Project Proposal Guide – MATHWORKS TRACK

Disclaimer:

The sample proposal below is to give an idea of how a proposal should be formatted. Our main objective is to illustrate the Artificial Intelligent section - we would like to see an overall system of your proposed project. In this document we have also provided information on what Artificial Intelligent is. Please go though it carefully.

Note that this proposal is not representative of the level of complexity/innovativeness of a winning project. Also, we would expect a more elaborate level of literature review.

You may email Siti Safwana, Application Engineer (siti.safwana@techsource-asia.com) if you have any questions. Please mention IMDC 2020 in your subject header.

Project Title: Real-time Object Detection on GPU

Abstract
In recent years, Convolutional Neural Network (CNN) models on graphics processing units (GPU) have been increased with limitation of computing power and memory resource encourages numbers of studies to develop efficient model design. In this proposed work, Nvidia Jetson TX2 as main platform of GPU to test accuracy of real-time object detection using CNN model architecture (e.g. Yolo, Yolo2, Alexnet). Others than accuracy, computational execution time also will be tested with involved CNN model architecture. For main software platform, MATLAB software will be used together with Deep Learning Toolbox, Parallel Computing Toolbox, GPU Coder and MATLAB Coder.

1. Introduction

Humans glance at an image and instantly know what objects are in the image, where they are, and how they interact. The human visual system is fast and accurate, allowing us to perform complex tasks like driving with little conscious thought. Fast, accurate algorithms for object detection would allow computers to drive cars without specialized sensors, enable assistive devices to convey real-time scene information to human users, and unlock the potential for general purpose, responsive robotic systems [1].

2. Objective

Objective of this project is to analysis:

- The accuracy of Alexnet, YOLO, YOLO2 on real-time implementation.
- Computational time (FPS) of the involved CNN model architecture in real-time implementation.

3. Project methodology

Deep Learning was (and still is) commonly used for object detection if big data involved. It is often used to analyse individual words, whereas deep learning can be applied to complete sentences, greatly increasing its accuracy.

This project can be divided into:

- Labelling data – label classes/groups of objects to detect
- Training data – training the labelled data in involved CNN model architecture
- Deploy the model into hardware platform (Nvidia Jetson TX2)

Deep Learning Toolbox

| Labelling data | Training data | Hardware deployment |

Figure 1 shows the overall project process.

Labelling Data

Labelling typically takes a set of unlabelled data and augments each piece of that unlabelled data with meaningful tags that are informative. For example, labels might indicate whether a photo contains an object, animals etc [2].

Training Data

Training the labelled data using selected CNN model architecture. For this work, YOLO, YOLO2 and Alexnet are the CNNs' model. To get better accuracy result, some hypermeters need to tune.

Hardware Deployment

Once the best result achieves, code from MATLAB environment will convert to CUDA language using GPU Coder Toolbox. The CUDA code will deploy into Nvidia Jetson TX2 to use in real-time environment object detection.

4. Expected results

Accuracy result of input image (object) from camera. The input image later will be processed by Nvidia Jetson TX2 board where the training CNN model already deploy to the hardware. For accuracy measurement, the CNN model was evaluating in MATLAB software using Deep Learning Toolbox while computational time measurement (FPS) in deployment hardware. Figure 2 shows the overall project implementation. Other than that, this project expected to deliver finding below:

- Novel theories & New findings
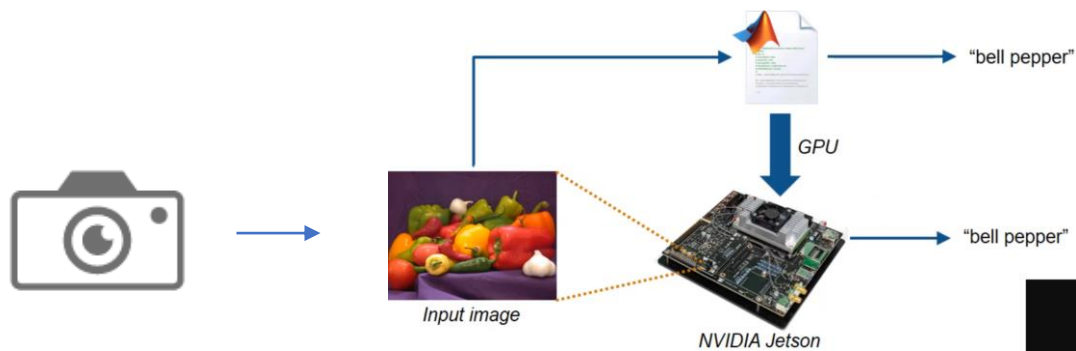- Specific or Potential Applications
- Impact on Society

Figure 2: Overall project implementation process

5. References

[1]   J. Redmon, S. Divvala, R. Girshick, and A. Farhadi, "You only look once: Unified, real-time object detection," in *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, 2016.

[2]   "Labeled data - Wikipedia." [Online]. Available: https://en.wikipedia.org/wiki/Labeled_data. [Accessed: 13-Sep-2019].

Deep Learning with GPU Coder

In a typical Convolutional Neural Networks (CNN) workflow, you start with constructing a CNN architecture by using the Deep Learning Toolbox™ and train the network in tandem with the Parallel Computing Toolbox™. Alternatively, you can import a ConvNet already trained on a large dataset and transfer the learned features. Transfer learning implies taking a CNN trained for one set of classification problems and retraining it to classify a different set of classes. Here the last few layers of the CNN are relearned. Again, Parallel Computing Toolbox is used in the learning phase. You can also import a trained CNN network from other frameworks like Caffe or MatConvNet into a SeriesNetwork object.

