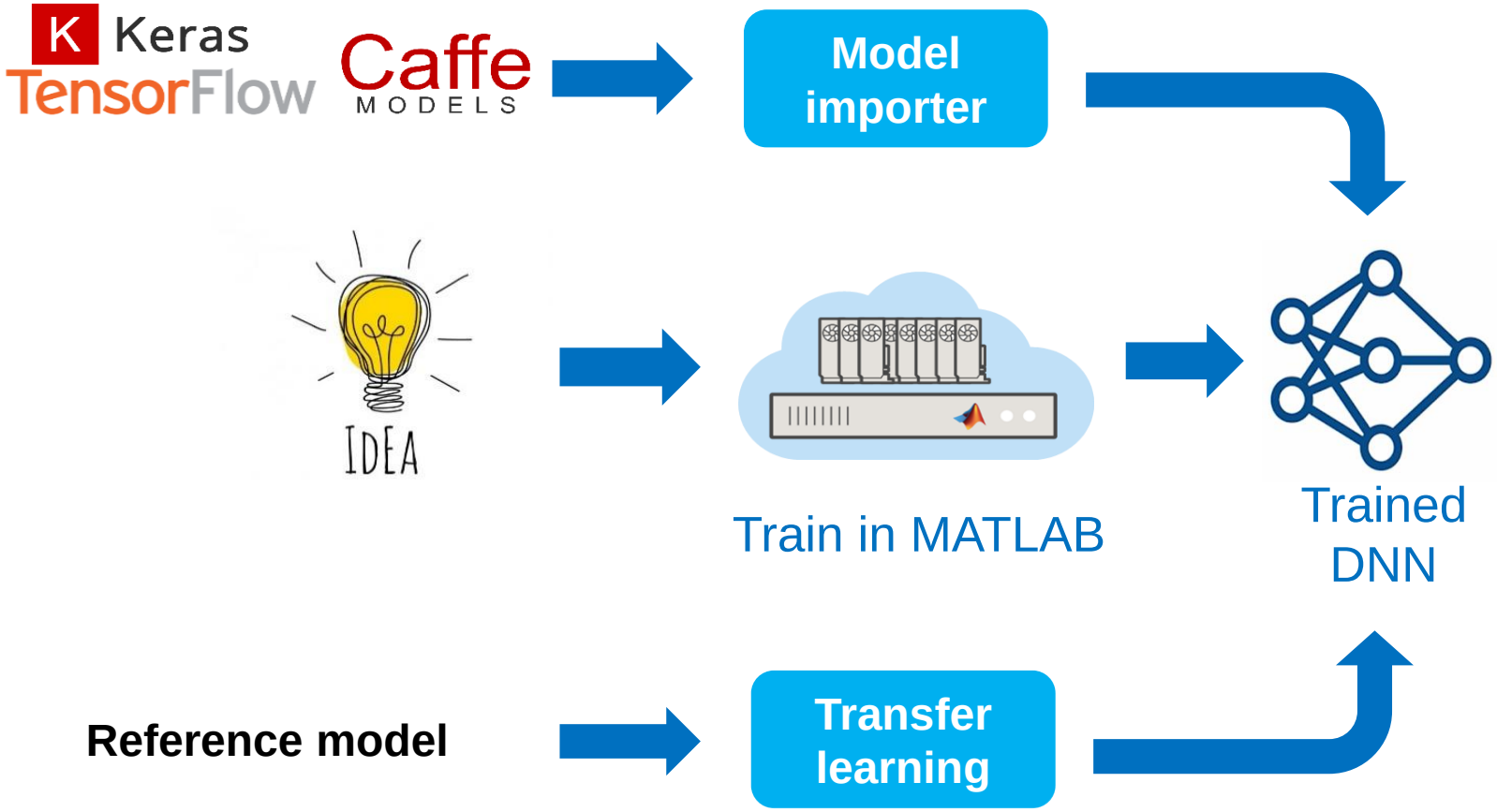
```
>> make -C ./codegen -f …mk
```

4. Use hand written main function to call inference engine

5. Generate the exe and test the executable

```
>> make -C ./ ……
```

How to get a Trained DNN into MATLAB?



Deep Learning Inference Deployment

Target Libraries

Keras
TensorFlow
Caffe
MODELS

Model importer

Train in MATLAB

Trained DNN

Reference model

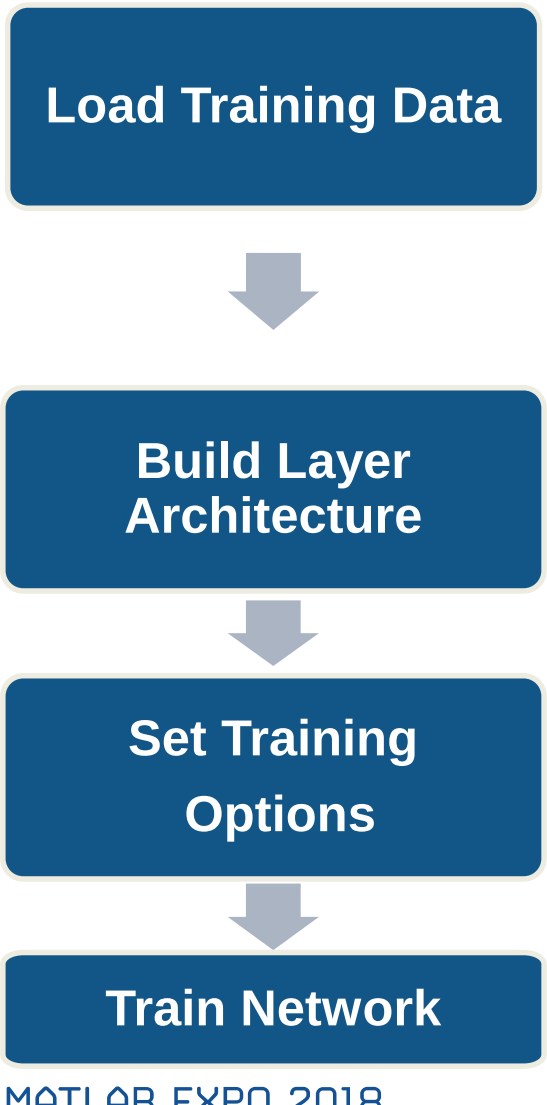
Transfer learning

Intel MKL-DNN Library

NVIDIA TensorRT & cuDNN Libraries

ARM Compute Library

Building DNN from Scratch




```
%% Create a datastore
imds = imageDatastore('Data',...
 'IncludeSubfolders',true,'LabelSource','foldernames');
num_classes = numel(unique(imds.Labels));
%% Build layer architecture
layers = [imageInputLayer([64 32 3])
    convolution2dLayer(5,20)
    reluLayer()
    maxPooling2dLayer(2,'Stride',2)
    fullyConnectedLayer(512)
    fullyConnectedLayer(2)
    softmaxLayer()
    classificationLayer()];
%% Set Training Options
    trainOpts = trainingOptions( 'sgdm',...
    'MiniBatchSize', miniBatchSize,...
    'Plots', 'training-progress');
%% Train Network
net = trainNetwork(imds, layers, trainOpts);
```

