

# **IMAGE STITCHING**



**Dosen Pengampu :**

**Adi Hermansyah, M. T**

**Disusun Oleh :**

**M. Athallah rafif aldera**

**09011282227066**

**PROGRAM STUDI SISTEM KOMPUTER FAKULTAS ILMU**

**KOMPUTER UNIVERSITAS SRIWIJAYA**

**2023/2024**

## Langkah Pertama: Menyiapkan *Module*

Pertama, kita pastikan bahwa sudah memiliki python beserta pip. Lalu jalan kan beberapa command untuk menginstall beberapa *module* ke dalam linux anda. Berikut beberapa command yang harus Anda jalankan:

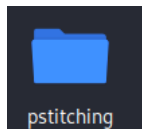
```
$ pip install opencv-python
```

```
$ pip install stitching
```

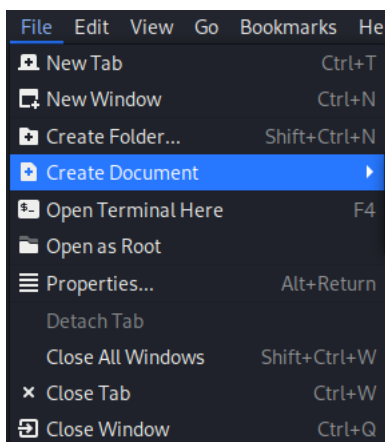
```
$ pip install imutils
```

## Langkah kedua: Menyiapkan Foto dan Program

Siapkan direktori baru untuk menyimpan foto dan program yang akan digunakan untuk menjakankan image stitching. Disini saya membuatnya dengan nama “*pstitching*”, seperti berikut:



Lalu, kita akan membuat file yang berformat “.py” untuk menuliskan program image stitching kita ke dalamnya,



Setelah itu salin program berikut kedalam file yang berformat “.py” tadi.

```
from imutils import paths
```

```
import numpy as np
```

```
import argparse
```

```
import imutils
```

```
import cv2
```

```
ap = argparse.ArgumentParser()
```

```
ap.add_argument("-i", "--images", type=str, required=True,
```

```
                help="path to input directory of images to stitch")
```

```

ap.add_argument("-o", "--output", type=str, required=True,
                help="path to the output image")
args = vars(ap.parse_args())

print("[INFO] loading images...")
imagePaths = sorted(list(paths.list_images(args["images"])))
images = []

for imagePath in imagePaths:
    image = cv2.imread(imagePath)
    images.append(image)

print("[INFO] stitching images...")
stitcher = cv2.createStitcher() if imutils.is_cv3() else cv2.Stitcher_create()
(status, stitched) = stitcher.stitch(images)

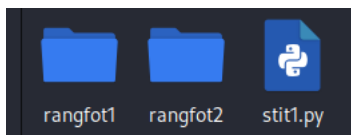
if status == 0:
    # write the output stitched image to disk
    cv2.imwrite(args["output"], stitched)

    # display the output stitched image to our screen
    cv2.imshow("Stitched", stitched)
    cv2.waitKey(0)

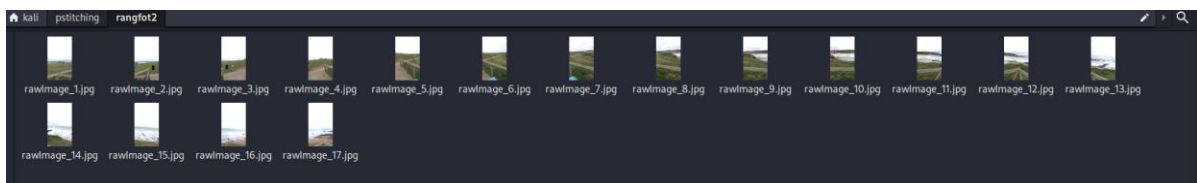
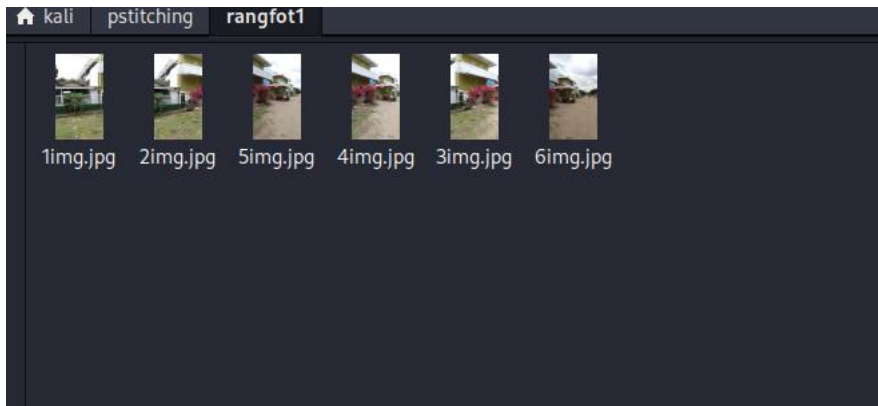
else:
    print("[INFO] image stitching failed ({}).format(status))

```

Setelah program telah dibuat maka kita akan memindahkan foto kedalam direktori “*pstitching*” tadi dengan menambah direktori baru, seperti yang ada di lampiran berikut:



Untuk kasus kali ini saya memiliki 2 rangkaian foto yang dimasukan kedalam direktori berbeda, yang cuplikan dari kedua direktori nya seperti berikut:



## Langkah Ketiga: Menjalankan Program

Buka terminal dan pindah ke direktori yang kita buat tadi dengan “`cd pstitching`”,

```
(kali@kali)-[~]
$ cd pstitching
```

lalu jalankan command berikut:

```
$ python3 stit2.py --images rangfot1 --output hasil1.jpg
```

```
(kali@kali)-[~/pstitching]
$ python3 stit2.py --images rangfot1 --output hasil1.jpg
[INFO] loading images ...
[INFO] stitching images ...
```

Maka outup nya yaitu hasil1.jpg akan seperti berikut:



Lalu bisa dilanjutkan juga dengan menggunakan rangkaian foto yang ke dua dengan hanya mengubah command nya menjadi:

```
$ python3 stit2.py --images rangfot2 --output hasil2.jpg
```

```
(kali@kali)-[~/pstitching]
$ python3 stit2.py --images rangfot2 --output hasil2.jpg
[INFO] loading images ...
[INFO] stitching images ...
```

Dan output dari rangkaian foto yang kedua adalah seperti, berikut:



Dapat dilihat dari kedua output yang sudah dibuat terlihat, bahwa output yang dihasilkan tidak sepenuhnya rata menutupi semua frame, hal ini terjadi karena rangkaian foto sebelum dirangkai tidak lah rapi dan lurus seperti foto panorama pada umumnya, dan hal ini merupakan hal yang normal karena terjadi dikarenakan rangkainan foto yang tidak lurus dan simetris.