

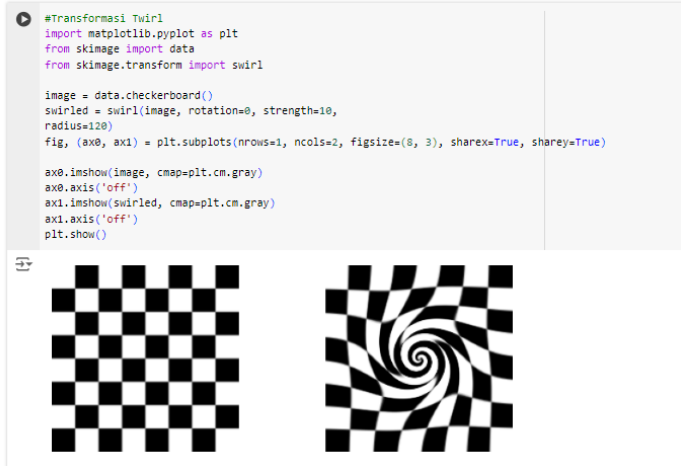
Nama : Vilda Azizah Wiguna

Nim : 1217070085

## Tugas Praktikum 7 Pengolahan Citra digital

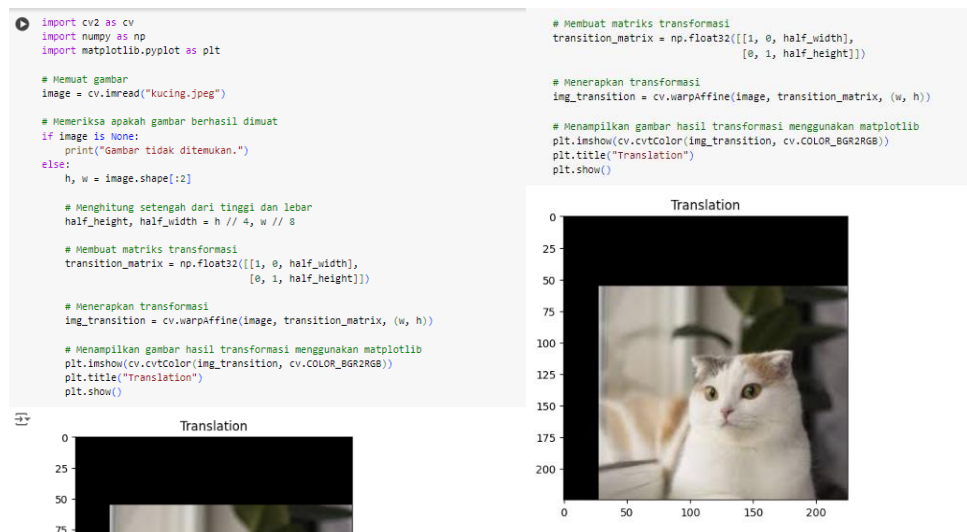
### 1. Tranformasi Twirl

✓ Vilda Azizah Wiguna (1217070085)



Transformasi twirl merupakan salah satu jenis transformasi geometris yang menghasilkan efek "berputar" pada suatu gambar. Transformasi ini sering digunakan dalam pengolahan gambar untuk menghasilkan efek artistic.

