

Tutorial Setup Environment Pengolahan Citra

Prerequisites:

1. Python 3.6+
2. Pip
3. Virtualenv
4. Scikit-image: 0.15.0

Note: Jika Anda ingin menggunakan Jupyter Notebook sebagai *coding tool*, langsung lanjut ke tahap 3.

1. Install Pip

- a. Download file **get-pip.py** <https://bootstrap.pypa.io/get-pip.py>
- b. Buka console atau terminal kesayangan anda, lalu jalankan perintah ini.

```
python get-pip.py
```

- c. Kita juga bisa melihat versi pip dengan menjalankan perintah ini.

```
pip --version
```

2. Install virtualenv

Sangat disarankan membuat sebuah *virtual environment* sebelum memulai sebuah proyek, agar library yang diinstall tidak disimpan secara global dalam komputer.

Note: Seluruh library yang diinstall dalam sebuah *virtual environment* hanya akan bekerja apabila dinyalakan.

- a. Pertama-tama install dulu *virtualenv* menggunakan tools pip.

```
pip install virtualenv
```

- b. Buat sebuah *virtualenv* dengan menjalankan perintah ini.

```
virtualenv <nama_env>
```

- c. Jalankan *virtualenv* yang sudah dibuat.

Windows:

```
env\Scripts\activate
```

MacOS/Ubuntu

```
source env/bin/activate
```

- d. Jika ingin menonaktifkan *virtualenv*, jalankan perintah ini.

```
deactivate
```

3. Install Jupyter Notebook

Jupyter Notebook adalah sebuah aplikasi *web open source* yang digunakan untuk membuat dokumen yang berisi kode, persamaan, visualisasi, dan teks. Selama perkuliahan pengolahan citra, Anda dapat menggunakan Jupyter Notebook untuk mengolah data dengan Python dan memvisualisasikannya.

- a. Install Jupyter Notebook.

```
pip install jupyter
```

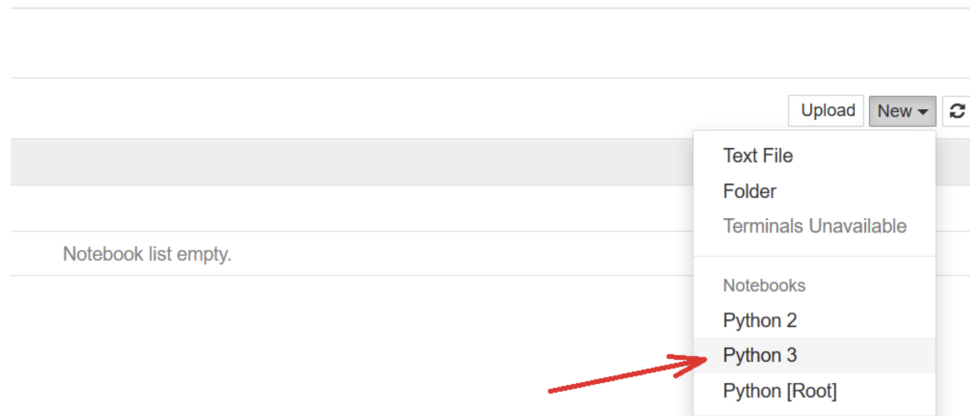
Setelah menjalankan perintah ini, maka pip akan menginstall beberapa library lainnya yang dibutuhkan oleh scikit-image.

