



Penerapan *Transfer Learning* Untuk Klasifikasi Citra Bunga Menggunakan Model *MobileNet-V2*

Aulia Putri^{1*}, Budi Santoso², Bunga Intan³

Program Studi Informatika, Universitas Bina Insan Lubuklinggau

e-mail: *¹2102020005@mhs.univbinainsan.ac.id, ²budisantoso@univbinainsan.ac.id,
³bungaintan@univbinainsan.ac.id

Abstrak

Proses identifikasi manual jenis bunga yang memiliki kemiripan visual seperti bunga Alamanda, Mandevilla, dan Ruellia seringkali menghadapi tantangan dalam hal konsistensi dan efisiensi. Penelitian ini bertujuan untuk menganalisis performa model *MobileNetV2* dengan pendekatan *Transfer Learning* dalam mengklasifikasikan ketiga jenis bunga tersebut. Dataset yang digunakan berjumlah 5321 gambar yang terdiri dari 1968 gambar Alamanda, 1502 gambar Mandevilla, dan 1851 gambar Ruellia. Dataset dibagi menjadi data *training* (80%), *validation* (10%), dan *testing* (10%). Proses pengolahan data mencakup *pre-processing* dengan mengubah ukuran gambar menjadi 224x224 piksel, *augmentasi* data, dan *standarisasi* melalui *normalisasi* nilai piksel. Hasil penelitian menunjukkan bahwa model mencapai performa yang sangat baik dengan akurasi 100% pada data *testing* dan nilai *loss* 0.0394. Model juga menunjukkan hasil yang konsisten pada data *training* dengan akurasi 99.79% dan *loss* 0.0409. *Confusion Matrix* menunjukkan model berhasil mengklasifikasikan seluruh sampel dengan tepat tanpa ada kesalahan klasifikasi silang. Hasil ini mengindikasikan bahwa arsitektur *MobileNetV2* mampu mengenali dan membedakan karakteristik visual dari ketiga jenis bunga dengan sangat akurat, meskipun ketiganya memiliki bentuk dasar

Kata kunci : *MobileNetV2*, *Transfer Learning*, Klasifikasi Bunga, *Augmentasi* data, *Normalisasi* Nilai Piksel

Abstract

Manual identification of flower species with visual similarities, such as Alamanda, Mandevilla, and Ruellia flowers, often faces challenges in terms of consistency and efficiency. This research aims to analyze the performance of the *MobileNetV2* model with a transfer learning approach in classifying these three flower species. The dataset consists of 5,321 images, comprising 1,968 Alamanda images, 1,502 Mandevilla images, and 1,851 Ruellia images. The dataset was divided into training (80%), validation (10%), and testing (10%) data. Data processing includes pre-processing by resizing images to 224x224 pixels, data augmentation, and standardization through pixel value normalization. The results show that the model achieved excellent performance with 100% accuracy on testing data and a loss value of 0.0394. The model also demonstrated consistent results on training data with 99.79% accuracy and 0.0464 loss, as well as on validation data with 100% accuracy and 0.0409 loss. The confusion matrix shows that the model successfully classified all samples correctly without any cross-classification errors. These results indicate that the *MobileNetV2* architecture can recognize and differentiate the visual characteristics of the three flower species with high accuracy, despite their similar basic shapes.

Keywords : *MobileNetV2*, *Transfer Learning*, Flower Classification, Data Augmentation, Pixel Normalization

I. PENDAHULUAN

Bunga merupakan salah satu bagian tumbuhan yang memiliki keragamab bentuk, warnam dan karakteristik visual lainnya. Dalam proses idnetifikasinya, beberapa jenis bunga memiliki kemiripan yang signifikan sehingga sulit untuk dibedakan, terutama bagi orang yang tidak memiliki pengetahuan khusus tentang bunga. Hal ini terlihat pada kelompok bunga yang berbentuk terompet seperti Alamanda, Mandevilla, dan Ruellia yang memiliki struktur dasar serupa namun merupakan spesies yang berbeda. Proses klasifikasi manual bunga-bunga ini menjadi tantangan tersendiri karena membutuhkan waktu yang lama dan sering terjadi kesalahan dalam proses identifikasinya terutama karena perbedaan pada ukuran, warna, dan makhota bunga.

