Laffe2使用说明

LLVision

编 制：顾博

最新版本：V\_0.0.3（修订中）

**版本说明：**

|  |  |  |  |
| --- | --- | --- | --- |
| 版本 | 修订说明 | 修订人 | 日期 |
| V\_0.0.1 | 初稿 | 顾博 | 2019-07-26 |
| V\_0.0.2 | 同步更新OpenVino\_R2 | 顾博 | 2019-08-26 |
| V\_0.0.3 | 同步更新OpenVino\_R3 | 顾博 | 2019-10-26 |
|  |  |  |  |

目录

[Laffe2使用说明 1](#_Toc28967069)

[一、概述： 3](#_Toc28967070)

[二、Laffe2工作流程： 5](#_Toc28967071)

[三、模型转换： 5](#_Toc28967072)

[1 转换工具介绍： 5](#_Toc28967073)

[2 转换工具安装： 6](#_Toc28967074)

[3 模型转换： 8](#_Toc28967075)

[四、模型加载并计算Output 10](#_Toc28967076)

[1 安装demo APK： 10](#_Toc28967077)

[2. Laffe binary路径 10](#_Toc28967078)

[3. demo APK执行Laffe binary 10](#_Toc28967079)

[4. How to get model output in APK/SDK 10](#_Toc28967080)

[五、附录： 11](#_Toc28967081)

## 一、概述：

* Laffe2基于OpenVino，在计算机视觉场景中快速实现深度神经网络（DNN）的原型设计，验证和部署。
* 依赖于LLvisionCompileTool环境来编译模型文件，这些文件将在Laffe2中运行，以通过机载相机、host相机、host推送图片、处理过的数据进行检测/分类操作。
* SDK API允许用户加载图形文件并获取NN输出。

Laffe2目前支持74种layer如下

|  |
| --- |
| **Support Layers\_VPU** |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Activation-Clamp | Crop | Eltwise-Mul | LSTMSequence | RNN |
| Activation-ELU | CTCGreedyDecoder \* | Eltwise-NotEqual | MVN \* | ROIPooling |
| Activation-Leaky ReLU | Deconvolution | Eltwise-Pow | Normalize \* | ScaleShift \* |
| Activation-PReLU | DetectionOutput \* | Eltwise-Prod | Pad \* | Slice |
| Activation-ReLU | Eltwise-Div | Eltwise-SquaredDiff | Permute | SoftMax |
| Activation-ReLU6 | Eltwise-Equal | Eltwise-Sub | Pooling(AVG,MAX) | Split |
| Activation-Sigmoid/Logistic | Eltwise-FloorMod | Eltwise-Sum | Power | Squeeze |
| Activation-TanH | Eltwise-Greater | Flatten | PriorBox | TensorIterator |
| ArgMax | Eltwise-GreaterEqual | FullyConnected (Inner Product) | PriorBoxClustered | Tile |
| BatchNormalization | Eltwise-Less | Gather | Proposal | Unsqueeze |
| Broadcast | Eltwise-LessEqual | Gemm | PSROIPooling |  |
| Concat | Eltwise-LogicalAnd | GRN | RegionYolo |  |
| Const | Eltwise-LogicalOr | Interp | ReorgYolo |  |
| Convolution-Dilated | Eltwise-LogicalXor | Log | Resample |  |
| Convolution-Grouped | Eltwise-Max | LRN (Norm) | Reshape |  |
| Convolution-Ordinary | Eltwise-Min | LSTMCell | ReverseSequence |  |

\*- support is limited to the specific parameters. Refer to "Known Layers Limitation" section for the device [from the list of supported](https://docs.openvinotoolkit.org/2019_R3.1/_docs_IE_DG_supported_plugins_Supported_Devices.html).

## 二、Laffe2工作流程：

Laffe2 binary

Model

转换工具(ubuntu)

SDK 加载计算

图形展现

应用解析Output

Output

## 三、模型转换：

### 1 转换工具介绍：

LLvisionCompileTool

版本信息：

|  |  |  |  |
| --- | --- | --- | --- |
| version | commiter | date | message |
| V\_0.0.3 | Gubo | 19/12/10 | Version\_0.0.3 release |

