

MODUL 1 – Penggunaan Github dan Google Colaboratory, Membaca dan Menampilkan Citra

A. TUJUAN

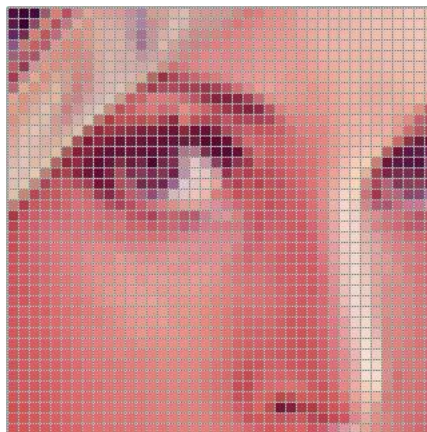
1. Mampu membuat *Repository* di *Github*
2. Mampu membuat *Python Notebook* di *Google Colaborator*
3. Mampu menyimpan dan mengakses *Python Notebook* dari *Google Colaborator*
4. Mampu membuka *File* Citra dari *Code Python*
5. Mampu mengakses pixel pada citra

B. ALAT DAN BAHAN

1. PC/LAPTOP
2. Github
3. *Google Colaborator*

C. ULASAN TEORI

Citra Digital (Citra Raster) adalah representasi numerik dari citra dua dimensi. Nilai numerik yang direpresentasikan umumnya adalah nilai biner 8 bit. Nilai biner ini disimpan pada elemen citra yang sering disebut sebagai pixel. Citra digital berisi pixel yang jumlah baris dan kolomnya tetap. Pixel adalah elemen gambar terkecil dari citra digital. Pixel disimpan pada memory computer sebagai map raster, yaitu array dua dimensi bertipe integer.

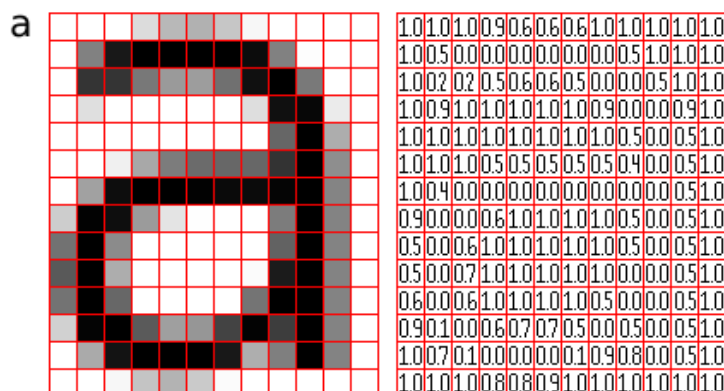


Citra Raster diakuisisi menggunakan berbagai macam perangkat input dan teknik, seperti digital camera, scanner, radar, camera infra merah, dan lain sebagainya. Pengolahan Citra Digital adalah ilmu yang mempelajari algoritma transformasi citra.

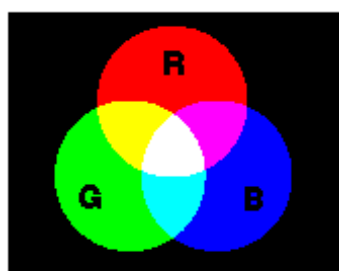
Citra digital yang umum digunakan pada pengolahan citra adalah Citra biner, Citra Keabuan, dan Citra Berwarna RGB (Red, Green, dan Blue).

Citra biner adalah citra yang memiliki 2 warna saja, yaitu hitam dan putih. Jika direpresentasikan dengan nilai biner 8bit adalah warna hitam bernilai 0000 0000, dan putih bernilai 1111 1111. biasa ditampilkan dengan nilai normalisasi 0 dan 1, atau decimal 0 dan 255, atau heksadesimal 00x dan FFx.

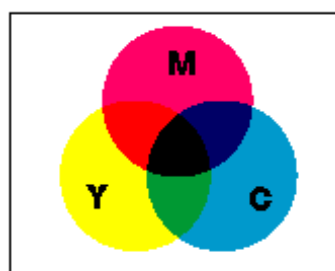
Citra keabuan adalah citra yang memiliki derajat keabuan sebanyak 256 warna. Dimulai dengan warna terkecilnya yaitu hitam, dan warna terbesarnya adalah putih. Pada gambar berikut ditunjukkan representasi nilai normalisasi citra keabuan dimana 1.0 menyatakan warna putih, 0.0 menyatakan warna hitam, dan nilai antara 0.0 – 1.0 menyatakan warna derajat keabuannya.



