



Edge Computing AI Application

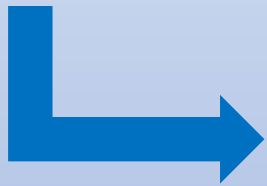
OpenMMLab Kneron Edition

Presenter : Andy Hseih

Kneron Edge Computing AI Application Flow

Model Training

- Data preparation
- Model training
- Model output



Model Convert

- Model convert
- Model quantization
- Model compiling



Model Deployment

- Inference



Kneron Edge Computing AI Application Flow

Model
Training

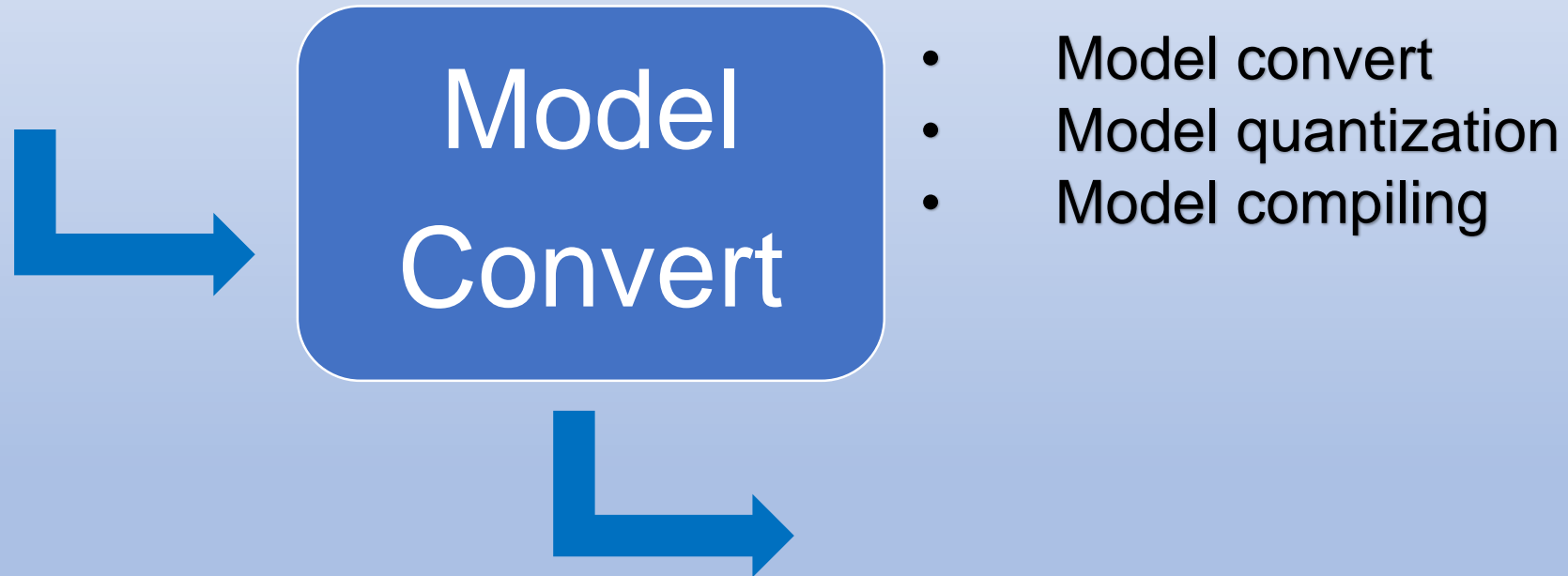
- Data preparation
- Model training
- Model output



On Self Computer !