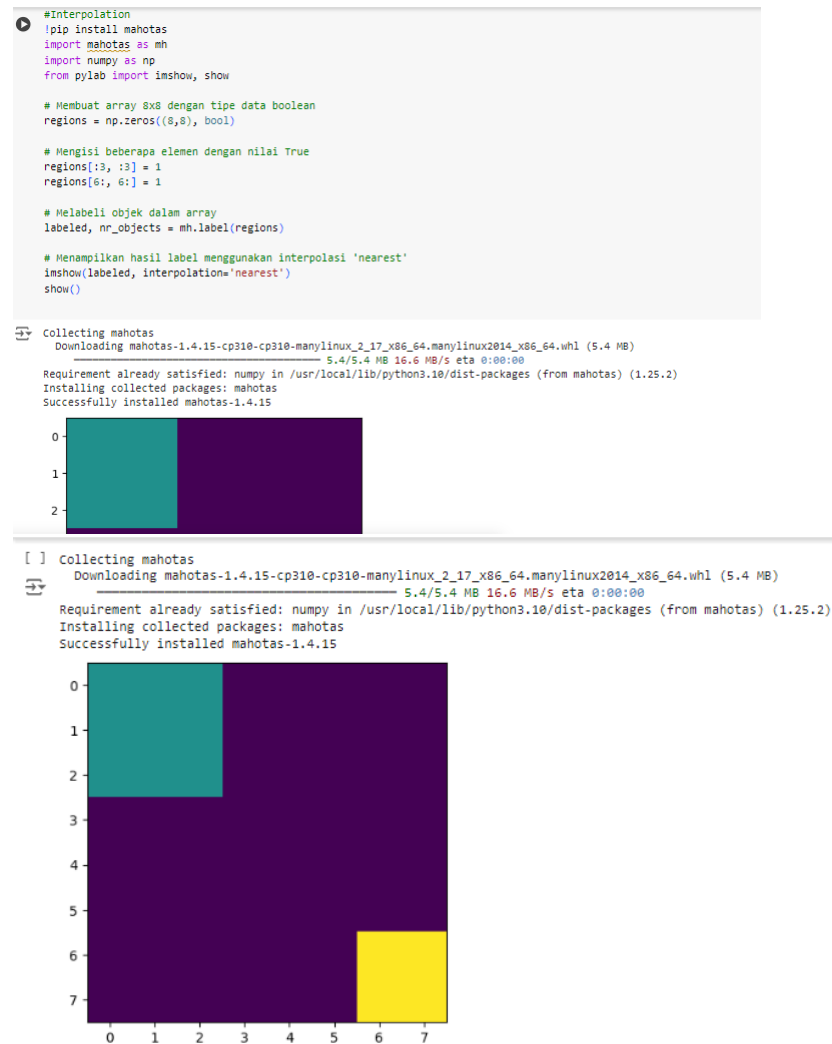
### 2. Translation



### 3. Rotation



### 4. Interpolation



```
# Install mahotas jika belum terpasang
!pip install mahotas

import mahotas as mh
import numpy as np
from pylab import imshow, show

# Membuat array 8x8 dengan tipe data boolean
regions = np.zeros((8,8), bool)

# Mengisi beberapa elemen dengan nilai True
regions[:3, :3] = 1
regions[6:, 6:] = 1

# Melabeli objek dalam array
labeled, nr_objects = mh.label(regions)

# Menampilkan hasil label menggunakan interpolasi 'nearest'
imshow(labeled, interpolation='nearest')
show()

# Melabeli objek dengan struktur yang lebih besar
labeled, nr_objects = mh.label(regions, np.ones((3,3), bool))
sizes = mh.labeled.labeled_size(labeled)

print('Background size:', sizes[0])
print('Size of first region:', sizes[1])

# Membuat array acak dengan ukuran yang sama dengan regions
array = np.random.random_sample(regions.shape)

# Menghitung jumlah nilai pada region tertentu dalam array
sums = mh.labeled_sum(array, labeled)
print('Sum of first region:', sums[1])
```

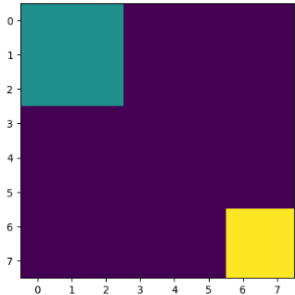
Requirement already satisfied: mahotas in /usr/local/lib/python3.10/dist-packages (1.4.15)

```
print('Size of first region:', sizes[1])

# Membuat array acak dengan ukuran yang sama dengan regions
array = np.random.random_sample(regions.shape)

# Menghitung jumlah nilai pada region tertentu dalam array
sums = mh.labeled_sum(array, labeled)
print('Sum of first region:', sums[1])
```

