Embedded GPGPU and Deep Learning for Industrial Market

Author:
Dan Mor
GPGPU and HPEC Product Line Manager
September 2018

Table of Contents

1. Introduction

In today's world, when NVIDIA CUDA and Deep-learning have become de-facto tools for parallel GPGPU computing, many industries are trying to catch up and move towards incorporating this exciting technology.

These are the emerging market segments, with fast-growing investments and deployment of sophisticated technologies:

- Industrial Automation
- Security and Surveillance
- Aerospace Equipment and Systems
- Boating and Marine Industry
- Smart Cities

Manufacturers in these segments are looking for small, low-power assemblies with powerful data processing capabilities--surveillance, image processing, communications and video analytics--outdoor/indoor industrial systems.

In this white paper, we will look at the Industrial Market verticals and show how they can benefit from embedded GPGPU technology.

2. Difficulties in Current Embedded Industrial Market Systems

Let's examine five major verticals in the Industrial market, and try to find a common denominator for them.

2.1 Industrial Automation

Industrial automation is the use of control systems (such as computers or robots), and information technologies, for handling different processes and machinery in an industry, in order to replace a human worker. Today's Industrial Automation complex processes involve image processing and video analytics. These processes include, but are not limited to:

Image Classification, Image Location, Image Segmentation

2.2 Security and Surveillance

2.2.1 Smart Cities

In the current digital era, security cameras are common in most industries around the globe. The industrial market is not an exception. These cameras monitor inside and outside of buildings (Industrial Indoor/Outdoor), record/stream video from remote locations or analyze high quality video to identify suspicious persons and activities. These processes include but are not limited to:

Image Classification, Image Location, Image Segmentation

2.3 Aerospace Equipment/Systems, Boating and Marine Industry

Demand for industrial and civilian air and marine programs is growing continuously. This means that industrial helicopters, aircraft and marine systems are also in demand. These systems always include some type of image/graphic processing, starting from capturing images/video, weather condition analysis, and ending up with complex navigation systems. These processes include but are not limited to:

Image Classification, Image Location, Image Segmentation

Looking at above Industrial market verticals clearly shows that "*Image Classification, Image Location, Image Segmentation*" are the common denominator for all these market branches.

Let's drill down into the "common denominator".

- Image Classification - pattern or object recognition and identifying classes
- Image Location - locating and extracting image coordinates, finding where in the video those objects are located
- Image Segmentation - locate objects boundaries lines, curves, etc. in images/video

All the above processes involve heavy-duty calculation power. If you attempt this kind of calculation using conventional CPU or FPGAs, the results very quickly become:

CPU Choking - Slow OS

↓

Complex CPU/System Load Balancing

↓

CPU Upgrading and Over Clocking

↓

Power Consumption and Heat Dissipation Issues

“*Image Classification, Image Location and Image Segmentation*” –are perfect candidates for deploying NVIDIA Deep-learning inference networks (which can benefit from hundreds of parallel CUDA cores calculations). The industrial market is always looking for Small Form Factor (SFF) and Size, Weight and Power (SWaP) optimized systems, so let’s see if we can find the perfect GPGPU system for the above market verticals.

3. Embedded GPGPU Deep Learning Process

Once we understand that we need to process our data using "*Image Classification, Image Location and Image Segmentation*" algorithms, it becomes pretty obvious that we need to use NVIDIA Deep-learning inference networks.

Deep-learning is a subset of AI (Artificial Intelligence) and Machine Learning that uses multi-layered artificial neural networks to deliver state-of-the-art accuracy in tasks such as object detection, speech recognition, language translation and others.