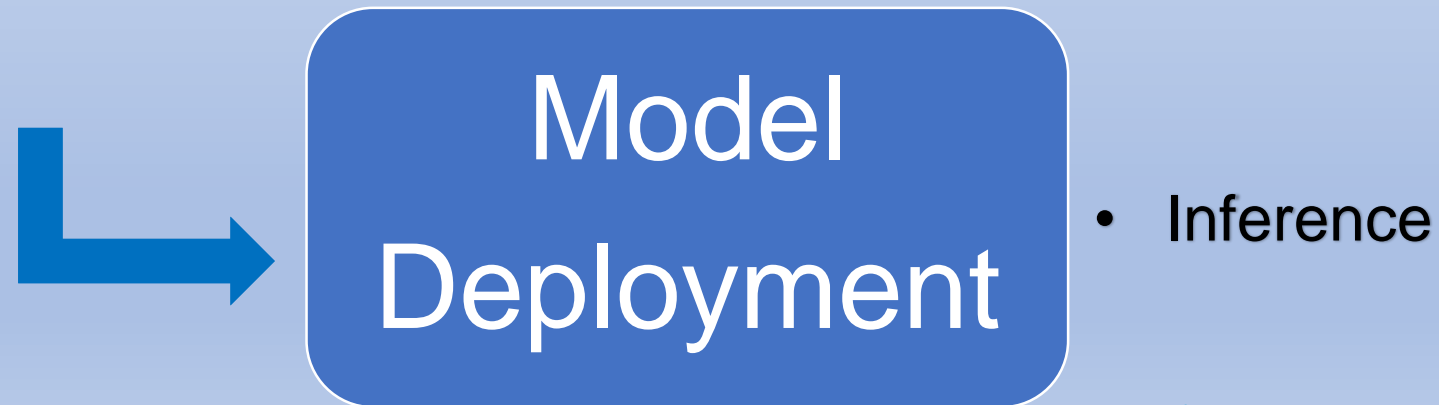
Kneron Edge Computing AI Application Flow

On Self Computer & Kneron ToolChain Docker!



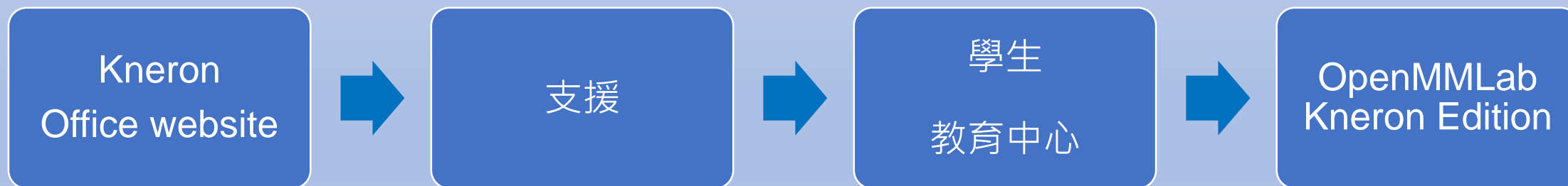
Kneron Edge Computing AI Application Flow

On Self Computer & Use dongle KL720!



OpenMMLab Kneron Edition

- kneron-mmdetection (mmdetection-based)
- kneron-mmsegmentation (mmsegmentation-based)
- kneron-mmtracking (mtracking-based)
- kneron-mmpose (mmpose-based)
- kneron-mmclassification (mmclassification-based)



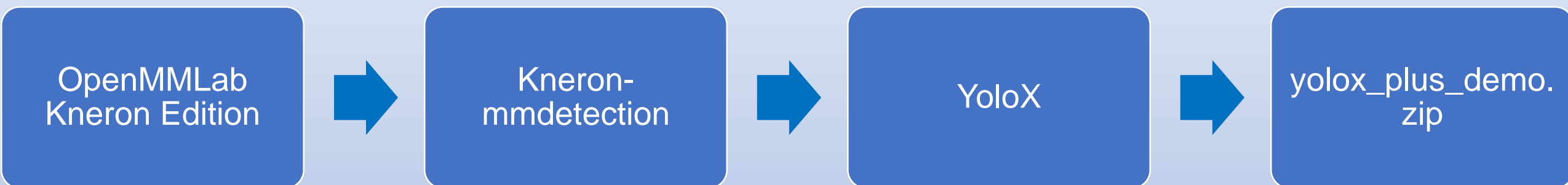
Link : <https://www.kneron.com/tw/support/education-center/>

