

LAMPIRAN

Lampiran : Surat Keputusan Rektor IIB Darmajaya
 Nomor : SK. 0543/DM/DFIK/BAAK/X-23
 Tanggal : 16 Oktober 2023
 Perihal : Pembimbing Penulisan Skripsi Semester Ganjil TA. 2023/2024
 Program Studi Strata Satu (S1) Teknik Informatika

**Judul Skripsi Dan Dosen Pembimbing Skripsi Semester Ganjil TA. 2023/2024
 Program Studi Strata Satu (S1) Teknik Informatika**

No	NAMA	NPM	JUDUL	PEMBIMBING
1	Yoga Budiman	2011010070	Pengenalan Informasi Pakaian dan Alat Tradisional Kebudayaan Pesisir Barat Menggunakan Virtual Reality (VR) Berbasis Android (Studi Kasus : Lamban Balak)	R.Z. Abdul Aziz, M.T., Ph.D
2	Muhammad Alvian	2011010071	Implementasi Pembelajaran Bacaan Tajwid dan Iqro Teknologi Augmented Reality	
3	Latifah Nabilah	2011010045	Diagnosis Penyakit Daun Mawar Menggunakan Pendekatan Deep Learning	Nisar, M.T.
4	Indah Amelia	2011010023	Diagnosa Penyakit Daun Tomat Menggunakan Metode VGG-19	
5	Putri Diah Agustina	2011010065	Implementasi Sistem Rekomendasi Pemilihan Mata Kuliah Peminatan Mahasiswa Baru Menggunakan Algoritma Support Vector Machine Berbasis Website (Studi Kasus : Teknik Informatika IIB Darmajaya)	Suhendro Yusuf Irianto, M.Kom., Ph.D
6	Deani Intan Sari	2011010011	Analisis Sentimen Terhadap Penggunaan Aplikasi EdLink Menggunakan Algoritma Support Vector Machine	
7	Dila Mutiara	2011010005	Deteksi Cyberbullying Pada Komentar Youtube Dengan Metode Naive Bayes	
8	Welda Syahfira	2011010096	Penerapan Metode Knowledge-Based Recommendation dalam Menentukan Perekrutan Mahasiswa Magang Berdasarkan Kesesuaian Jurusan dan Kebutuhan Bisnis UMKM	DR. Chairani, S.Kom., M.Eng
9	Irfan Adigunanto	1911010079	Penerapan Sistem File Encryption pada Sistem Operasi Linux untuk Menjaga Kerahasiaan Data Pengguna	
10	Adella Marsha Safira	2011010016	Sistem Pemilihan Sampah Dengan Scale Invariant Feature Transform (SIFT)	
11	Bhagus Adi Pratama	2011010134	Pengembangan Aplikasi Deteksi Ekspresi Wajah Secara Real Time Menggunakan Convolution Neural Network	
12	Irham Febriansyah	2011010121	Implementasi Algoritma K-Means Untuk Klastering Destinasi Wisata Di Kabupaten Pesawaran Berbasis Web (Studi Kasus: Dinas Pariwisata Kabupaten Pesawaran)	DR. Sri Lestari, S.Kom., M.Cs
13	Muhammad Fattan Abdul Aziz	1911010110	Aplikasi Pengenalan Kebudayaan Lampung Menggunakan Metode Augmented Reality Berbasis Android	
14	Hawali Nur Aqil	2011010026	Sistem Informasi dan Monitoring Industry Kreatif (SIMOKRAF) pada Kabupaten Pesawaran	
15	Muhammad Fattan Abdul Aziz	1911010110	Aplikasi Pengenalan Kebudayaan Lampung Menggunakan Metode Augmented Reality Berbasis Android	DR. Yuni Arkhiansyah, M.Kom.
16	Chandry Vivia Khoirunnisa	2011010105	Sistem Rekomendasi Program Studi Menggunakan Machine Learning Untuk Siswa Sekolah Menengah Atas (Studi Kasus: Institut Informatika dan Bisnis Darmajaya)	
17	Riko Dwi Saputra	2011010079	Pengembangan Permainan Edukasi Interaktif Untuk Anak Pada Mata Pelajaran Matematika Berbasis Mobile Menggunakan Metode ADDIE	
18	Audina Tazkia	2011010090	Implementasi Metode Naive Bayes Classifier Dalam Menganalisis Sentimen Pelanggan Terhadap Mie Gacoan Pada Instagram	

Submissions | Bulletin of Computer Science and Electrical Engineering

bcsee.org/index.php/bcsee/submissions#myQueue

Bulletin of Computer Science and Electrical Engineering

Submissions

My Queue 1 Archives Help

My Assigned

Search Filters New Submission

1185	indah_amelia et al.	0/0	Review	View	⌵
Comparison of Tomato Leaf Disease Detection Using Transfer Learning Architecture with the ...					