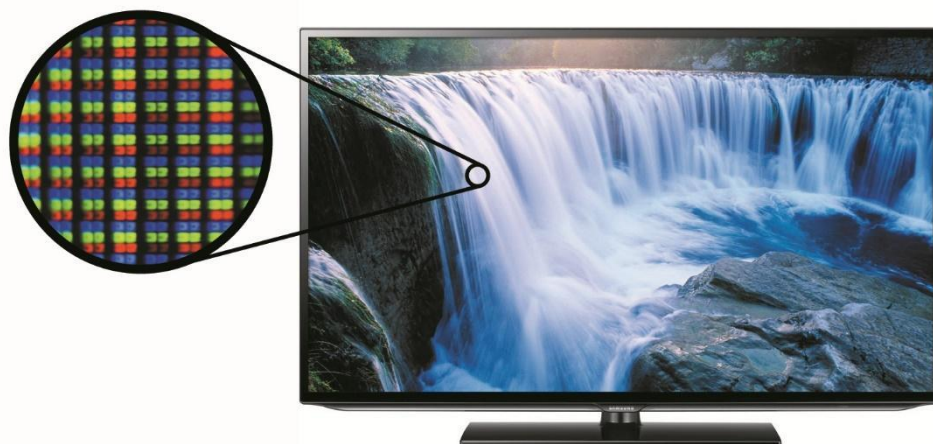
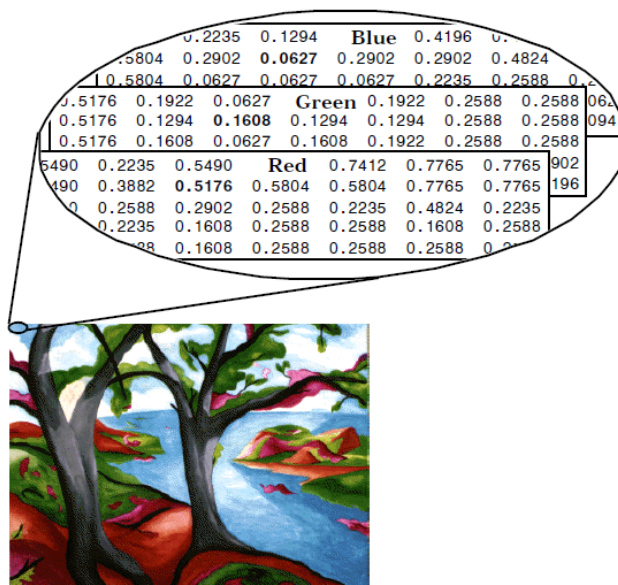
Citra berwarna RGB adalah citra yang memiliki 3 level / channel warna direpresentasikan dengan resolusi citra 3 dimensi. Pada citra digital, level pertama digunakan untuk menyimpan warna R (Red / merah), Level kedua digunakan untuk menyimpan warna G (Green / hijau), dan level ketiga digunakan untuk menyimpan warna B (Blue / biru). Pada perangkat keluaran seperti LCD Monitor, RGB disusun secara array dan berukuran sangat kecil. Representasi nilainya disimpan dalam nilai biner. Warna Hitam direpresentasikan dengan R = 0000 0000, G = 0000 0000, dan B = 0000 0000. Warna Merah direpresentasikan dengan R = 1111 1111, G = 0000 0000, dan B = 0000 0000. Karena memiliki 3 level, maka ukuran file citra RGB dibandingkan dengan citra keabuan adalah 3:1 dengan jumlah pixel (resolusi citra) yang sama. Warna lain selain Merah, hijau, dan biru adalah campuran dari ketiga warna tersebut. Perhatikan gambar berikut, warna kuning adalah campuran dari warna merah dan hijau. Kuning direpresentasikan dengan nilai R = 1111 1111, G = 1111 1111, dan B = 0000 0000.



RGB: TV's and Monitors
Use Additive Color



CMY: Color Printing Press
Use Subtractive Color



Citra digital dapat ditampilkan pada berbagai macam perangkat lunak penampil citra (Image Viewer). Web Browser dapat menampilkan standard citra format internet secara langsung seperti GIF, JPEG, dan PNG. Beberapa browser dapat menampilkan SVG yang merupakan format standard W3C. Beberapa citra sains saat ini dapat berukuran sangat besar (sebagai contoh 46 giga pixel ukuran citra dari galaxy BimaSakti, berukuran 194 GB).

Bahasa Python termasuk bahasa pemrograman trend di beberapa tahun terakhir ini. Pada awalnya Python merupakan kelanjutan bahasa pemrograman ABC yang dikembangkan oleh Guido van Rossum pada tahun 1990 di CWI, Amsterdam. Python memiliki ciri khusus jika dibandingkan dengan bahasa lain dalam hal penulisan kode program yang memiliki aturan mengenai indentasi, tipe data, tuple, dan dictionary. Kelebihan Python dibandingkan dengan bahasa pemrograman lain terutama terlihat dalam hal penanganan modul, serta keunggulan Python yang merupakan produk yang opensource, free, dan multiplatform. Python memiliki beberapa keunggulan antara lain : Terdapat modul-modul yang telah disediakan oleh Python; tata bahasa lebih mudah dipahami; layout yang dimiliki lebih mudah untuk ditinjau ulang dan dikembangkan; berorientasi obyek; pengolahan memori dilakukan secara modular; dapat dibangun dengan bahasa lain Python sendiri ataupun C/C++.modul-modul tersebut dapat dibangun dengan bahasa Python maupun C/C++; serta kelebihan lain.

