

```
In [1]: import tensorflow as tf
print(tf.__version__)

from tensorflow.keras.preprocessing.image import ImageDataGenerator, array_to_img

datagen = ImageDataGenerator(
    rotation_range=40,
    width_shift_range=0.4,
    height_shift_range=0.4,
    shear_range=0.4,
    zoom_range=0.4,
    horizontal_flip=True,
    fill_mode='reflect') # Other parameters are: contant, wrap, nearest

img = load_img(r"C:\Users\lenovo\Desktop\Bam\Bam.jpg")
img
```

2.18.0

Out[1]:





```
In [2]: x = img_to_array(img) # this is a Numpy array with shape (3, 150, 150)
x = x.reshape((1,) + x.shape) # this is a Numpy array with shape (1, 3, 150, 150)

# the .flow() command below generates batches of randomly transformed images
# and saves the results to the `preview/` directory
i = 0
for batch in datagen.flow(x, batch_size=1,
                          save_to_dir=r"C:\Users\lenovo\Desktop\Data agumentatio
                          i += 1
                          if i > 27:
                              break # otherwise the generator would loop indefinitely
```