

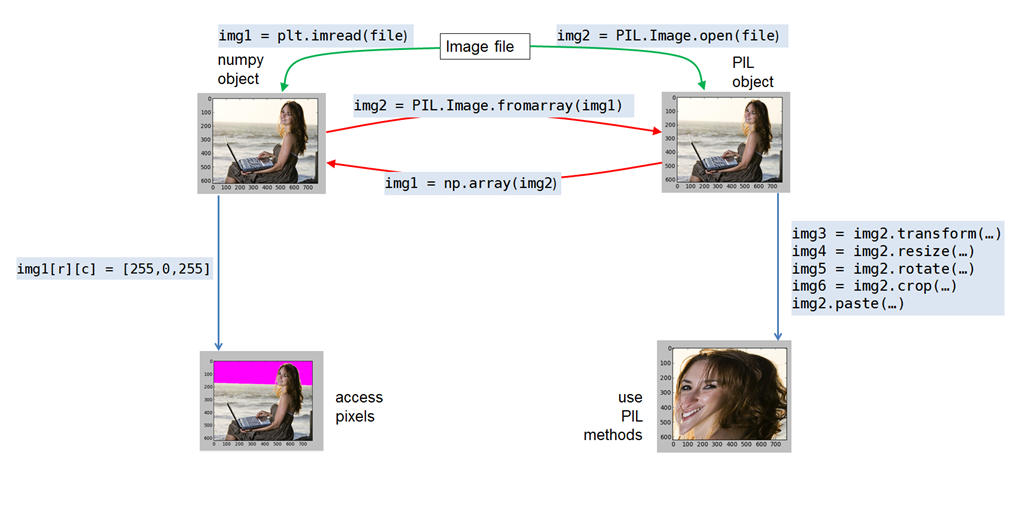
**Lesson 1.4 Reference Card for Pyplot and PIL**

|  |  |
| --- | --- |
| **Method** | **Description** |
| axis('on' | 'off') | Show/hide axes (and their titles and ticks)  *In documentation, the vertical line | usually means “or”. Here it shows that the argument can be either* 'on' *or* 'off'*.* |
| set\_xlim(*xmin*, *xmax*) | Set lower and upper limits to x-axis |
| set\_ylim(*ymin*, *ymax*) | Set lower and upper limits to y-axis |
| cla() | Clear axes |
| imshow(*img*) | Place an image on an axis |
| minorticks\_on() | Show minor ticks |
| minorticks\_off() | Hide minor ticks |
| set\_xlabel(*string*) | Set x-axis title |
| set\_ylabel(*string*) | Set y-axis title |
| set\_xticks(*list*) | Set which tick marks to label |
| set\_title(*string*) | Set subplot title |

Also useful: fig, ax = plt.subplots(*rows*, *columns*)

fig.canvas.draw()

**Figure 1. Methods for converting between PIL and numpy objects**



**Table 2. Methods of the PIL.Image class**

|  |  |
| --- | --- |
| **Method** | **Example and Comments** |
| open() | img = PIL.Image.open(*filename*)  *Creates a new PIL.Image object from a standard image file.* |
| new() | new\_img = PIL.Image.new(*mode, size, color*)  *Creates a new PIL.Image object of a solid color. For example, to create a 100 x 200 pixel image with all pixels set to RGB=5,5,5 with transparency (alpha=0):*  PIL.Image.new('RGBA', (100, 200), (5, 5, 5, 0) ) |
| crop() | new\_img = img2.crop( (*x0*, *y0*, *x1*, *y1*) )  *Creates a new image cropped to the coordinates within img2.* |
| convert() | new\_img = img2.convert('RGBA')  *Creates a new image like img2 but with the format specified – a good way to add an alpha channel.* |
| resize() | new\_img = img2.resize( (*new*\_*width, new\_height*))  *Creates a new image interpolating from img2.* |
| rotate() | new\_img = img2.rotate(angle, expand=False)  *Creates a new image rotated by angle degrees. Use expand=True to avoid cropping.* |
| paste() | img2.paste(*other\_img,(x,y), mask=none*)  *Changes img2 without creating a new image, pasting the top left of the other\_img*  *at the given coordinates in img2. Optionally, uses mask’s alpha channel.* |
| transform() | new\_img = img2.transform( (new\_width, new\_height), PIL.Image.QUAD,  (*x0*,*y0*,*x1*,*y1,x2*,*y2*,*x3,y3,x4*,*y4*) )  *Creates a new image by distorting a quadrilateral in img2 into a rectangle.* |
| save() | img2.save(filename)  *Saves image. Filename should be a string with a standard extension like .jpg or .png .* |

**Table 3. Methods of the PIL.ImageDraw class**

|  |  |
| --- | --- |
| This module must be imported separately from PIL: import PIL.ImageDraw | |
| **Method** | **Example and Comments** |
| Draw() | draw = PIL.ImageDraw.Draw(img2) *makes a drawing object* |
| ellipse() | draw.ellipse( *(left, top, bottom, right)*, fill=None, outline=None) |
| line() | draw.line(*[ (x1, y1), (x2, y2) ]*, *fill*=None) f |
| polygon() | draw.polygon( *[(x1, y1)*, *(x2, y2), (x3, y3),…],* fill=None, outline=None) |
| text() c | draw.text(*(x1, y1)*, *message*, fill=None, font=None) |