Google Collaboratory atau Google Colab merupakan *tools* yang berbasis *cloud* dan bersifat *free*. Google Colab dibuat dengan *environment* jupyter dan mendukung banyak pustaka

(*library*) yang dibutuhkan dalam lingkungan pengembangan *Artificial Intelligence* (AI). Google Colab memungkinkan penggunaanya untuk menulis dan mengeksekusi Python di *browser* (Chrome, Firefox dan Safari) tanpa memerlukan konfigurasi, dapat mengakses GPU secara gratis, serta dapat berbagi kode program (kolaborasi tim) dengan mudah. Google Colab dapat mengeksekusi, menulis, menyimpan bahkan membagikan kode program yang telah dibuat melalui google drive. Penulisan kode program Python serta eksekusi kode program tersebut tidak memerlukan proses instalasi dengan mendownload installer pada komputer. Instalasi tidak perlu dilakukan karena semua proses dilakukan di cloud. Selain itu fungsionalitas tambahan Python dapat juga memanfaatkan built-in library yang ada pada Google Colab.

D. PRAKTIKUM

1. Bagi yang belum mempunyai akun Github, bisa membuat akun baru di Github (https://github.com/join?ref_cta=Sign+up&ref_loc=header+logged+out&ref_page=%2F&source=header-home)

Join GitHub

Create your account

Username *

Email address *


Password *

Make sure it's at least 15 characters OR at least 8 characters including a number and a lowercase letter. [Learn more.](#)


Email preferences

☒ Send me occasional product updates, announcements, and offers.


Verify your account



School work and student projects



Use the GitHub API



Other

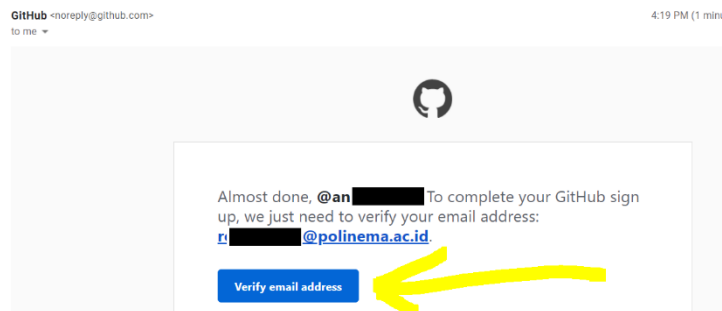
I am interested in:

We'll connect you with communities and projects that fit your interests.

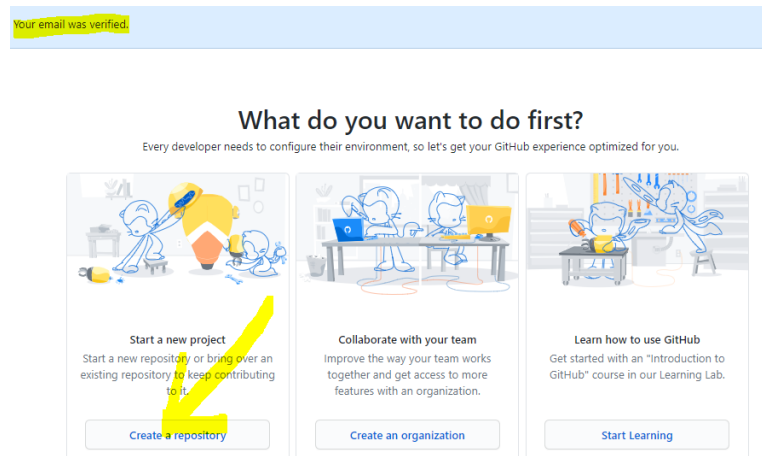
For example: [lineageos](#) [actions](#) [producthunt](#)

Complete setup

Setelah melakukan pengisian semua *mandatory field*, klik “complete setup”. Langkah selanjutnya adalah verifikasi alamat email yang digunakan untuk membuat akun baru.




2. Masuk ke dalam akun github yang telah dibuat, dan buatlah repositori baru, dengan memilih “Create a Repository”.



Isikan nama repositori, deskripsi (opsional), dan pilih apakah repositori yang akan dibuat bersifat *public* atau *private*. Pilih *Private* jika project tidak ingin diakses oleh publik. Setelah itu klik “Create repository” untuk membuat repositori baru.