Before diving into Deep-learning, here are some definitions:

- DNN - Deep Neural Network
- NVIDIA DIGITS – Deep-learning GPU Training System - is a web application for training Deep-learning models. The currently supported frameworks are: Caffe, Torch, and Tensorflow. These are used to interactively train network models on annotated datasets in the cloud or PC.
- NVIDIA TensorRT™ - is a high-performance Deep-learning inference optimizer and runtime platform.
- CUDA® - is a parallel computing platform and programming model developed by NVIDIA for general computing on graphical processing units (GPGPU)
- Inference - live data evaluation at runtime. Predicts and applies reasoning based on the examples it learned
- NVIDIA Jetson TX2/TX2i/Xavier - embedded AI computing device
- Docker - Computer program that performs operating-system-level virtualization, also known as "containerization
- NVIDIA docker - Docker containers leveraging NVIDIA GPUs

There is a very good, comprehensive guide made by NVIDIA, which explains in detail how to deploy a "Jetson-inference" open source project. There is a link to this project in the "References" section. Such a project can be confusing for those not familiar with GPGPUs and Deep-learning. This document will try to make it simple, so readers without any previous experience in programming can understand the process, flow, and benefits of this technology.

First, let's understand the data process flow and identify main the process nodes in a Deep-learning architecture.

Step 1 - **NVIDIA DIGITS** – Webb App, training Deep Learning

NVIDIA DIGITS - How it works?

So it's a web application, where I can find it?

You need to install it. Since Deep Learning is a performance-intensive process, you need a PC with an NVIDIA GPU (which will use CUDA parallel computing API model). If you have this kind of PC, you can install NVIDIA DIGITS on it. Local installation requires a lot of package dependencies, so for beginners, it is recommended to use "NVIDIA GPU Cloud"(NGC)" known as "nvidia-docker". During "nvidia-docker" installation, you will install a Linux container, with all the SW required to run "NVIDIA-DIGITS". NVIDIA adds to this Linux container an ability to use local PC NVIDIA GPU and expose it to this container.

Step 2 - **NVIDIA Jetson SoM** – Embedded System on Module

NVIDIA Jetson is an embedded AI computing device. It has as a core, ARM CPUs and NVIDIA GPU. Jetson is used to deploy the runtime inference in the field. Jetson supports TensorRT, with which you can optimize neural network models, calibrate for lower precision with high accuracy, and finally deploy the models to hyper scale data centers, embedded or automotive product platforms. TensorRT-based applications on GPUs perform up to x100 faster than a regular CPU during inference for models trained in all major frameworks.

Once you load Deep Learning model from DIGITS to NVIDIA Jetson platform, you can use this model as a parameter to NVIDIA example applications which perform "*Image Classification, Image Location, Image Segmentation*".

Step 3 - Input Camera – Video Camera Input

Input camera is used for capturing an image or real-time video stream. It can be any camera, HD-SDI, Composite, etc. It should support V4L2 interface to be able to easily integrate specific cameras into existing NVIDIA samples.

Step 4 - Captured Frame – Video frame captured from the

Captured frame is transferred to system memory for the further GPGPU and inference process.

Step 5 - Classified Object – Object analyzed by Deep Learning

This is the final process – inference.

Once we have an image in memory, we can use Deep-learning model to classify, locate or segment the image.

Conceptual block diagram for above process is shown below.

