

Optimasi Algoritma SVM dengan Ekstraksi Fitur Warna pada Klasifikasi Biji Kopi Sangrai

Gde Krishna Sankya Yogeswara¹, I Ketut Gede Suhartana², I Gusti Ngurah Anom Cahyadi Putra³, I Wayan Supriana⁴

^aProgram Studi Informatika, Fakultas Matematika dan Ilmu Pengetahuan Alam
Jalan Raya Kampus Unud, Jimbaran, Bali, 80361, Indonesia

¹krishnasankya@gmail.com

²ikq.suhartana@unud.ac.id

³anom.cahyadi@unud.ac.id

⁴wayan.supriana@unud.ac.id

Abstract

Manual classification of roasted coffee beans is often hampered by subjectivity and inconsistency. This study aims to compare the performance of the Support Vector Machine (SVM) algorithm in classifying the ripeness of roasted coffee beans using two different feature approaches: raw pixel features and extracted hybrid features. The research focuses on finding the best hyperparameter combination for each approach and determining which method yields superior classification performance. The dataset used consists of 900 augmented coffee bean images, evenly distributed across three classes (Dark, Light, Medium). In the first approach, an SVM model was trained directly on flattened raw pixel data. In the second approach, an SVM model was trained using combined features extracted via a Convolutional Neural Network (CNN) and Color Histogram. The experimental results show a significant performance difference. The SVM model using raw pixel features achieved a maximum accuracy of 88.33% with the best parameters {kernel: 'rbf', C: 10, gamma: 0.01}. Meanwhile, the hybrid model utilizing feature extraction from the deeper_wider CNN architecture and color histograms drastically improved performance, reaching an accuracy of 98.33% with parameters {kernel: 'rbf', C: 10, gamma: 0.1}. These results demonstrate that employing high-level feature extraction through CNN is significantly superior to using raw pixels for the task of roasted coffee bean classification.

Keywords: Support Vector Machine, Image Classification, Roasted Coffee Beans, Feature Extraction, Convolutional Neural Network, Color Histogram.

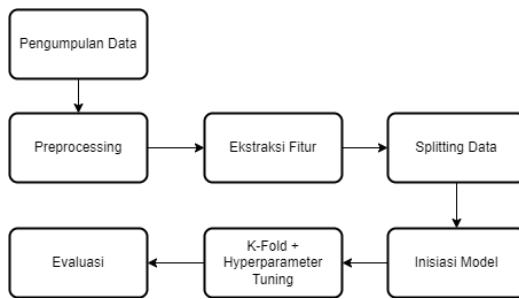
1. Pendahuluan

Biji kopi sangrai merupakan komoditas dengan nilai ekonomi tinggi yang kualitasnya sangat ditentukan oleh presisi proses penyangraian. Penilaian tingkat kematangan sangrai (roast level) secara konvensional mengandalkan persepsi visual manusia terhadap warna biji, sebuah metode yang rentan terhadap subjektivitas dan inkonsistensi [1]. Kesalahan penilaian dapat merugikan penyangrai dan konsumen, sehingga diperlukan solusi yang lebih objektif dan efisien.

Teknologi pengolahan citra digital menawarkan solusi menjanjikan untuk otomatisasi proses ini. Beberapa penelitian sebelumnya telah menerapkan arsitektur machine learning untuk penilaian kualitas biji kopi, salah satunya menggunakan Support Vector Machine (SVM) [6]. SVM adalah algoritma klasifikasi yang efektif dalam memisahkan data dengan mencari hyperplane optimal [2]. Namun, kinerja SVM sangat bergantung pada kualitas fitur yang diekstraksi. Di sisi lain, Convolutional Neural Network (CNN) unggul dalam mengekstraksi fitur-fitur hierarkis yang kompleks (warna, bentuk, tekstur) secara otomatis dari citra mentah [3]. Beberapa studi telah menunjukkan keberhasilan model hibrida CNN-SVM dalam berbagai domain, yang mengindikasikan potensi peningkatan kinerja dengan menggabungkan kekuatan keduanya [4].

Penelitian ini bertujuan untuk menganalisis dan membandingkan secara komprehensif kinerja dua pendekatan: (1) model SVM yang dilatih langsung menggunakan data piksel mentah, dan (2) model SVM hibrida yang menggunakan gabungan fitur hasil ekstraksi dari CNN dan Histogram Warna. Tujuannya adalah untuk menentukan metode mana yang paling unggul dan menemukan kombinasi hyperparameter terbaik untuk klasifikasi tingkat kematangan biji kopi sangrai (Dark, Light, Medium).

2. Metode Penelitian



Gambar 1. Alur Metode Penelitian

Sesuai pada Gambar 1, dalam rangka memperoleh keluaran dari penelitian ini, diterapkan sejumlah tahapan metodologis yang mencakup proses pengumpulan data, analisis, hingga tahap akhir berupa perancangan sistem sebagai bentuk demonstrasi implementasi metode yang telah dikembangkan.

