S9545 - USING THE DEEPSTREAM SDK FOR AI-BASED VIDEO ANALYTICS

Anudeep Nallamothu - NVIDIA Solutions Architect
Andrew Bull - NVIDIA Solutions Architect
AGENDA

• Realtime Streaming Video Analytics
• Framework for Analyzing Video
• Understand the Basics: DeepStream SDK 3.0
• Hardware Platforms
• An Overview of TensorRT 5.0
• Transfer Learning Toolkit
• Build with DeepStream: Example Applications
• Getting Started Resources
REALTIME STREAMING VIDEO ANALYTICS
REALTIME STREAMING VIDEO ANALYTICS FROM EDGE TO CLOUD

Access Control  Managing operations  Parking Management  Traffic Engineering

Retail Analytics  Optical Inspection  Managing Logistics  Content Filtering
FRAMEWORK FOR ANALYZING VIDEO
FRAMEWORK FOR ANALYZING VIDEO

Perception

DATA

REMOTE DISPLAY

MULTIMEDIA APIs

CUDA

TENSORRT, CUDA

MULTIMEDIA APIs

COMPOSITE

LOCAL DISPLAY

DATA

REMOTE DISPLAY

METADATA

PROCESSING

DECODE

PRE-PROCESS

TRACK, DETECT, CLASSIFY

Metadata

Data Analytics
DEEPTream FOR Ai APPLICATION PERFORMANCE AND SCALE

<table>
<thead>
<tr>
<th>Scalability</th>
<th>Perception</th>
<th>Perception - edge to cloud</th>
<th>Perception and Analytics</th>
<th>Solution framework</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Perception</td>
<td>Perception - edge to cloud</td>
<td>Perception and Analytics</td>
<td>Solution framework</td>
</tr>
<tr>
<td></td>
<td>Platform specific APIs</td>
<td>Unified APIs across platforms</td>
<td>Multi-GPU containerized applications</td>
<td>Optical flow</td>
</tr>
<tr>
<td></td>
<td>Streams: Multi (Tesla), single (Jetson)</td>
<td>Multi-streams/ multi-DNNs</td>
<td>360D cameras</td>
<td>Remote display</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Custom graphs</td>
<td>Dynamic stream management</td>
<td>Multi-GPU dynamic orchestration</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>IOT services</td>
<td>Indexed video storage and retrieval</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>Workflow templates for full solutions</td>
</tr>
</tbody>
</table>

DeepStream Next - POR can change
DEEP LEARNING FOR IVA

End-to-end workflow
DEEPSTREAM SDK

**USER APPLICATIONS**

- ACCESS CONTROL
- SMART PARKING
- RETAIL ANALYTICS/CHECKOUT
- INTELLIGENT TRAFFIC SYSTEMS
- LAW ENFORCEMENT

**DEEPSTREAM SDK**

**PLUG-INS (BUILD WITH OPEN SOURCE, THIRD PARTY, NVIDIA)**
- DNN Inference/TensorRT Plug-Ins
- Communications Plug-Ins
- Video/Image Capture and Processing Plug-Ins
- Third-Party Library Plug-Ins

**ANALYTICS: MULTI-CAMERA, MULTI-SENSOR FRAMEWORK**
- DeepStream in Containers, Multi-GPU Orchestration
- Tracking & Analytics Across Large Scale Multi-Camera
- Streaming and Batch Analytics
- Event Fabric

**DEVELOPMENT TOOLS**
- End-to-End Reference Applications
- App Building/Configuration Tools
- Plug-In Templates and Adaptation Guides
- Profiling and Performance Tuning

**TENSORRT**
**MULTIMEDIA APIs / VIDEO CODEC SDK**
**IMAGING & DEWARPING LIBRARY**
**METADATA & MESSAGING**
**NVIDIA CONTAINERS**
**MESSAGE BUS CLIENTS**
**MULTI-CAMERA TRACKING LIB**

**LINUX, CUDA**

**PERCEPTION INFRA - JETSON, TESLA SERVER (EDGE AND CLOUD)**
**ANALYTICS INFRA - EDGE SERVER, NGC, AWS, AZURE**
NVIDIA IVA PLATFORM
Deploy from the edge to the cloud