Create a new repository

Owner * Repository name *

 andrie2000 / PCVK_Genap_2021 ✓

Great repository names are short and memorable. Need inspiration? How about [fuzzy-octo-disco](#)?

Description (optional)

Berisi Project-project untuk praktikum mata kuliah PCVK Genap 20/21 JTI Polinema

☐ Public
Anyone on the internet can see this repository. You choose who can commit.


☒ Private
You choose who can see and commit to this repository.

Initialize this repository with:
Skip this step if you're importing an existing repository.

☒ Add a README file
This is where you can write a long description for your project. [Learn more.](#)

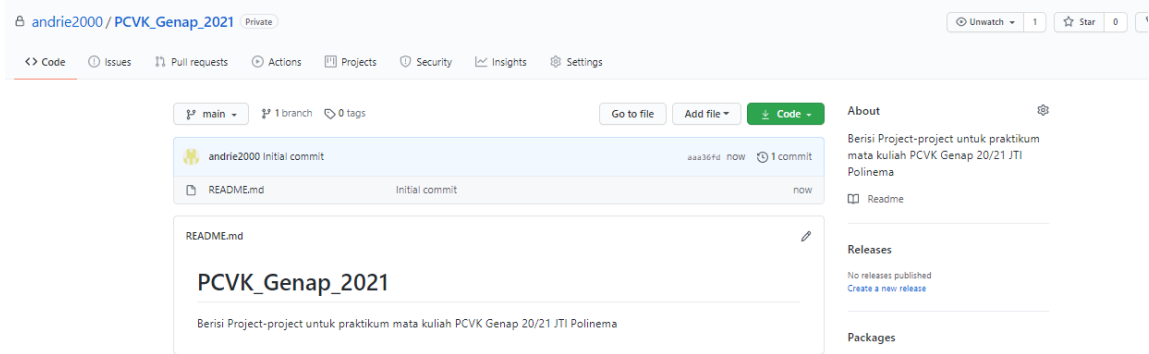
☐ Add .gitignore
Choose which files not to track from a list of templates. [Learn more.](#)

☐ Choose a license
A license tells others what they can and can't do with your code. [Learn more.](#)

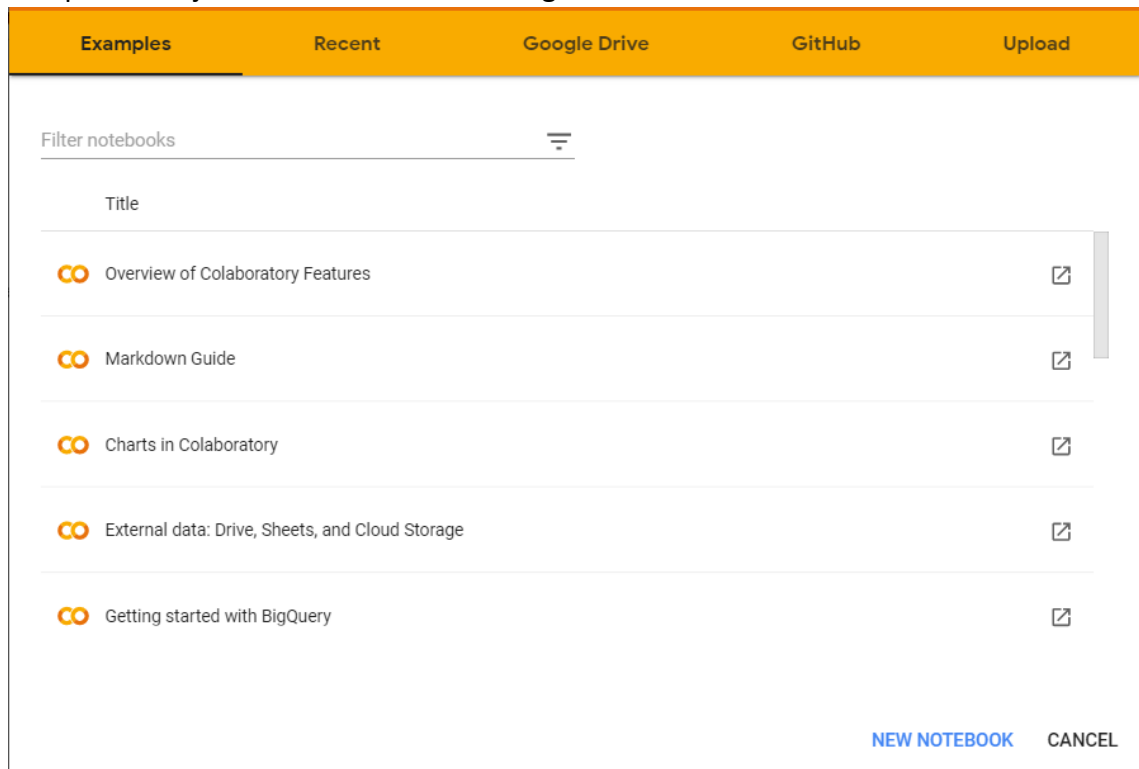
This will set  main as the default branch. Change the default name in your [settings](#).

