## **Xception模型**

keras.applications.xception.Xception(include\_top=**True**, weights='imagenet',

input\_tensor=**None**, input\_shape=**None**,

pooling=**None**, classes=1000)

Xception V1 模型, 权重由ImageNet训练而言

在ImageNet上,该模型取得了验证集top1 0.790和top5 0.945的正确率

注意,该模型目前仅能以TensorFlow为后端使用,由于它依赖于"SeparableConvolution"层,目前该模型只支持channels\_last的维度顺序(width, height, channels)

默认输入图片大小为299x299

### **参数**

* include\_top：是否保留顶层的3个全连接网络
* weights：None代表随机初始化，即不加载预训练权重。'imagenet'代表加载预训练权重
* input\_tensor：可填入Keras tensor作为模型的图像输出tensor
* input\_shape：可选，仅当include\_top=False有效，应为长为3的tuple，指明输入图片的shape，图片的宽高必须大于71，如(150,150,3)

pooling：当include\_top=False时，该参数指定了池化方式。None代表不池化，最后一个卷积层的输出为4D张量。‘avg’代表全局平均池化，‘max’代表全局最大值池化。

classes：可选，图片分类的类别数，仅当include\_top=True并且不加载预训练权重时可用。

### **返回值**

Keras 模型对象

### **参考文献**

* [Xception: Deep Learning with Depthwise Separable Convolutions](https://arxiv.org/abs/1610.02357)

### **License**

预训练权重由我们自己训练而来，基于MIT license发布

## **VGG16模型**

keras.applications.vgg16.VGG16(include\_top=**True**, weights='imagenet',

input\_tensor=**None**, input\_shape=**None**,

pooling=**None**,

classes=1000)

VGG16模型,权重由ImageNet训练而来

该模型再Theano和TensorFlow后端均可使用,并接受channels\_first和channels\_last两种输入维度顺序

模型的默认输入尺寸是224x224

### **参数**

* include\_top：是否保留顶层的3个全连接网络
* weights：None代表随机初始化，即不加载预训练权重。'imagenet'代表加载预训练权重
* input\_tensor：可填入Keras tensor作为模型的图像输出tensor
* input\_shape：可选，仅当include\_top=False有效，应为长为3的tuple，指明输入图片的shape，图片的宽高必须大于48，如(200,200,3)

### **返回值**

* pooling：当include\_top=False时，该参数指定了池化方式。None代表不池化，最后一个卷积层的输出为4D张量。‘avg’代表全局平均池化，‘max’代表全局最大值池化。
* classes：可选，图片分类的类别数，仅当include\_top=True并且不加载预训练权重时可用。

Keras 模型对象

### **参考文献**

* [Very Deep Convolutional Networks for Large-Scale Image Recognition](https://arxiv.org/abs/1409.1556)：如果在研究中使用了VGG，请引用该文

### **License**

预训练权重由[牛津VGG组](http://www.robots.ox.ac.uk/~vgg/research/very_deep/)发布的预训练权重移植而来，基于[Creative Commons Attribution License](https://creativecommons.org/licenses/by/4.0/)

## **VGG19模型**

keras.applications.vgg19.VGG19(include\_top=**True**, weights='imagenet',

input\_tensor=**None**, input\_shape=**None**,

pooling=**None**,

classes=1000)

VGG19模型,权重由ImageNet训练而来

该模型在Theano和TensorFlow后端均可使用,并接受channels\_first和channels\_last两种输入维度顺序

模型的默认输入尺寸是224x224

### **参数**

* include\_top：是否保留顶层的3个全连接网络
* weights：None代表随机初始化，即不加载预训练权重。'imagenet'代表加载预训练权重
* input\_tensor：可填入Keras tensor作为模型的图像输出tensor
* input\_shape：可选，仅当include\_top=False有效，应为长为3的tuple，指明输入图片的shape，图片的宽高必须大于48，如(200,200,3)
* pooling：当include\_top=False时，该参数指定了池化方式。None代表不池化，最后一个卷积层的输出为4D张量。‘avg’代表全局平均池化，‘max’代表全局最大值池化。
* classes：可选，图片分类的类别数，仅当include\_top=True并且不加载预训练权重时可用。