Once you have obtained the trained network, you can use GPU Coder™ to generate CUDA® code that can be deployed to an embedded platform such as NVIDIA® Tegra® TK1, TX1, or TX2. The generated code calls the cuDNN and TensorRT libraries (when specified) to leverage high performance.

Once you have obtained the trained network, you can use GPU Coder™ to generate CUDA® code that can be deployed to an embedded platform such as NVIDIA® Tegra® TK1, TX1, or TX2. The generated code calls the cuDNN and TensorRT libraries (when specified) to leverage high performance.
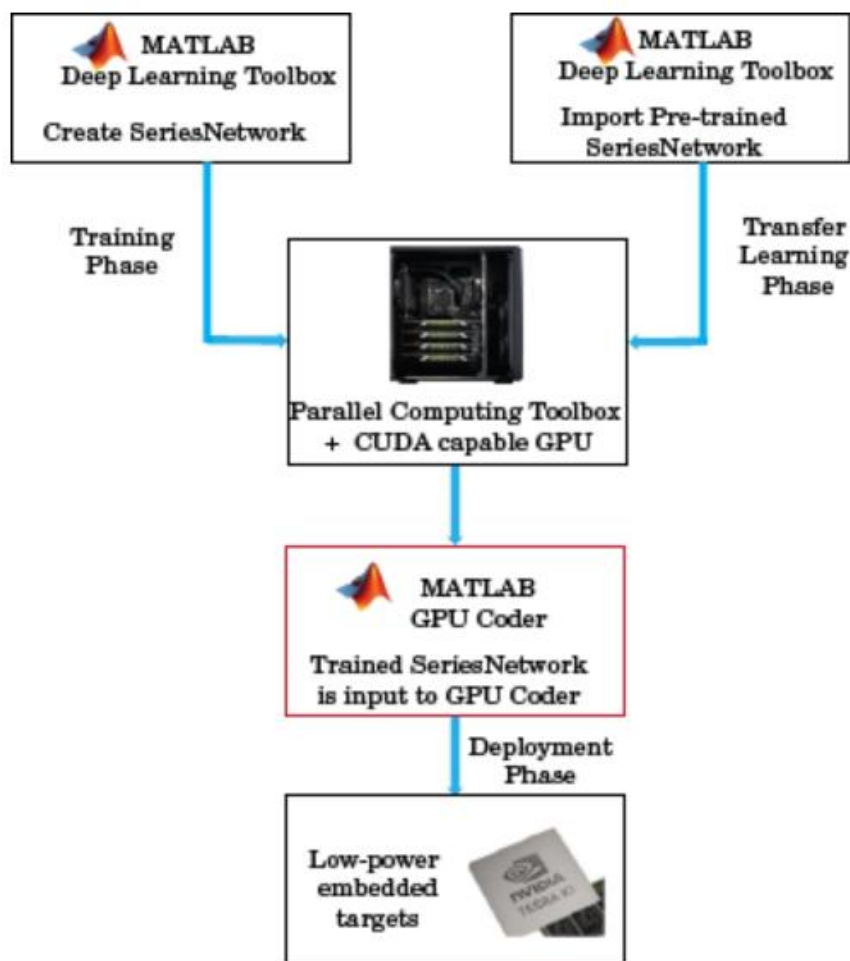
MATLAB for Artificial Intelligence

AI is everywhere. It's not just powering applications like smart assistants, machine translation, and automated driving, it's also giving engineers and scientists a set of techniques for tackling common tasks in new ways. And yet, according to recent estimates, while many organizations recognize the value and potential of AI, few are using it—Gartner's recent survey of 3,000 companies indicated that, of the 50% that are beginning to plan for AI, only 4% have implemented it [1]. Figure below shows an AI model is just one part of the complete workflow for developing a fully engineered system.
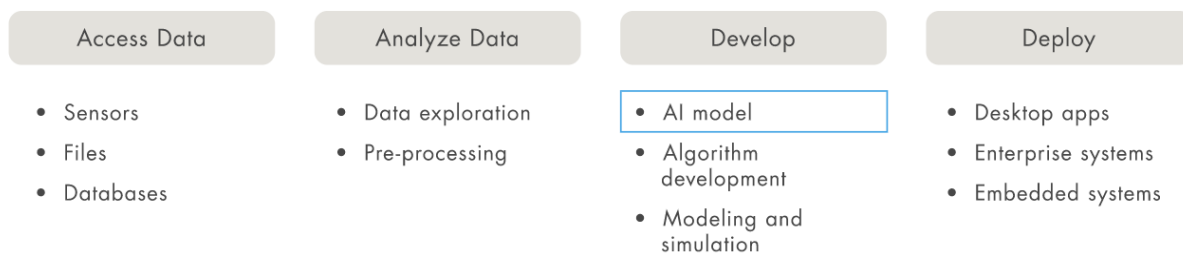

Figure 4: An AI model is just one part of the complete workflow for developing a fully engineered system.

With MATLAB, you are ready for AI even if you have no experience with machine learning. You can use apps to quickly try out different approaches and apply your domain expertise to prepare the data.

If it's not feasible to identify features in your data, you can use deep learning, which identifies features for you as part of the training process. Deep learning requires lots of data, but you can use transfer learning to extend an existing network to work with the data you have.

1 *"The Real Truth of Artificial Intelligence." Presented at Gartner Data & Analytics Summit, March 2018.*

Useful resources:

MATLAB for Deep Learning
https://www.mathworks.com/solutions/deep-learning.html

Introduction to Deep Learning
https://www.mathworks.com/videos/series/introduction-to-deep-learning.html

Getting Started with GPU Coder
https://www.mathworks.com/help/gpucoder/getting-started-with-gpu-coder.html?s_tid=CRUX_lftnav