<LUMOSHIVE>
MAKING GOOD DEVELOPERS GREAT
PT. LUMOS INISIATIF INDONESIA

CHAPTER 7 07

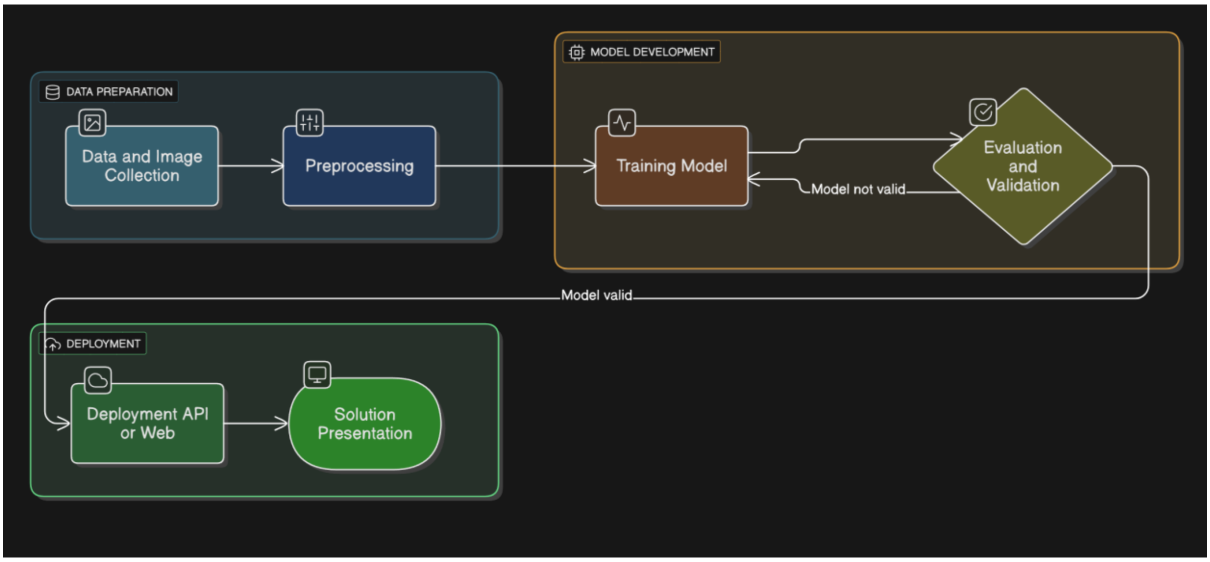
Studi Kasus Proyek Computer Vision

Pendahuluan

Computer Vision (CV) adalah bidang ilmu komputer yang mempelajari bagaimana komputer bisa “melihat” dan menginterpretasikan gambar atau video layaknya manusia. Tujuannya adalah mengambil informasi dari gambar untuk keperluan otomatisasi, klasifikasi, deteksi, segmentasi, dan lain sebagainya.

Contoh penerapan:
- Deteksi wajah di kamera smartphone
- Klasifikasi penyakit tanaman dari foto daun
- Sistem pengenalan plat nomor kendaraan
- Analisis citra medis (misal, deteksi tumor)

A. Tahapan Proyek Computer Vision End-to-End
Studi kasus CV biasanya melibatkan alur berikut:
1. Pengolahan Citra (Image Preprocessing)
2. Training Model (Pelatihan Model)
3. Evaluasi Model
4. Deployment (Penerapan Model)
5. Presentasi Hasil dan Solusi



1. Pengolahan Citra (Image Preprocessing)
Tujuan: Membersihkan, menstandarkan, dan mempersiapkan data citra agar siap digunakan dalam pelatihan model.

Langkah-langkah umum:
- Resize: Menyamakan ukuran gambar (misal, 224x224 pixel).

- Augmentasi: Menambah variasi data (rotasi, flip, zoom, dst).
- Normalisasi: Mengubah rentang pixel menjadi standar (0-1).

Contoh kode:

```
from PIL import Image
import numpy as np
img = Image.open('sample.jpg')
img = img.resize((224, 224))
img_array = np.array(img) / 255.0  # normalisasi
```

2. Training Model
Tujuan: Melatih model agar bisa mengenali pola dalam gambar.
Model populer: Convolutional Neural Network (CNN)
Transfer Learning: Menggunakan model pra-latih (misal, ResNet, MobileNet, dll).

