Getting started for running event camera ML application in Petalinux 2022.2

1. Overview of Kria-App

This Kria-App consists of setup steps for running the event camera based Machine learning inference in Kria KV260 Board. Here we are taking an event camera from Prophesee and also performing recorded event file based ML inference.

This Kria-App has Kria-Ubuntu and Kria-Petalinux both OS support.

2. Prerequisites

Following are the prerequisites of this Kria-App:

- KV260 board and power supply
- Ethernet cable for Network connection
- USB cable (for serial connection to host machine)
- HDMI monitor or DP monitor and cable
- Prophesee CCAM5 (optional if using event file based ML)
- Mouse and Keyboard
- SD card (32 GB)

3. Get the SD card wic image

Get the sdcard wic image zip file from following download link :
https://logictronix.com/wp-content/uploads/2024/07/kv260-event-cam-ml-2022-2-test.wic.zip
Unzip the above downloaded file to **kv260-event-cam-ml-2022-2-test.wic,xz** file

Burn the wic.xz image in the sd card with application like Balena Etcher.
Also update the KV260 boot firmware to Xilinx download - 2022.2_update1_BOOT.BIN by using image recovery tool or xmutil app.

Next boot SD card in the KV260 board connected to HDMI display, keyboard and mouse.
Open the terminal in the Desktop and change to /home/petalinux directory by running following commands.

```
cd /home/petalinux
```

4. Get RAW video file

Get raw video file for testing by running following wget command:
Here you will need internet connection in KV260 board to download the file.

```
wget
https://logictronix.com/wp-content/uploads/2024/07/traffic_cam_day_0007.zip
```

Then unzip the files to get the raw file:

```
unzip traffic_cam_day_0007.zip
```

Change to application directory located at /home/petalinux/event_cam_ml_apps/ directory

```
cd event_cam_ml_apps
```

5. Setup hardware and camera

Run **load.sh** script to load the overlay and setting the camera pipeline
Now change to **bin** directory present in **event_cam_ml_apps** folder to run applications.

```
cd bin
```

6. Run the application:

Application can takes input from file or event camera.

a. Running application with file input :

In bin directory run following commands to run the applications :

For YoloV7 :

```
./event_cam_ml_app_yolov7 -i /home/petalinux/traffic_cam_day_0007.raw -m
../xmodels/Yolov7_320_.xmodel -c 0.3 -x 3 -y 3
```

For YoloV7 tiny :

```
./event_cam_ml_app_yolov7_tiny -i /home/petalinux/traffic_cam_day_0007.raw
-m ../xmodels/Yolov7_tiny_320.xmodel -c 0.3 -x 3 -y 3
```

For YoloV4 tiny:

```
./event_cam_ml_app_yolov4_tiny -i /home/petalinux/traffic_cam_day_0007.raw
-m ../xmodels/yolov4_pe_car.xmodel -c 0.3 -x 3 -y 3
```

After running any of above command, output will be displayed in the Desktop monitor.

b. Running application with camera input :

First enable the camera by running following command:

```
echo on > /sys/class/video4linux/v4l-subdev<X>/device/power/control
```

