

LECTURE 24

PILLOW

MCS 275 Spring 2021

Emily Dumas

LECTURE 24: PILLOW

Course bulletins:

- Worksheet 9 will be ready soon.
- Read the [Project 3 description](#).
- Project 3 due 6:00pm CDT on Friday March 19.
- We'll discuss Project 3 a bit today.

INSTALL PILLOW

It is a package you can install with `pip`.

```
python3 -m pip install pillow
```

Or check the [official install instructions](#).

PILLOW AND PIL

Pillow provides a module called `PIL`.

The name difference is for historical reasons. Pillow replaces a module named `PIL` that was created for Python 2, and which hasn't been updated since 2011.

IMAGES

There are two basic types of image files you will encounter: **vector images** and **bitmap images**.

VECTOR IMAGES

Store instructions about what to draw (a circle here, a line there, etc.); can be viewed at any size without loss of sharpness.

- e.g. *SVG*, *PS*, *EPS*, *DXF*, *PDF*
- Good for drawings, diagrams, text

Displaying a vector image is a complicated operation!

BITMAP IMAGES

A rectangular grid of colors, meant to be displayed with each color corresponding to one pixel on the display device; becomes blurry or blocky if you zoom in.

- e.g. PNG, JPEG, GIF, BMP, TIFF
- Good for photos, screenshots

JPEG (ASIDE)

JPEG is for photos. That's what the P stands for.

JPEG compresses the image data, discarding some of it. Images with sharp edges will look bad as a result.

Use PNG for anything other than photos, unless it is essential to have a small file size.

PILLOW IS FOR BITMAPS

Pillow is for working with bitmap images. It can read and write PNG, JPEG, GIF, BMP, TIFF, and more.

It is useful for format conversion, low-level image operations (e.g. make this pixel red), and provides some high-level operations too (e.g. blur, sharpen, convert to grayscale, ...).

LOAD, SAVE, CREATE

PIL.Image is a class that represents bitmap images.

```
from PIL import Image

img = Image.open("adorable_kitten.png") # load
img.save("discord_avatar.jpg") # save

# new color image, 1920x1080 resolution, all white
img = Image.new("RGB", (1920,1080), color=(255,255,255))
```

MODES

An image file can store various amounts and types of color data. Pillow encodes this in a **mode** string:

- "1" - 1 bit per pixel, 0=black, 1=white
- "L" - 8 bits per pixel, 256 shades of gray. 0=black, 255=white. Also called "grayscale".
- "RGB" - 24 bits per pixel, 8 each for red, green, blue. Also called "true color". Most common.

These are common modes, but there are [lots more](#).

PIXEL COORDINATES

A location in a bitmap image is specified by a pair of integers (x, y) . The upper left corner is $(0, 0)$.

Coordinate x increases as you move right, and y increases as you move down.

Note the y direction is opposite from mathematics.

WORKING WITH PIXELS

Suppose `img` is a `PIL.Image` object.

Set a pixel color (draw a tiny dot):

```
# make pixel at (10,20) magenta
img.putpixel( (10,20), (255,0,255) )
```

Get a pixel color:

```
# returns color of pixel at (10,20)
img.getpixel( (10,20) )
```

OPERATIONS

Some other methods of `PIL.Image`:

- `convert` - Conversion to a different mode, e.g. from true color to grayscale.
- `crop` - Crop (remove all but a smaller rectangle).
- `resize` - Stretch or compress to a new size.
- `paste` - Draw another image on this one.
- `transpose` - Do any combination of mirroring and rotating by multiples of 90 degrees.

Lots more in the documentation.

OTHER OPTIONS

There are many Python image processing libraries, and for a particular purpose it may be best to use something other than Pillow. Examples:

- [OpenCV](#) is targeted at computer vision and machine learning applications (e.g. face detection)
- [Scikit-image](#) aims to be high-performance and to support video files. It uses `numpy` arrays extensively.

Generally, Pillow tends to emphasize minimal dependencies and doing basic things well.

REFERENCES

- [Pillow home page](#)
- [Pillow tutorial](#)
- [Pillow API reference](#)

REVISION HISTORY

- 2021-03-08 Fix name of local time zone at Chicago on March 19
- 2021-03-08 Initial publication