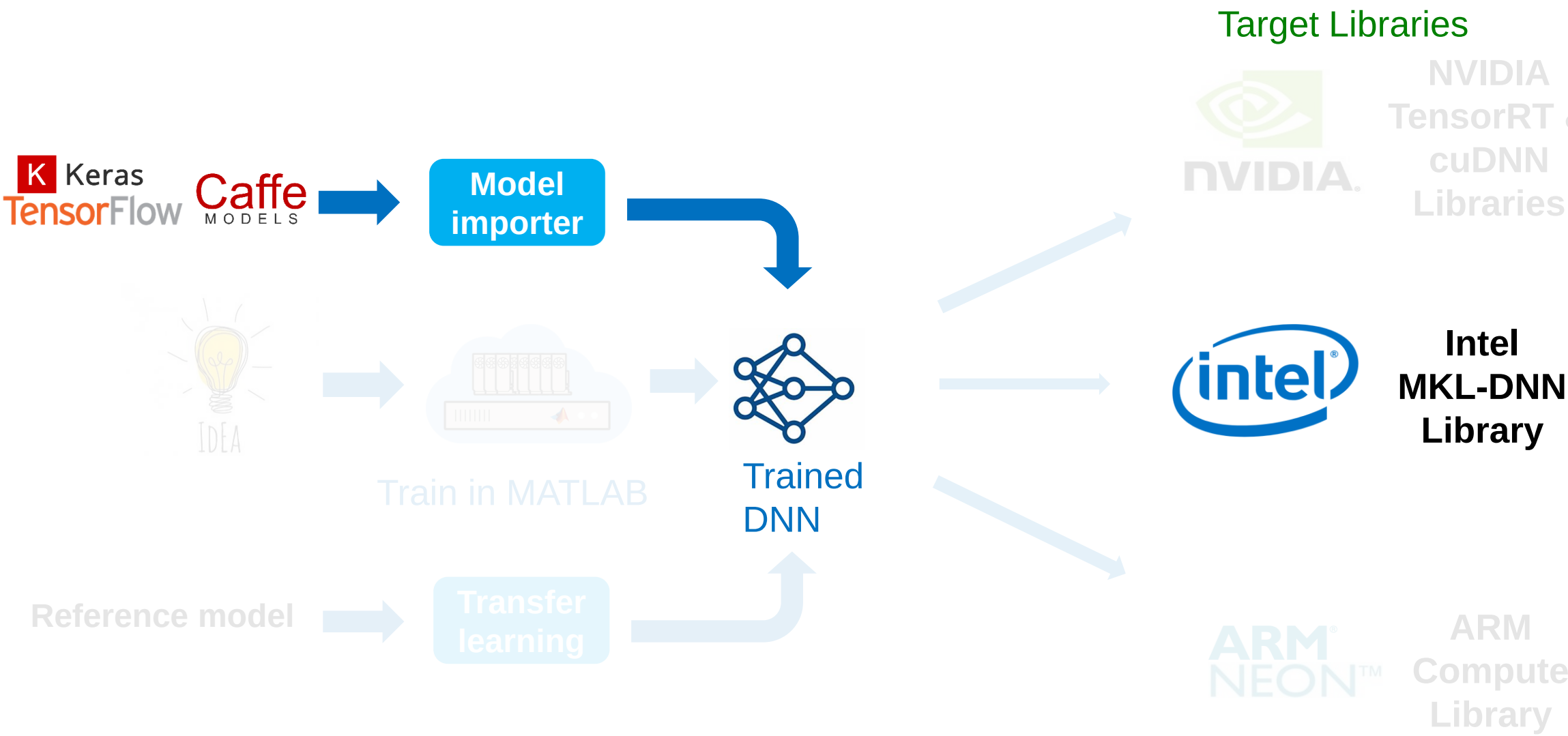
Pedestrian Detection DNN Deployment on ARM Processor

```
layers = [imageInputLayer([64 32 3])
convolution2dLayer(5,20)
reluLayer()
maxPooling2dLayer(2,'Stride',2)
CrossChannelNormalizationLayer(5,'K',1);
convolution2dLayer(5,20)
reluLayer()
maxPooling2dLayer(2,'Stride',2)
fullyConnectedLayer(512)
fullyConnectedLayer(2)
softmaxLayer()
classificationLayer()];
```

Pedestrian Detection DNN Deployment on ARM Processor

- ARM Neon instruction set architecture
  - Example: ARM Cortex A
- ARM Compute Library
  - Low-level Software functions
  - Computer vision, machine learning etc…
- Pedestrian detection on Raspberry pi

Deep Learning Inference Deployment

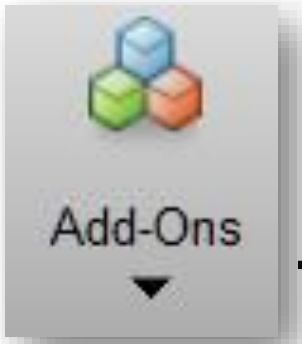


Importing DNN from Open Source Framework

Caffe Model Importer
(including Caffe Model Zoo)

- `importCaffeLayers`
- **`importCaffeNetwork`**

```
network = importCaffeNetwork(protofile, 'yolo.caffemodel');
```