Where <X> is the v4l2 subdevice number for imx636 device in media graph. One can get this subdevice number by running '**media-ctl -p'** command in terminal. This will list the v4l2 subdevices in media graph.
Here is the media graph listed by running 'media-ctl -p' command:

```
driver          psee-video
model           Prophesee Video Pipeline
serial
bus info
hw revision     0x0
driver version  5.15.136

Device topology
- entity 1: ps_host_if output 0 (1 pad, 1 link)
            type Node subtype V4L flags 0
            device node name /dev/video0
        pad0: SINK
                <- "a0050000.event_stream_smart_tra":1 [ENABLED]

- entity 5: a0050000.event_stream_smart_tra (2 pads, 2 links, 0 routes)
            type V4L2 subdev subtype Unknown flags 0
            device node name /dev/v4l-subdev0
        pad0: SINK
                [stream:0 fmt:PSEE_EVT21ME/1280x720 field:none colorspace:raw xfer:none]
                <- "a0040000.axis_tkeep_handler":1 [ENABLED]
        pad1: SOURCE
                [stream:0 fmt:PSEE_EVT21ME/1280x720 field:none colorspace:raw xfer:none]
                -> "ps_host_if output 0":0 [ENABLED]

- entity 8: a0040000.axis_tkeep_handler (2 pads, 2 links, 0 routes)
            type V4L2 subdev subtype Unknown flags 0
            device node name /dev/v4l-subdev1
        pad0: SINK
                [stream:0 fmt:PSEE_EVT21ME/1280x720 field:none colorspace:raw xfer:none]
                <- "a0010000.mipi_csi2_rx_subsystem":1 [ENABLED]
        pad1: SOURCE
                [stream:0 fmt:PSEE_EVT21ME/1280x720 field:none colorspace:raw xfer:none]
                -> "a0050000.event_stream_smart_tra":0 [ENABLED]

- entity 11: a0010000.mipi_csi2_rx_subsystem (2 pads, 2 links, 0 routes)
             type V4L2 subdev subtype Unknown flags 0
             device node name /dev/v4l-subdev2
        pad0: SINK
                [stream:0 fmt:PSEE_EVT21ME/1280x720 field:none colorspace:raw xfer:none]
                <- "imx636 6-003c":0 [ENABLED]
        pad1: SOURCE
                [stream:0 fmt:PSEE_EVT21ME/1280x720 field:none colorspace:raw xfer:none]
                -> "a0040000.axis_tkeep_handler":0 [ENABLED]

- entity 14: imx636 6-003c (1 pad, 1 link, 0 routes)
             type V4L2 subdev subtype Sensor flags 0
             device node name /dev/v4l-subdev3
        pad0: SOURCE
                [stream:0 fmt:PSEE_EVT21ME/1280x720 field:none colorspace:raw xfer:none]
                -> "a0010000.mipi_csi2_rx_subsystem":0 [ENABLED]
```

So in above media graph, imx636 v4l2 subdevice number is 3. So we will need to set v4l-subdev<X> to v4l-subdev3 in above **5.b. Running application with camera input step**

Next run following commands to run application with camera input :

For YoloV7 :

```
MV_LOG_LEVEL=TRACE V4L2_HEAP=reserved  V4L2_SENSOR_PATH=/dev/v4l-subdev<X>
./event_cam_ml_app_yolov7 -m ../xmodels/Yolov7_320_.xmodel -c 0.3 -x 3 -y 3
```

For YoloV7 tiny :

```
MV_LOG_LEVEL=TRACE V4L2_HEAP=reserved  V4L2_SENSOR_PATH=/dev/v4l-subdev<X>
./event_cam_ml_app_yolov7_tiny -m ../xmodels/Yolov7_tiny_320.xmodel -c 0.3
-x 3 -y 3
```

For YoloV4 tiny:

```
MV_LOG_LEVEL=TRACE V4L2_HEAP=reserved  V4L2_SENSOR_PATH=/dev/v4l-subdev<X>
./event_cam_ml_app_yolov4_tiny -m ../xmodels/yolov4_pe_car.xmodel -c 0.3 -x
3 -y 3
```

In above commands set v4l-subdev<X> depending upon imx636 device v4l2 subev number as in above step.
After running any of above commands, output will be displayed in Desktop monitor.

7. Known Issues and Solutions

a. For testing camera input, object in motion should be kept at least 1 meter away from camera. If object in motion is near to the camera, application stops or start with no display.
b. While running application with camera input, if there is no display then rerun the application after few seconds.
c. Also camera is prone to static electricity and stops running if touched with bare hands. So when camera is running don't touch with bare hands.

Document Revision History

Date	Version	Revision	Author
17/07/2024	1.0	Updated for closed release	Sanam Shakya