Klasifikasi adalah metode dalam pembejarian mesin yang digunakan untuk memprediksi kelas atau kategori dari suatu data berdasarakan pola atau fitur yang dimilikinya. Klasifikasi objek sangat berguna untuk mengenali jenis objek tertentu dari sekian objek yang harus di klasifikasinya (Putri etl al., 2024)

Untuk mengatasi tantangan ini, deep learning hadir sebagai Solusi yang menjanjikan. *Deep learning* adalah cabang ilmu machine learning berbasis Jaringan Syaaraf Tiruan (JST) atau bisa dikatakan sebagai perkembangan dari JST, dengan kemampuannya menginterpretasikan data visual menyerupai cara kerja otak manusia (Ilahiyah & Nilogiri, 2018). Deep learning menggunakan jaringan saraf tiruan berlapis-lapis untuk memberikan akurasi yang tinggi dalam melakukan deteksi objek, pengenalan suara, terjemahan Bahasa dan lainnya (Rosalina & Wijaya).

Dalam kasus klasifikasi bunga, pendekatan *transfer learning* memungkinkan pemanfaatan model pra-terlatih (*pre-trained model*). Teknik ini memungkinkan pembelajaran dengan data yang terbatas karena memanfaatkan pengetahuan yang sudah diperoleh model dari dataset sebelumnya. Pendekatan ini sangat berguna terutama Ketika data training yang teresdia terbatas, karena model tidak

perlu belajar dari nol melainkan dapat mengadaptasi pengetahuan yang sudah dimiliki untuk tugas baru (Fahcruroji et al., 2024)

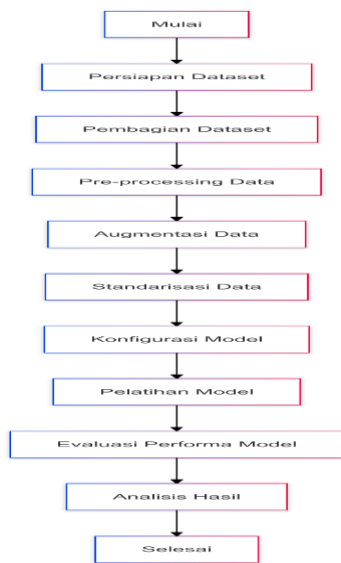
Salah satu model yang relevan dengan masalah ini adalah *MobileNetV2*. *MobileNetV2* dirancang khusus untuk perangkat dengan kapasitas komputasi terbatas (Butar- Butar & Marpaung, 2023). Model ini menggunakan teknologi *inverted residual* dan *linear bottlenecks* yang memungkinkan ekstraksi fitur visual yang efisien. Teknologi ini sangat penting dalam membedakan karakteristik halus antar spesies bunga, seperti susunan kelopak, mahkota bunga, dan detail-detail kecil lainnya yang menjadi pembeda antar jenis bunga. Model ini juga dirancang mempertimbangkan efisiensi komputasi, sehingga dapat diimplementasikan dalam berbagai perangkat (Maulana et al., 2024)

Perapan *MobileNetV2* telah dilakukan dalam beberapa penelitian klasifikasi citra dengan hasil yang menjanjikan. Rachburee menunjukkan keunggulan *MobileNetV2* dalam klasifikasi spesies teratai dengan akurasi mencapai 99.5% (Rachburee & Punlumjeak, 2022). Penelitian lain oleh Butar juga menunjukkan keefektifan *MobileNetV2* dalam mengklasifikasikan daun tanaman obat dengan akurasi 97% (Butar-Butar & Marpaung, 2023). Berdasarkan penelusuran literatur yang dilakukan, masih terbatas penelitian yang menganalisis kemampuan *MobileNetV2* dalam membedakan bunga-bunga dengan stuktur dasar yang mirip.

Berdasarkan hal tersebut, penelitian ini akan menganalisis “penerapan transfer learning untuk klasifikasi citra bunga menggunakan *MobileNetV2*”. Penelitian ini diharapkan dapat menganalisis performa model dalam mengklasifikasikan jenis-jenis bunga yang memiliki kemiripan visual dengan akurat dan efisien.

II. METODOLOGI PENELITIAN

Penelitian ini menerapkan metode penelitian kuantitatif dengan pendekatan eksperimental untuk menganalisis performa model *MobileNetV2* dalam klasifikasi citra bunga menggunakan transfer learning.



