

```

import numpy as np

from skimage import io
import matplotlib.pyplot as plt

def imshow(image_RGB, title=''):
    io.imshow(image_RGB)

    plt.title(title)
    io.show()

def random_crop(img, crop_size):
    # Note: image_data_format is 'channel_last'
    assert img.shape[2] == 3
    height, width = img.shape[0], img.shape[1]
    dy, dx = crop_size
    x = np.random.randint(0, width - dx + 1)
    y = np.random.randint(0, height - dy + 1)
    return img[y:(y+dy), x:(x+dx), :]

def center_crop(img, crop_size):
    #width, height = img.size
    assert img.shape[2] == 3
    height, width = img.shape[0], img.shape[1]

    new_width, new_height = crop_size [0], crop_size[1]

    x = int((width - new_width)/2)
    y = int((height - new_height)/2)
    dx = new_width
    dy = new_height
    #right = int((width + new_width)/2)
    #bottom = int((height + new_height)/2)

    return img[y:(y+dy), x:(x+dx), :]

def crop_generator(batches, crop_length):
    """Take as input a Keras ImageGen (Iterator) and generate random
    crops from the image batches generated by the original iterator.
    https://jkjung-avt.github.io/keras-image-cropping/
    """
    while True:
        batch_x, batch_y = next(batches)
        batch_crops = np.zeros((batch_x.shape[0], crop_length, crop_length, 3))
        for i in range(batch_x.shape[0]):
            #batch_crops[i] = random_crop(batch_x[i], (crop_length, crop_length))
            batch_crops[i] = center_crop(batch_x[i], (crop_length, crop_length))
        yield (batch_crops, batch_y)

```

```
import datetime
def GetTimeString(m = -1):
    if m==0:
        s1 = datetime.datetime.now().strftime("%Y%m%d%H%M%S")
    else:
        s1 = datetime.datetime.now().strftime("%Y%m%d_%H%M%S")

    return s1
```