### **返回值**

### **返回值**

Keras 模型对象

### **参考文献**

* [Very Deep Convolutional Networks for Large-Scale Image Recognition](https://arxiv.org/abs/1409.1556)：如果在研究中使用了VGG，请引用该文

### **License**

预训练权重由[牛津VGG组](http://www.robots.ox.ac.uk/~vgg/research/very_deep/)发布的预训练权重移植而来，基于[Creative Commons Attribution License](https://creativecommons.org/licenses/by/4.0/)

## **ResNet50模型**

keras.applications.resnet50.ResNet50(include\_top=**True**, weights='imagenet',

input\_tensor=**None**, input\_shape=**None**,

pooling=**None**,

classes=1000)

50层残差网络模型,权重训练自ImageNet

该模型在Theano和TensorFlow后端均可使用,并接受channels\_first和channels\_last两种输入维度顺序

模型的默认输入尺寸是224x224

### **参数**

* include\_top：是否保留顶层的全连接网络
* weights：None代表随机初始化，即不加载预训练权重。'imagenet'代表加载预训练权重
* input\_tensor：可填入Keras tensor作为模型的图像输出tensor
* input\_shape：可选，仅当include\_top=False有效，应为长为3的tuple，指明输入图片的shape，图片的宽高必须大于197，如(200,200,3)
* pooling：当include\_top=False时，该参数指定了池化方式。None代表不池化，最后一个卷积层的输出为4D张量。‘avg’代表全局平均池化，‘max’代表全局最大值池化。
* classes：可选，图片分类的类别数，仅当include\_top=True并且不加载预训练权重时可用。

### **返回值**

Keras 模型对象

### **参考文献**

* [Deep Residual Learning for Image Recognition](https://arxiv.org/abs/1512.03385)：如果在研究中使用了ResNet50，请引用该文

### **License**

预训练权重由[Kaiming He](https://github.com/KaimingHe/deep-residual-networks)发布的预训练权重移植而来，基于[MIT License](https://github.com/KaimingHe/deep-residual-networks/blob/master/LICENSE)

## **InceptionV3模型**

keras.applications.inception\_v3.InceptionV3(include\_top=**True**,

weights='imagenet',

input\_tensor=**None**,

input\_shape=**None**,

pooling=**None**,

classes=1000)

InceptionV3网络,权重训练自ImageNet

该模型在Theano和TensorFlow后端均可使用,并接受channels\_first和channels\_last两种输入维度顺序

模型的默认输入尺寸是299x299

### **参数**

* include\_top：是否保留顶层的全连接网络
* weights：None代表随机初始化，即不加载预训练权重。'imagenet'代表加载预训练权重
* input\_tensor：可填入Keras tensor作为模型的图像输出tensor
* input\_shape：可选，仅当include\_top=False有效，应为长为3的tuple，指明输入图片的shape，图片的宽高必须大于197，如(200,200,3)
* pooling：当include\_top=False时，该参数指定了池化方式。None代表不池化，最后一个卷积层的输出为4D张量。‘avg’代表全局平均池化，‘max’代表全局最大值池化。
* classes：可选，图片分类的类别数，仅当include\_top=True并且不加载预训练权重时可用。

### **返回值**

Keras 模型对象

### **参考文献**

* [Rethinking the Inception Architecture for Computer Vision](http://arxiv.org/abs/1512.00567)：如果在研究中使用了InceptionV3，请引用该文

### **License**

预训练权重由我们自己训练而来，基于[MIT License](https://github.com/KaimingHe/deep-residual-networks/blob/master/LICENSE)

## **InceptionResNetV2模型**

keras.applications.inception\_resnet\_v2.InceptionResNetV2(include\_top=**True**, weights='imagenet', input\_tensor=**None**, input\_shape=**None**, pooling=**None**, classes=1000)