EDGE / ON-PREMISE
Inference

CORE/CLOUD
Training and Inference

Camera
NVR / APPLIANCE
SERVER
data center

DEEPSTREAM • TENSORRT • JETPACK

JETSON
QUADRO / TESLA
TESLA / DGX
WHAT’S NEW IN DEEPSTREAM 3.0

LATEST GPUs - TESLA T4, JETSON XAVIER

TensorRT 5, CUDA 10

NEW PLUGINS

PLUGIN

LOW LEVEL LIB

GPU

Increased capability and throughput

DYNAMIC STREAM MANAGEMENT

Add, remove, modify streams on the fly

CONNECT EDGE TO CLOUD

Stream and Batch Analytics on Metadata

EASY TO SCALE AND MANAGE

Deploy in Docker Containers

HIGH EFFICIENCY AND THROUGHPUT WITH TLT

TLT model files are plug-n-play
DEEPSTREAM STREAMING ARCHITECTURE

- CAPTURE
- DECODE, CAMERA PROCESS
- SCALE, DEWARP, CROP
- STREAM MGMT
- DETECT & CLASSIFY
- TRACKING
- ON SCREEN DISPLAY
- OUTPUT

Components:

- GigE
- NVDEC
- GPU
- CPU
- GPU
- HDMI
- ISP
- VPA
- TC
- VPA
- SATA
- VIC
- DLA
- CPU
DEEPSTREAM BUILDING BLOCK

- A plugin model based pipeline architecture
- Graph-based pipeline interface to allow high-level component interconnect
- Heterogenous processing on GPU and CPU
- Hides parallelization and synchronization under the hood
- Inherently multi-threaded
**NVIDIA-ACCELERATED PLUGINS**

<table>
<thead>
<tr>
<th>Plugin Name</th>
<th>Functionality</th>
</tr>
</thead>
<tbody>
<tr>
<td>gst-nvvideocoders</td>
<td>Accelerated video decoders</td>
</tr>
<tr>
<td>gst-nvstreammux</td>
<td>Stream aggregator - muxer and batching</td>
</tr>
<tr>
<td>gst-nvinfer</td>
<td>TensorRT based inference for detection &amp; classification</td>
</tr>
<tr>
<td>gst-nvtracker</td>
<td>Reference KLT tracker implementation</td>
</tr>
<tr>
<td>gst-nvosd</td>
<td>On-Screen Display API to draw boxes and text overlay</td>
</tr>
<tr>
<td>gst-tiler</td>
<td>Renders frames from multi-source into 2D grid array</td>
</tr>
<tr>
<td>gst-eglglessink</td>
<td>Accelerated X11 / EGL based renderer plugin</td>
</tr>
<tr>
<td>gst-nvvidconv</td>
<td>Scaling, format conversion, rotation</td>
</tr>
<tr>
<td>Gst-nvdewarp</td>
<td>Dewarping for 360 Degree camera input</td>
</tr>
<tr>
<td>Gst-nvmsgconv</td>
<td>Meta data generation</td>
</tr>
<tr>
<td>Gst-nvmsgbroker</td>
<td>Messaging to Cloud</td>
</tr>
</tbody>
</table>
SCALE WITH DEEPSTREAM IN DOCKER

Discover GPU-Accelerated Containers

Innovate in Minutes, Not Weeks

Stay Up to Date

DEEPSTREAM WITH AZURE IOT
HARDWARE PLATFORMS
NVIDIA T4 UNIVERSAL INERENCE ACCELERATOR

320 Turing Tensor Cores
2,560 CUDA Cores
65 FP16 TFLOPS | 130 INT8 TOPS | 260 INT4 TOPS
16GB | 320GB/s
70 W

H.264 Decode Throughput (Streams)

H.265 Decode Throughput (Streams)
THE JETSON FAMILY
From AI at the Edge to Autonomous Machines