Kneron office website: <https://www.kneron.com/tw/>



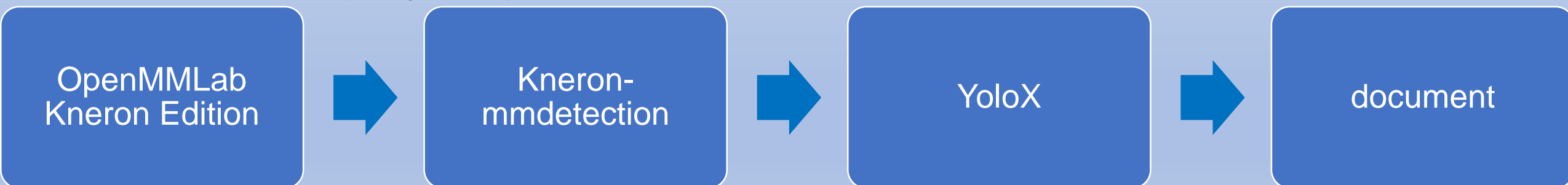
kneron-mmdetection (mmdetection-based)

- Kneron PLUS demo - YoloX



Link : [download link](#)

- YoloX step by step



Link : https://github.com/kneron/kneron-mmdetection/blob/main/docs_kneron/yolox_step_by_step.md



Model Training – YOLOX (YOLOX-s)

- Data preparation
- Model training
- Model output

On Self Computer !

Model Training – YOLOX

- **Environment**

(Python 3.6+ 、 PyTorch 1.3+ 、 CUDA 9.2+ 、 mmdcv-full)

- **kneron-mmdetection**

```
git clone https://github.com/kneron/kneron-mmdetection
```

```
pip install -r requirements/build.txt
```

```
pip install -v -e . # or "python setup.py develop"
```

```
pip install onnx onnxoptimizer onnx-simplifier
```

Pytorch versions download link : <https://pytorch.org/get-started/previous-versions>

mmdcv-full install guide : <https://github.com/open-mmlab/mmdcv>



Model Training – YOLOX (Data preparation)

- Data preparation

```
mmdetection
├── mmdet
├── tools
├── configs
├── data (this folder should be made beforehand)
│   ├── coco (symlink)
│   │   ├── annotations
│   │   ├── train2017
│   │   ├── val2017
│   │   └── test2017
│   └── ...
```

COCO Dataset link : <https://cocodataset.org/#download>

Model Training – YOLOX (Model training)

