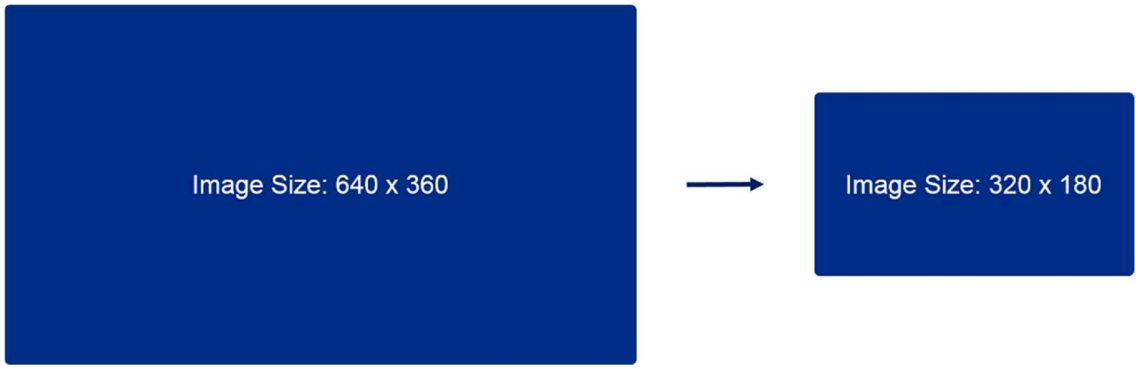
amoments_PS

1 Moment Calculator in Processing System (PS)

1.1 Contents

- Moment Calculating with using PIL library
- Import libraries
- Create an Image object
- Display the image
- Change the image size
- Display final image
- References



1.2 Import libraries

1. **PIL** library to load and resize the imag
2. **numpy** to store the pixel array of the image
3. **IPython.display** to show the image in the notebook

[0]:
```
from PIL import Image
import numpy as np
from IPython.display import display
```

1.3 Create an Image object

We will load image and create an Image object.

[2]:
```
image_path = "images/beuth.png"
original_image = Image.open(image_path)
original_image.load()
```

[2]: <PixelAccess at 0xb03bf3a0>

We now create a numpy array of the pixels.

[3]:
```
input_array = np.array(original_image)
```

1.4 Display the image

[4]:
```
input_image = Image.fromarray(input_array)
display(input_image)
```

[4]:

Let's double-check the original image size.

[5]:
```
old_width, old_height = original_image.size

print("Image size: {}x{}pixels.".format(old_width,old_height))
```

Image size: 640x360 pixels.

1.5 Change the image size

We will set image resize dimensions.

[0]:
```
resize_factor = 2
new_width = int(old_width/resize_factor)
new_height = int(old_height/resize_factor)
```

We will use resize () method from the PIL library. We map multiple input pixels to single output pixels to downscale the image
The Python Imaging Library provides different resampling filters. We use the default: NEAREST
Pick one nearest pixel from the input image. Ignore all other input pixels.

[7]:
```
resized_image = original_image.resize((new_width, new_height))
```

1.6 Display final image

[8]:
```
output_array = np.array(resized_image)
result = Image.fromarray(output_array)
display(result)
```

[8]:

Let's check the change the image size.

[9]:
```
width, height = resized_image.size
print("Resized image size: {}x{} pixels.".format(width, height))
```

Resized image size: 320x180 pixels.

We can time of the process in software operation.

[10]:
```
%%timeit
resized_image = original_image.resize((new_width, new_height), Image.BILINEAR)
```

10 loops, best of 3: 24.6 ms per loop

References

https://pillow.readthedocs.io/en/3.1.x/index.html

https://github.com/Xilinx/PYNQ/blob/master/docs/source/python_environment.ipynb
https://github.com/Xilinx/PYNQ/blob/master/docs/source/jupyter_notebooks.ipynb
https://github.com/Xilinx/PYNQ/blob/master/docs/source/jupyter_notebooks_advanced_features.ipyn
https://pillow.readthedocs.io/en/latest/handbook/concepts.html#filters