[Create repository](#)

3. Repository telah siap diisikan dengan project-project untuk perkuliahan PCVK Genap 20/21.



4. Pada praktikum pengolahan citra dan visi komputer, kita akan melakukan *editing* dan *build code* untuk pengolahan citra menggunakan *Google Colaboratory*. *Google Colaboratory* dapat dicari menggunakan *search engine* atau dapat langsung dibuka pada link berikut: <https://colab.research.google.com/>
Tampilan dari jendela utama adalah sebagai berikut:



5. Aplikasi baru akan dibuat dalam format notebook python (tipe file adalah **.ipynb*), dimana file tersebut dapat disimpan pada drive cloud Google Colab, Google Drive, ataupun dapat terhubung pada Github secara langsung. Pada Kuliah ini kita akan menggunakan Github. Pilih menu Github pada jendela utama, kemudian lanjutkan untuk terhubung dengan Github personal anda.

Contoh

Terbaru

Google Drive

GitHub

Upload

Masukkan URL GitHub atau telusuri menurut organisasi atau pengguna

☐ Sertakan repositori pribadi

<https://github.com/milyun> 🔍

Repositori: [🔗](#)
 milyun/react-pwa

Cabang: [🔗](#)
 master

Jalur


Tidak ada hasil

NOTEBOOK BARU BATAL


Masukkan URL akun github Anda (https://github.com/nama_akun), kemudian tekan enter atau tekan icon search, kemudian akan muncul daftar repositori dan cabang yang terdapat pada akun Anda. Jika repository yang dibuat pada tahap 2 bersifat private, maka pilih checkbox “Sertakan repositori pribadi”, sehingga muncul jendela untuk memberikan otorisasi pada Google Colab, dan tekan tombol “*Authorize googlecolab*”

CO

✓




Authorize Colaboratory



Colaboratory by googlecolab


wants to access your milyun account



Gists

Read and write access

⌵




Repositories

Public and private

⌵


Organization access



jti-polinema

×


Request



polinema-mobile

×

Request



sipeno

×

Request

Cancel

Authorize googlecolab

Authorizing will redirect to
<https://colab.research.google.com>

- Setelah google colab terhubung dengan Github, Anda bisa memilih repositori dan membuat notebook baru.

Contoh

Terbaru

Google Drive

GitHub

Upload

Masukkan URL GitHub atau telusuri menurut organisasi atau pengguna ☒ Sertakan repositori pribadi

milyun

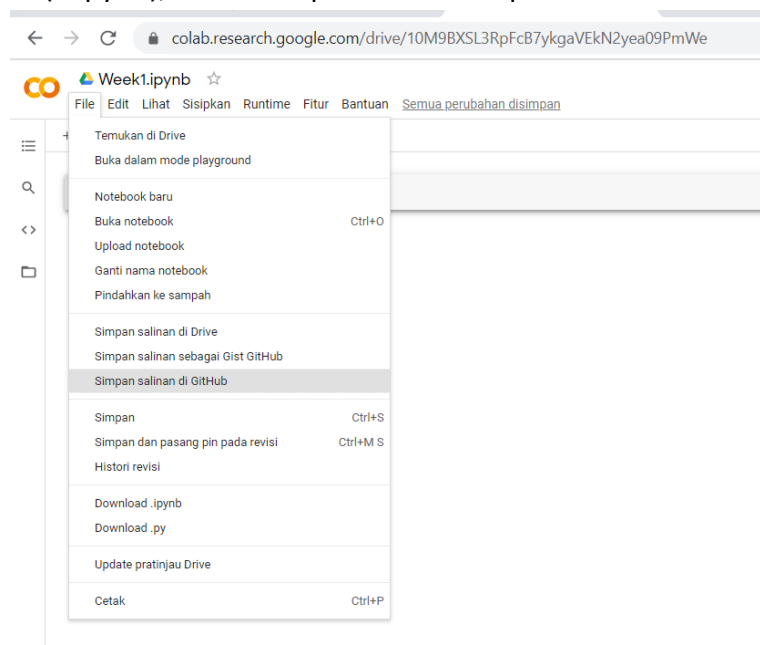
Repositori: [milyun/PCVK_Genap_2021](#) Cabang: [main](#)

Jalur

Tidak ada hasil

[NOTEBOOK BARU](#) [BATAL](#)

Ganti nama file (*.ipynb), kemudian pilih File → Simpan Salinan ke Github



Sehingga akan muncul jendela seperti berikut:

Salin ke Github

Repositori: [milyun/PCVK_Genap_2021](#)

Cabang: [main](#)