- Model training

python tools/train.py configs/yolox/yolox_s_8x8_300e_coco_img_norm.py

```
andy_nsin@sw1: ~/data/andy_nsin/kneron_openMLab/kneron-mmddetection
檔案(F) 編輯(E) 檢視(V) 搜尋(S) 終端機(T) 求助(H)
16.6197, grad_norm: 653.0741
2022-08-31 11:19:18,145 - mmdet - INFO - Epoch [1][6450/59144] lr: 5.947e-07, eta: 23 days, 21:06:32, time: 0.127, data_time: 0.026, memory: 1754, loss_cls: 1.8164, loss_bbox: 4.7492, loss_obj: 12.0290, loss:
18.5946, grad_norm: 688.2076
2022-08-31 11:19:24,252 - mmdet - INFO - Epoch [1][6500/59144] lr: 6.039e-07, eta: 23 days, 21:19:40, time: 0.122, data_time: 0.022, memory: 1754, loss_cls: 1.8256, loss_bbox: 4.7396, loss_obj: 11.3638, loss:
17.9290, grad_norm: 661.5175
2022-08-31 11:19:30,232 - mmdet - INFO - Epoch [1][6550/59144] lr: 6.132e-07, eta: 23 days, 21:26:50, time: 0.120, data_time: 0.022, memory: 1754, loss_cls: 1.8296, loss_bbox: 4.7605, loss_obj: 12.8102, loss:
19.4083, grad_norm: 708.4950
2022-08-31 11:19:36,362 - mmdet - INFO - Epoch [1][6600/59144] lr: 6.226e-07, eta: 23 days, 21:40:39, time: 0.123, data_time: 0.023, memory: 1754, loss_cls: 1.7988, loss_bbox: 4.7650, loss_obj: 14.9048, loss:
21.4687, grad_norm: 803.0702
2022-08-31 11:19:42,353 - mmdet - INFO - Epoch [1][6650/59144] lr: 6.321e-07, eta: 23 days, 21:48:01, time: 0.120, data_time: 0.019, memory: 1754, loss_cls: 1.8431, loss_bbox: 4.7462, loss_obj: 10.9454, loss:
17.5347, grad_norm: 690.7887
2022-08-31 11:19:48,808 - mmdet - INFO - Epoch [1][6700/59144] lr: 6.417e-07, eta: 23 days, 22:15:49, time: 0.129, data_time: 0.029, memory: 1754, loss_cls: 1.8223, loss_bbox: 4.7510, loss_obj: 9.8698, loss: 1
6.4431, grad_norm: 612.4499
2022-08-31 11:19:54,732 - mmdet - INFO - Epoch [1][6750/59144] lr: 6.513e-07, eta: 23 days, 22:19:54, time: 0.118, data_time: 0.018, memory: 1754, loss_cls: 1.8050, loss_bbox: 4.7472, loss_obj: 10.9392, loss:
17.4914, grad_norm: 634.0925
2022-08-31 11:20:00,517 - mmdet - INFO - Epoch [1][6800/59144] lr: 6.609e-07, eta: 23 days, 22:17:53, time: 0.116, data_time: 0.018, memory: 1754, loss_cls: 1.8204, loss_bbox: 4.7335, loss_obj: 10.8468, loss:
17.4007, grad_norm: 730.3221
2022-08-31 11:20:06,270 - mmdet - INFO - Epoch [1][6850/59144] lr: 6.707e-07, eta: 23 days, 22:14:31, time: 0.115, data_time: 0.016, memory: 1754, loss_cls: 1.8406, loss_bbox: 4.7458, loss_obj: 12.6950, loss:
19.2814, grad_norm: 654.2009
2022-08-31 11:20:12,183 - mmdet - INFO - Epoch [1][6900/59144] lr: 6.805e-07, eta: 23 days, 22:18:04, time: 0.118, data_time: 0.017, memory: 1754, loss_cls: 1.8272, loss_bbox: 4.7434, loss_obj: 14.7574, loss:
21.3200, grad_norm: 700.3311
2022-08-31 11:20:17,995 - mmdet - INFO - Epoch [1][6950/59144] lr: 6.904e-07, eta: 23 days, 22:17:17, time: 0.116, data_time: 0.015, memory: 1754, loss_cls: 1.8068, loss_bbox: 4.7768, loss_obj: 12.2277, loss:
18.8112, grad_norm: 659.7582
2022-08-31 11:20:24,363 - mmdet - INFO - Exp name: yolox_s_8x8_300e_coco_img_norm.py
2022-08-31 11:20:24,363 - mmdet - INFO - Epoch [1][7000/59144] lr: 7.004e-07, eta: 23 days, 22:39:57, time: 0.127, data_time: 0.028, memory: 1754, loss_cls: 1.8140, loss_bbox: 4.7566, loss_obj: 11.3943, loss:
17.9650, grad_norm: 685.0185
2022-08-31 11:20:29,987 - mmdet - INFO - Epoch [1][7050/59144] lr: 7.104e-07, eta: 23 days, 22:31:08, time: 0.112, data_time: 0.015, memory: 1754, loss_cls: 1.8281, loss_bbox: 4.7644, loss_obj: 11.4635, loss:
18.0480, grad_norm: 699.2608
2022-08-31 11:20:35,749 - mmdet - INFO - Epoch [1][7100/59144] lr: 7.206e-07, eta: 23 days, 22:28:08, time: 0.115, data_time: 0.017, memory: 1754, loss_cls: 1.7954, loss_bbox: 4.7566, loss_obj: 13.7553, loss:
20.3073, grad_norm: 715.6911
2022-08-31 11:20:41,542 - mmdet - INFO - Epoch [1][7150/59144] lr: 7.307e-07, eta: 23 days, 22:26:29, time: 0.116, data_time: 0.018, memory: 1754, loss_cls: 1.8096, loss_bbox: 4.7525, loss_obj: 13.2080, loss:
19.7701, grad_norm: 691.0692
2022-08-31 11:20:47,207 - mmdet - INFO - Epoch [1][7200/59144] lr: 7.410e-07, eta: 23 days, 22:19:37, time: 0.113, data_time: 0.017, memory: 1754, loss_cls: 1.8310, loss_bbox: 4.7506, loss_obj: 9.9137, loss: 1
6.4952, grad_norm: 608.1882
2022-08-31 11:20:53,024 - mmdet - INFO - Epoch [1][7250/59144] lr: 7.513e-07, eta: 23 days, 22:19:01, time: 0.116, data_time: 0.016, memory: 1754, loss_cls: 1.8682, loss_bbox: 4.7399, loss_obj: 10.1970, loss:
16.8051, grad_norm: 616.0910
2022-08-31 11:20:59,798 - mmdet - INFO - Epoch [1][7300/59144] lr: 7.617e-07, eta: 23 days, 22:57:13, time: 0.135, data_time: 0.039, memory: 1754, loss_cls: 1.8500, loss_bbox: 4.7412, loss_obj: 10.7017, loss:
17.2929, grad_norm: 711.0497
2022-08-31 11:21:05,462 - mmdet - INFO - Epoch [1][7350/59144] lr: 7.722e-07, eta: 23 days, 22:50:12, time: 0.113, data_time: 0.017, memory: 1754, loss_cls: 1.8060, loss_bbox: 4.7605, loss_obj: 11.9429, loss:
18.5094, grad_norm: 685.6117
2022-08-31 11:21:11,301 - mmdet - INFO - Epoch [1][7400/59144] lr: 7.827e-07, eta: 23 days, 22:50:18, time: 0.117, data_time: 0.020, memory: 1754, loss_cls: 1.8708, loss_bbox: 4.7207, loss_obj: 11.0901, loss:
17.6816, grad_norm: 706.9231
2022-08-31 11:21:17,010 - mmdet - INFO - Epoch [1][7450/59144] lr: 7.933e-07, eta: 23 days, 22:45:15, time: 0.114, data_time: 0.016, memory: 1754, loss_cls: 1.8218, loss_bbox: 4.7604, loss_obj: 12.0195, loss:
18.6018, grad_norm: 690.7105
2022-08-31 11:21:23,119 - mmdet - INFO - Epoch [1][7500/59144] lr: 8.040e-07, eta: 23 days, 22:56:01, time: 0.122, data_time: 0.023, memory: 1754, loss_cls: 1.8635, loss_bbox: 4.7331, loss_obj: 12.0937, loss:
18.6904, grad_norm: 669.8295
2022-08-31 11:21:28,987 - mmdet - INFO - Epoch [1][7550/59144] lr: 8.148e-07, eta: 23 days, 22:57:11, time: 0.117, data_time: 0.017, memory: 1754, loss_cls: 1.9196, loss_bbox: 4.7457, loss_obj: 11.5573, loss:
18.2226, grad_norm: 691.8984
2022-08-31 11:21:34,925 - mmdet - INFO - Epoch [1][7600/59144] lr: 8.256e-07, eta: 23 days, 23:01:05, time: 0.119, data_time: 0.020, memory: 1754, loss_cls: 1.8486, loss_bbox: 4.7487, loss_obj: 12.0398, loss:
18.6370, grad_norm: 679.5872
2022-08-31 11:21:40,620 - mmdet - INFO - Epoch [1][7650/59144] lr: 8.365e-07, eta: 23 days, 22:55:31, time: 0.114, data_time: 0.019, memory: 1754, loss_cls: 1.9245, loss_bbox: 4.7164, loss_obj: 8.8063, loss: 1
5.4472, grad_norm: 654.1462
```