Gambar 1. Alur Penelitian

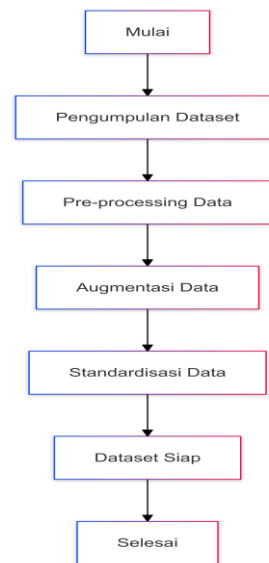
Dataset yang digunakan adalah dataset yang melalui proses perekaman video kemudian di ekstrak dan menghasilkan 5.321 gambar yang terdiri dari tiga jenis bunga yaitu Alamanda, Mandevilla, dan Ruellia. Dataset tersebut dibagi menggunakan metode *train_test_split* dengan proporsi 80% data *training* sebanyak 4254 gambar, 10% data *validation* sebanyak 533 gambar, dan 10% data *testing* sebanyak 534 gambar.

Distribusi detail setiap split:

	Train	Validation	Test
Alamanda	1725	459	296
Mandevilla	1319	355	226
Ruellia	1614	425	278

Gambar 1. Distribusi detail setiap split

Proses pengolahan data mencakup pengumpulan dataset yang terdiri dari foto-foto alamanda, mandevilla, dan ruellia. Dataset yang terkumpul kemudian melalui tahap *pre-processing* dengan mengubah ukuran gambar menjadi 224x224 piksel, *augmentasi* data, dan standarisasi melalui normalisasi nilai piksel. Dilanjutkan dengan *augmentasi* data khusus untuk data training yang meliputi rotasi, pergeseran, flip horizontal, *zoom*, dan perubahan perspektif. Kemudian dilakukan standarisasi data melalui normalisasi nilai piksel.



Gambar 2. Alur Pengolahan Data

Setelah data siap, dilakukan konfigurasi model menggunakan arsitektur *mobilenetv2* sebagai base model yang dikembangkan dengan menambahkan beberapa layer tabahan untuk meningkatkan kemampuan klasifikasi.

III. HASIL DAN PEMBAHASAN

Tujuan dari penelitian ini adalah mengetahui kinerja (performance) dari model *MobileNetV2* untuk klasifikasi citra bunga. Adapun model *MobileNetV2* yang digunakan adalah *MobileNetV2* dengan output shape yaitu (None, 7, 7, 1280) dan jumlah parameter output yaitu 2,257,984. Layer selanjutnya adalah *global average pooling2d* dengan output shape yaitu (None, 1280). Layer selanjutnya *batch_normalization* dengan output shape (None, 1280) dan jumlah parameter 5,120 dan *batch_normalization_2* dengan output shape (None, 128) dan jumlah parameter 512. Layer selanjutnya adalah *dropout* output shape (None, 256) dan *dropout_1* dengan output shape (None, 128). Layer selanjutnya adalah *dense* dengan output shape (None, 256) dan parameter 327, 936, *dense_1* dengan output shape (None, 128) dan parameter 32,896, *dense_2* dengan output shape (None, 3) dan jumlah parameter 387. Selanjutnya kita juga menggunakan *Optimizer Adam* dengan

learning rate 0.0001 dan loss function categorical crossentropy untuk multi class

Layer (type)	Output Shape	Param #
mobilenetv2_1.00_224 (Functional)	(None, 7, 7, 1280)	2,257,984
global_average_pooling2d (GlobalAveragePooling2D)	(None, 1280)	0
batch_normalization (BatchNormalization)	(None, 1280)	5,120
dense (Dense)	(None, 256)	327,936
batch_normalization_1 (BatchNormalization)	(None, 256)	1,024
dropout (Dropout)	(None, 256)	0
dense_1 (Dense)	(None, 128)	32,896
batch_normalization_2 (BatchNormalization)	(None, 128)	512
dropout_1 (Dropout)	(None, 128)	0
dense_2 (Dense)	(None, 3)	387

Total params: 2,625,859 (10.02 MB)

Trainable params: 364,547 (1.39 MB)

Non-trainable params: 2,261,312 (8.63 MB)