目录结构：

├── bin -------------- Executable file

├── example　　-------------- Models that have been tested

├── extensions -------------- Model preprocessed configuration file directory

├── inference\_engine　　-------------- inference\_engine

├── LLvisionCompile　　-------------- Model conversion script

└── setupvars.sh　　-------------- Environment profile

### 2 转换工具安装：

**安装环境：**

**Development computer with supported OS**

**x86-64 with Ubuntu (64 bit) 16.04/18.04Desktop**

1. 安装转换工具
2. 解压工具包

*tar -xzvf LLvisionXXX.tar.gz*

1. 配置环境变量

*cd LLvisionCompile/ # 进入目录*

*ln -s ./bin/LLvisionCompile\_V3/LLvisionCompile\_V3 LLvisionCompile # 建立软连接*

*source setupvars.sh # 配置安装文件*

*提示：[setupvars.sh] LLvisionCompile environment initialized*

***Optional****:*

*The OpenVINO environment variables are removed when you close the shell. As an option, you can permanently set the environment variables as follows:*

1. *修改setupvars.sh line16*

*修改前：INSTALLDIR=$(pwd)*

*修改后：INSTALLDIR=/<LLvisionCompile\_directory>*

1. *Open the .bashrc file in <user\_directory>:*

*vi <user\_directory>/.bashrc*

1. *Add this line to the end of the file:*

*source / <LLvisionCompile\_directory>/setupvars.sh*

1. *Save and close the file: press the Esc key and type :wq.*
2. *To test your change, open a new terminal. You will see [setupvars.sh] Compile environment initialized.*
3. 检查是否安装成功

*./bin/myriad\_compile*

*若成功则显示：*

*Inference Engine:*

*API version ............ 2.1*

*Build .................. custom\_releases/2019/R3\_ac8584cb714a697a12f1f30b7a3b78a5b9ac5e05*

*Path to model xml file is required*

1. 可能出现的错误

可能会出现错误1： 检查安装成功出现报错找不到相关so库。

*./bin/myriad\_compile: error while loading shared libraries: libinference\_engine.so: cannot open shared object file: No such file or directory*

错误原因：inference\_engine的默认路径是<LLvisionCompile\_directory>，没有配置环境导致找不到inference\_engine

解决办法：

1. 重新配置环境变量
2. 忽略错误直接转换模型进行验证

可能会出现错误2： 执行文件LLvisionCompile运行失败

错误原因：软连接失效

解决办法：

执行命令：*ln -s ./bin/LLvisionCompile\_V3/LLvisionCompile\_V3 LLvisionCompile*

### 3 模型转换：

1. 模型转换示例

模型转换需要文件有：

* 1. 模型

例：./example/MobileNetSSD/MobileNetSSD\_deploy.caffemodel

* 1. proto 模型定义文件

例: ./example/MobileNetSSD/MobileNetSSD\_deploy.prototxt

准备完全以上文件就可以进行模型转换了：参加example

更多例子请参考./example/README.md

**注：1.** **加密模型请参见***[Compile-specific parameters](#Compile)***参数-e(--enc)**

**2. 模型输入仅支持NCHW(NHWC)，且H(W)<=720**

**3. 多模型串行务必配置相同的VPU\_NUMBER\_OF\_SHAVES**

*Example:*

*Caffe Example*

*./LLvisionCompile --input\_model ./example/MobileNetSSD/MobileNetSSD\_deploy.caffemodel --mean\_values [127.5,127.5,127.5] --scale\_value [127.5] --output\_dir ./example/MobileNetSSD -VPU\_NUMBER\_OF\_SHAVES 4*

*Model Optimizer arguments:*

*Common parameters:*

*- Path to the Input Model: /home/gubo/WorkSpace/G26X/LLvisionCompile/./example/MobileNetSSD\_deploy.caffemodel*

*- Path for generated IR: /home/gubo/WorkSpace/G26X/LLvisionCompile/./example/*

*- IR output name: MobileNetSSD\_deploy*

*- Log level: ERROR*

*- Batch: Not specified, inherited from the model*

*- Input layers: Not specified, inherited from the model*

*- Output layers: Not specified, inherited from the model*

*- Input shapes: Not specified, inherited from the model*

*- Mean values: [127.5,127.5,127.5]*

*- Scale values: [127.5]*

*- Scale factor: Not specified*

*- Precision of IR: FP16*

*- Enable fusing: True*

*…*

*Model Optimizer version: \*\*\**

*[ SUCCESS ] Generated IR model.*

*[ SUCCESS ] XML file: /home/gubo/WorkSpace/G26X/LLvisionCompile/./example/MobileNetSSD\_deploy.xml*

*[ SUCCESS ] BIN file: /home/gubo/WorkSpace/G26X/LLvisionCompile/./example/MobileNetSSD\_deploy.bin*