TensorFlow-Keras Model Importer

- `importKerasLayers`
- **`importKerasNetwork`**

KERAS IMPORTER

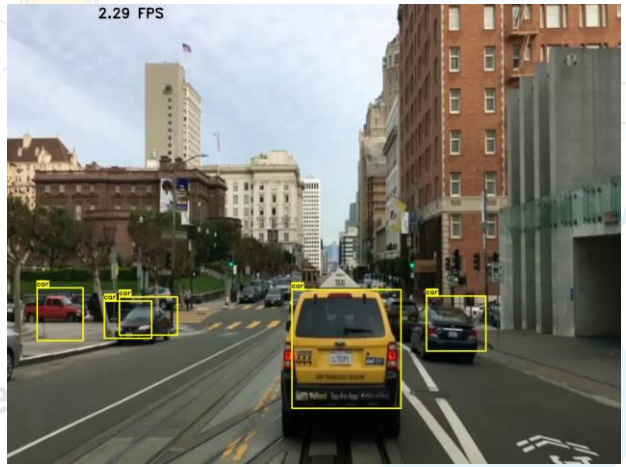
Importer for TensorFlow-Keras Models

Deep Learning Inference Deployment

Keras
TensorFlow
Caffe
MODELS

Model importer

Object Detection



Trained DNN

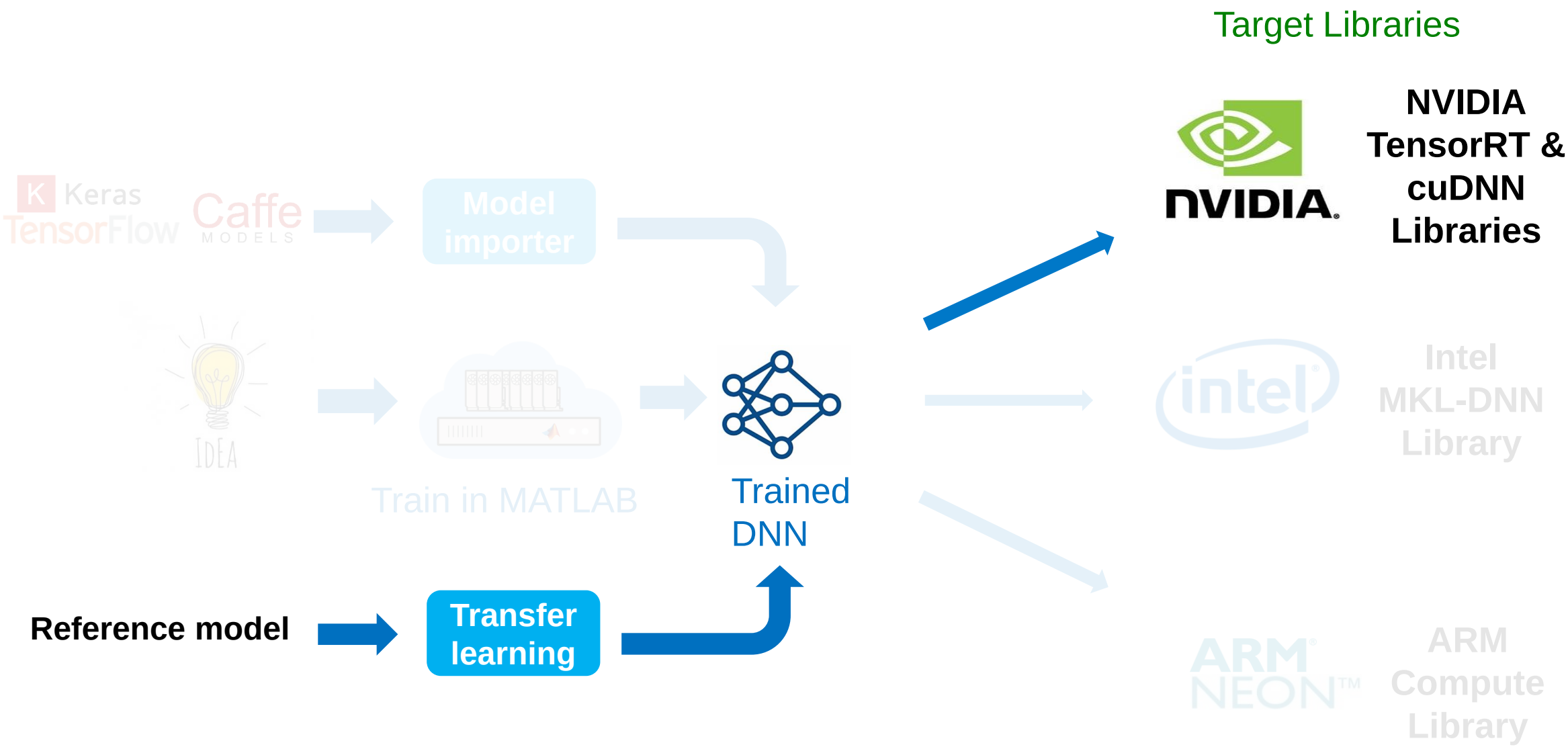
Target Libraries

NVIDIA
TensorRT &
cuDNN
Libraries

Intel MKL-DNN Library

ARM
Compute
Library

Deep Learning Inference Deployment



Deep Learning Inference Deployment

Target Libraries

NVIDIA TensorRT & cuDNN Libraries

Model importer

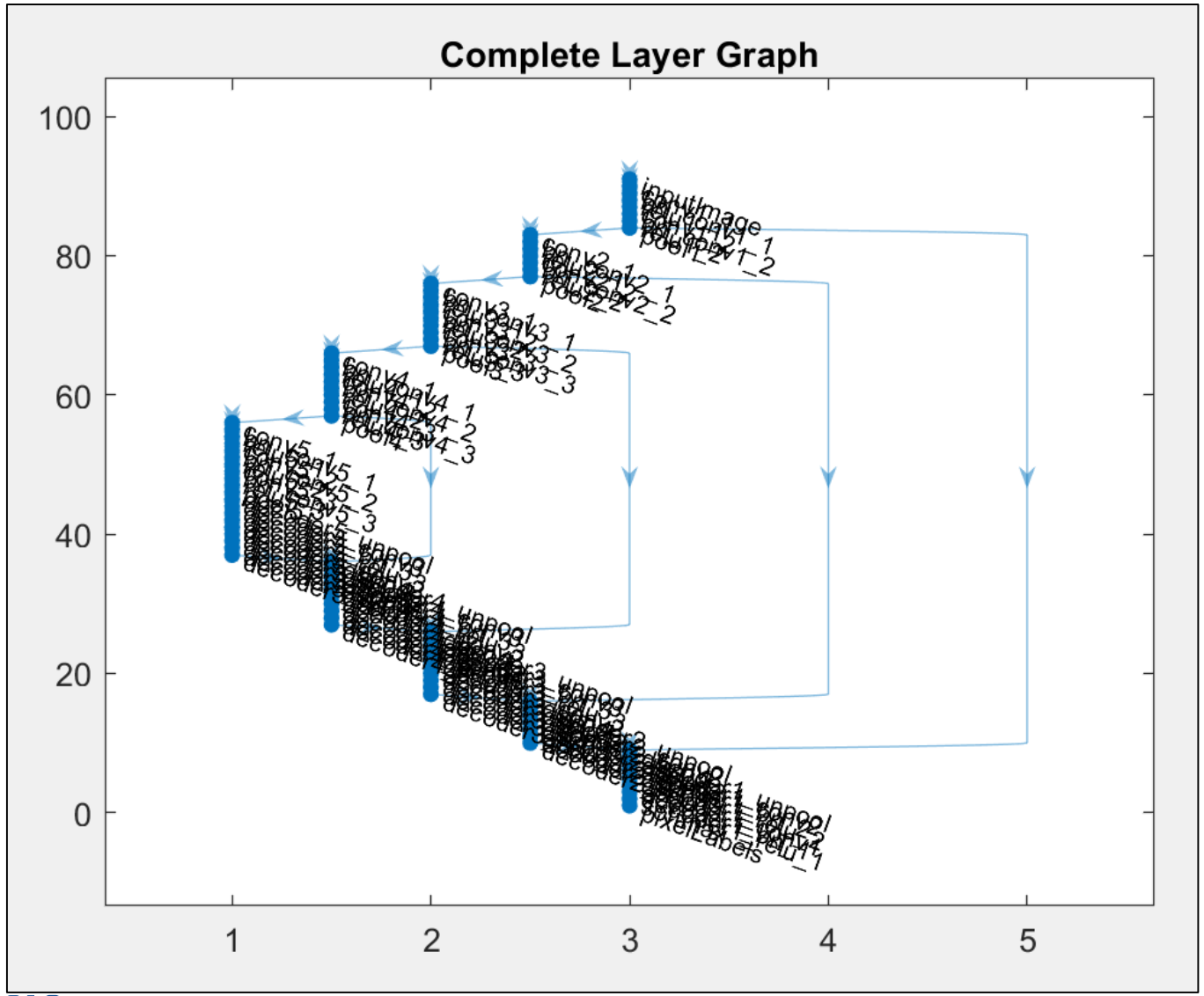
Train in MATLAB

Trained DNN

Reference model

Transfer learning

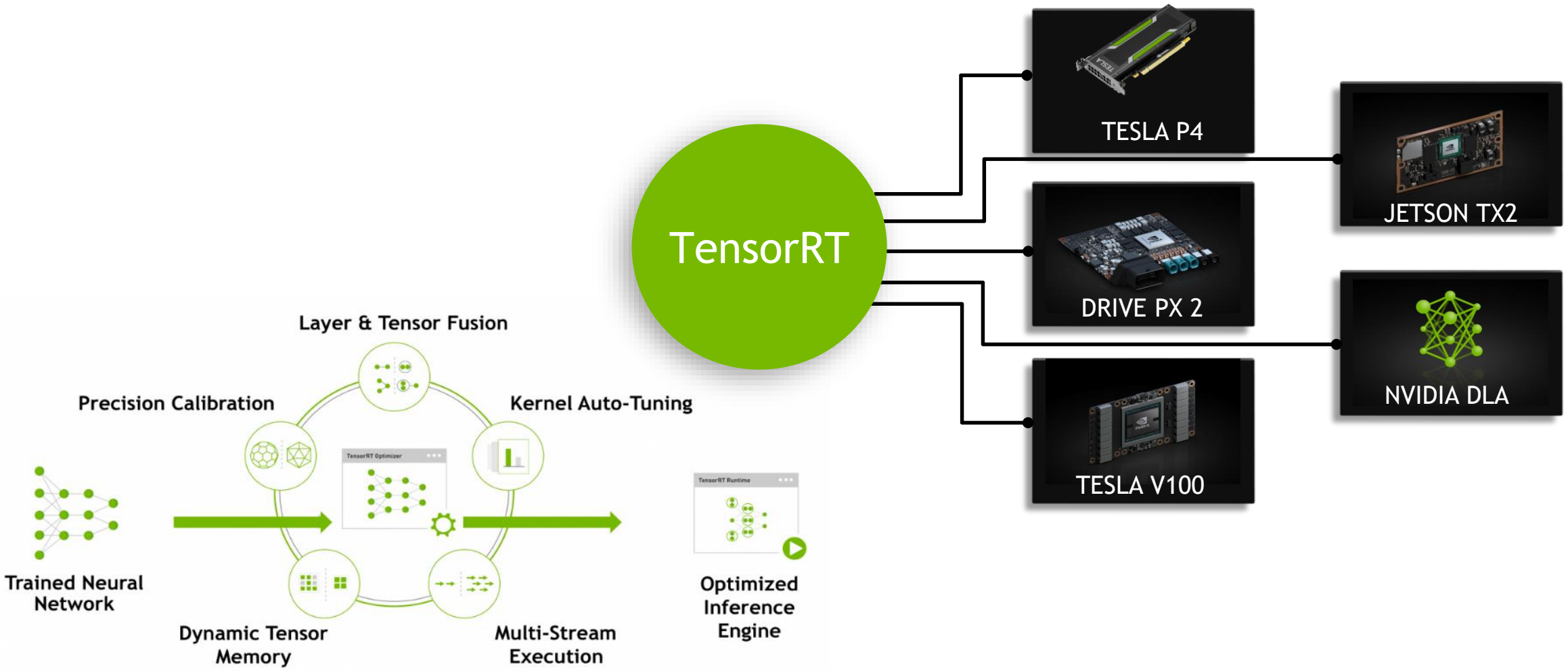
Layered Architecture for Segnet- Semantic Segmentation