- b. Run Jupyter Notebook.

```
jupyter notebook
```

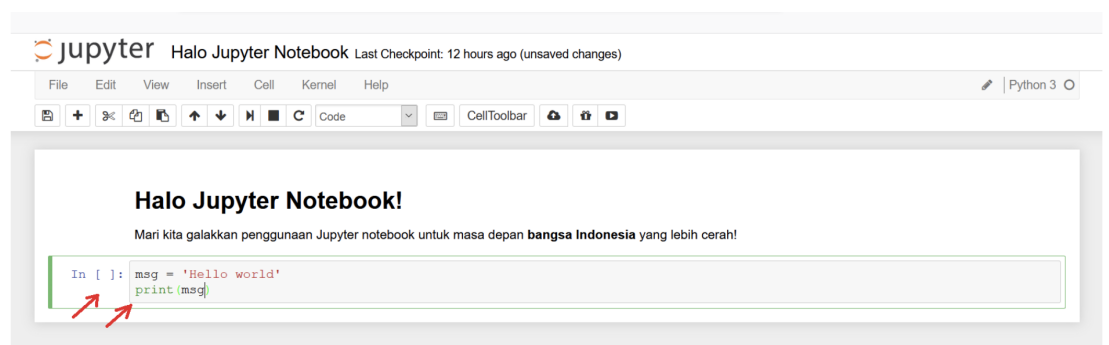
- c. Buat notebook baru.

Klik tombol **New**, lalu pilih **Python 3** seperti pada gambar di bawah ini:



d. Menulis dan Menjalankan Kode.

- Ubah jenis sel (terletak di *toolbar* atas) menjadi **Code**
- Tulis kode
- Jalankan dengan tekan **Shift + Enter**



e. Install library pada Jupyter Notebook.

Anda juga dapat langsung install library yang diperlukan dengan menjalankan perintah pip pada cell Jupyter Notebook.

```
pip install [package_name]
```

Contoh, Anda ingin install library NumPy.

```
pip install numpy
```

Note: Anda juga dapat menggunakan Google Colab atau Kaggle Notebook sebagai *coding tools*.

Pengolahan Citra pada Python

1. Import Library

Untuk menggunakan library pengolahan citra pada python, bisa menggunakan scikit-image dan matplotlib. Gunakan syntax berikut untuk mengimport library tersebut:

```
from skimage import color, io
from skimage.transform import rescale
import matplotlib.pyplot as plt
```

Apabila ada library yang belum terinstall, dapat dilakukan instalasi menggunakan pip.

2. Membaca Citra

Download gambar yang telah disediakan di SCell kemudian ekstrak dan letakkan seluruh gambar pada direktori tempat anda bekerja.

Lakukan pembacaan *fasilkom.png* dan *fasilkomgelap.png*. Untuk membaca gambar, gunakan perintah `io.imread('file_name')`, contoh:

```
i1 = io.imread('fasilkom.png')
```

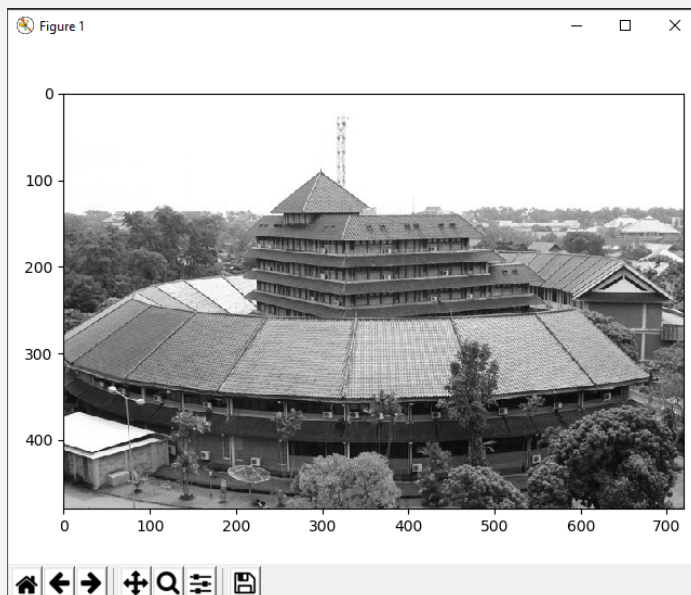
Anda juga dapat membaca gambar dengan memasukkan path dari file yang akan dibaca, contoh:

```
i1 = io.imread('C:\Image\fasilkomgelap.png')
```

3. Menampilkan Citra

Untuk menampilkan gambar, gunakan fungsi `plt.imshow(var_name)`, contoh:

```
plt.imshow(i1, cmap='gray', vmin=0, vmax=255)  
plt.show()
```