*[ SUCCESS ] Total execution time: 3.60 seconds.*

*The xml is /home/gubo/WorkSpace/G26X/LLvisionCompile/./example/MobileNetSSD\_deploy.xml!*

*The bolb is /home/gubo/WorkSpace/G26X/LLvisionCompile/./example/MobileNetSSD\_deploy.blob!*

*Inference Engine:*

*API version ............ \*\**

*Build .................. custom\_releases/\*\*\**

*Done*

*MO model file size = 14442816*

*LLvision Blob file size = 14443840*

*命令执行完就可以在/example/ 文件夹下生成*

*MobileNetSSD\_deploy.blob文件，这就是转换之后的模型。*

## 四、模型加载并计算Output

### 1 安装demo APK：

(1)apk file

路径: /LaffeDemo/Android\_Laffe\_SampleCode/laffedemo.apk

手机安装apk

(2)sample code

路径: /LaffeDemo/Android\_Laffe\_SampleCode/LaffeDemo.zip

通过开发环境将此示例代码运行到手机。

### 2. Laffe binary路径

路径: /LaffeDemo/app/src/main/assets/detect.bin

注意:

"detect.bin" 是测试过的Laffe模型.

### 3. demo APK执行Laffe binary

通过IAiClient interface method "loadModel(String filePath)"加载模型

请参阅示例代码中“MainActivity.java”文件中的调用模式。

### 4. How to get model output in APK/SDK

输出是通过IAiClient接口方法“setOnDetectResultListener（ResCallback callback）”获得的，然后通过“ParseCommonNet.cpp”文件解析数据。

请参阅示例代码中“MainActivity.java”文件中的获取模型计算结果的调用模式。

## 五、附录：

*./LLvisionCompile -h*

*[-h] [--framework {tf,caffe,mxnet,kaldi,onnx}]*

*[--input\_model INPUT\_MODEL]*

*[--model\_name MODEL\_NAME]*

*[--output\_dir OUTPUT\_DIR]*

*[--input\_shape INPUT\_SHAPE] [--scale SCALE]*

*[--reverse\_input\_channels]*

*[--log\_level {CRITICAL,ERROR,WARN,WARNING,INFO,DEBUG,NOTSET}]*

*[--input INPUT] [--output OUTPUT]*

*[--mean\_values MEAN\_VALUES]*

*[--scale\_values SCALE\_VALUES]*

*[--data\_type {FP16}] [--disable\_fusing]*

*[--disable\_resnet\_optimization]*

*[--finegrain\_fusing FINEGRAIN\_FUSING]*

*[--disable\_gfusing]*

*[--enable\_concat\_optimization]*

*[--move\_to\_preprocess] [--extensions EXTENSIONS]*

*[--batch BATCH] [--version] [--silent]*

*[--freeze\_placeholder\_with\_value FREEZE\_PLACEHOLDER\_WITH\_VALUE]*

*[--generate\_deprecated\_IR\_V2] [--keep\_shape\_ops]*

*[--steps] [--input\_model\_is\_text]*

*[--input\_checkpoint INPUT\_CHECKPOINT]*

*[--input\_meta\_graph INPUT\_META\_GRAPH]*

*[--saved\_model\_dir SAVED\_MODEL\_DIR]*

*[--saved\_model\_tags SAVED\_MODEL\_TAGS]*

*[--tensorflow\_subgraph\_patterns TENSORFLOW\_SUBGRAPH\_PATTERNS]*

*[--tensorflow\_operation\_patterns TENSORFLOW\_OPERATION\_PATTERNS]*

*[--tensorflow\_custom\_operations\_config\_update TENSORFLOW\_CUSTOM\_OPERATIONS\_CONFIG\_UPDATE]*

*[--tensorflow\_use\_custom\_operations\_config TENSORFLOW\_USE\_CUSTOM\_OPERATIONS\_CONFIG]*

*[--tensorflow\_object\_detection\_api\_pipeline\_config TENSORFLOW\_OBJECT\_DETECTION\_API\_PIPELINE\_CONFIG]*