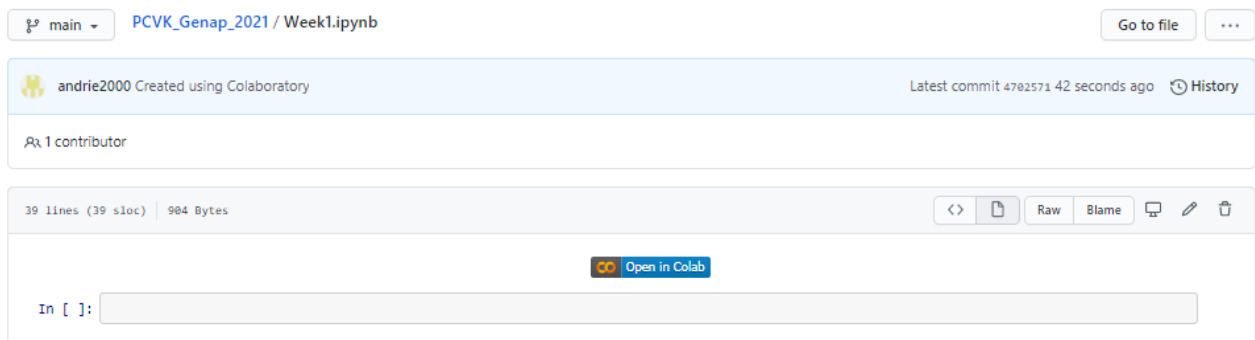
Jalur file
Week1.ipynb

Pesan pengubahan
Dibuat menggunakan Colaboratory

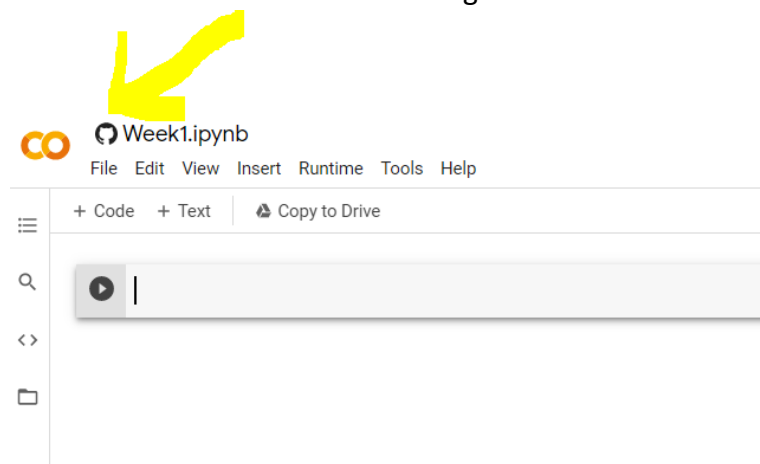
☒ Sertakan link ke Colaboratory

[BATAL](#) [OKE](#)

- Setelah menekan tombol “OKE” maka akan langsung terbuka halaman file notebook yang baru saja dibuat pada repositori Github.



Di pojok kiri atas dari Google Colab perhatikan jika icon Github sudah muncul, tandanya file tersebut berhasil diakses oleh Google Colab.



File notebook dapat berisi catatan terformat dan berisi code python yang bisa langsung di running secara langsung. Tombol “+ Code” untuk menambahkan code program, dan “+ Text” digunakan untuk menambahkan catatan terformat.

- Gunakan beberapa library berikut sebagai langkah pertama:

Langkah (Step) 1:

Pada code awal, beberapa library yang akan digunakan adalah: **numpy, pandas, cv2, skimage, PIL, matplotlib**

--Bahasa--

--English--

- [Numpy](#) is an array manipulation library, used for linear algebra, Fourier transform, and random number capabilities.
- [Pandas](#) is a library for data manipulation and data analysis.
- [CV2](#) is a library for computer vision tasks.
- [Skimage](#) is a library which supports image processing applications on python.
- [Matplotlib](#) is a library which generates figures and provides graphical user interface toolkit.

```
[9] import numpy as np
import pandas as pd
import cv2 as cv
from google.colab.patches import cv2_imshow # for image display
from skimage import io
from skimage import transform
from PIL import Image
import matplotlib.pyplot as plt
```

- Langkah 2 untuk membaca dan menampilkan image

Langkah (Step) 2:

Baca Image dari URLs - Read Image From URLs

--Bahasa--

Pada Langkah ini kita akan membaca image dari URLs, menampilkannya menggunakan OpenCV, perhatikan ada perbedaan ketika membuka image dalam RGB dan BGR. Channel Warna default untuk OpenCV adalah BGR.