2.1 Pengumpulan Data



Gambar 2. Citra Biji Kopi

Data yang digunakan adalah dataset primer yang terdiri dari 900 citra biji kopi sangrai. Dataset ini terbagi rata ke dalam tiga kelas kematangan: *Dark* (300 citra), *Light* (300 citra), dan *Medium* (300 citra). Contoh dataset seperti pada Gambar 2, pengambilan gambar dilakukan menggunakan kamera *smartphone* dengan latar belakang yang seragam untuk menjaga konsistensi.

2.2 Preprocessing

Preprocessing merupakan tahap krusial yang dapat dilakukan untuk menyelaraskan serta meningkatkan kualitas data. Beberapa tahap preprocessing dalam penelitian ini dilakukan semaksimal mungkin untuk memastikan bahwa data yang diinput terfokus secara khusus pada obyek penelitian, yaitu Biji Kopi. Berikut adalah langkah-langkah *preprocessing* yang akan diterapkan.

a. Cropping



Gambar 3. Cropping Citra Biji Kopi

Cropping adalah langkah pertama dari proses *preprocessing* yang agar model dapat mempelajari fitur gambar tanpa adanya gangguan terkait obyek lainnya. Berdasarkan contoh hasil *cropping* pada Gambar 3, hal ini akan mengurangi keberagaman objek dalam data yang diinputkan ke dalam model yang nantinya akan meningkatkan performa model dalam mengenali pola-pola penting pada citra biji kopi.

b. *Removing Background*



Gambar 4. *Removing Background* Citra Biji Kopi

Mengacu pada Gambar 4, langkah ini berfungsi sebagai lapisan pemrosesan lanjutan yang bertujuan untuk memastikan bahwa fokus utama citra input benar-benar tertuju pada objek biji kopi, tanpa adanya gangguan visual, termasuk latar belakang. Dengan demikian, model yang dikembangkan akan memiliki tingkat ketahanan (*robustness*) yang tinggi dan tetap dapat berfungsi secara optimal meskipun gambar diambil dalam kondisi atau lokasi yang berbeda-beda.

c. *Resizing*

Model memerlukan input citra dengan dimensi yang seragam agar proses pemrosesan dapat dilakukan secara konsisten dan menghasilkan output yang optimal. Oleh karena itu, pada penelitian ini dilakukan proses *resizing* citra menjadi ukuran 224x224 piksel.

2.3 Ekstraksi Fitur

Untuk model hibrida, dua jenis fitur diekstraksi dari setiap citra yang telah melalui *preprocessing*:

1. **Fitur CNN:** sebuah arsitektur CNN dirancang untuk mengekstraksi fitur visual tingkat tinggi seperti tekstur dan bentuk. Kemampuan jaringan syaraf tiruan konvolusional dalam mempelajari representasi data citra secara mendalam telah terbukti sangat efektif, seperti yang ditunjukkan pada penelitian fundamental oleh Krizhevsky, dkk. [5]. Fitur ini diambil dari output lapisan sebelum fully connected layer. Beberapa arsitektur dengan kedalaman dan lebar yang berbeda (simple, original, deeper_wider) diuji untuk menemukan ekstraktor fitur terbaik.
2. **Fitur Histogram Warna:** Histogram warna (RGB) dihitung untuk merepresentasikan distribusi warna global dari citra. Vektor histogram ini kemudian diratakan (*flattened*) untuk menjadi fitur numerik.

Kedua set fitur ini kemudian digabungkan menjadi satu vektor fitur tunggal untuk setiap citra yang akan menjadi input bagi *classifier* SVM.

2.4 Splitting Data

Keseluruhan dataset dibagi menjadi 2 bagian diantaranya data train dan data testing. Pembagian jumlah *splitting* data menggunakan proporsi yang umum yaitu 80% data train dan 20% data testing.

2.5 Iniasiasi Model

Penelitian ini mengevaluasi dan membandingkan performa di antara tiga jenis model yang diikutsertakan. Model-model tersebut diantaranya :

- a. SVM

Tabel 1. Algoritma SVM

Algoritma 1. Support Vector Machine

```
1: Input: X
2: Output: Y (class label)
3: Initialize kernel, C, gamma
4: Declare K
5: Process
6: If kernel == "linear" then
7:     K = x.xT
8: If kernel == "rbf" then
9:     K = K = (1 + x.xT)Q
10: If kernel == "poly" then
11:     K = e-y||x-xT||
12:
13: w → minw(1/2 * ||w||2) such that yn (wTxn + b) ≥ 1
14: w = α → dual optimization problem
15: a → maxa(a - 1/2 * a (yyT * K) a) (quadratic programming)
16: a → mina(1/2 * a (yyT * K) a - a), subject to yTa = 0, 0 ≤ a ≤ C (equality in dual
problem)
17: partial grad → minimize a
18: End
19: Output
20: Y = sign(sum(a * yisv * K) + b)
```