Model Training – YOLOX (Model output)

- Model output
(work_dirs/yolox_s_8x8_300e_coco_img_norm/latest.pth)
- (Optional) Test trained pytorch model

```
python tools/test_kneron.py \  
  configs/yolox/yolox_s_8x8_300e_coco_img_norm.py \  
  work_dirs/latest.pth \  
  --eval bbox \  
  --out-kneron output.json
```

Model Convert – YOLOX

- Model convert
- Model quantization
- Model compiling

On Self Computer & Kneron ToolChain Docker!



Model Convert – .onnx (Model convert)

- Model convert (.pth -> .onnx)

```
python tools/deployment/pytorch2onnx_kneron.py \  
  configs/yolox/yolox_s_8x8_300e_coco_img_norm.py \  
  work_dirs/yolox_s_8x8_300e_coco_img_norm/latest.pth \  
  --output-file work_dirs/latest.onnx \  
  --skip-postprocess \  
  --shape 640 640
```

Model Convert – .onnx (Model convert)

- (Optional) Test exported onnx model

```
python tools/test_kneron.py \  
  configs/yolox/yolox_s_8x8_300e_coco_img_norm.py \  
  work_dirs/latest.onnx \  
  --eval bbox \  
  --out-kneron output.json
```

Model Convert – .onnx (Model convert)

- Model optimizer (.onnx -> .onnx) -> **Kneron Toolchain Docker**

➤ Python API Work Flow

import ktc

```
onnx_path = '/data1/latest.onnx'  
m = onnx.load(onnx_path)  
m = ktc.onnx_optimizer.onnx2onnx_flow(m)  
onnx.save(m, 'latest.opt.onnx')
```

Kneron Toolchain Docker guide link : <https://doc.kneron.com/docs/#toolchain/manual/#1-installation>



Model Convert – .bie (Model quantization)

- Configure and load data necessary for ktc, and check onnx

```
# npu (only) performance simulation
km = ktc.ModelConfig(20008, "0001", "720", onnx_model=m)
eval_result = km.evaluate()
print("\nNpu performance evaluation result:\n" + str(eval_result))
```

- Model quantization

```
# fixed-point analysis
bie_model_path = km.analysis({"input": img_list})
print("\nFixed-point analysis done. Saved bie model to '" + str(bie_model_path) + "'")
```

Model Convert – .nef (Model compiling)

- Model compiling

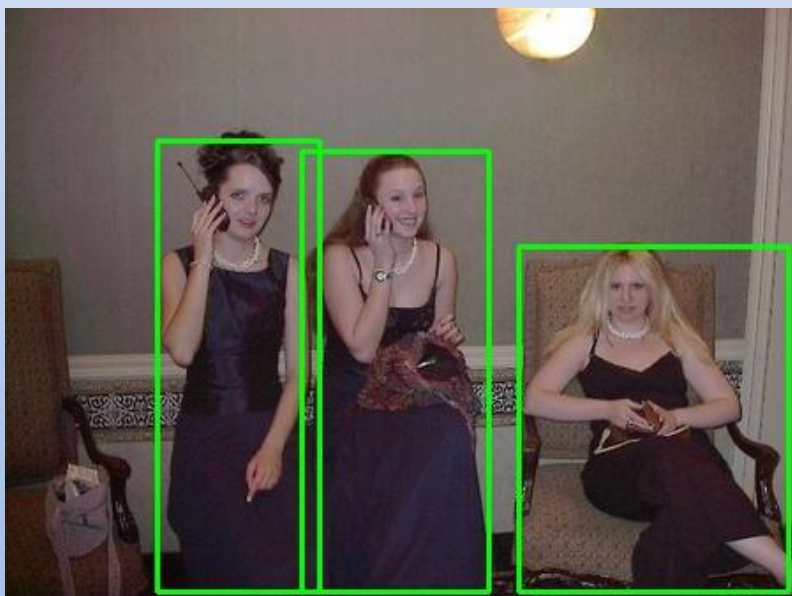
```
# compile
nef_model_path = ktc.compile([km])
print("\nCompile done. Saved Nef file to '" + str(nef_model_path) + "'")
```