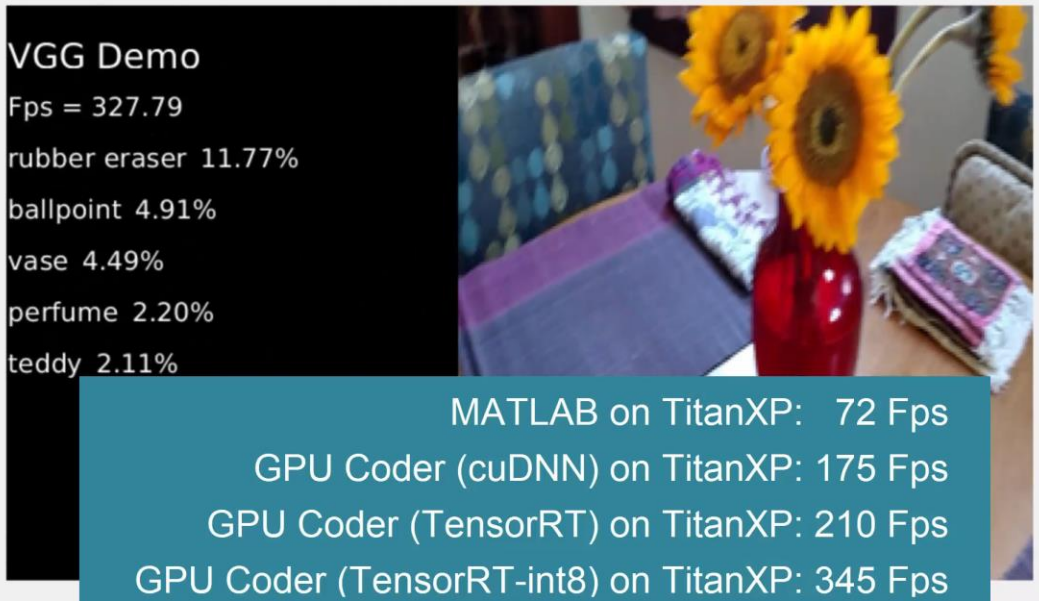
DAG Network
Total number of layers: 91

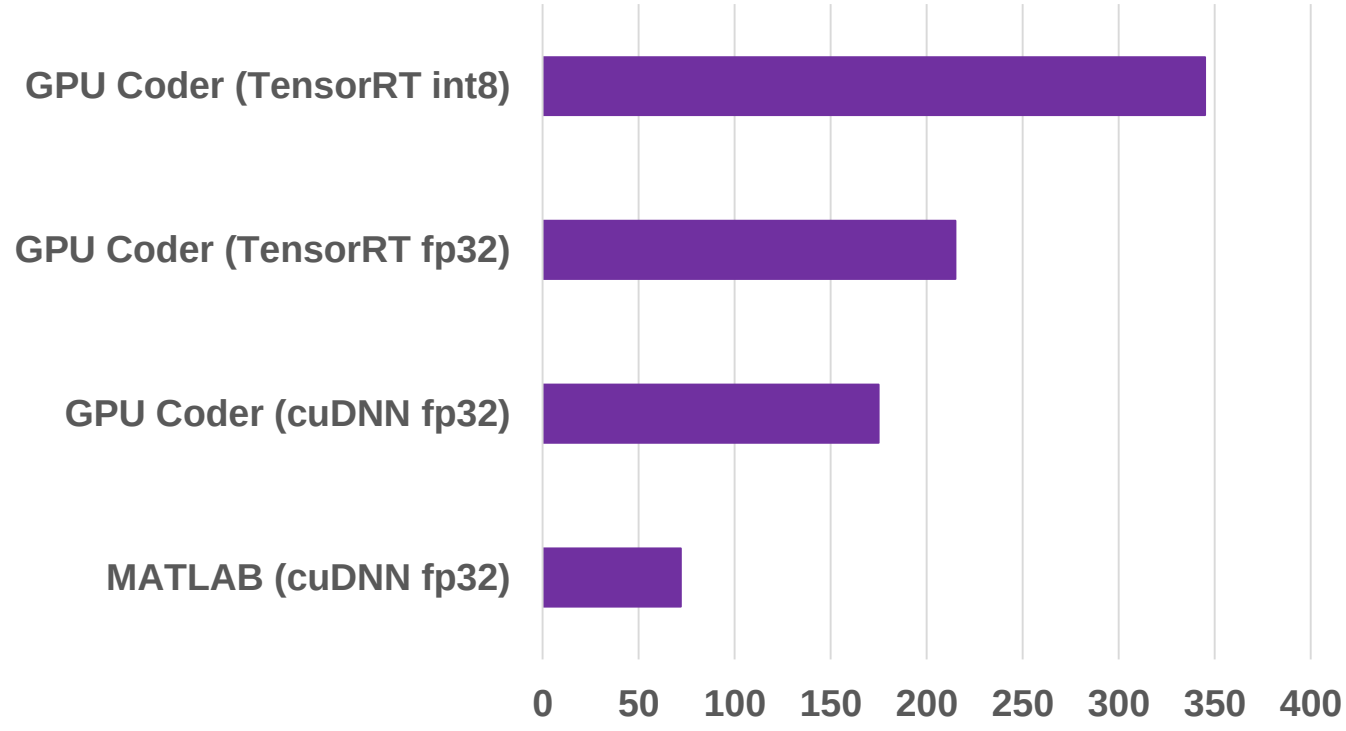
NVIDIA TensorRT

PROGRAMMABLE INFERENCE ACCELERATOR



Performance Summary (VGG-16) on TitanXP



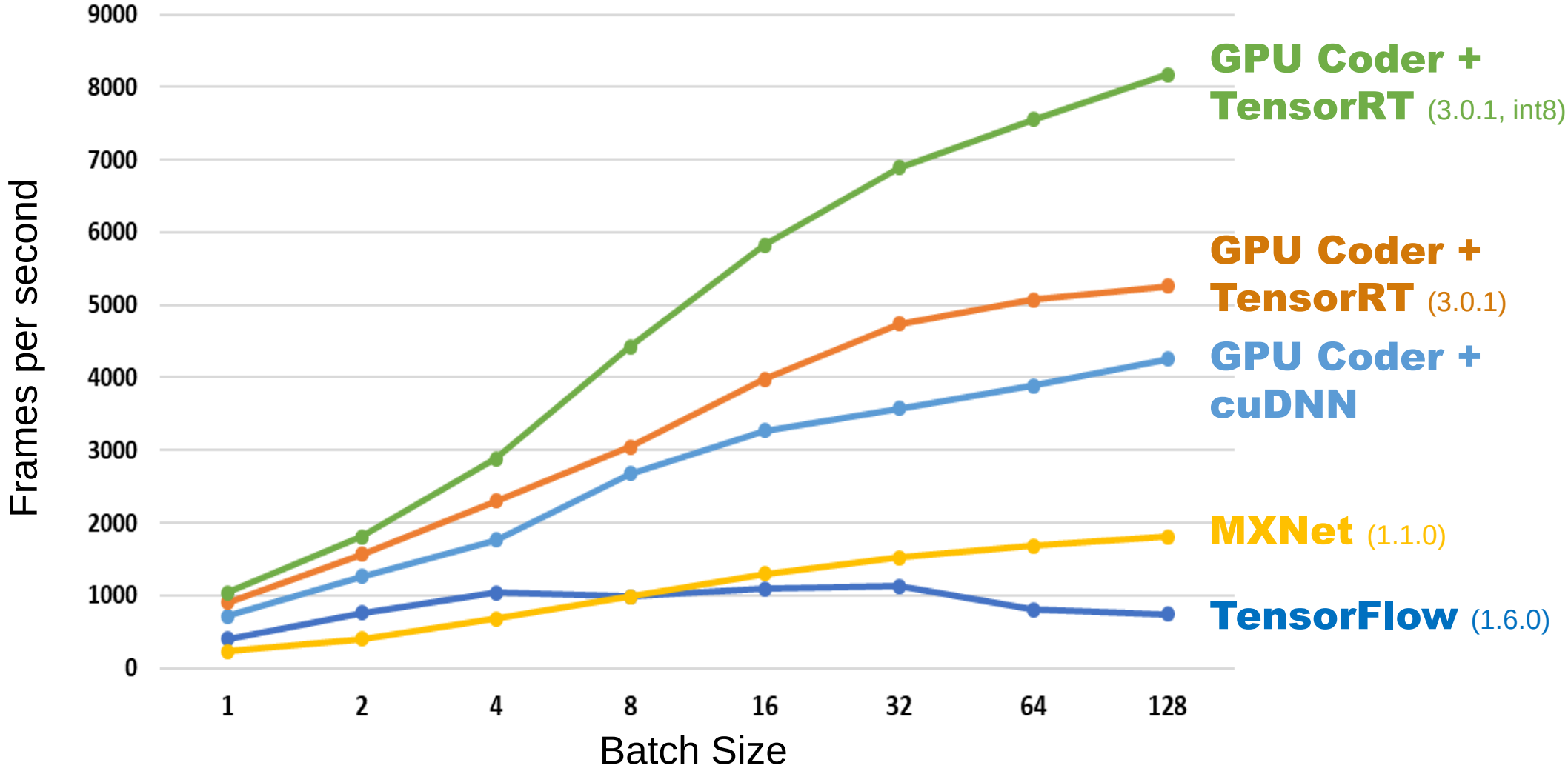


How Good is Generated Code Performance?