**JETSON NANO**
- 5 - 10W
- 0.5 TFLOPS (FP16)
- 45mm x 70mm
- $129 AVAILABLE IN Q2

**JETSON TX2**
- 7 - 15W
- 1.3 TFLOPS (FP16)
- 50mm x 87mm
- $299 - $749

**JETSON AGX XAVIER**
- 10 - 30W
- 10 TFLOPS (FP16) | 32 TOPS (INT8)
- 100mm x 87mm
- $1099

Multiple devices - Same software

AI at the edge

Fully autonomous machines
<table>
<thead>
<tr>
<th>JETSON NANO</th>
<th>JETSON TX2</th>
<th>JETSON AGX XAVIER</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>GPU</strong></td>
<td>128 Core Maxwell 0.5 TFLOPs (FP16)</td>
<td>256 Core Pascal 1.3 TFLOPS (FP16)</td>
</tr>
<tr>
<td><strong>CPU</strong></td>
<td>4 core ARM A57 @ 1.43 GHz</td>
<td>6 core Denver and A57 @ 2GHz</td>
</tr>
<tr>
<td><strong>Memory</strong></td>
<td>4 GB 64 bit LPDDR4 25.6 GB/s</td>
<td>4 GB 128 bit LPDDR4 51 GB/s</td>
</tr>
<tr>
<td><strong>Storage</strong></td>
<td>16 GB eMMC</td>
<td>16 GB eMMC</td>
</tr>
<tr>
<td><strong>Video Encode</strong></td>
<td>4K @ 30</td>
<td>2x 4K @ 60</td>
</tr>
<tr>
<td></td>
<td>4x 1080p @ 30</td>
<td>4x 4K @ 30</td>
</tr>
<tr>
<td></td>
<td>8x 720p @ 30 (H.264/H.265)</td>
<td>14x 1080p @ 30 (H.264/H.265)</td>
</tr>
<tr>
<td><strong>Power mode</strong></td>
<td>5W</td>
<td>7.5W</td>
</tr>
<tr>
<td><strong>Video Decode</strong></td>
<td>4K @ 60</td>
<td>2x 4K @ 60</td>
</tr>
<tr>
<td></td>
<td>2x 4K @ 30</td>
<td>4x 4K @ 30</td>
</tr>
<tr>
<td></td>
<td>8x 1080p @ 30</td>
<td>14x 1080p @ 30</td>
</tr>
<tr>
<td><strong>Camera</strong></td>
<td>12 (3x4 or 4x2) MIPI CSI-2 DPHY 1.1 lanes (1.5 Gbps)</td>
<td>12 (3x4 or 6x2) MIPI CSI-2 D-PHY 1.2 lanes (30 Gbps)</td>
</tr>
<tr>
<td><strong>WiFi/BT</strong></td>
<td>Requires external chip</td>
<td>Requires external chip</td>
</tr>
<tr>
<td><strong>Display</strong></td>
<td>HDMI 2.0 or DP1.2</td>
<td>HDMI 2.0 or DP 1.2</td>
</tr>
<tr>
<td></td>
<td>eDP 1.4</td>
<td>eDP 1.4</td>
</tr>
<tr>
<td></td>
<td>DSI (1 x2) 2 simultaneous</td>
<td>DSI (2 x4) 3 simultaneous</td>
</tr>
<tr>
<td><strong>UPHY</strong></td>
<td>1 x1/2/4 PCIE</td>
<td>1+ 1 x4 or 1+1+1 x1/x2 PCIe or 3xUSB 3.0</td>
</tr>
<tr>
<td><strong>SATA</strong></td>
<td>None</td>
<td>1x</td>
</tr>
<tr>
<td><strong>Power mode</strong></td>
<td>5W</td>
<td>7.5W</td>
</tr>
<tr>
<td><strong>USB OTG</strong></td>
<td>Not supported</td>
<td>Not Supported</td>
</tr>
<tr>
<td><strong>Mechanical</strong></td>
<td>69.6mm x 45mm 260 pin edge connector, No TTP</td>
<td>87mm x 50mm 400 pin connector, Integrated TTP</td>
</tr>
</tbody>
</table>
JETSON NANO RUNS MODERN AI

Inference

<table>
<thead>
<tr>
<th>Model</th>
<th>Img/sec</th>
</tr>
</thead>
<tbody>
<tr>
<td>Resnet50</td>
<td>40</td>
</tr>
<tr>
<td>Inception v4</td>
<td>0</td>
</tr>
<tr>
<td>VGG-19</td>
<td>0</td>
</tr>
<tr>
<td>SSD</td>
<td>0</td>
</tr>
<tr>
<td>SSD (300x300)</td>
<td>0</td>
</tr>
<tr>
<td>SSD (960x544)</td>
<td>0</td>
</tr>
<tr>
<td>SSD (1920x1080)</td>
<td>0</td>
</tr>
<tr>
<td>Tiny Yolo</td>
<td>0</td>
</tr>
<tr>
<td>Unet</td>
<td>0</td>
</tr>
<tr>
<td>Super resolution</td>
<td>0</td>
</tr>
<tr>
<td>OpenPose</td>
<td>0</td>
</tr>
</tbody>
</table>

Coral dev board (Edge TPU) • Raspberry Pi 3 + Intel Neural Compute Stick 2 • Jetson Nano • Not supported/DNR
NVIDIA TensorRT
From Every Framework, Optimized For Each Target Platform
TENSORRT OVERVIEW