The .nef model will generate at
/data1/batch_compile/models_720.nef

Model Deployment – KL720

- yolox_plus_demo

```
python KL720DemoGenericInferenceYoloX_BypassHwPreProc.py -img ./000000000536.jpg -nef ./example_yolox_720.nef
```



```
(kneronmmlab) E:\kneron_plus\OpenMMLab_Kneron_Demo\yolox_plus_demo>python KL720DemoGenericInferenceYoloX_BypassHwPreProc.py
[Connect Device]
- Success
[Set Device Timeout]
- Success
[Upload Model]
- Success
===== NEF Info =====
Toolchain ver= kneron/toolchain:v0.16.0
Schema ver = v0.9.1
=====
[Read Image]
- Success
[Starting Inference Work]
- Starting inference loop 1 times
- .
[Retrieve Inference Node Output ]
- Success
[Output Result Image]
- Output bounding boxes on 'output_000000000536.jpg'
(291.7100891113281,135.5036407470703,444.81996459960936,331.6263565063477)
(86.29214324951171,75.08173713684083,178.1478744506836,331.3882736206055)
(168.07337951660156,81.79956779479981,274.36656799316404,331.9453491210938)
(kneronmmlab) E:\kneron_plus\OpenMMLab_Kneron_Demo\yolox_plus_demo>
```

On Self Computer & Use dongle KL720!



Kneron Forum

The screenshot shows the Kneron Forum website interface. At the top, there is a dark blue navigation bar with the Kneron logo on the left and links for 'Document Center', 'Categories', and 'Discussions' in the center. On the right side of the navigation bar are 'Sign In' and 'Register' buttons. Below the navigation bar, the main content area is divided into several sections. The first section is 'KL520 related discussion', followed by 'AI Model migration', 'KL720 related discussion', and 'Innoserve area'. The 'Innoserve area' section is highlighted with a red border. To the right of the main content area, there is a search bar, a 'Howdy, Stranger!' message with 'Sign In' and 'Register' buttons, a 'Categories' section with 'Recent Discussions' and 'Activity' links, and a 'Popular Tags' section with various tags like 'KL720 32', 'KL520 31', etc.

Kneron Document Center Categories Discussions Sign In Register

KL520 related discussion
Any Kneron KL520 HW or SW issues, bugs, release notices, announcements etc....to be discussed here.
93 discussions 390 comments Most recent: 模型轉換在KL520上面的問題 by Andy Hsieh on August 30

AI Model migration
Any issues, bugs, discussions related to AI model migration to Kneron's SoC
63 discussions 352 comments Most recent: Converting Keras with TimeDistributed layer to ONNX failed by Andy Hsieh on August 24

KL720 related discussion
Any Kneron KL720 HW or SW issues, bugs, release notices, announcements etc....to be discussed here.
60 discussions 253 comments Most recent: Customize Dataset by Chen-Pocky on August 30

Innoserve area
For Innoserve related topic and data
28 discussions 115 comments Most recent: yolov5 Object Detection by Maria Chen on August 29

Search

Howdy, Stranger!
It looks like you're new here. If you want to get involved, click one of these buttons!
Sign In Register

Categories
Recent Discussions
Activity

Popular Tags
KL720 32 KL520 31
KL720 Dongle Dev-board 14
Convert 14 Model 13 toolchain 13
ubuntu 18.04 8 inference 8
yolov3-tiny 8 nef 7 onnx 7
KL720 sdk 7 quantization 6
python 6 host lib 5 yolov3 5
tensorflow 5 pytorch 5 uart 4
ESP8266 4 PUS 4
Windows 3 kic 3

Link : <https://www.kneron.com/tw/forum/>





Thank you for listening.