Gambar 3. Arsitektur MobileNetV2

Setelah kita membuat model langkah selanjutnya adalah melakukan proses *training*, dimana *training* adalah merupakan proses di mana *machine learning* bekerja sehingga algoritma yang sudah kita definisikan dapat mengingat pola dari masing-masing kelas pada data yang kita train.

```
print("\nMemulai proses training...")
history = model.fit(
    train_generator,
    validation_data=validation_generator,
    epochs=50,
    callbacks=[model_checkpoint, early_stopping, reduce_lr]
)
```

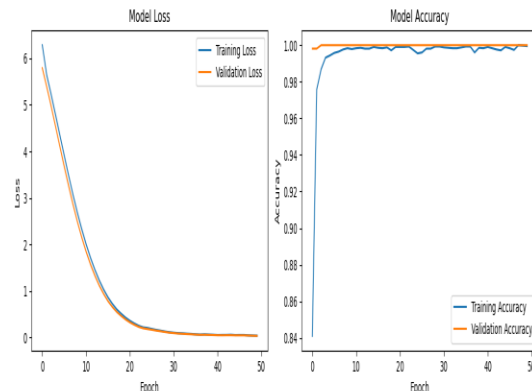
```
146/146 — 20ms 1s/step - accuracy: 0.9994 - loss: 1.4004 - val_accuracy: 1.0000 - val_loss: 1.4104 - learning_rate: 1.0000e-04
Epoch 13/50
146/146 — 222s 1s/step - accuracy: 0.9951 - loss: 1.2230 - val_accuracy: 1.0000 - val_loss: 1.0556 - learning_rate: 1.0000e-04
...
Epoch 40/50
146/146 — 204s 1s/step - accuracy: 0.9988 - loss: 0.8487 - val_accuracy: 1.0000 - val_loss: 0.8360 - learning_rate: 1.0000e-04
Epoch 50/50
146/146 — 237s 1s/step - accuracy: 0.9988 - loss: 0.8383 - val_accuracy: 1.0000 - val_loss: 0.8341 - learning_rate: 1.0000e-04
```

Dari hasil diatas bisa kita lihat bahwa akurasi pada data *training* adalah 0.9979 dan nilai lossnya adalah 0.0464. selanjutnya untuk akurasi validasi data adalah 1.0000 dan lossnya adalah 0.0409. Selanjutnya melakukan *testing* menggunakan data test. Di mana data test ini tidak diikutsertakan ke dalam proses training. Hasilnya adalah sebagai berikut.

=== Model Evaluation ===

```
Validation and Test Scores:
39/39 — 28s 718ms/step - accuracy: 1.0000 - loss: 0.0341
Validation loss: 0.03413493558764458
Validation accuracy: 1.0
25/25 — 21s 856ms/step - accuracy: 1.0000 - loss: 0.0341
Test loss: 0.03412279859185219
Test accuracy: 1.0
```

Selanjutnya kita tampilkan grafik loss untuk melihat konvergensi model dan grafik accuracy untuk melihat performa model

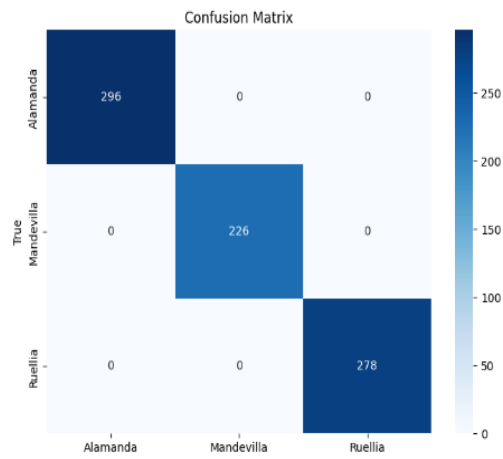


Gambar 4. Grafik History Training Model

Dari grafik diatas tidak ada tanda *overfitting* karena kurva *training* dan *validation* berjalan beriringan, loss terus menurun dan stabil di akhir training.

Selain akurasi, kita juga akan menghitung *F1-Score*, *precision* dan *recall* serta *confusion matrix* pada masing-masing kelas. *Confusion matrix* adalah table yang menyatakan klasifikasi jumlah data uji yang benar dan jumlah data uji yang salah. Hasil *confusion matrix* digunakan untuk menghitung *performance metrics* untuk mengukur kinerja model yang telah dibuat (Tri Susilo et al., 2024). Dimana *True Positive* (TP) adalah jumlah data positif yang diklasifikasikan dengan benar oleh sistem, *True Negative* (TN) adalah jumlah data *negative* yang diklasifikasikan dengan benar oleh sistem, *False Positive* (FP) adalah jumlah data positif yang diklasifikasikan dengan salah oleh sistem dan *False Negative* (FN) adalah jumlah data *negative* yang diklasifikasikan dengan salah oleh sistem. Hasil prediksi dan *confusion matrix* dan *Classification report* dari arsitektur *mobilenetv2* dapat dilihat pada Gambar berikut ini.

