MiiVii Accelerator SDK User Manual V1.2

MiiVii Dynamics Co., Ltd.

2019.4

Status

[ ] Draft [ ] Formal Revision [√] Formal Release	Name	Instruction for MiiVii accelerator SDK
	Version	V1.2
	Author	Juns
	Date	2019/04/17

Revision History

Version	Date	Release Note	Author
V1.0	2019/01/04	First create	Juns
V1.1	2019/03/01	Add instruction to enter maximum performance mode	Haoran
V1.2	2019/04/17	Amendment	Haoran

Contents

Chapter 1 Instruction

This SDK provides acceleration functions for deep learning models below:

Model name	**Deep learning framework**	**Model input image size**
yolov3	darknet	416x416
yolov3 tiny	darknet	416x416
yolov2 tiny	darknet	416x416

Darknet Version

Git repo	git clone https://github.com/AlexeyAB/darknet.git
Commit version	git checkout 2c5e383c04655fe45f3f533eb3a69a80acbf3561

Chapter 2 Instructions

It is recommended to set your MiiVii Brain device to max mode for the best acceleration performance.

```
sudo nvpmodel -m 0
sudo ~/jetson_clocks.sh
```

1. **Run demo**

   ```
   cd /opt/miivii/features/miivii-accelerator/
   bash bin/demo
   ```

2. **Compile**

   ```
   cp -r /opt/miivii/features/miivii-accelerator /home/nvidia/
   cd /home/nvidia/miivii-accelerator
   bash build.sh
   ```

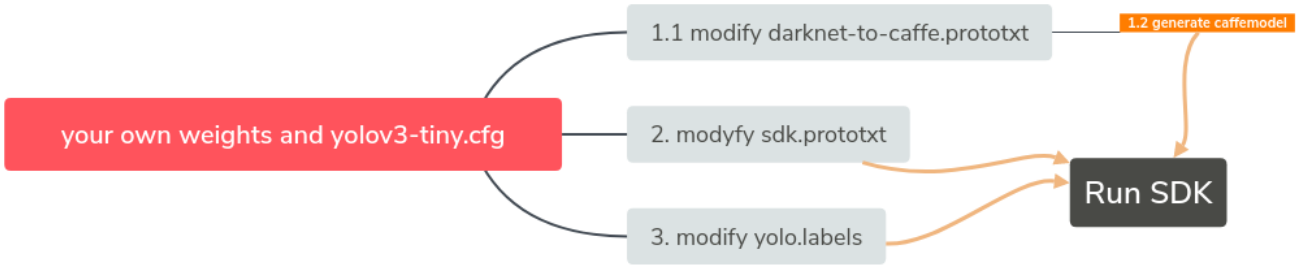
3. **Check Sample codes**

   Sample code is in directory of /home/nvidia/miivii-accelerator/src.

   Files with '–min' like 'yolov3-tiny-min.cpp' are the minimum sample program to use our SDK.

Chapter 3 Example for accelerate your own model

Here's an example of how to use our SDK to accelerate yolov3-tiny model. Basic process is showed as below.



1. Generate caffemodel file

1.1 Modify prototxt file

Modify below file:

/home/nvidia/miivii-accelerator/networks/yolov3-tiny/yolov3-tiny_darknet_to_caffe.prototxt

to transfer your weights into caffemodel. Change the num output parameter according to your own model's object category number, which should be equal to your cfg file's last convolutional filters.

```
436 layer {
437   name: "conv10"
438   type: "Convolution"
439   bottom: "relu9"
440   top: "conv10"
441   convolution_param {
442     num_output: 255
443     kernel_size: 1
444     stride: 1
445     pad: 0
446   }
447 }
    ...
561 layer {
562   name: "conv13"
563   type: "Convolution"
564   bottom: "relu12"
565   top: "conv13"
566   convolution_param {
567     num_output: 255
568     kernel_size: 1
569     stride: 1
570     pad: 0
```

```
571   }
572 }
```

1.2 Generate caffemodel file

Use the modified prototxt file and your weights file to generate caffemodel file.

If your weights file is /home/nvidia/own.weights, generate caffemodel file by:

cd /home/nvidia/miivii-accelerator/scripts/

modify script model_transfer.sh: change –w parameter value into /home/nvidia/own.weights

bash model_transfer.sh

then file yolov3-tiny.caffemodel will be generated under: /home/nvidia/miivii-accelerator/scripts/.

2. Modify yolov3-tiny_SDK.prototxt

File:

/home/nvidia/miivii-accelerator/networks/yolov3-tiny/yolov3-tiny_SDK.prototxt

is used with the generated caffemodel file above by our SDK to accelerate your model.

Change the num_output parameter according to your own model's object category number like you just did with the other prototxt file, which should be equal to your cfg file's last convolutional filters..

```
410 layer {
411   name: "conv10"
412   type: "Convolution"
413   bottom: "leaky9"
414   top: "conv10"
415   convolution_param {
416     num_output: 255
417     kernel_size: 1
418     stride: 1
419     pad: 0
420   }
421 }
...
526   convolution_param {
527     num_output: 255
528     kernel_size: 1
```

```
529     stride: 1
530     pad: 0
531   }
```

3. **Modify yolo.labels**

   Copy the label file:

   `cp /opt/miivii/models/yolo/yolov3-tiny/yolo.labels /home/nvidia/miivii-accelerator/`

   If our object categories are dog and cat, then the yolo.labels file should be:

   ```
   cat
   dog
   ```

4. **Run SDK to accelerate your model**

   Now we have:

   - Modified `yolov3-tiny_SDK.prototxt`
   - Generated `yolov3-tiny.caffemodel`
   - Modified `yolo.labels`

   run:

   ```
   /home/nvidia/miivii-accelerator/bin/yolov3-tiny-video\
     /opt/miivii/data/yolov3_1016.mp4 <path to>yolo.labels \
     <path to>yolov3-tiny.caffemodel \
     <path to>yolov3-tiny_SDK.prototxt
   ```

   Now your model has been accelerated, enjoy!

5. **Notes**

   After running the SDK first time, a tensorcache file will be generated under the directory of the caffemodel file, you can use this file according to the sample code to save the initializing time.

Chapter 4 Other examples

Demonstrate yolov3 model acceleration using GMSL or USB camera input:

```
cd /opt/miivii/features/miivii-accelerator/
bin/yolov3-video 0
```

0 is your device name, normally GMSL camera is 0 and 1, and USB camera is 2.