- Performance of CNN inference (Alexnet) on Titan XP GPU

- Performance of CNN inference (Alexnet) on Jetson (Tegra) TX2

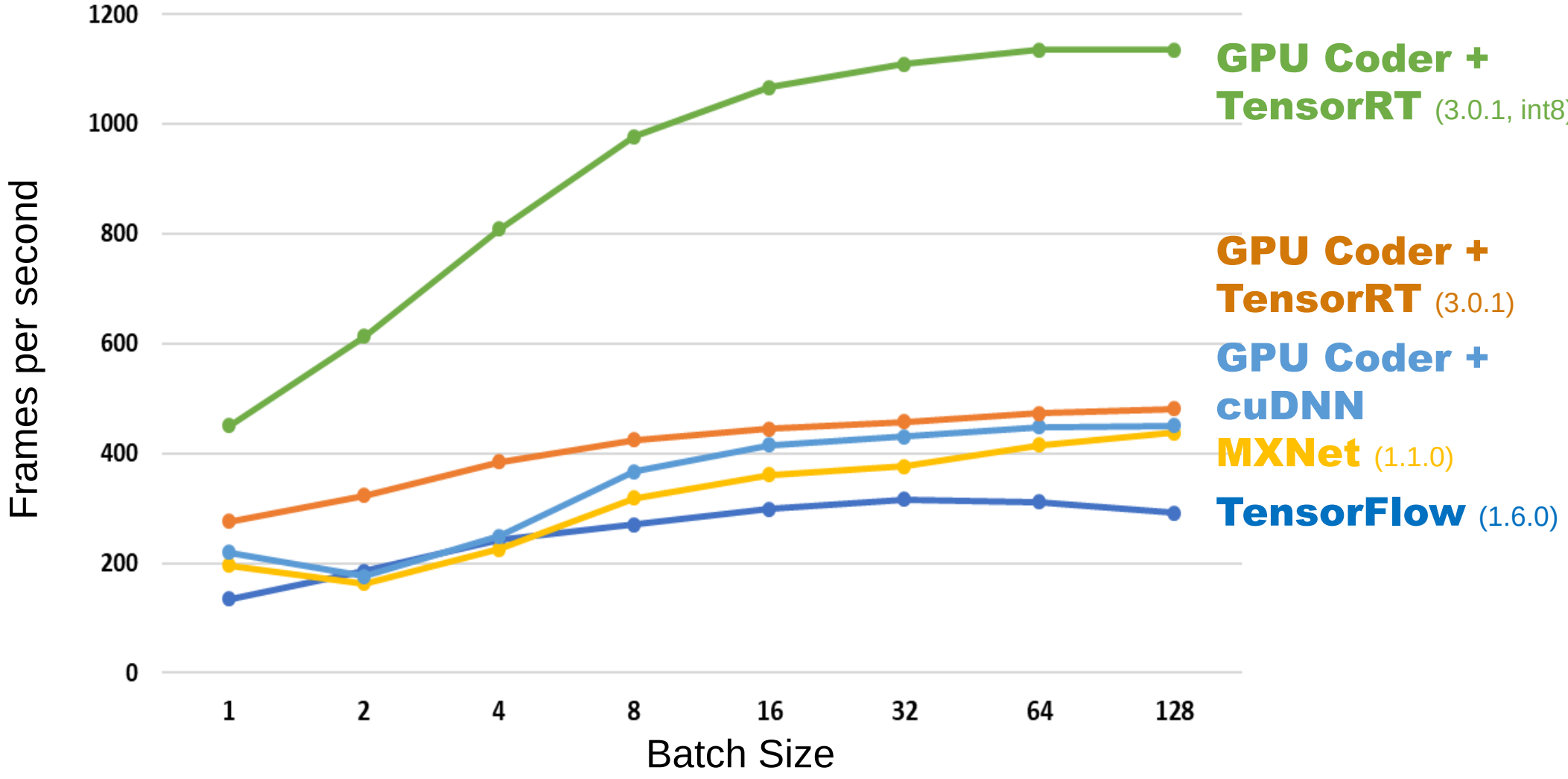
Alexnet Inference on NVIDIA Titan Xp



CPU	Intel(R) Xeon(R) CPU E5-1650 v4 @ 3.60GHz
GPU	Pascal Titan Xp
cuDNN	v7

Testing platform

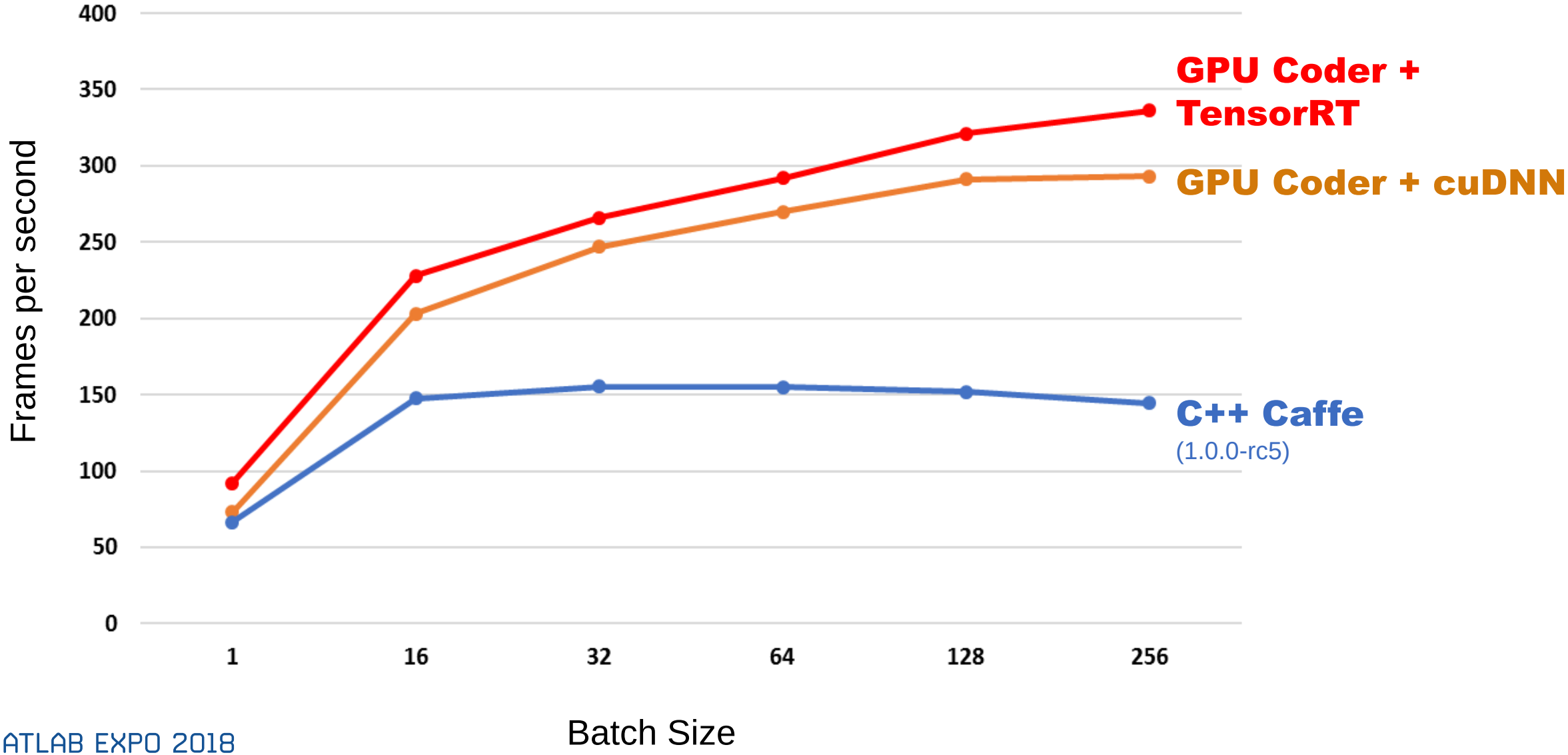
VGG-16 Inference on NVIDIA Titan Xp



CPU	Intel(R) Xeon(R) CPU E5-1650 v4 @ 3.60GHz
GPU	Pascal Titan Xp
cuDNN	v7

Testing platform

Alexnet Inference on Jetson TX2: Frame-Rate Performance



Brief Summary

