



ABOUT PROJECT

Melakukan perancangan sistem untuk mengklasifikasi buah tomat berdasarkan gambar sample buah tomat yang diambil dan dimasukkan ke dalam sistem, dengan memberikan user tampilan, apakah buah tomat tersebut termasuk dalam kategori buah tomat matang, setengah matang . belum matang atau busuk.

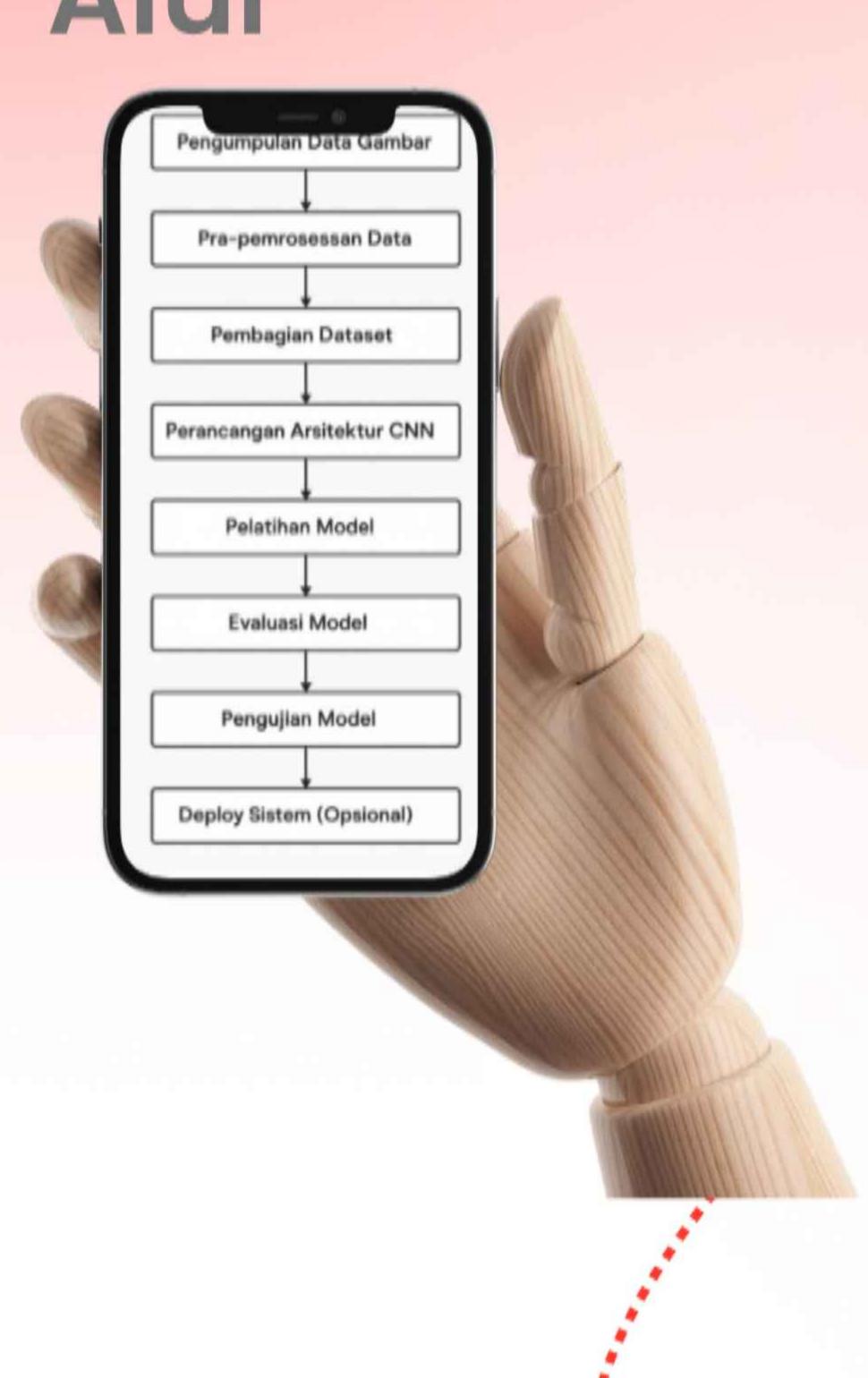


Matang









1. Pengumpulan Data Gambar – Mengambil

Alur Pengerjaan