High-performance Deep Learning Inference Engine for Production Deployment

Data

Data Tools

- Labeler

Training

Training Frameworks

- Caffe
- torch
- Theano
- TensorFlow

ONNX

MODEL REPOSITORY

Deployment

Inference Frameworks

- TensorRT
  - Optimized Runtime Engine

We Are Here

DATA CENTER / AUTOMOTIVE / EMBEDDED

MANAGE / AUGMENT

DEEP NEURAL NETWORK

TEST

TRAIN
Inference Optimizer and Runtime

**NVIDIA TensorRT 5**

Data center, embedded & automotive
In-framework support for TensorFlow
Support for all other frameworks and ONNX
TensorRT inference server microservice with Docker and Kubernetes integration
New layers and APIs
New OS support for Windows and CentOS

*New in TRT5*
MODEL IMPORTING

AI Researchers
Data Scientists

Caffe
TensorFlow

Python/C++ API
Model Importer

ML Researchers
Data Scientists

Other Frameworks

Python/C++ API
Network Definition API

Runtime inference
C++ or Python API

Example: Importing a TensorFlow model

```python
import tensorrt as trt
import uff
from tensorrt.parsers import uffparser

G_LOGGER = trt.infer.ConsoleLogger(trt.infer.LogSeverity.INFO)

uff_model = uff.from_tensorflow_frozen_model("frozen_model.pb", "dense_2/Softmax")

parser = uffparser.create_uff_parser()
parser.register_input("input_1", (3, 224, 224), 0)
parser.register_input("dense_2/Softmax")

gEngine = trt.infer.uff_to_trt_engine(G_LOGGER,
                                       uff_model,
parser,
                                       INFEERENCE_BATCH_SIZE,
                                       1<<20,
                                       trt.infer.DataType.FLOAT)

gRuntime = trt.infer.create_infer_runtime(G_LOGGER)
context = gEngine.create_execution_context()
```

developer.nvidia.com/tensorrt
FP16, INT8 PRECISION CALIBRATION

<table>
<thead>
<tr>
<th></th>
<th>FP32 Top 1</th>
<th>INT8 Top 1</th>
<th>Difference</th>
</tr>
</thead>
<tbody>
<tr>
<td>Googlenet</td>
<td>68.87%</td>
<td>68.49%</td>
<td>0.38%</td>
</tr>
<tr>
<td>VGG</td>
<td>68.56%</td>
<td>68.45%</td>
<td>0.11%</td>
</tr>
<tr>
<td>Resnet-50</td>
<td>73.11%</td>
<td>72.54%</td>
<td>0.57%</td>
</tr>
<tr>
<td>Resnet-152</td>
<td>75.18%</td>
<td>74.56%</td>
<td>0.61%</td>
</tr>
</tbody>
</table>

Precision calibration for INT8 inference:
- Minimizes information loss between FP32 and INT8 inference on a calibration dataset
- Completely automatic
WORLD’S MOST PERFORMANT INFERNECE PLATFORM
Up To 36X Faster Than CPUs | Accelerates All AI Workloads

For all three graphs:
- Dual-Socket Xeon Gold 6140 @ 3.6GHz with single GPU as shown
- 19.01-py3 for T4 ResNet-50, 18.11-py3 for T4| TensorRT 5.0 | CPU FP32, P4 & T4: INT8 | Batch Size = 128
TensorRT INTEGRATED WITH TensorFlow
8x faster Inference Than TensorFlow Only

Throughput at < 7ms latency (TensorFlow ResNet-50)

Available in TensorFlow 1.7 and above
https://github.com/tensorflow/tensorflow

* Min CPU latency measured was 70 ms. It is not < 7 ms.
CPU: Skylake Gold 6140, 2.5GHz, Ubuntu 16.04; 18 CPU threads.
Pascal P4; CUDA (384.111; v9.0.176); Batch size: CPU=1, TF_GPU=1 (latency 12 ms) , TF-TRT=4 w/ latency=6ms
TRANSFER LEARNING TOOLKIT
TRANSFER LEARNING TOOLKIT

PRE-TRAINED MODEL → DATA → PYTHON API

- PRUNE
- SCENE ADAPTATION
- ADD CLASSES

EVALUATION
PRUNING
RE-TRAINING
EXPORT

OUTPUT MODEL
End to End NVIDIA Deep Learning Workflow

Pre-Trained model access from NGC * Training & adaptation * Applications ready to integrate with DeepStream

Accelerate time to market and save on compute resources!
Pruning Models

1. Reduce model size and increase throughput
2. Incrementally retrain model after pruning to recover accuracy

**EXAMPLE**

- Network: ResNet 18 4-class (Car, Person, Bicycle, Road sign)
- Memory size: 46.2 MB to 6.7 MB
- FPS: 16 fps to 30 fps