--English--

In this step we will read images from urls, and display them using openCV, please note the difference when reading image in RGB and BGR format. The default input color channels are in BGR format for openCV. RGB?

```
# Membuat list untuk menyimpan url dari beberapa image
urls = ["https://iiif.lib.ncsu.edu/iiif/0052574/full/800,/0/default.jpg",
        "https://iiif.lib.ncsu.edu/iiif/0016007/full/800,/0/default.jpg",
        "https://placekitten.com/800/571"]

# baca dan tampilkan image
# loop pada tiap url image, beberapa image dapat disimpan pada list
for url in urls:
    image = io.imread(url)           #read image
    image = cv.resize(image, (0,0), fx=0.5, fy=0.5) #resize image to half size
    image_2 = cv.cvtColor(image, cv.COLOR_BGR2RGB) #convert color to RGB
    final_frame = cv.hconcat((image, image_2))    #concatenate image
    cv2.imshow(final_frame)                       #show image
    print('\n')
```

Pada tahap 2 kita akan membuat sebuah list untuk menyimpan URL beberapa citra, dimana untuk setiap citra akan dilakukan: pembacaan citra, resize ukuran citra menjadi setengahnya, konversi citra berwarna menjadi format RGB, menggabungkan citra asli dan citra hasil konversi, dan yang terakhir adalah menampilkan citra tersebut. Untuk url citra yang digunakan silahkan copy paste URL berikut, atau Anda juga bisa menggunakan URL citra yang lain:

```
urls = ["https://iiif.lib.ncsu.edu/iiif/0052574/
/full/800,/0/default.jpg", "https://iiif.lib.ncsu.edu/iiif
/0016007/full/800,/0/default.jpg", "https://placekitten.co
m/800/571"]
```

Hasil dari potongan kode di atas adalah sebagai berikut:



10. Langkah 3 pada praktikum ini adalah melihat ukuran file image, dengan cara sebagai berikut:

Langkah (Step) 3

melihat ukuran File image

```
[51] tinggi = image_2.shape[0]
lebar = image_2.shape[1]
print("resolusi image: tinggi x lebar = ",tinggi," x ",lebar)
cv2_imshow(image_2)
```

resolusi image: tinggi x lebar = 286 x 400



11. Langkah 4 berikut digunakan untuk mengakses pixel dengan memberikan garis horizontal berwarna putih di tengah image

Langkah (Step) 4

mengakses pixel dengan memberi garis diagonal menyilang

```
image_2 = cv.cvtColor(image, cv.COLOR_BGR2RGB)
image_3 = cv.cvtColor(image, cv.COLOR_BGR2RGB)

#membuat garis horizontal ditengah image
for y in range (lebar):
    image_3[int((tinggi)/2),y] = [255,255,255]

final_frame = cv.hconcat((image_2, image_3))
cv2.imshow(final_frame)
```



Pertanyaan

1. Jelaskan, mengapa pada modul praktikum ini eksekusi kode Python dilakukan menggunakan Google Colab?
2. Jelaskan mengenai kegunaan setiap library pada praktikum langkah ke delapan? Apakah semua library tersebut harus digunakan dalam praktikum sesi ini?
3. Pada uji coba langkah ke-9 terdapat potongan kode program sebagai berikut :
`image = cv.resize(image, (0,0), fx=0.5, fy=0.5)`

Apakah kegunaan kode program tersebut? dan apa pengaruhnya jika tidak dilakukan?