- gambar untuk pelatihan.
- 2. Pra-pemrosesan Menyiapkan gambar (resize, normalisasi).
- 3. Pembagian Dataset Memisah data jadi latih, validasi, dan uji.
- 5. Pelatihan Melatih model dengan data latih.

4. Desain CNN – Membuat arsitektur model.

- 6. Evaluasi Mengukur performa model.
- 8. Deploy (Opsional) Mengintegrasikan model

7. Pengujian – Menguji model dengan data baru.

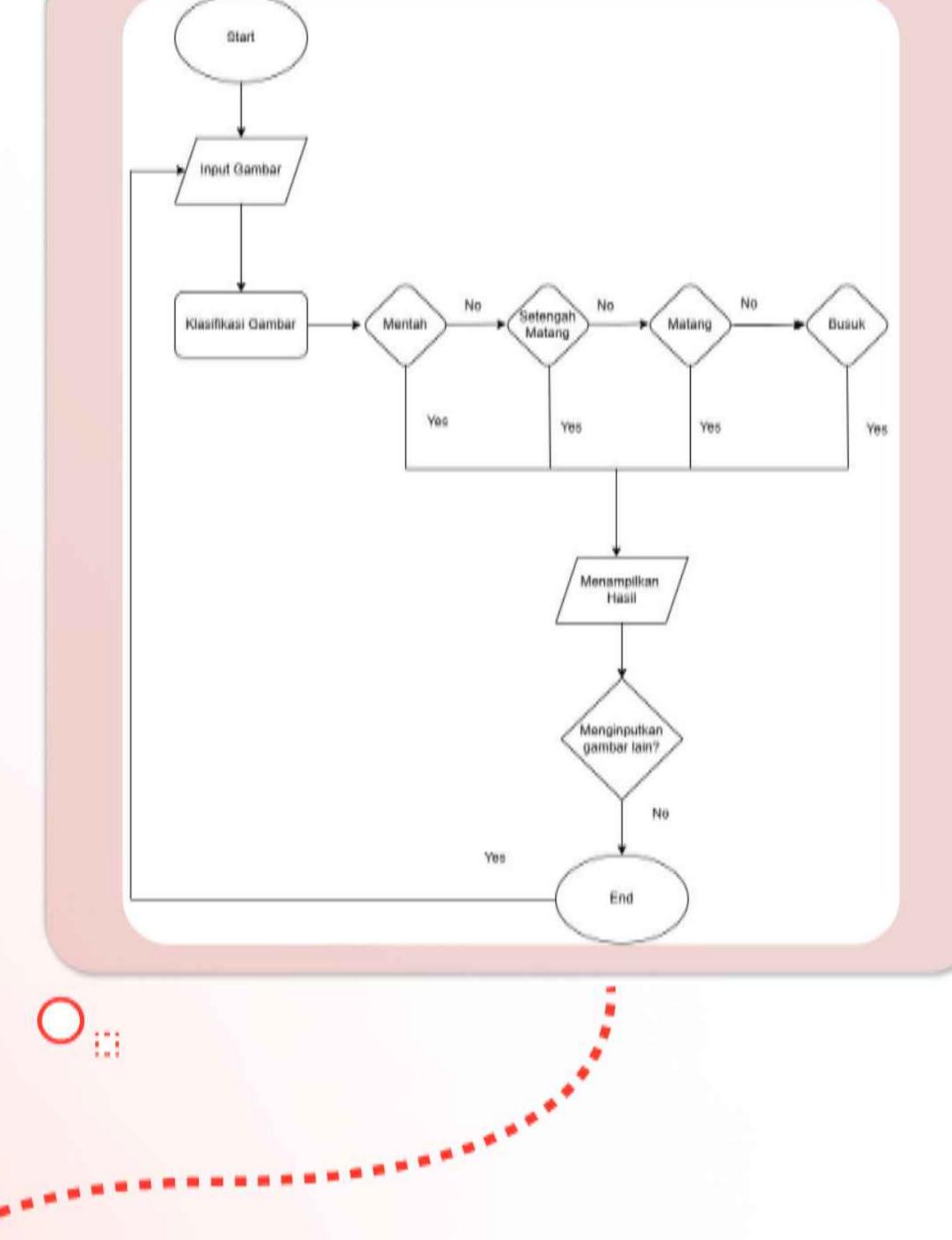
ke aplikasi.