Algoritma *Support Vector Machine* (SVM) pada tabel 2 diawali dengan proses menerima input berupa matriks fitur X yang telah melalui tahap ekstraksi, dengan total sembilan fitur per sampel. Pada tahap awal, proses dimulai dengan inisialisasi *hyperparameter*, yaitu jenis kernel, nilai regularisasi C, dan parameter gamma yang berperan penting dalam pengendalian kompleksitas model dan fleksibilitas fungsi kernel. Selanjutnya, dilakukan pembuatan fungsi kernel untuk memproyeksikan data ke dalam ruang berdimensi lebih tinggi. Tujuan utama dari metode SVM adalah memaksimalkan margin antara kelas-kelas, yang secara matematis dicapai dengan meminimalkan norma vektor bobot w melalui formulasi *dual optimization problem*. Untuk meningkatkan efisiensi komputasi, digunakan teknik *kernel trick* dan *quadratic programming*, sehingga tidak perlu menghitung seluruh transformasi eksplisit ke ruang berdimensi tinggi. Formulasi dual ini dikonversi ke bentuk kuadratik untuk memperoleh solusi optimal terhadap parameter α. Setelah nilai-nilai α optimal diperoleh, model SVM dapat digunakan untuk melakukan prediksi dengan menghasilkan nilai label kelas yang telah di *encode* secara numerik.

2.6 K-Fold Cross Validation dan Hyperparameter Tuning

Nilai K yang akan digunakan pada proses *K-Fold Cross Validation* yaitu 5, sehingga akan terdapat lima iterasi dimana setiap iterasi akan menggunakan sebagian dataset untuk menjadi data validation dan sebagian lagi menjadi data pelatihan. Penggunaan metode validasi ini bertujuan untuk mencegah terjadinya overfitting serta memberikan performa yang valid pada sebuah model karena keseluruhan dataset akan berperan dalam proses klasifikasi dan pengujian.

Selanjutnya, *hyperparameter tuning* akan diterapkan pada setiap *fold* data untuk mencari kombinasi *hyperparameter* yang memiliki performa terbaik. Berikut adalah nilai-nilai *hyperparameter* yang akan digunakan dalam penelitian ini.

- a. C = [0.01, 0.1, 1, 10, 100, 1000]
- b. Kernel = [*linear*, *polynomial*, *rbf*]
- c. Gamma = [0.001, 0.01, 0.1, 1, 10, 100]

2.7 Evaluasi

Setiap hasil prediksi pada model-model yang diteliti akan melalui tahap evaluasi dengan menggunakan beberapa metrik diantaranya *precision*, *recall*, *f1-score*, dan *accuracy*. Keempat metrik tersebut dapat dikalkulasikan dengan *formula* berikut.

$$\text{Akurasi} : \frac{TP + TN}{TP + FP + FN + TN} \quad (1)$$

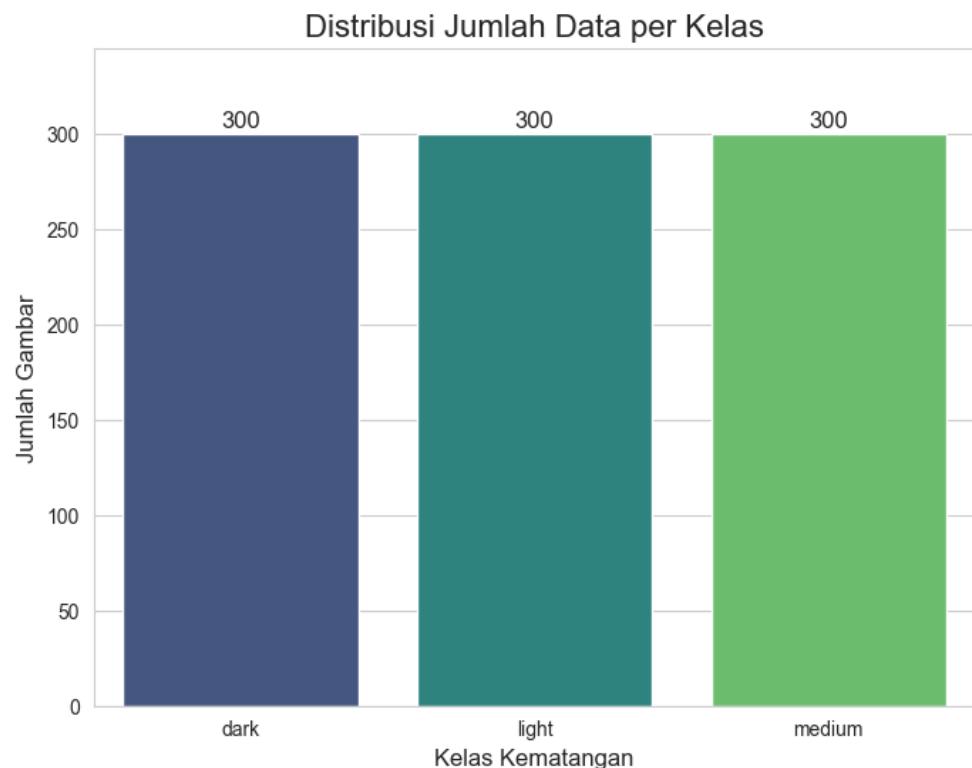
$$\text{Presisi} : \frac{TP}{TP + FP} \quad (2)$$