Confusion Matrix:



Gambar 5. Confusion matrix

Classification Report:

	precision	recall	f1-score	support
Alamanda	1.00	1.00	1.00	296
Mandevilla	1.00	1.00	1.00	226
Ruellia	1.00	1.00	1.00	278
accuracy			1.00	800
macro avg	1.00	1.00	1.00	800
weighted avg	1.00	1.00	1.00	800

Gambar 6. Classification report

Dari hasil *classification report*, performa model klasifikasi untuk 3 kelas Alamanda, Mandevilla, dan Ruellia semua kelas memiliki *precision* 1.00 (100%) yang artinya Ketika model memprediksi suatu sampel sebagai kelas tertentu prediksi tersebut selalu benar dan tidak ada false positive. *Recall*, semua kelas memiliki recall 1.00 (100%) yang artinya model berhasil mengidentifikasi semua sampel dari masing-masing kelas dengan benar dan tidak ada *false negative*. *F1-Score*, nilai 1.00 untuk semua kelas, *F1-score* adalah rata-rata harmonis dari *precision* dan *recall* dimana nilai yang sempurna ini menunjukkan keseimbangan yang baik antara *precision* dan *recall*.

Dari hasil *confusion matrix* dapat dilihat bahwa model berhasil mengklasifikasikan dengan sempurna dimana Alamanda 197 sampel diprediksi dengan benar, Mandevilla 151 sampel diprediksi dengan benar, dan Ruellia 186 sampel diprediksi dengan benar.

Setelah menentukan model arsitektur *MobileNetV2* kemudian dilanjutkan menghitung akurasi training, akurasi

validasi, akurasi *testing*, *F1-score*, *precision* dan *recall*, serta *confusion matrix*. Langkah selanjutnya Kita akan melihat visualisasi hasil prediksi nya melalui gambar berikut



Gambar 7. Hasil Klasifikasi 3 Kelas

Pada gambar diatas digunakan untuk memvisualisasikan hasil prediksi dengan menampilkan sampel gambar dari setiap kelas, label sebenarnya (True), hasil prediksi (Pred), dan Tingkat kepercayaan (Conf). Selain itu

IV. KESIMPULAN

Berdasarkan hasil penelitian dan pengujian yang sudah dilakukan pada penelitian ini maka dapat ditarik Kesimpulan sebagai berikut:

1. Model *mobilenetv2* yang dikembangkan berhasil melakukan klasifikasi dengan sangat baik, mencapai akurasi 100% pada data *testing* dengan nilai *loss* yang sangat rendah yaitu 0.0348. Perlu dicatat bahwa tingginya akurasi ini bisa dipengaruhi oleh ukuran dataset yang relative terbatas (5321 gambar)

2. Proses pengolahan dataset meliputi pembagain data (80% *training*, 10% validasi, 10% testing), pre-processing, *augmentasi*, dan standarisasi dan terbukti efektif dalam menghasilkan model yang akurat
3. *Confusion matrix* menunjukkan bahwa model berhasil mengklasifikasikan seluruh sampel dengan tepat, 197 sampel alamanda, 151 sampel mandevilla, dan 186 sampel ruellia tanpa ada kesalahan klasifikasi silang.
4. Performa model yang sangat baik ini menunjukkan bahwa arsitektur *mobilenetv2* mampu mengenali dan membedakan karakteristik visual dari ketiga jenis bunga dengan sangat akurat, meskipun ketiganya memiliki kemiripan bentuk dasar yaitu seperti terompet.