![Diagram showing memory size and average frame rate/second comparison between unpruned and pruned networks.](image)

- **6.5x Model Size Reduction**
- **>2x Throughput Increase**
# FEATURES

<table>
<thead>
<tr>
<th>Efficient Pre-trained Models</th>
<th>Faster Inference with Model Pruning</th>
<th>Training with Multiple GPUs</th>
</tr>
</thead>
<tbody>
<tr>
<td>GPU-accelerated models trained on large scale public datasets.</td>
<td>Model pruning reduces size of the model resulting in faster inference</td>
<td>Re-training models, adding custom data for multi GPU training using an easy to use tool</td>
</tr>
<tr>
<td>Abstraction</td>
<td>Containerization</td>
<td>Integration</td>
</tr>
<tr>
<td>Abstraction from having deep knowledge of frameworks, simple intuitive interface to the features</td>
<td>Packaged in a container easily accessible from NVIDIA GPU Cloud website. All code dependencies are managed automatically</td>
<td>Models exported using TLT are easily consumable for inference with Deep Stream SDK</td>
</tr>
</tbody>
</table>
BUILD WITH DEEPSTREAM: EXAMPLE APPLICATIONS
NVIDIA ENDEAVOR - SMART GARAGE SOLUTION
DEEPSTREAM 3.0 END-TO-END APPLICATION

PERCEPTION - MULTI-GPU APPS

EVENTS AND MESSAGING

ANALYTICS - MULTI-CAMERA ANALYTICS AND TRACKING FRAMEWORK

Containers

Perception graph → Metadata

Containers

Perception graph → Metadata

Stream Processing

Batch Processing

REST APIs

Search & Query

NoSQL DB

Search Indexer

Static Orchestration and management
PERCEPTION GRAPH

COMM PLUGIN
- RTSP

PREPROCESSING PLUGINS
- Decoder
- Dewarp library
  - Camera calibration

DETECTION, CLASSIFICATION & TRACKING PLUGINS
- Detection and classification
  - ROI calibration
  - Global positioning
- Tracker

COMMUNICATIONS PLUGINS
- Transmit Metadata
- Analytics server

360d feeds
- Dewarping

ROI: Lines

ROI: Polygon
ENABLING 360D CAMERA PROCESSING

- Equirectangular
- Cylindrical
- Panini
- Perspective
- Rotated cylinder
- Pushbroom

Tesla only
DYNAMIC STREAM MANAGEMENT

1. Add/ Remove camera streams
2. Change FPS
3. Change resolutions
THIRTY STREAMS
MULTI-STREAM REFERENCE APPLICATION
REFERENCE APPLICATION VIDEO
START DEVELOPING WITH DEEPSTREAM

DEEPSTREAM . EXPLORE METROPOLIS . SUPPORT FORUMS
ONLINE RESOURCES

• NVIDIA DeepStream SDK
  • Product Page

• Blogs
  • Breaking the Boundaries of Intelligent Video Analytics with DeepStream SDK 3.0
  • Multi-Camera Large-Scale Intelligent Video
  • Using Calibration to Translate Video Data to the Real World
  • Accelerating Intelligent Video Analytics using Transfer Learning Toolkit
  • Accelerate Video Analytics Development with DeepStream SDK 2.0

• Webinars
  • Use Nvidia’s DeepStream and Transfer Learning Toolkit to Deploy Streaming Analytics at Scale
  • Streamline Deep Learning for Video Analytics with DeepStream SDK 2.0
ONLINE RESOURCES

• **Forums**
  - Tesla Forum
  - Jetson Forum

• **Software**
  - DeepStream Container for Tesla and Sample Applications
  - JetPack (installer to flash your Jetson Developer Kit)
  - TensorRT

• **GitHub Repositories**
  - Reference Apps for Video Analytics using TensorRT 5 and DeepStream SDK 3.0
  - An Example of Using DeepStream SDK for Redaction
  - DeepStream 3.0 - 360 Degree Smart Parking Application

• **Gstreamer Plugin and Application Development Guide**
  - https://gstreamer.freedesktop.org/documentation/
Getting Started: Transfer Learning Toolkit

- Try Transfer Learning Toolkit. Access Open Beta today!
- Deploy end to end IVA solution with NVIDIA DeepStream 3.0. Download DeepStream 3.0
- Sign up for NVIDIA Developer Zone to access downloads, documentation and user tutorials
- Blogs:
  - What is Transfer Learning?
  - Pruning Models with Transfer Learning Toolkit
  - Accelerate IVA Applications with Transfer Learning Toolkit