



tf.keras.applications.vgg19.preprocess_input

 [TensorFlow 1 version](#) [\(/versions/r1.15/api_docs/python/tf/keras/applications/vgg19/preprocess_input\)](https://www.tensorflow.org/api_docs/python/tf/keras/applications/vgg19/preprocess_input)

 [View source](#) (<https://github.com/tensorflow/tensorflow/blob/r1.15/tensorflow/python/keras/applications/vgg19.py#L234>) [on GitHub](#)

Preprocesses a tensor or Numpy array encoding a batch of images.

 [View aliases](#)

Compat aliases for migration

See [Migration guide](https://www.tensorflow.org/guide/migrate) (<https://www.tensorflow.org/guide/migrate>) for more details.

tf.compat.v1.keras.applications.vgg19.preprocess_input

(https://www.tensorflow.org/api_docs/python/tf/keras/applications/vgg19/preprocess_input)

```
tf.keras.applications.vgg19.preprocess_input(  
    x, data_format=None  
)
```

Used in the notebooks

Used in the tutorials

- [Neural style transfer](https://www.tensorflow.org/tutorials/generative/style_transfer) (https://www.tensorflow.org/tutorials/generative/style_transfer)

Usage example with applications.MobileNet

(https://www.tensorflow.org/api_docs/python/tf/keras/applications/MobileNet):

```
i = tf.keras.layers.Input([None, None, 3], dtype = tf.uint8)  
x = tf.cast(i, tf.float32)  
x = tf.keras.applications.mobilenet.preprocess_input(x)  
core = tf.keras.applications.MobileNet()  
x = core(x)  
model = tf.keras.Model(inputs=[i], outputs=[x])  
  
image = tf.image.decode_png(tf.io.read_file('file.png'))  
result = model(image)
```

Args

x	A floating point <code>numpy.array</code> or a <code>tf.Tensor</code> (https://www.tensorflow.org/api_docs/python/tf/Tensor), 3D or 4D with 3 color channels, with values in the range [0, 255]. The preprocessed data are written over the input data if the data types are compatible. To avoid this behaviour, <code>numpy.copy(x)</code> can be used.
data_format	Optional data format of the image tensor/array. Defaults to None, in which case the global setting <code>tf.keras.backend.image_data_format()</code> (https://www.tensorflow.org/api_docs/python/tf/keras/backend/image_data_format) is used (unless you changed it, it defaults to "channels_last").

Returns

Preprocessed `numpy.array` or a `tf.Tensor` (https://www.tensorflow.org/api_docs/python/tf/Tensor) with type `float32`.

The images are converted from RGB to BGR, then each color channel is zero-centered with respect to the ImageNet dataset, without scaling.

Raises

ValueError	In case of unknown <code>data_format</code> argument.
-------------------	---

Except as otherwise noted, the content of this page is licensed under the [Creative Commons Attribution 4.0 License](https://creativecommons.org/licenses/by/4.0/) (<https://creativecommons.org/licenses/by/4.0/>), and code samples are licensed under the [Apache 2.0 License](https://www.apache.org/licenses/LICENSE-2.0) (<https://www.apache.org/licenses/LICENSE-2.0>). For details, see the [Google Developers Site Policies](https://developers.google.com/site-policies) (<https://developers.google.com/site-policies>). Java is a registered trademark of Oracle and/or its affiliates. Some content is licensed under the [numpy license](https://numpy.org/doc/stable/license.html) (<https://numpy.org/doc/stable/license.html>).

Last updated 2021-05-14 UTC.