4. Mengecek Dimensi Citra

```
dimension = i1.shape
```

5. Menyimpan Gambar

Untuk menyimpan gambar, gunakan fungsi `io.imsave('file_name', var_name)`, Contoh:

```
io.imsave('fasilkombaru.png', i1)
```

6. Visualisasi Citra

| Fungsi | Keterangan |
|--|------------------------------|
| <code>plt.plot(x, x)</code> <code>plt.plot(x, np.sin(x))</code> | Plot 1D |
| <code>plt.axis([xmin, xmax, ymin, ymax])</code> | Merubah koordinat |
| <code>plt.title("Image 1")</code> | Menambahkan judul pada plot. |

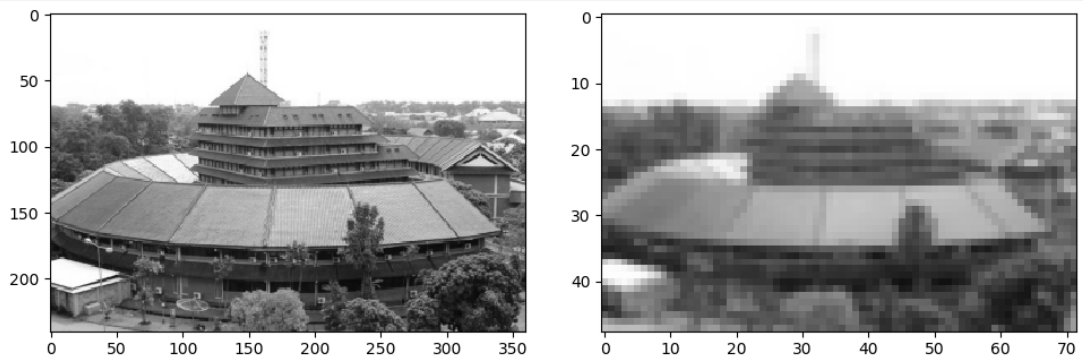
```
plt.subplot(n_row, n_col, index)
```

Menampilkan beberapa citra, dengan index dimulai dari 1 hingga n.

7. Resolusi

Resolusi adalah kerapatan pixel pada suatu gambar, semakin besar resolusi spasial maka semakin banyak pixel yang digunakan dalam menampilkan suatu citra.

```
i3 = rescale(i1, 0.5, anti_aliasing=True)
i4 = rescale(i1, 0.1, anti_aliasing=True)
```



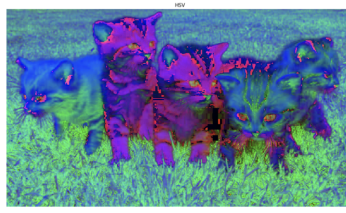
8. Konversi Citra

Untuk mengkonversi citra, dapat menggunakan beberapa fungsi yang sudah disediakan oleh scikit-image, yaitu sebagai berikut.

| Fungsi | Kegunaan | Format |
|-----------------------|--|---|
| <code>rgb2hsv</code> | RGB ke HSV. | <code>y=color.rgb2hsv(x)</code> |
| <code>hsv2rgb</code> | HSV ke RGB. | <code>y=color.hsv2rgb(x)</code> |
| <code>rgb2gray</code> | RGB ke grayscale. | <code>y=color.rgb2gray(x)</code> |
| <code>gray2rgb</code> | Membuat representasi RGB dari image grayscale. | <code>y=color.gray2rgb(x[, alpha])</code> |



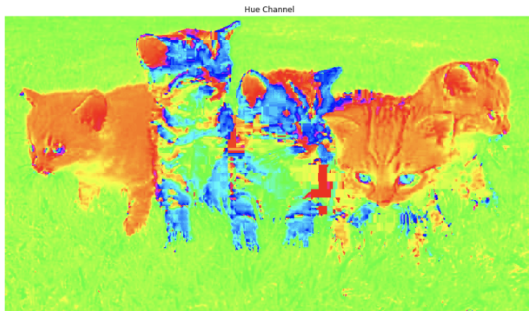
Original



HSV



Grayscale



Hue Channel



Value Channel

Selamat mencoba!