Object Detection with Jetson TX1: Challenges and Insights

Oron Port Gil Shomron
{soronpo, gilsho}@tx.technion.ac.il
Electrical Engineering
Technion — Israel Institute of Technology

ABSTRACT

1. INTRODUCTION

[1] [2]

2. RESULTS

To compare Caffe SSD performance versus YOLO performance on Jetson TX1, we used their supplied applications for object detection on image files. We downloaded 600 images from MSCOCO dataset [3], and measured the total execution time. We made three comparison between the two algorithms: when both run on (1) CPU, (2) GPU without cuDNN, and (3) GPU with cuDNN[1].

Figure 1 shows the execution time of both SSD and YOLO for each of the options described above. Not surprisingly, running object detection algorithms on a CPU is not as efficient as running them on a GPU. The GPU built-in parallelism is advantageous for such algorithms. [TODO: comparison between SSD and YOLO on CPU]

[TODO: comparison between GPU with and without cuDNN]

[TODO: comparison between SSD and YOLO on GPU]

3. PROFILING

4. INSTALLATION AND EXECUTION

YOLO compiles out-of-the-box on Jetson TX1. Unfortunately, Caffe does not. In this section we will describe the problems we encountered during installation. Hopefully, this information will ease the process for others. In addition, we will share some tips, gathered during this project, regarding the execution of the SSD implementation.

Makefile.config. To use cuDNN acceleration:

```
USE_CUDNN := 1
```

[1]The NVIDIA CUDA Deep Neural Network library (cuDNN) is a GPU-accelerated library of primitives for deep neural networks. cuDNN provides highly tuned implementations for standard routines such as forward and backward convolution, pooling, normalization, and activation layers.

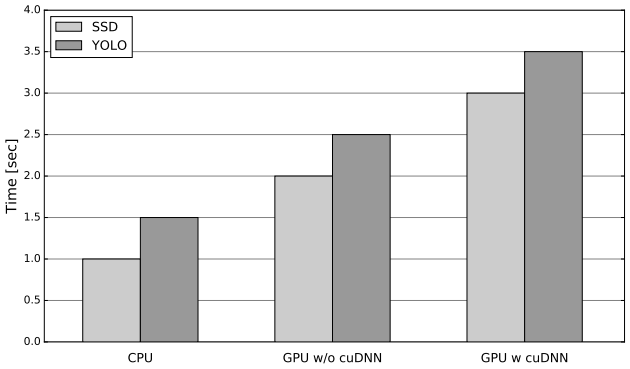

Figure 1: Execution time comparison

Tegra X1 has CUDA capability 5.3, therefore append to *CUDA_ARCH*:

```
CUDA_ARCH := -gencode arch=compute_53,
    ↪ code=sm_53
```

HDF5 directories should be added to *INCLUDE_DIRS* and *LIBRARY_DIRS*:

```
INCLUDE_DIRS := $(PYTHON_INCLUDE) /usr/
    ↪ local/include /usr/include/hdf5/
    ↪ serial
LIBRARY_DIRS := $(PYTHON_LIB) /usr/
    ↪ local/lib /usr/lib/aarch64-linux-
    ↪ gnu/serial
```

Makefile. HDF5 libraries need to be added to the Makefile also:

```
LIBRARIES += glog gflags protobuf
    ↪ boost_system boost_filesystem m
    ↪ hdf5_serial_hl hdf5_serial
```

Python. Caffe also has Python libraries. Running Python scripts can fail due to unset Python path. By running:

```
export PYTHONPATH=$CAFFE_ROOT/python
```

where *$CAFFE_ROOT* is the Caffe home directory, we managed to fix the issues.

Web Camera. Jetson TX1 on-board CSI camera does not work straightaway. On the other hand, plugging a dedicated web-camera almost does. To run Caffe SSD web-camera demo, add the following line before the command:

```
LD_PRELOAD=/usr/lib/aarch64-linux-gnu/
    ↪ libv4l/v4l2convert.so
```

Performance Tuning. The new Tegra Linux driver package releases include *jetson_clocks.sh* script, this is able to maximize performance by disabling DVFS, CPU idle, and CPU quit [4]. To toggle performance:

```
sudo ./jetson_clocks.sh
```

We recommend reading the manual first.

We also noticed that Jetson TX1 fan does not work on default. The script above turns it on. Enabling the fan without running the *jetson_clocks.sh* script, can be achieved with:

```
echo 255 > /sys/kernel/debug/tegra_fan/
    ↪ target_pwm
```

5. CONCLUSION

6. REFERENCES

[1] W. Liu, D. Anguelov, D. Erhan, C. Szegedy, S. Reed, C.-Y. Fu, and A. C. Berg, "SSD: Single Shot Multibox Detector," in *European Conference on Computer Vision*, pp. 21–37, Springer, 2016.

[2] J. Redmon, S. Divvala, R. Girshick, and A. Farhadi, "You Only Look Once: Unified, Real-Time Object Detection," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 779–788, 2016.

[3] "COCO - Common Objects in Context." http://mscoco.org.

[4] Nvidia, "Tegra Linux Driver Package R24.2," 2016.