

Lampiran

Lampiran 01 Surat Pernyataan Validasi

SURAT PERNYATAAN VALIDASI

Yang bertandatangan di bawah ini:

Nama : IPUTU YASA ARBAWA
Instansi : BALAI TANAMAN NASIONAL BALI BARAT
Jabatan : PENGELOLA EKOSISTEM HUTAN (PEH)

Setelah dilakukan analisis yang mendalam dan revisi seperlunya terkait data yang digunakan dalam penelitian skripsi yang berjudul "KLASIFIKASI PENYAKIT DAUN TANAMAN BAKAU MENGGUNAKAN CONVOLUTIONAL NEURAL NETWORK (CNN)" oleh peneliti:

Nama : I Kadek Prasta Yudhantara
Nomor Induk Mahasiswa : 2115101004
Program Studi : Ilmu Komputer
Perguruan Tinggi : Universitas Pendidikan Ganesha

Maka saya selaku validator yang ditunjuk, dengan ini menyatakan bahwa data tersebut valid dan layak digunakan dalam penelitian.

Demikian pernyataan ini dibuat agar digunakan sebagaimana mestinya.

JEMBERA, 2 - MEI - 2025
Validator

(IPUTU YASA ARBAWA)
NIP. 196712311989031023

Lampiran 02 Dokumentasi validasi data



Lampiran 03 Proses Pengambilan Dataset



Lampiran 04 Potongan Code MobilnetV2 dan VGG16

```
Untitled0.ipynb
File Edit Lihat Sisipkan Runtime Fitur Bantuan

Q Perintah + Kode + Teks ▶ Jalankan semua ▼

[ ] ▶
# ===== BUILD MODEL MOBILENETV2 =====
def build_mobilenetv2(input_shape=(224,224,3), num_classes=4, dropout_rate=0.5):
    base_model = MobileNetV2(weights="imagenet", include_top=False, input_shape=input_shape)
    base_model.trainable = False # freeze base layer

    model = Sequential([
        base_model,
        GlobalAveragePooling2D(),
        Dense(256, activation='relu'),
        Dropout(dropout_rate),
        Dense(num_classes, activation='softmax')
    ])
    return model
```

```
Untitled0.ipynb
File Edit View Insert Runtime Tools Help

Q Commands + Code + Text ▶ Run all ▼

[ ] ▶
# ===== BUILD VGG16 (16 LAYER) =====
def build_vgg16(input_shape=(224,224,3), num_classes=4, dropout_rate=0.5):
    model = Sequential()

    # Block 1
    model.add(Conv2D(64, (3,3), activation='relu', padding='same', input_shape=input_shape))
    model.add(Conv2D(64, (3,3), activation='relu', padding='same'))
    model.add(MaxPooling2D((2,2)))

    # Block 2
    model.add(Conv2D(128, (3,3), activation='relu', padding='same'))
    model.add(Conv2D(128, (3,3), activation='relu', padding='same'))
    model.add(MaxPooling2D((2,2)))

    # Block 3
    model.add(Conv2D(256, (3,3), activation='relu', padding='same'))
    model.add(Conv2D(256, (3,3), activation='relu', padding='same'))
    model.add(Conv2D(256, (3,3), activation='relu', padding='same'))
    model.add(MaxPooling2D((2,2)))

    # Block 4
```