note.nkmk.me

Top > Python > Pillow

Generate square or circular thumbnail images with Python, Pillow

Date: 2019-05-14 / tags: Python, Pillow, Image Processing

Create square or circular thumbnail images using Python's image processing library Pillow (PIL).

Although there is a method called `thumbnail()` in the Pillow's `Image` module, it just resizes the image to fit within the specified size. Here I define my own function as an example.

The following contents will be described.

- How to make rectangular image square
  - Crop a square from a rectangular image
  - Add margins to make rectangle square
- Crop a square image into a circle
  - Make the background a solid color
  - Make the background transparent
- Sample code for batch processing

Please refer to the following articles for the installation and basic usage of Pillow (PIL).

- **Related post:** How to use Pillow (PIL: Python Imaging Library)

Import `Image`, `ImageDraw`, and `ImageFilter` from `PIL`. `ImageDraw` and `ImageFilter` are used to draw and process circles. When creating a square thumbnail image, they may be omitted.

For batch processing, import `os` and `glob`.

Read the image to be used in the following example, and decide the width (= height) of the thumbnail image you want to finally obtain.

```
import os
import glob

from PIL import Image, ImageDraw, ImageFilter

im = Image.open('data/src/lena.jpg')
thumb_width = 150
```

source: image_my_thumbnail.py

How to make rectangular image square

Because resizing a rectangular image to a square with `resize()` of `Image` module changes the aspect ratio, use one of the following methods.

- Crop a square from a rectangular image
- Add margins to make rectangle square

Crop a square from a rectangular image

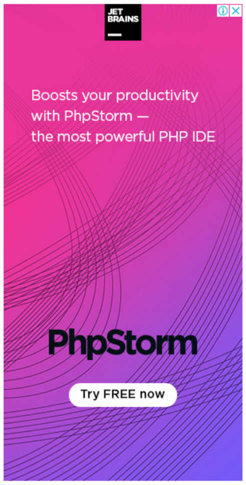
Categories

Python
NumPy pandas
OpenCV **Pillow**

Image Processing File
String Dictionary
Regular expression List

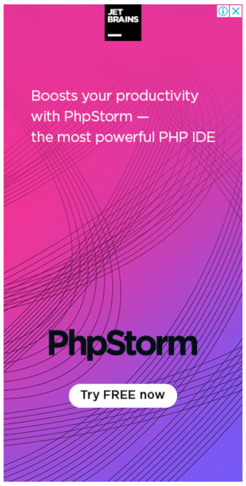
About

- GitHub: nkmk



Related Posts

- Composite two images according to a mask image with Python, Pillow
- Create and save animated GIF with Python, Pillow
- Invert image with Python, Pillow (Negative / positive inversion)
- Python, Pillow: Flip image
- Paste another image into an image with Python, Pillow



Crop an area of an image with `crop()`.

- **Related post:** Crop a part of the image with Python, Pillow (trimming)

Define a function to crop the central area of the image.

```python
def crop_center(pil_img, crop_width, crop_height):
    img_width, img_height = pil_img.size
    return pil_img.crop(((img_width - crop_width) // 2,
                         (img_height - crop_height) // 2,
                         (img_width + crop_width) // 2,
                         (img_height + crop_height) // 2))
```

source: imagelib.py

Use as follows.

```python
im_thumb = crop_center(im, thumb_width, thumb_width)
im_thumb.save('data/dst/lena_thumbnail_center_square.jpg', quality=95)
```

source: image_my_thumbnail.py

Instead of cropping the area of the thumbnail size, it is possible to crop the largest size square (= rectangular short side square) and then resize it.

Define a function to crop the largest sized square. It uses a function to crop the center area of the image.

```python
def crop_max_square(pil_img):
    return crop_center(pil_img, min(pil_img.size), min(pil_img.size))
```

source: imagelib.py

Use as follows. After making the rectangular image into a square, it is reduced by `resize()` to the size of the desired thumbnail image.

```python
im_thumb = crop_max_square(im).resize((thumb_width, thumb_width), Image.LANCZOS)
im_thumb.save('data/dst/lena_thumbnail_max_square.jpg', quality=95)
```

source: image_my_thumbnail.py

Add margins to make rectangle square

If you want to keep the entire original rectangular image, add margins at the top, bottom, left or right to make it square.

`new()` can be used to generate a solid image and paste it with `paste()`.

- **Related post:** Add margins to the image with Python, Pillow like enlarging the canvas

Define a function that adds a margin so that it will eventually become a square with the size of the long side of the rectangle.

```python
def expand2square(pil_img, background_color):
    width, height = pil_img.size
    if width == height:
        return pil_img
    elif width > height:
        result = Image.new(pil_img.mode, (width, width), background_color)
        result.paste(pil_img, (0, (width - height) // 2))
        return result
    else:
        result = Image.new(pil_img.mode, (height, height), background_color)
        result.paste(pil_img, ((height - width) // 2, 0))
        return result
```

source: imagelib.py

Use as follows. After making the rectangular image into a square, it is reduced by `resize()` to the size of the desired thumbnail image.

```python
im_thumb = expand2square(im, (0, 0, 0)).resize((thumb_width, thumb_width), Image.LANCZOS)
im_thumb.save('data/dst/lena_thumbnail_expand.jpg', quality=95)
```

source: image_my_thumbnail.py

Crop a square image into a circle

If you want to generate a circular thumbnail image, crop from square to circle.