DNN libraries are great for inference, …

- GPU coder generates code that takes advantage of:

NVIDIA® CUDA libraries, including cuDNN, and TensorRT

Intel® Math Kernel Library for Deep Neural Networks (MKL-DNN)

ARM® Compute libraries for mobile platforms

Brief Summary

DNN libraries are great for inference, ...

- GPU coder generates code that takes advantage of:

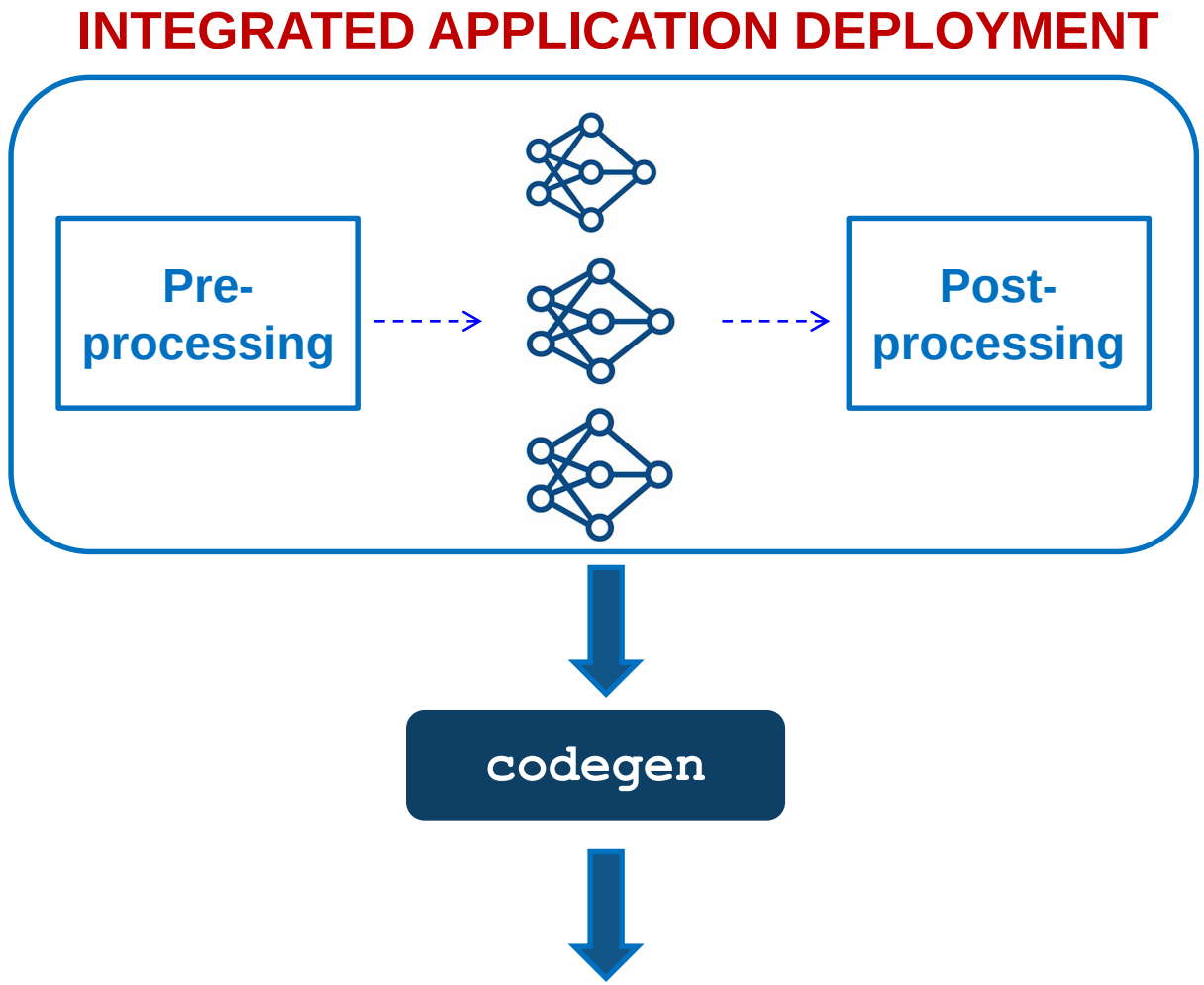
NVIDIA® CUDA libraries, including cuDNN, and TensorRT

but, applications require more than just inference

Intel® Math Kernel Library for Deep Neural Networks (MKL-DNN)