Requirement already satisfied: mahotas in /usr/local/lib/python3.10/dist-packages (1.4.15)  
Requirement already satisfied: numpy in /usr/local/lib/python3.10/dist-packages (from mahotas) (1.25.2)



Background size: 51  
Size of first region: 9  
Sum of first region: 4.490559240029162

## 5. Skala Interpolasi Miring

```
import cv2 as cv
import numpy as np
import matplotlib.pyplot as plt

# Memuat gambar
image = cv.imread("kucing.jpg")

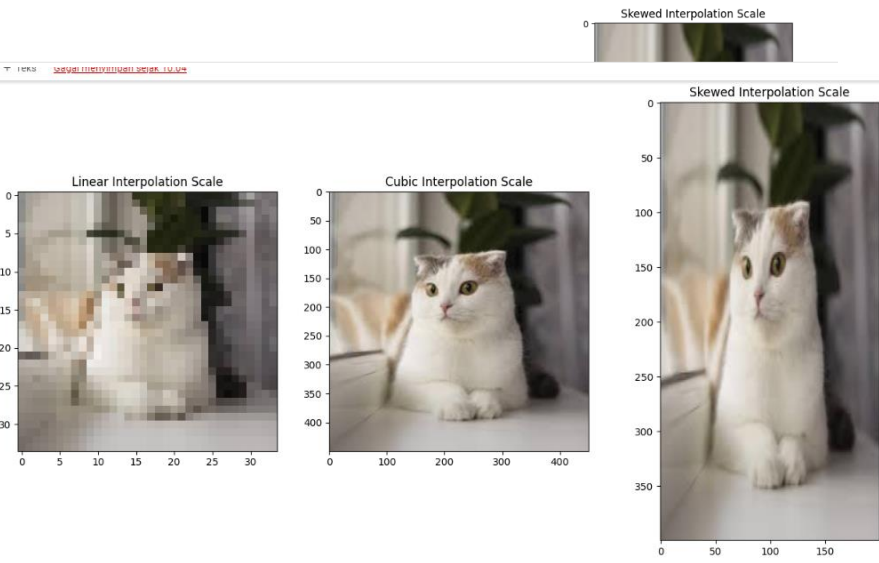
# Memeriksa apakah gambar berhasil dimuat
if image is None:
    print("Gambar tidak ditemukan.")
else:
    # Membuat subplot
    fig, ax = plt.subplots(1, 3, figsize=(16, 8))

    # Skala gambar menjadi 0.15 kali dari ukuran aslinya
    image_scaled = cv.resize(image, None, fx=0.15, fy=0.15, interpolation=cv.INTER_LINEAR)
    ax[0].imshow(cv.cvtColor(image_scaled, cv.COLOR_BGR2RGB))
    ax[0].set_title("Linear Interpolation Scale")

    # Skala gambar menjadi 2 kali dari ukuran aslinya
    image_scaled_2 = cv.resize(image, None, fx=2, fy=2, interpolation=cv.INTER_CUBIC)
    ax[1].imshow(cv.cvtColor(image_scaled_2, cv.COLOR_BGR2RGB))
    ax[1].set_title("Cubic Interpolation Scale")

    # Skewed scale: merubah ukuran gambar ke 200x400
    image_scaled_3 = cv.resize(image, (200, 400), interpolation=cv.INTER_AREA)
    ax[2].imshow(cv.cvtColor(image_scaled_3, cv.COLOR_BGR2RGB))
    ax[2].set_title("Skewed Interpolation Scale")

    # Menampilkan plot
    plt.show()
```



Skewed Interpolation Scale

Linear Interpolation Scale

Cubic Interpolation Scale

Skewed Interpolation Scale

Terhubung ke backend Google Compute Engine Python 3