FLOWCHART

mendeteksi apakah buat tomat tersebut Mentah, Setengah Matang atau Matang, setelah itu akan menampilkan hasilnya dimana dapat juga presentasi dari kategori buah tomat tersebut, setelah itu jika user ingin menginputkan kembali gambar maka akan kembali ke proses seperti sebelumnya.

User diharuskan memasukkan citra tomat

yang akan dideteksi, lalu aplikasi akan



testing +

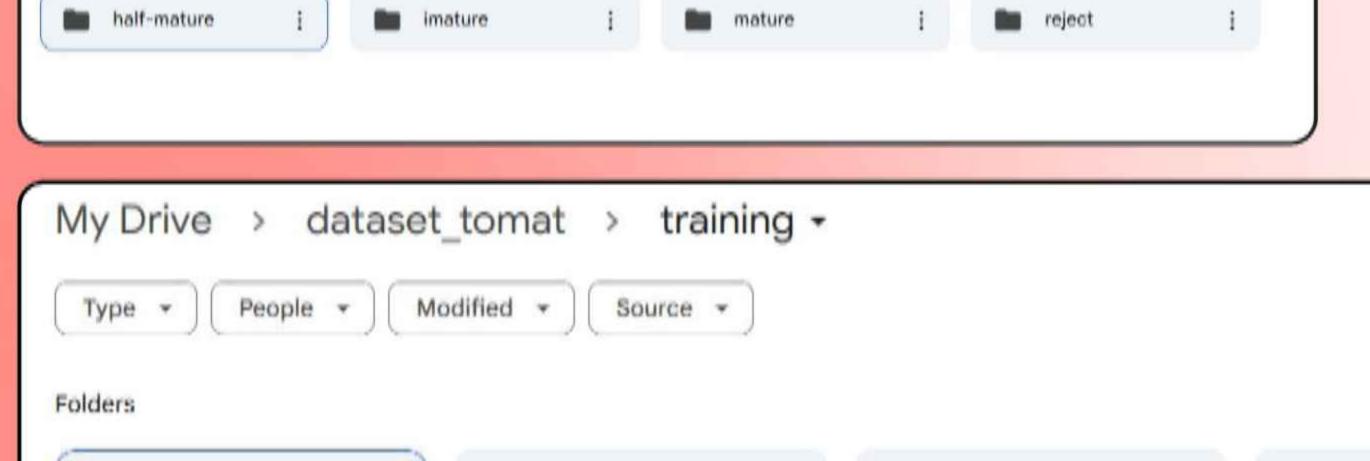
dataset_tomat >

Modified -

My Drive

Folders

DATASET



imature

berupa gambar dimana didapatkan dari kaggle Link Dataset:

DATASET

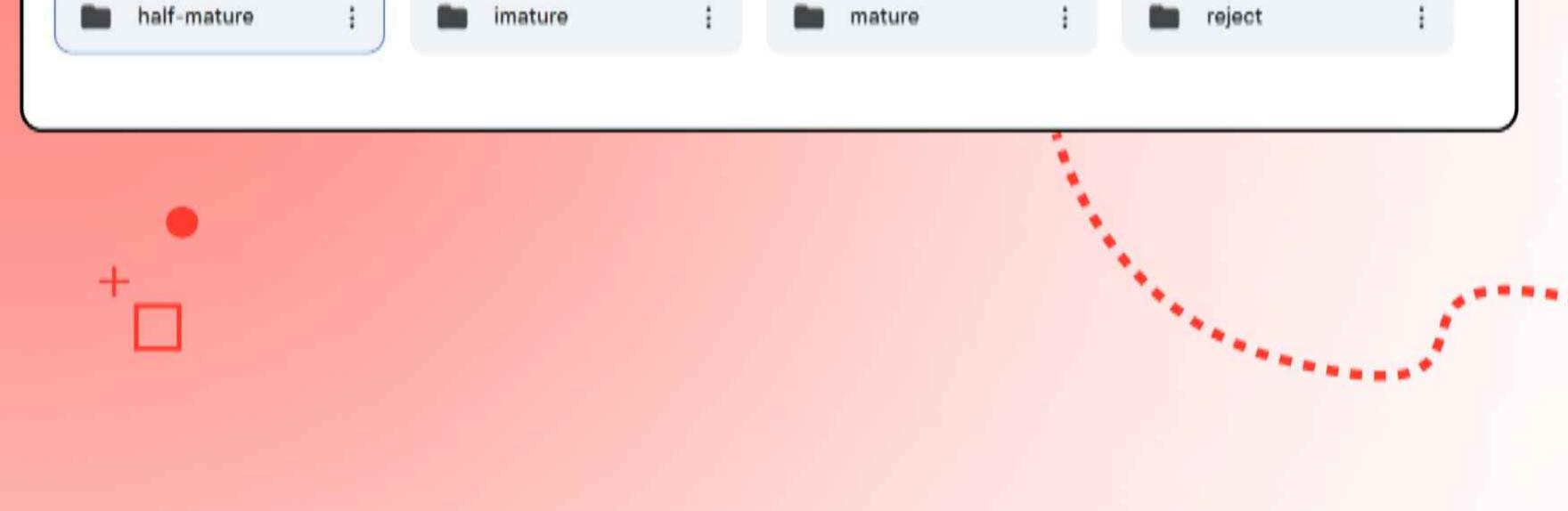
https://llnk.dev/datasettomat

Training loss

validation loss

Dataset yang digunakan merupakan dataset

0.5



mature

Tahapan:

Google Colab

Link Google Colab:

2. Pra-pemrosesan Data 3. Pembagian Dataset

5. Pelatihan Model di Google Colab

7. Pengujian Model

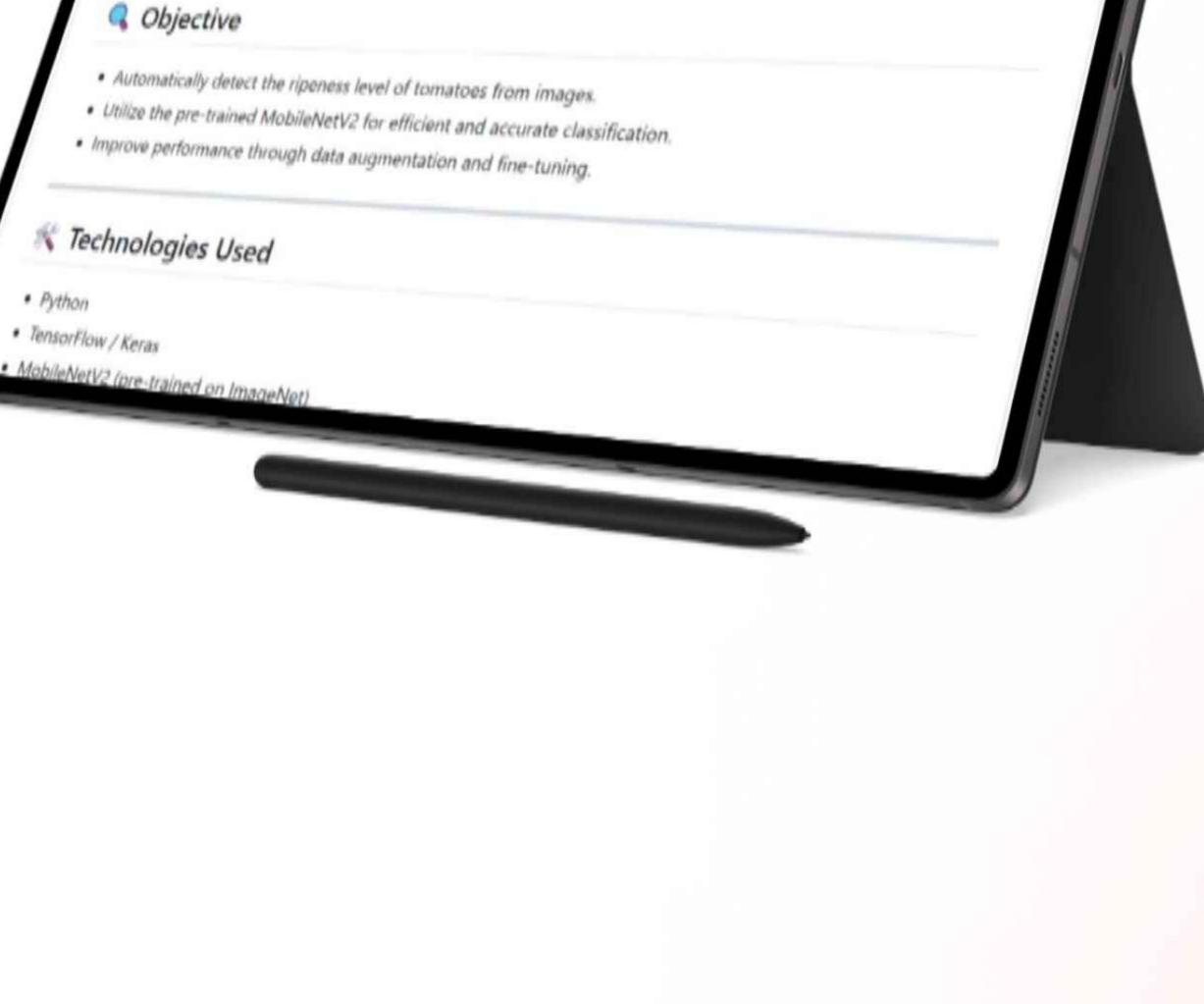
- 6. Evaluasi Model
- 8. Deploy Model (Opsional)

0.90 https://bit.ly/ColabIdentifikasiTomat Validation accuracy 1. Pengumpulan Data Gambar 4. Perancangan Model CNN (MobileNetV2)

Tomato Ripeness Classification using Transfer Learning (MobileNetV2)

as mobile and embedded systems.

Github



This project aims to classify the ripeness level of tomatoes (unripe, semi-ripe, ripe and reject) from images using

Transfer Learning based on the MobileNetV2 architecture. The model is optimized for lightweight environments such

sarana berbagi hasil penelitian/model. Repository berisi:

Github

Link Github:

1. Kode program Python untuk pelatihan model di Google Colab

https://bit.ly GithubldentifikasiTomat

2. Arsitektur MobileNetV2 dan proses training 3. File model hasil pelatihan (.tflite) untuk kebutuhan deployment 4. Kodingan aplikasi Android (Kotlin) yang sudah

GitHub sebagai bentuk dokumentasi terbuka dan

- terintegrasi dengan model 5. Dokumentasi dan penjelasan alur sistem