ARM® Compute libraries for mobile platforms

Deep learning Workflows- Integrated Application Deployment



Traffic sign detection and recognition

YOLO

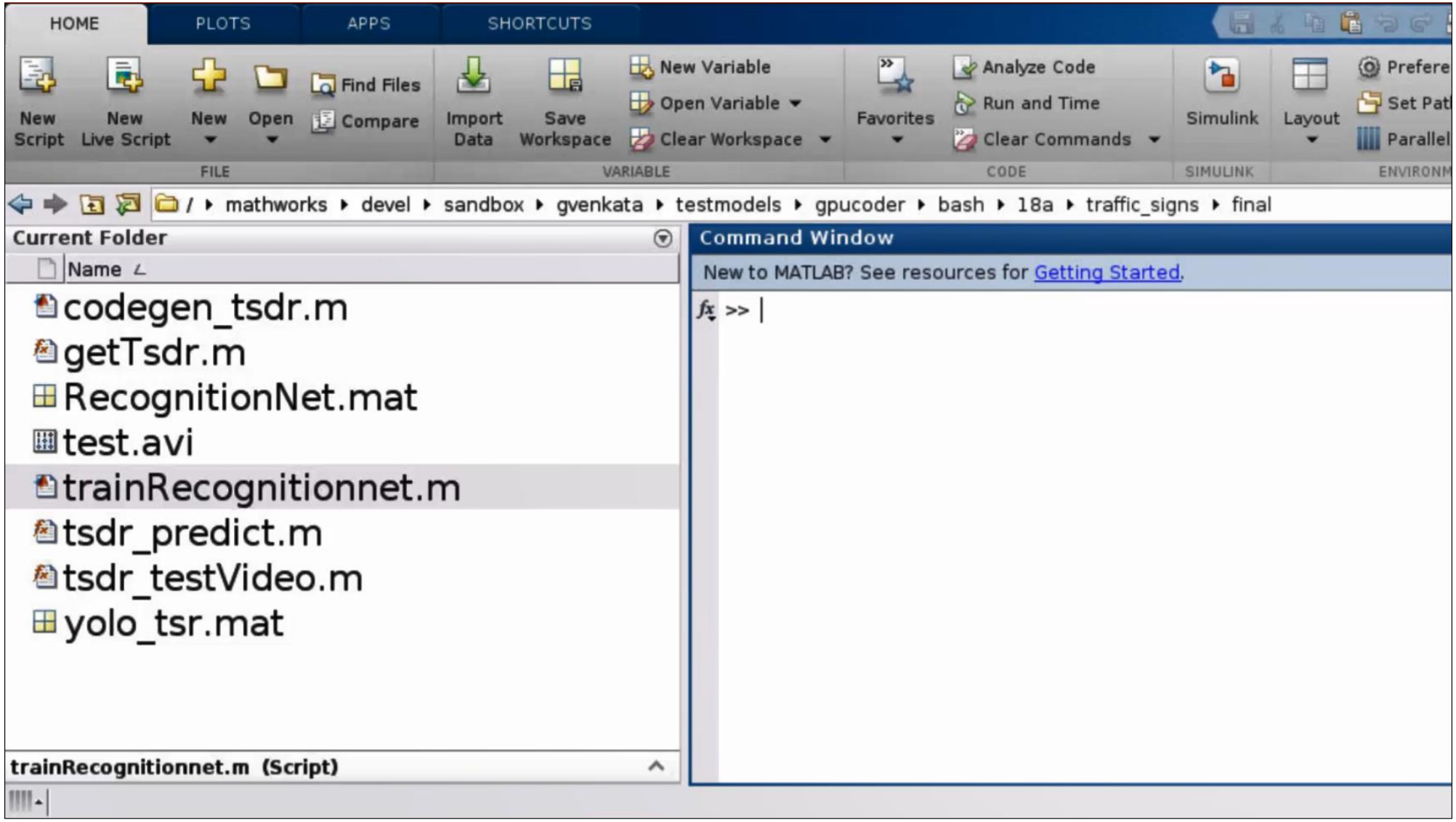
Recognition net

Object detection DNN

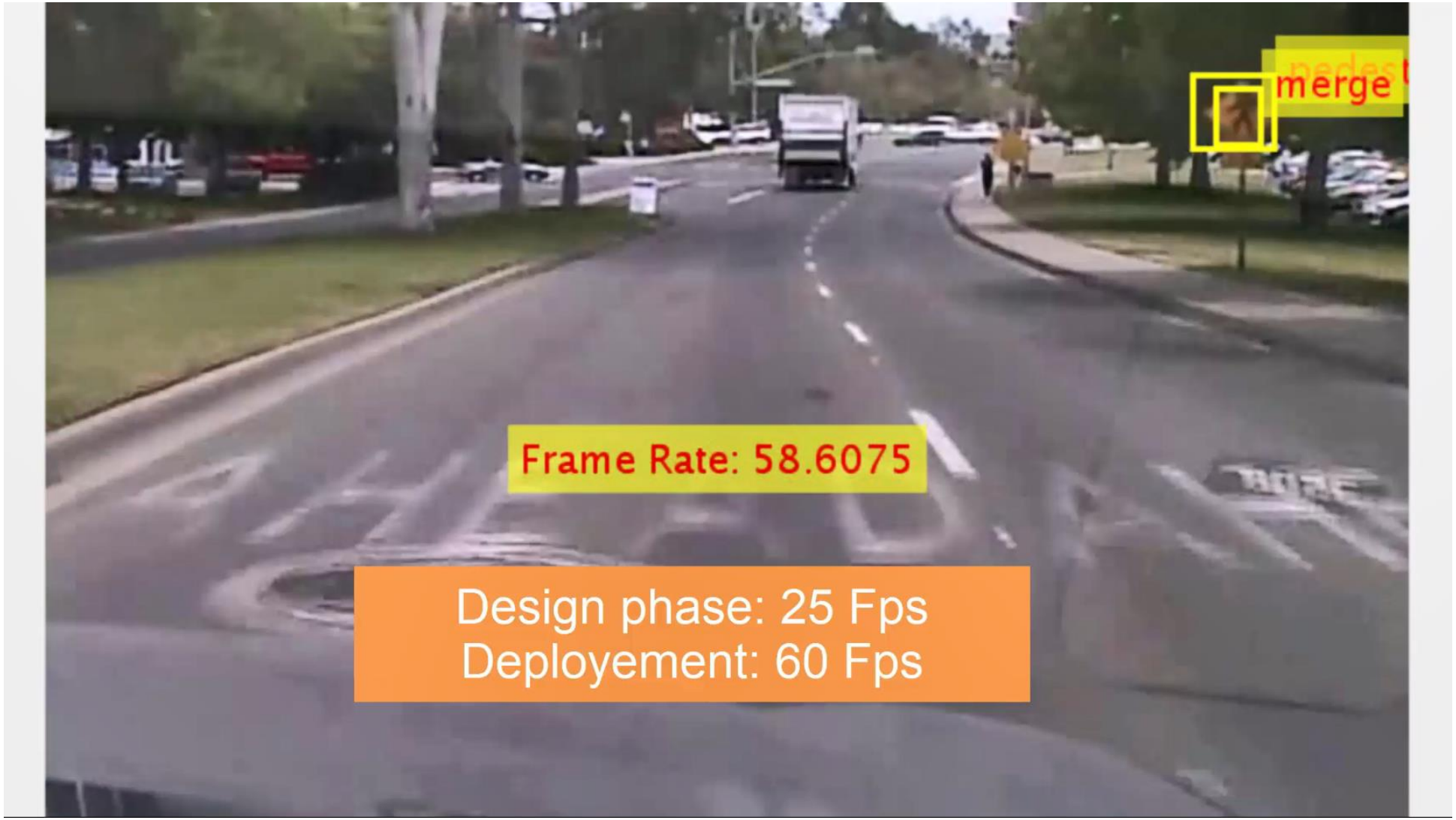
Strongest Bounding Box

Classifier DNN

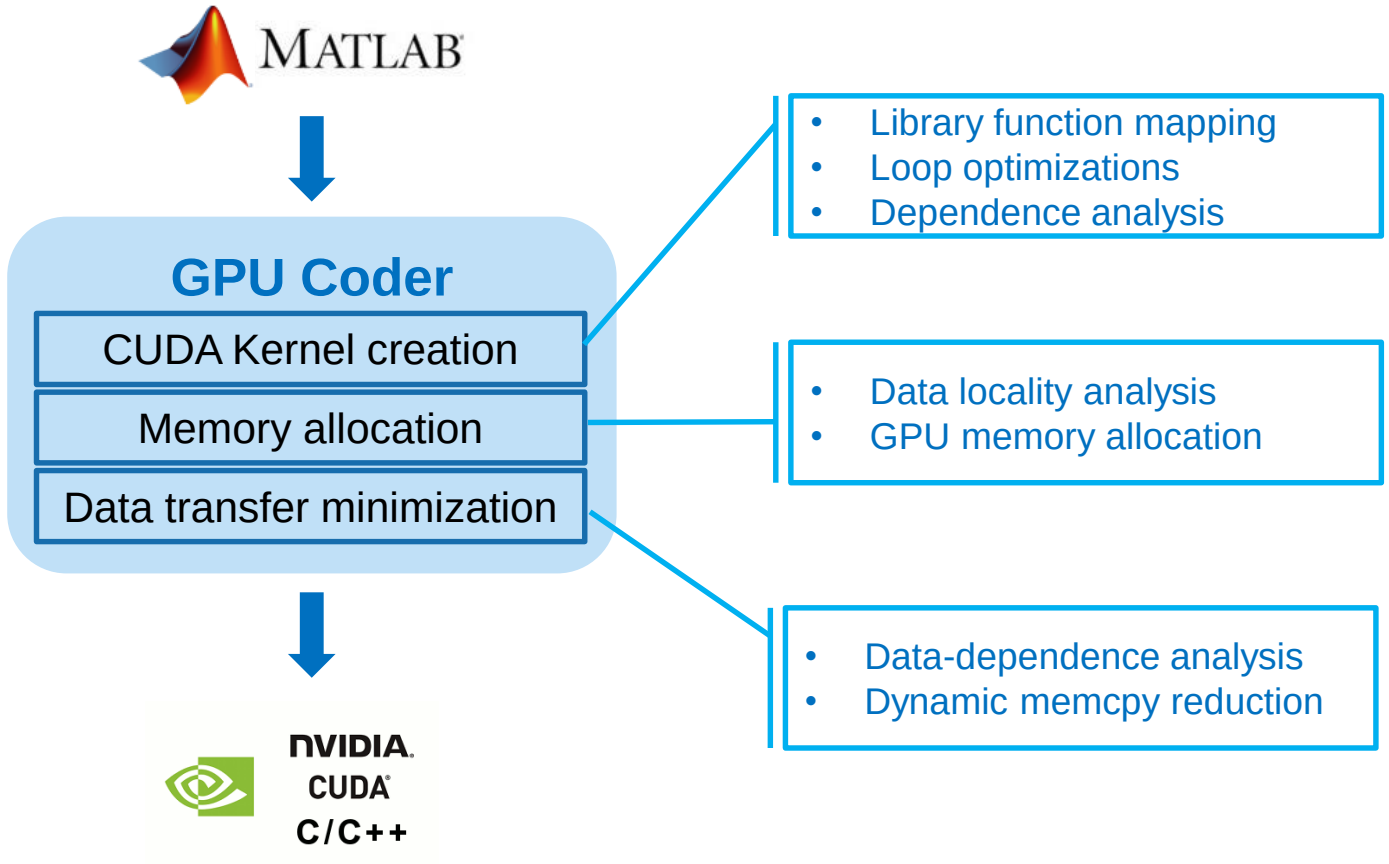
Traffic sign detection and recognition



Traffic sign detection and recognition

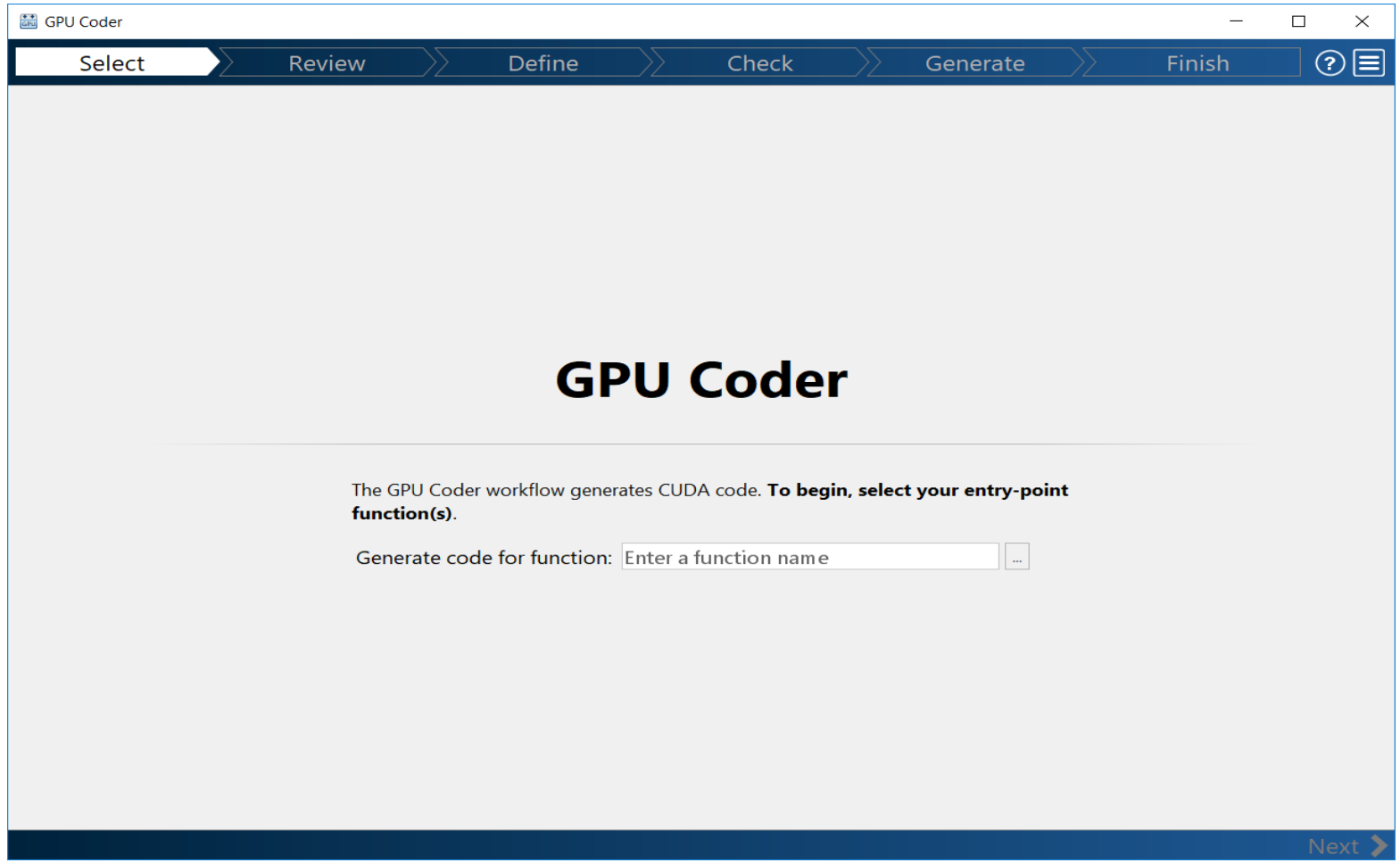


GPU Coder Helps You Deploy Applications to GPUs Faster



CUDA Code Generation from GPU Coder app

Integrated editor and simplified workflow for code generation



Summary- *GPU Coder*

MATLAB algorithm
(functional reference)

GPU Coder

Build type

Call CUDA from MATLAB directly

.mex

Call CUDA from (C++) hand-coded main()

.lib

Call CUDA from (C++) hand-coded main().

Cross-compiled .lib

Desktop GPU

Desktop GPU

Embedded GPU

C++

C++

1 Functional test

2 Deployment unit-test

3 Deployment integration-test

4 Real-time test

MATLAB Deep Learning Framework

- **Manage** large image sets
- **Automate** image labeling
- **Easy access** to models

- **Acceleration** with GPU's
- **Scale** to clusters

DEPLOYMENT

Intel MKL-DNN Library

NVIDIA TensorRT & cuDNN Libraries

ARM Compute Library

Speaker Details

Email: Rishu.Gupta@mathworks.in

LinkedIn: https://www.linkedin.com/in/rishu-gupta-72148914/

Contact MathWorks India

Products/Training Enquiry Booth

Call: 080-6632-5749

Email: info@mathworks.in

- **Share your experience with MATLAB & Simulink on Social Media**
  - Use #MATLABEXPO
  - I use #MATLAB because…… Attending #MATLABEXPO
  - Examples
    - I use #MATLAB because it helps me be a data scientist! Attending #MATLABEXPO
    - Learning new capabilities in #MATLAB and #Simulink at #MATLABEXPO.

- **Share your session feedback:**

  Please fill in your feedback for this session in the feedback form