There are two ways, one is to make the background cropped into a circle a solid color (white, black etc), and the other is to make it transparent (to make it transparent png).

Make the background a solid color

Use `composite()` to composite two images according to the mask image.

- **Related post:** Composite two images according to a mask image with Python, Pillow

Draw a circle and use it as a mask image. For details on drawing, see the following post.

- **Related post:** Draw circle, rectangle, line etc with Python, Pillow

Create a single color plain image of the desired background color with `new()` and composite it with the square image with a circular mask.

The border is smoothed by blurring the mask image with `ImageFilter`. Since the area of the circle spreads when it blurs, it is necessary to draw a smaller circle first.

Define the following function. Specify the background color `background_color`, the size of the blur `blur_radius`, and the offset `offset`. No blur with `blur_radius=0`.

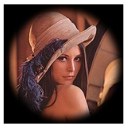
```
def mask_circle_solid(pil_img, background_color, blur_radius, offset=0):
    background = Image.new(pil_img.mode, pil_img.size, background_color)

    offset = blur_radius * 2 + offset
    mask = Image.new("L", pil_img.size, 0)
    draw = ImageDraw.Draw(mask)
    draw.ellipse((offset, offset, pil_img.size[0] - offset, pil_img.size[1] - offset), fi
    mask = mask.filter(ImageFilter.GaussianBlur(blur_radius))

    return Image.composite(pil_img, background, mask)
```

source: imagelib.py

Use as follows.

```
im_square = crop_max_square(im).resize((thumb_width, thumb_width), Image.LANCZOS)
im_thumb = mask_circle_solid(im_square, (0, 0, 0), 4)
im_thumb.save('data/dst/lena_thumbnail_mask_circle_solid.jpg', quality=95)
```

source: image_my_thumbnail.py



Make the background transparent

Use `putalpha()` which adds an alpha channel to the image.

- **Related post:** Create transparent png image with Python, Pillow (putalpha)

The flow is the same as when using a single color plain background.

```
def mask_circle_transparent(pil_img, blur_radius, offset=0):
    offset = blur_radius * 2 + offset
    mask = Image.new("L", pil_img.size, 0)
    draw = ImageDraw.Draw(mask)
    draw.ellipse((offset, offset, pil_img.size[0] - offset, pil_img.size[1] - offset), fi
    mask = mask.filter(ImageFilter.GaussianBlur(blur_radius))

    result = pil_img.copy()
    result.putalpha(mask)

    return result
```

source: imagelib.py

Transparent images are saved with `png`.

```
im_square = crop_max_square(im).resize((thumb_width, thumb_width), Image.LANCZOS)
im_thumb = mask_circle_transparent(im_square, 4)
im_thumb.save('data/dst/lena_thumbnail_mask_circle_transparent.png')
```

source: image_my_thumbnail.py

Sample code for batch processing

Create thumbnail images collectively from image files in any directory (folder).

Generate thumbnail images of image files in `src_dir` and save them in `dst_dir` .

```
src_dir = 'data/src'
dst_dir = 'data/dst'

files = glob.glob(os.path.join(src_dir, '*.jpg'))

for f in files:
    im = Image.open(f)
    im_thumb = crop_max_square(im).resize((thumb_width, thumb_width), Image.LANCZOS)
    ftitle, fext = os.path.splitext(os.path.basename(f))
    im_thumb.save(os.path.join(dst_dir, ftitle + '_thumbnail' + fext), quality=95)
```

source: image_my_thumbnail.py

Share

Tweet แชร์

Related Categories

- Python
- Pillow
- Image Processing

Related Posts

- Composite two images according to a mask image with Python, Pillow
- Create and save animated GIF with Python, Pillow
- Invert image with Python, Pillow (Negative / positive inversion)
- Python, Pillow: Flip image
- Paste another image into an image with Python, Pillow
- Create transparent png image with Python, Pillow (putalpha)
- Add margins to the image with Python, Pillow like enlarging the canvas
- Draw circle, rectangle, line etc with Python, Pillow
- Crop a part of the image with Python, Pillow (trimming)
- Python, Pillow: Rotate image
- Concatenate images with Python, Pillow
- Get image size (width, height) with Python, OpenCV, Pillow (PIL)
- Generate QR code image with Python, Pillow, qrcode
- How to use Pillow (PIL: Python Imaging Library)
- OpenCV, NumPy: Rotate and flip image