*[--tensorboard\_logdir TENSORBOARD\_LOGDIR]*

*[--tensorflow\_custom\_layer\_libraries TENSORFLOW\_CUSTOM\_LAYER\_LIBRARIES]*

*[--disable\_nhwc\_to\_nchw]*

*[--input\_proto INPUT\_PROTO]*

*[--caffe\_parser\_path CAFFE\_PARSER\_PATH] [-k K]*

*[--mean\_file MEAN\_FILE]*

*[--mean\_file\_offsets MEAN\_FILE\_OFFSETS]*

*[--disable\_omitting\_optional]*

*[--enable\_flattening\_nested\_params]*

*[--input\_symbol INPUT\_SYMBOL]*

*[--nd\_prefix\_name ND\_PREFIX\_NAME]*

*[--pretrained\_model\_name PRETRAINED\_MODEL\_NAME]*

*[--save\_params\_from\_nd] [--legacy\_mxnet\_model]*

*[--enable\_ssd\_gluoncv] [--counts COUNTS]*

*[--remove\_output\_softmax] [--remove\_memory]*

*[-i {HOST,CAM}] [-Channels\_Order\_RGB] [-F FPS]*

*[-Mt MAXTIME] [-e] [-key KEY] [-c CRYPT\_SIZE]*

*[-oo OUTPUT\_BLOB]*

*[-VPU\_NUMBER\_OF\_SHAVES {1,2,3,4,5,6,7,8}]*

*optional arguments:*

*-h, --help show this help message and exit*

*--framework {tf,caffe,mxnet,kaldi,onnx}*

*Name of the framework used to train the input model.*

***Framework-agnostic parameters:***

*--input\_model INPUT\_MODEL, -w INPUT\_MODEL, -m INPUT\_MODEL*

*Tensorflow\*: a file with a pre-trained model (binary*

*or text .pb file after freezing). Caffe\*: a model*

*proto file with model weights*

*--model\_name MODEL\_NAME, -n MODEL\_NAME*

*Model\_name parameter passed to the final create\_ir*

*transform. This parameter is used to name a network in*

*a generated IR and output .xml/.bin files.*

*--output\_dir OUTPUT\_DIR, -o OUTPUT\_DIR*

*Directory that stores the generated IR. By default, it*

*is the directory from where the Model Optimizer is*

*launched.*

*--input\_shape INPUT\_SHAPE*

*Input shape(s) that should be fed to an input node(s)*

*of the model. Shape is defined as a comma-separated*

*list of integer numbers enclosed in parentheses or*

*square brackets, for example [1,3,227,227] or*

*(1,227,227,3), where the order of dimensions depends*

*on the framework input layout of the model. For*

*example, [N,C,H,W] is used for Caffe\* models and*

*[N,H,W,C] for TensorFlow\* models. Model Optimizer*

*performs necessary transformations to convert the*

*shape to the layout required by Inference Engine*

*(N,C,H,W). The shape should not contain undefined*

*dimensions (? or -1) and should fit the dimensions*

*defined in the input operation of the graph. If there*

*are multiple inputs in the model, --input\_shape should*

*contain definition of shape for each input separated*

*by a comma, for example: [1,3,227,227],[2,4] for a*

*model with two inputs with 4D and 2D shapes.*

*Alternatively, you can specify shapes with the --input*

*option.*

*--scale SCALE, -s SCALE*

*All input values coming from original network inputs*

*will be divided by this value. When a list of inputs*

*is overridden by the --input parameter, this scale is*

*not applied for any input that does not match with the*

*original input of the model.*

*--reverse\_input\_channels*

*Switch the input channels order from RGB to BGR (or*

*vice versa). Applied to original inputs of the model*

*if and only if a number of channels equals 3. Applied*

*after application of --mean\_values and --scale\_values*

*options, so numbers in --mean\_values and*

*--scale\_values go in the order of channels used in the*

*original model.*

*--log\_level {CRITICAL,ERROR,WARN,WARNING,INFO,DEBUG,NOTSET}*

*Logger level*

*--input INPUT Quoted list of comma-separated input nodes names with*

*shapes and values for freezing. The shape and value*

*are specified as space-separated lists. For example,*

*use the following format to set input port 0 of the*

*node `node\_name1` with the shape [3 4] as an input*

*node and freeze output port 1 of the node `node\_name2`*

*with the value [20 15] and the shape [2]:*

*"0:node\_name1[3 4],node\_name2:1[2]->[20 15]".*

*--output OUTPUT The name of the output operation of the model. For*

*TensorFlow\*, do not add :0 to this name.*

*--mean\_values MEAN\_VALUES, -ms MEAN\_VALUES*

*Mean values to be used for the input image per*

*channel. Values to be provided in the (R,G,B) or*

*[R,G,B] format. Can be defined for desired input of*

*the model, for example: "--mean\_values*