V. SARAN

Untuk pengembangan penelitian selanjutnya, beberapa saran yang dapat diberikan adalah sebagai berikut:

1. Memperluas dataset dengan menambahkan lebih banyak variasi gambar dalam berbagai kondisi pencahayaan, sudut pengambilan untuk meningkatkan robustness model. Hal ini penting mengingat dataset saat ini masih relative kecil, yang mungkin membatasi kemampuan generalisasi model pada situasi nyata yang lebih beragam
2. Melakukan pengujian model pada spesies bunga lain yang memiliki karakteristik visual serupa untuk menguji kemampuan generalisasi model
3. Membandingkan performa model dengan arsitektur deep learning lainnya untuk menentukan arsitektur mana yang paling optimal untuk kasus klasifikasi bunga

VI. DAFTAR PUSTAKA

Andika Maulana, S., Husna Batubara, S., Permata Putri Pasaribu, Y., Syahputra, H., & Ramadhani, F. (2024). Deteksi Burung Menggunakan Convolutional Neural Network (Cnn) Dengan Model Arsitektur Mobilenetv2. *JATI (Jurnal Mahasiswa Teknik Informatika)*, 8(4), 6108–6114.

<https://doi.org/10.36040/jati.v8i4.10126>

Chan, J. (2017). *Learn C# in One Day and Learn It Well Python for Beginners with Hands-on Project The only book you need to start coding in Python immediately (Second Edition)*. <http://www.learncodingfast.com/python>

Dwi Lestari, M., & Erika, L. (2021). Science Midwifery Purple Ruellia Flower (Ruellia Simplex Wright) Ethanol Extract Lotion as Skin Moisturizer. *Science Midwifery*, 10(1), 132–139.

www.midwifery.iocspublisher.org

Fahcuroji, A. R., Madona Yunita Wijaya, & Irma Fauziah. (2024). Implementasi Algoritma Cnn Mobilenet Untuk Klasifikasi Gambar Sampah Di Bank Sampah. *PROSISKO: Jurnal Pengembangan Riset Dan Observasi Sistem Komputer*, 11(1), 45–51. <https://doi.org/10.30656/prosisko.v11i1.8101>

Hendri Butar-Butar, R. J., & Marpaung, N. L. (2023). Deep Learning untuk Identifikasi Daun Tanaman Obat Menggunakan Transfer Learning MobileNetV2. *Jurnal Informatika: Jurnal Pengembangan IT*, 8(2), 142–148.

<https://doi.org/10.30591/jpit.v8i2.5217>

Ilahiyah, S., & Nilogiri, A. (2018). Implementasi Deep Learning Pada Identifikasi Jenis Tumbuhan Berdasarkan Citra Daun Menggunakan Convolutional Neural Network _

Ilahiyah _ JUSTINDO (Jurnal Sistem dan Teknologi Informasi Indonesia). *JUSTINDO(Jurnal Sistem & Teknologi Informasi Indonesia)*, 3(2), 49–56.

Moolayil, J. (2018). Learn Keras for Deep Neural Networks: A Fast-Track Approach to Modern Deep Learning with Python. In *Learn Keras for Deep Neural Networks: A Fast-Track Approach to Modern Deep Learning with Python*.
<https://doi.org/10.1007/978-1-4842-4240-7>

Putri, R. A., Suryadi Satyawan, A., Prihantono, J. A., Linggi, R. S., Paramita, I. G. A. P. S., Iswarawati, N. K. E., Akbar, F., & Utomo, P. A. (2024). Model Deep Learning Untuk Klasifikasi Objek Pada Gambar Fisheye. *Jurnal Teknologi Informasi Dan Ilmu Komputer*, 11(3), 519–528.
<https://doi.org/10.25126/jtiik.938047>

Rachburee, N., & Punlumjeak, W. (2022). Lotus species classification using transfer learning based on VGG16, ResNet152V2, and MobileNetV2. *IAES International Journal of Artificial Intelligence*, 11(4), 1344–1352.
<https://doi.org/10.11591/ijai.v11.i4.pp1344-1352>

Rosalina, R., & Wijaya, A. (2020). Pendeteksian Penyakit pada Daun Cabai dengan Menggunakan Metode Deep Learning. *Jurnal Teknik Informatika Dan Sistem Informasi*, 6(3), 452–461.
<https://doi.org/10.28932/jutisi.v6i3.2857>

A. A. Tri Susilo, H. O. L. Wijaya, and E. Elmayati, “Implementasi Algoritma Naïve Bayes Pada Klasifikasi Penentuan Jenis Kartu Kredit,” *JUTIM (Jurnal Tek. Inform. Musirawas)*, vol. 8, no. 2, pp. 76–85, 2024, doi: 10.32767/jutim.v1i1.2161.