9:19 PM 3/9/2024

```
[5] from google.colab import drive
import tensorflow as tf
drive.mount('/content/drive')

images_dataset = tf.keras.preprocessing.image_dataset_from_directory(
    '/content/drive/MyDrive/datasettomato',
    shuffle=True,
    image_size=(IMAGE_SIZE, IMAGE_SIZE),
    batch_size=BATCH_SIZE,
)
```

```
[2] from tensorflow import keras
from tensorflow.keras.layers import Dense, Flatten
from tensorflow.keras.models import Model, Sequential, load_model
from tensorflow.keras.preprocessing import image
from tensorflow.keras.preprocessing.image import ImageDataGenerator
from tensorflow.keras.applications import VGG19
import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
from glob import glob
```

```
[8] training_set=train_datagen.flow_from_directory("/content/drive/MyDrive/datasettomato/Train",
                                                target_size=(224,224),class_mode="categorical",batch_size=50,shuffle=True)
test_set=test_datagen.flow_from_directory("/content/drive/MyDrive/datasettomato/Test",
                                         target_size=(224,224),batch_size=50,class_mode="categorical",shuffle=True)
validation_set=test_datagen.flow_from_directory("/content/drive/MyDrive/datasettomato/Validation",
                                               target_size=(224,224),batch_size=50,class_mode="categorical",shuffle=True)
```

```
[12] x=Flatten()(vgg19.output)
pred_vgg19 = Dense(3, activation='softmax')(x)
# Create the model
vgg19_model = Model(inputs=vgg19.input, outputs=pred_vgg19)
```

```
[13] vgg19_model.compile(optimizer="Adam",loss="categorical_crossentropy",metrics=["accuracy"])
```

```
[ ] # Melatih model
history = vgg19_model.fit(
    train_generator,
    validation_data=test_generator,
    epochs=18
)
```

```
▶ # Plot the loss
plt.plot(history.history['loss'], label='train loss')
plt.plot(history.history['val_loss'], label='val loss')
plt.legend()
plt.savefig('LossVal_loss') # Simpan plot sebelum menampilkannya
plt.show()

# Plot the accuracy
plt.plot(history.history['accuracy'], label='train acc')
plt.plot(history.history['val_accuracy'], label='val acc')
plt.legend()
plt.savefig('AccVal_acc') # Simpan plot sebelum menampilkannya
plt.show()
```

```
[ ] import numpy as np
import matplotlib.pyplot as plt
from sklearn.metrics import accuracy_score, confusion_matrix, classification_report
from tensorflow.keras.preprocessing.image import ImageDataGenerator
from tensorflow.keras.models import load_model

# Set path to your test dataset
test_path = "/content/drive/MyDrive/datasettomat/test"

# Evaluasi model pada set data uji
predictions = vgg19_model.predict(test_generator)
predicted_labels = np.argmax(predictions, axis=1)
true_labels = test_generator.classes

# Hitung akurasi
accuracy = accuracy_score(true_labels, predicted_labels)
print(f"Accuracy: {accuracy * 100:.2f}%")

# Hitung confusion matrix
confusion_mtx = confusion_matrix(true_labels, predicted_labels)
print("Confusion Matrix:")
print(confusion_mtx)

# Tampilkan classification report
target_names = list(test_generator.class_indices.keys())
class_report = classification_report(true_labels, predicted_labels, target_names=target_names)
print("Classification Report:")
print(class_report)
```

```
▶ from google.colab import files
uploaded = files.upload()

for fn in uploaded.keys():
    # predict images
    path = '/content/' + fn
    img = image.load_img(path, target_size=(224,224))
    x = image.img_to_array(img)
    x = np.expand_dims(x, axis =0)

    preds = modelvgg19.predict(x)
    preds=np.argmax(preds, axis=1)
    if preds==0:
        preds="Bacterial_spot"
    elif preds==1:
        preds="Early_blight"
    else:
        preds="Late_blight"
    print(preds)
    plt.imshow(img)
```