InceptionResNetV2网络,权重训练自ImageNet

该模型在Theano、TensorFlow和CNTK后端均可使用,并接受channels\_first和channels\_last两种输入维度顺序

模型的默认输入尺寸是299x299

### **参数**

* include\_top：是否保留顶层的全连接网络
* weights：None代表随机初始化，即不加载预训练权重。'imagenet'代表加载预训练权重
* input\_tensor：可填入Keras tensor作为模型的图像输出tensor
* input\_shape：可选，仅当include\_top=False有效，应为长为3的tuple，指明输入图片的shape，图片的宽高必须大于197，如(200,200,3)
* pooling：当include\_top=False时，该参数指定了池化方式。None代表不池化，最后一个卷积层的输出为4D张量。‘avg’代表全局平均池化，‘max’代表全局最大值池化。
* classes：可选，图片分类的类别数，仅当include\_top=True并且不加载预训练权重时可用。

### **返回值**

Keras 模型对象

### **参考文献**

* [Inception-v4, Inception-ResNet and the Impact of Residual Connections on Learning](https://arxiv.org/abs/1602.07261)：如果在研究中使用了InceptionV3，请引用该文

### **License**

预训练权重基于[the Apache License](https://github.com/tensorflow/models/blob/master/LICENSE)

## **MobileNet模型**

keras.applications.mobilenet.MobileNet(input\_shape=**None**, alpha=1.0, depth\_multiplier=1, dropout=1e-3, include\_top=**True**, weights='imagenet', input\_tensor=**None**, pooling=**None**, classes=1000)

MobileNet网络,权重训练自ImageNet

该模型仅在TensorFlow后端均可使用,因此仅channels\_last维度顺序可用。当需要以load\_model()加载MobileNet时，需要在custom\_object中传入relu6和DepthwiseConv2D，即：

model = load\_model('mobilenet.h5', custom\_objects={

'relu6': mobilenet.relu6,

'DepthwiseConv2D': mobilenet.DepthwiseConv2D})

模型的默认输入尺寸是224x224

### **参数**

* include\_top：是否保留顶层的全连接网络
* weights：None代表随机初始化，即不加载预训练权重。'imagenet'代表加载预训练权重
* input\_tensor：可填入Keras tensor作为模型的图像输出tensor
* input\_shape：可选，仅当include\_top=False有效，应为长为3的tuple，指明输入图片的shape，图片的宽高必须大于197，如(200,200,3)
* pooling：当include\_top=False时，该参数指定了池化方式。None代表不池化，最后一个卷积层的输出为4D张量。‘avg’代表全局平均池化，‘max’代表全局最大值池化。
* classes：可选，图片分类的类别数，仅当include\_top=True并且不加载预训练权重时可用。
* alpha: 控制网络的宽度：
* 如果alpha<1，则同比例的减少每层的滤波器个数
* 如果alpha>1，则同比例增加每层的滤波器个数
* 如果alpha=1，使用默认的滤波器个数
* depth\_multiplier：depthwise卷积的深度乘子，也称为（分辨率乘子）
* dropout：dropout比例

### **返回值**

Keras 模型对象

### **参考文献**

* [MobileNets: Efficient Convolutional Neural Networks for Mobile Vision Applications](https://arxiv.org/pdf/1704.04861.pdf)：如果在研究中使用了MobileNet，请引用该文

### **License**

预训练基于[Apache License](https://github.com/tensorflow/models/blob/master/LICENSE)发布

## **DenseNet模型**

keras.applications.densenet.DenseNet121(include\_top=**True**, weights='imagenet', input\_tensor=**None**, input\_shape=**None**, pooling=**None**, classes=1000)

keras.applications.densenet.DenseNet169(include\_top=**True**, weights='imagenet', input\_tensor=**None**, input\_shape=**None**, pooling=**None**, classes=1000)

keras.applications.densenet.DenseNet201(include\_top=**True**, weights='imagenet', input\_tensor=**None**, input\_shape=**None**, pooling=**None**, classes=1000)

DenseNet网络,权重训练自ImageNet

该模型既可以使用channels\_first维度顺序(channels, height, width)，也可以使用channels\_last维度顺序(height, width, channels)