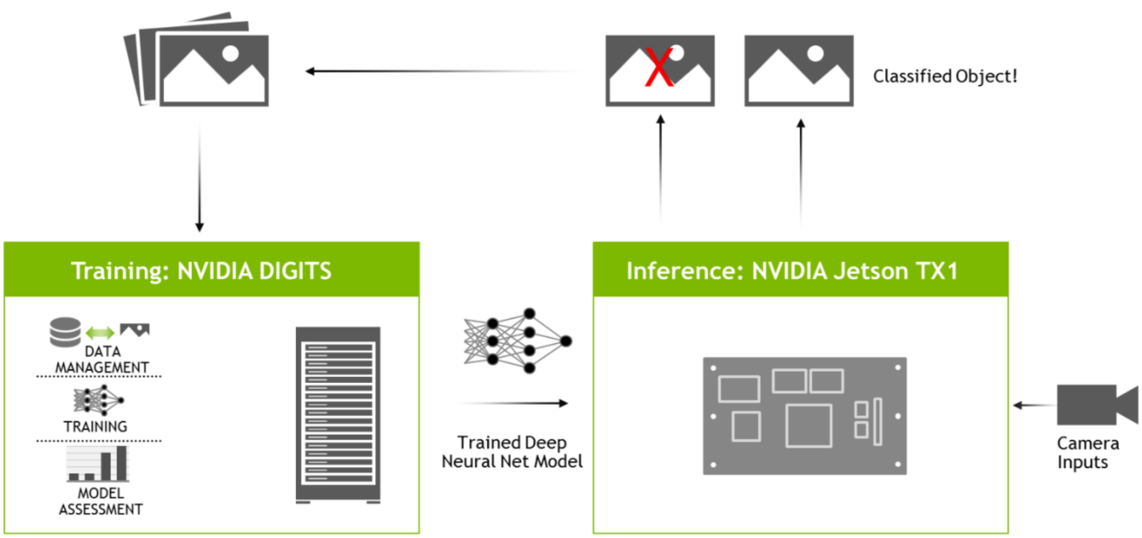

Figure 1 - Jetson-Inference block diagram

4. Embedded GPGPU Deep-learning benefits in Industrial Market

- SFF System (based on NVIDIA Jetson SoM) can be used for “Image Classification, Image Location, Image Segmentation” processes

- SWaP optimized system - NVIDIA Jetson is low power consumption device

- NVIDIA CUDA can be used as a common platform framework, this mean that the customer can share already trained Deep-learning models between devices

- Different use cases can be easily covered by simply retrain the model with different data sets

- Detection precision can be constantly approved by fault detection, adding additional images to the data set and retrain model

- Provision for next generation systems – more powerful GPU (embedded SoM) will be available, this mean more performance, this mean more complex and precise detection and

5. Use Cases

There are many use cases which can be covered by Embedded GPGPU Deep-learning in Industrial Markets, below are the five major verticals:

- Industrial Automation
- Security and Surveillance
- Smart Cities
- Aerospace Equipment and Systems
- Boating and Marine Industry

6. Aitech Embedded GPGPU Industrial Approach

Having seen all these benefits from Industrial GPGPU and Deep-learning, you are probably wondering if there is an embedded system which can leverage all this wonderful technology...

There is such a system!

Figure 2 - A177 Twister - GPGPU Fanless SFF Industrial Supercomputer

6.1 A177 Twister - GPGPU Fanless SFF Industrial Supercomputer

The A177 Twister is the smallest and most powerful GPGPU, ideally suited for next-generation industrial indoor/outdoor systems, Unmanned Aircraft Systems (UAS), Autonomous Ground Systems (UGV), Surveillance, Autonomous Robotic Systems, Smart Cities, Industrial Automation, Boating and Marine Industry and many other use cases.

Benefits	Aitech A177 System
Industrial Grade System	✓
IP67	✓
GPGPU	✓
Plenty of calculation power	✓
CUDA Framework	✓
NVIDIA Deep-learning	✓
SWaP Optimized	✓
SFF and Heat Dissipation	✓
Powered by NVIDIA Jetson TX2/TX2i/Xavier	✓
Image Classification, Image Location, Image Segmentation	✓

7. References

Jetson-interference open source project by NVIDIA:
https://github.com/dusty-nv/jetson-inference

nvidia-docker
https://github.com/NVIDIA/nvidia-docker

LXC vs Docker Comparison
https://robinsystems.com/blog/containers-deep-dive-lxc-vs-docker-comparison/

Image Folder Specification (for NVIDIA DIGITS Deep Learning process)
https://github.com/NVIDIA/DIGITS/blob/master/docs/ImageFolderFormat.md

8. Contact Details

Dan Mor

Product Line Manager

Tel: +972(9) 960-0648
Cell: +972(54) 795-0035
Fax: +972(9) 954-4315
E-mail: dmor@rugged.com

Company website: www.rugged.com

Embedded Computing *without* Compromise