*data[255,255,255],info[255,255,255]". The exact*

*meaning and order of channels depend on how the*

*original model was trained.*

*--scale\_values SCALE\_VALUES*

*Scale values to be used for the input image per*

*channel. Values are provided in the (R,G,B) or [R,G,B]*

*format. Can be defined for desired input of the model,*

*for example: "--scale\_values*

*data[255,255,255],info[255,255,255]". The exact*

*meaning and order of channels depend on how the*

*original model was trained.*

*--data\_type {FP16} All model weights and biases are quantized to FP16.No*

*need to fill in*

*--disable\_fusing Turn off fusing of linear operations to Convolution*

*--disable\_resnet\_optimization*

*Turn off resnet optimization*

*--finegrain\_fusing FINEGRAIN\_FUSING*

*Regex for layers/operations that won't be fused.*

*Example: --finegrain\_fusing Convolution1,.\*Scale.\**

*--disable\_gfusing Turn off fusing of grouped convolutions*

*--enable\_concat\_optimization*

*Turn on concat optimization*

*--move\_to\_preprocess Move mean values to IR preprocess section*

*--extensions EXTENSIONS*

*Directory or a comma separated list of directories*

*with extensions. To disable all extensions including*

*those that are placed at the default location, pass an*

*empty string.*

*--batch BATCH, -b BATCH*

*Input batch size*

*--version Version of Model Optimizer*

*--silent Prevent any output messages except those that*

*correspond to log level equals ERROR, that can be set*

*with the following option: --log\_level. By default,*

*log level is already ERROR.*

*--freeze\_placeholder\_with\_value FREEZE\_PLACEHOLDER\_WITH\_VALUE*

*Replaces input layer with constant node with provided*

*value, for example: "node\_name->True". It will be*

*DEPRECATED in future releases. Use --input option to*

*specify a value for freezing.*

*--generate\_deprecated\_IR\_V2*

*Force to generate legacy/deprecated IR V2 to work with*

*previous versions of the Inference Engine. The*

*resulting IR may or may not be correctly loaded by*

*Inference Engine API (including the most recent and*

*old versions of Inference Engine) and provided as a*

*partially-validated backup option for specific*

*deployment scenarios. Use it at your own discretion.*

*By default, without this option, the Model Optimizer*

*generates IR V3.*

*--keep\_shape\_ops [ Experimental feature ] Enables `Shape` operation*

*with all children keeping. This feature makes model*

*reshapable in Inference Engine*

*--steps Enables model conversion steps display*

**TensorFlow\*-specific parameters:**

*--input\_model\_is\_text*

*TensorFlow\*: treat the input model file as a text*

*protobuf format. If not specified, the Model Optimizer*

*treats it as a binary file by default.*

*--input\_checkpoint INPUT\_CHECKPOINT*

*TensorFlow\*: variables file to load.*

*--input\_meta\_graph INPUT\_META\_GRAPH*

*Tensorflow\*: a file with a meta-graph of the model*

*before freezing*

*--saved\_model\_dir SAVED\_MODEL\_DIR*

*TensorFlow\*: directory representing non frozen model*

*--saved\_model\_tags SAVED\_MODEL\_TAGS*

*Group of tag(s) of the MetaGraphDef to load, in string*

*format, separated by ','. For tag-set contains*

*multiple tags, all tags must be passed in.*

*--tensorflow\_subgraph\_patterns TENSORFLOW\_SUBGRAPH\_PATTERNS*

*TensorFlow\*: a list of comma separated patterns that*

*will be applied to TensorFlow\* node names to infer a*

*part of the graph using TensorFlow\*.*

*--tensorflow\_operation\_patterns TENSORFLOW\_OPERATION\_PATTERNS*

*TensorFlow\*: a list of comma separated patterns that*

*will be applied to TensorFlow\* node type (ops) to*

*infer these operations using TensorFlow\*.*

*--tensorflow\_custom\_operations\_config\_update TENSORFLOW\_CUSTOM\_OPERATIONS\_CONFIG\_UPDATE*

*TensorFlow\*: update the configuration file with node*

*name patterns with input/output nodes information.*

*--tensorflow\_use\_custom\_operations\_config TENSORFLOW\_USE\_CUSTOM\_OPERATIONS\_CONFIG*

*TensorFlow\*: use the configuration file with custom*

*operation description.*

*--tensorflow\_object\_detection\_api\_pipeline\_config TENSORFLOW\_OBJECT\_DETECTION\_API\_PIPELINE\_CONFIG*