模型的默认输入尺寸是224x224

### **参数**

* include\_top：是否保留顶层的全连接网络
* weights：None代表随机初始化，即不加载预训练权重。'imagenet'代表加载预训练权重。或须载入的权重文件的路径。
* input\_tensor：可填入Keras tensor作为模型的图像输出tensor
* input\_shape：可选，仅当include\_top=False有效，应为长为3的tuple，指明输入图片的shape，图片的宽高必须大于32，如(200,200,3)
* pooling：当include\_top=False时，该参数指定了池化方式。None代表不池化，最后一个卷积层的输出为4D张量。‘avg’代表全局平均池化，‘max’代表全局最大值池化。
* classes：可选，图片分类的类别数，仅当include\_top=True并且不加载预训练权重时可用。

### **返回值**

Keras 模型对象

### **参考文献**

* [Densely Connected Convolutional Networks](https://arxiv.org/abs/1608.06993)(CVPR 2017 Best Paper Award)

### **License**

预训练基于[Apache License](https://github.com/tensorflow/models/blob/master/LICENSE)发布

## **NasNet模型**

keras.applications.nasnet.NASNetLarge(input\_shape=**None**, include\_top=**True**, weights='imagenet', input\_tensor=**None**, pooling=**None**, classes=1000)

keras.applications.nasnet.NASNetMobile(input\_shape=**None**, include\_top=**True**, weights='imagenet', input\_tensor=**None**, pooling=**None**, classes=1000)

神经结构搜索(NasNet)网络,权重训练自ImageNet

NASNetLarge模型的默认输入尺寸是331x331，NASNetMobile模型的默认输入尺寸是224x224

### **参数**

* input\_shape：可选，仅当include\_top=False有效，应为长为3的tuple，指明输入图片的shape，图片的宽高必须大于32，如(200,200,3)
* include\_top：是否保留顶层的全连接网络
* weights：None代表随机初始化，即不加载预训练权重。'imagenet'代表加载预训练权重
* input\_tensor：可填入Keras tensor作为模型的图像输出tensor
* pooling：当include\_top=False时，该参数指定了池化方式。None代表不池化，最后一个卷积层的输出为4D张量。‘avg’代表全局平均池化，‘max’代表全局最大值池化。
* classes：可选，图片分类的类别数，仅当include\_top=True并且不加载预训练权重时可用。

### **返回值**

Keras 模型对象

### **参考文献**

* [Learning Transferable Architectures for Scalable Image Recognition](https://arxiv.org/abs/1707.07012)

### **License**

预训练基于[Apache License](https://github.com/tensorflow/models/blob/master/LICENSE)发布

## **MobileNet模型**

keras.applications.mobilenetv2.MobileNetV2(input\_shape=**None**, alpha=1.0, depth\_multiplier=1, include\_top=**True**, weights='imagenet', input\_tensor=**None**, pooling=**None**, classes=1000)

MobileNet网络,权重训练自ImageNet

该模型仅channels\_last维度顺序(height, width, channels)可用。

模型的默认输入尺寸是224x224

### **参数**

* input\_shape：可选，仅当include\_top=False有效，应为长为3的tuple，指明输入图片的shape，如(200,200,3)
* alpha: 控制网络的宽度：
* 如果alpha<1，则同比例的减少每层的滤波器个数
* 如果alpha>1，则同比例增加每层的滤波器个数
* 如果alpha=1，使用默认的滤波器个数
* depth\_multiplier：depthwise卷积的深度乘子，也称为（分辨率乘子）
* include\_top：是否保留顶层的全连接网络
* weights：None代表随机初始化，即不加载预训练权重。'imagenet'代表加载预训练权重。或须载入的权重文件的路径。
* input\_tensor：可填入Keras tensor作为模型的图像输出tensor
* pooling：当include\_top=False时，该参数指定了池化方式。None代表不池化，最后一个卷积层的输出为4D张量。‘avg’代表全局平均池化，‘max’代表全局最大值池化。
* classes：可选，图片分类的类别数，仅当include\_top=True并且不加载预训练权重时可用。

### **返回值**

Keras 模型对象