4. Perhatikan potongan kode program berikut :

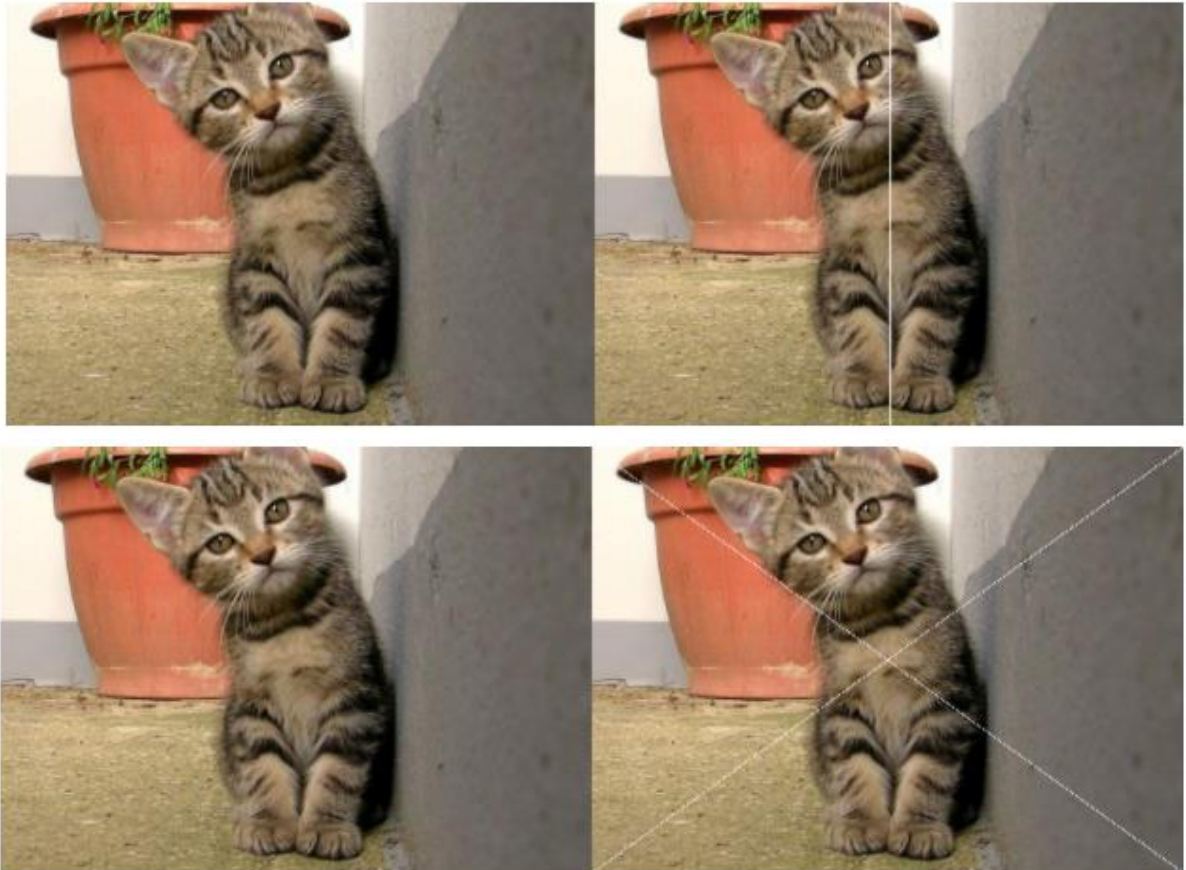
```
#membuat garis horizontal ditengah image
for y in range (lebar):
    image_3[int((tinggi)/2),y] = [255,255,255]
```

Apakah kegunaan kode [255,255,255] ? Jelaskan!

5. Jelaskan keterkaitan antara pixel dan juga resolusi gambar yang tinggi ataupun rendah!

E. TUGAS

1. Lakukan langkah-langkah praktikum seperti diatas
2. Buat garis vertikal dan garis menyilang diagonal pada image keluaran



3. Buat garis horisontal berwarna putih dibagian tengah gambar dengan panjang tertentu



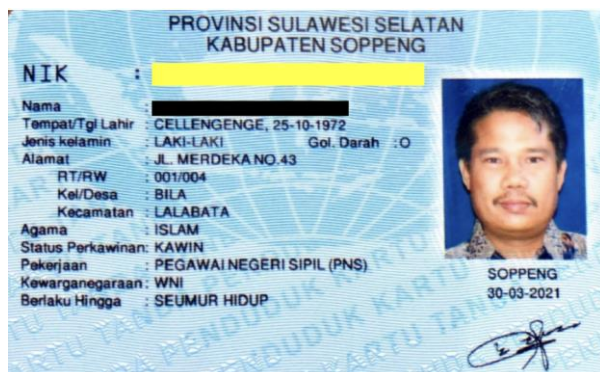
4. Buat kotak menggunakan kumpulan pixel warna putih di sembarang tempat dalam gambar



Bentuk kelompok masing-masing terdiri dari 2-3 orang. tiap kelompok lakukan langkah berikut:

1. Akses file image lokal (KTP/SIM) dan tampilkan menggunakan OpenCV dan Matplotlib
2. Identifikasi kenapa menggunakan konversi warna BGR2RGB pada matplotlib dan tanpa konversi pada opencv
3. Tutup tiap bagian tertentu dari kartu identitas tersebut. Kelompok 1 tutup bagian nomor NIK dan nama. Kelompok 2 tutup bagian TTL dan Jenis Kelamin. Kelompok 3 tutup bagian Gol. Darah dan TTD. Kelompok 4 tutup bagian Alamat dan Kecamatan. Kelompok 5 tutup bagian Agama dan Status Kawin. Kelompok 6 tutup Pekerjaan dan kewarganegaraan. Kelompok 7 tutup bagian berlaku hingga dan foto. Kelompok 8 tutup bagian kabupaten dan tanggal dikeluarkan KTP yang ada dibagian bawah foto. Kelompok 9 tutup bagian nama propinsi dan kabupaten di bagian paling atas KTP. Kelompok 10 tutup bagian RT/RW dan Kel/desa. Tutup menggunakan 2 kotak warna yang berbeda. Tidak boleh menggunakan fungsi rectangle opencv, fungsi ini akan dipelajari pada pertemuan 2.

Contoh hasil:



--- SELAMAT BELAJAR ---