*TensorFlow\*: path to the pipeline configuration file*

*used to generate model created with help of Object*

*Detection API.*

*--tensorboard\_logdir TENSORBOARD\_LOGDIR*

*TensorFlow\*: dump the input graph to a given directory*

*that should be used with TensorBoard.*

*--tensorflow\_custom\_layer\_libraries TENSORFLOW\_CUSTOM\_LAYER\_LIBRARIES*

*TensorFlow\*: comma separated list of shared libraries*

*with TensorFlow\* custom operations implementation.*

*--disable\_nhwc\_to\_nchw*

*Disables default translation from NHWC to NCHW*

***Caffe\*-specific parameters:***

*--input\_proto INPUT\_PROTO, -d INPUT\_PROTO*

*Deploy-ready prototxt file that contains a topology*

*structure and layer attributes*

*--caffe\_parser\_path CAFFE\_PARSER\_PATH*

*Path to Python Caffe\* parser generated from*

*caffe.proto*

*-k K Path to CustomLayersMapping.xml to register custom*

*layers*

*--mean\_file MEAN\_FILE, -mf MEAN\_FILE*

*Mean image to be used for the input. Should be a*

*binaryproto file*

*--mean\_file\_offsets MEAN\_FILE\_OFFSETS, -mo MEAN\_FILE\_OFFSETS*

*Mean image offsets to be used for the input*

*binaryproto file. When the mean image is bigger than*

*the expected input, it is cropped. By default, centers*

*of the input image and the mean image are the same and*

*the mean image is cropped by dimensions of the input*

*image. The format to pass this option is the*

*following: "-mo (x,y)". In this case, the mean file is*

*cropped by dimensions of the input image with offset*

*(x,y) from the upper left corner of the mean image*

*--disable\_omitting\_optional*

*Disable omitting optional attributes to be used for*

*custom layers. Use this option if you want to transfer*

*all attributes of a custom layer to IR. Default*

*behavior is to transfer the attributes with default*

*values and the attributes defined by the user to IR.*

*--enable\_flattening\_nested\_params*

*Enable flattening optional params to be used for*

*custom layers. Use this option if you want to transfer*

*attributes of a custom layer to IR with flattened*

*nested parameters. Default behavior is to transfer the*

*attributes without flattening nested parameters.*

**Mxnet-specific parameters:**

*--input\_symbol INPUT\_SYMBOL*

*Symbol file (for example, model-symbol.json) that*

*contains a topology structure and layer attributes*

*--nd\_prefix\_name ND\_PREFIX\_NAME*

*Prefix name for args.nd and argx.nd files.*

*--pretrained\_model\_name PRETRAINED\_MODEL\_NAME*

*Name of a pretrained MXNet model without extension and*

*epoch number. This model will be merged with args.nd*

*and argx.nd files*

*--save\_params\_from\_nd*

*Enable saving built parameters file from .nd files*

*--legacy\_mxnet\_model Enable MXNet loader to make a model compatible with*

*the latest MXNet version. Use only if your model was*

*trained with MXNet version lower than 1.0.0*

*--enable\_ssd\_gluoncv Enable pattern matchers replacers for converting*

*gluoncv ssd topologies.*

**Kaldi-specific parameters:**

*--counts COUNTS Path to the counts file*

*--remove\_output\_softmax*

*Removes the SoftMax layer that is the output layer*

*--remove\_memory Removes the Memory layer and use additional inputs*

*outputs instead*

**[Compile-specific parameters](#加密模型):**

*-i {HOST,CAM}, --inputfile {HOST,CAM}*

*File from host image(data) or camera frame,*

*-Channels\_Order\_RGB, --Channels\_Order\_RGB*

*Use this option if the order of channels in the*

*original model is RGB*

*-F FPS, --FPS FPS Calculation frequency*

*-Mt MAXTIME, --maxtime MAXTIME*

*Maximum time of conversion model runs*

*-e, --enc encrypt model*

*-key KEY, --key KEY 128 bits key of encrypt model,for*

*example'1234567812345678'*

*-c CRYPT\_SIZE, --crypt\_size CRYPT\_SIZE*

*encrypt or decrypt file size,size must 16 bytes*

*aligned and not exceed file size (unsigned*

*int),default is 8K*

*-oo OUTPUT\_BLOB, --output\_blob OUTPUT\_BLOB*

*Optional. Path to the output*

*file.default=<model\_xml\_file>.blob*

*-VPU\_NUMBER\_OF\_SHAVES {1,2,3,4,5,6,7,8}, --VPU\_NUMBER\_OF\_SHAVES {1,2,3,4,5,6,7,8} Optional. Specifies number of shaves.*