Contoh kode (Transfer Learning dengan Huggingface dan TensorFlow):

```
from transformers import ViTImageProcessor, TFViTForImageClassification
from PIL import Image
import requests
import tensorflow as tf

model = TFViTForImageClassification.from_pretrained('google/vit-base-patch16-224')
processor = ViTImageProcessor.from_pretrained('google/vit-base-patch16-224')

url = 'https://huggingface.co/datasets/huggingface/documentation-images/resolve/main/coco_sample.png'
image = Image.open(requests.get(url, stream=True).raw)
inputs = processor(images=image, return_tensors="tf")
```

```
outputs = model(**inputs)
logits = outputs.logits
predicted_class_idx = int(tf.math.argmax(logits, axis=-1)[0])
print("Prediksi kelas:", model.config.id2label[predicted_class_idx])
```

Link Colab siap pakai: https://colab.research.google.com/github/huggingface/notebooks/blob/main/examples/image_classification.ipynb

3. Evaluasi Model
Tujuan: Mengukur performa model pada data yang belum pernah dilihat.
Metode evaluasi umum: Akurasi, Precision, Recall, F1 Score, Confusion Matrix.

```
from sklearn.metrics import accuracy_score

# y_true dan y_pred adalah label asli dan prediksi
accuracy = accuracy_score(y_true, y_pred)
print("Akurasi model:", accuracy)
```

4. Deployment (Penerapan Model)
Tujuan: Menyajikan model agar dapat digunakan pengguna lain, biasanya melalui API, aplikasi web, atau integrasi ke perangkat.
Contoh tools deployment: Flask, FastAPI, Streamlit, Gradio.

```
import gradio as gr
from transformers import ViTImageProcessor, TFViTForImageClassification
from PIL import Image

model = TFViTForImageClassification.from_pretrained('google/vit-base-patch16-224')
```

```
processor = ViTImageProcessor.from_pretrained('google/vit-
base-patch16-224')


def predict(image):
    inputs = processor(images=image, return_tensors="tf")
    outputs = model(**inputs)
    logits = outputs.logits
    predicted_class_idx = int(tf.math.argmax(logits,
axis=-1)[0])
    return model.config.id2label[predicted_class_idx]


gr.Interface(fn=predict, inputs="image",
outputs="label").launch()
```

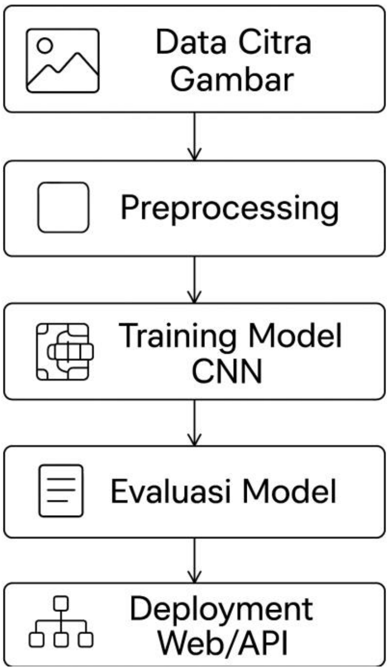
5. Presentasi Solusi Berbasis Computer Vision
Kiat presentasi solusi:
- Jelaskan masalah yang diselesaikan.
- Tunjukkan hasil demo (misal, web demo via Gradio/Streamlit).
- Visualisasikan data dan prediksi (pakai confusion matrix, grafik, dsb).
- Sajikan keunggulan dan batasan model.

Contoh slide: Latar belakang, contoh data & training, hasil evaluasi & demo, rencana deployment.

B. Contoh Studi Kasus Sederhana dari Huggingface
- Kasus: Klasifikasi gambar kucing vs anjing
Dataset: Huggingface cats_vs_dogs
(https://huggingface.co/datasets/cats_vs_dogs)

- Tutorial Jupyter/Colab siap pakai:
https://colab.research.google.com/github/huggingface/notebooks/blob/main/examples/image_classification_tensorflow.ipynb

5. Ilustrasi Proses dengan Diagram



C. Referensi & Link Tambahan

- Huggingface Computer Vision Examples:
https://huggingface.co/docs/transformers/main/en/tasks/image_classification
- Gradio Quickstart for Computer Vision:
https://gradio.app/get_started
- Google Colab: Image Classification with Transformers:
https://colab.research.google.com/github/huggingface/notebooks/blob/main/examples/image_classification.ipynb
- Huggingface Datasets for CV:
https://huggingface.co/datasets?task_categories=task_categories:image-classification