$$\text{Recall} : \frac{TP}{TP + FN} \quad (3)$$

$$\text{F1-score} : \frac{2 \times (\text{recall} \times \text{presisi})}{\text{recall} + \text{presisi}} \quad (4)$$

3. Hasil dan Pembahasan

3.1. Pengumpulan Data



Gambar 5. Barplot Jumlah Data Citra

Setelah proses validasi manual untuk menghindari anomali dan memastikan kualitas data, dataset akhir yang digunakan terdiri dari total 900 citra. Jumlah ini terbagi rata ke dalam tiga kelas, yaitu masing-masing 300 citra untuk kopi bisa dilihat pada gambar 5.

3.2. Preprocessing

Dataset citra yang telah melalui tahap *preprocessing* akan disimpan dalam sebuah folder dengan masing-masing kelas disimpan dalam sub-folder tersendiri sesuai label kelasnya.

3.3. Ekstraksi Fitur

Peneliti melakukan ekstraksi fitur khususnya berkaitan dengan fitur warna yang melibatkan rata-rata persebaran piksel (*mean*), variansi persebaran piksel (*variance*), dan warna piksel dominan (*mode/modus*). Setiap fitur ini akan diterapkan di masing-masing *channel* pada penelitian ini yaitu *Red*, *Green*, dan *Blue*. Hasil ekstraksi fitur dapat dilihat pada tabel 2.

Tabel 2. Hasil Ekstraksi Fitur

M_R	M_G	M_B	Var_R	Var_G	Var_B	Mo_R	Mo_G	Mo_B	Label
0.6504	0.48260	0.200912	0.148468	0.083375	0.021267	0.886275	0.678431	0.176471	1
0.6553	0.47399	0.17961	0.15856	0.08401	0.01702	0.91372	0.658824	0.180392	1
0.5972	0.4049	0.11463	0.13487	0.06341	0.00817	0.85098	0.53725	0.13333	1
0.6383	0.43215	0.16872	0.12292	0.05591	0.01302	0.90980	0.56470	0.18431	1
0.6242	0.42506	0.16622	0.12920	0.05972	0.01281	0.91372	0.57647	0.20000	1
...
0.1779	0.1589	0.12002	0.02180	0.01942	0.01588	0.20000	0.15686	0.09411	2
0.1796	0.18165	0.14832	0.01356	0.01493	0.01354	0.22352	0.19215	0.11764	2
0.1739	0.17312	0.13444	0.01117	0.01260	0.01250	0.20000	0.16078	0.09411	2
0.3203	0.30304	0.15060	0.02981	0.02835	0.01847	0.39215	0.36470	0.08235	2
0.3476	0.3246	0.14500	0.03825	0.03420	0.02168	0.41176	0.39215	0.05490	2

Rentang hasil ekstraksi fitur tidak melebihi nilai 1 karena nilai piksel pada citra telah dinormalisasi ke dalam rentang 0 hingga 1. Proses normalisasi ini dilakukan dengan membagi setiap nilai piksel dengan 255 sebelum dilakukan perhitungan untuk ekstraksi fitur.

3.4. Splitting Data

Dari total 900 citra biji kopi yang telah melalui proses pembersihan data, dilakukan pembagian data (*data splitting*) dengan proporsi 80% untuk data latih (*training data*), dan 20% untuk data uji (*testing data*).

3.5. Inisiasi Model

a. SVM

Algoritma Support Vector Machine (SVM) yang digunakan dalam penelitian ini tidak memanfaatkan library machine learning seperti scikit-learn, melainkan diimplementasikan secara mandiri (native) menggunakan library matematis seperti NumPy, statistics, dan cvxopt. Pada inisiasi awal dengan parameter {'kernel': 'linear', 'C': 1.0}, model ini dievaluasi performanya.

Berdasarkan hasil yang disajikan pada Tabel 3, model SVM dengan konfigurasi awal ini berhasil mencapai akurasi sebesar 88%, dengan rata-rata presisi 90%, recall 87%, dan F1-score 88%. Tabel ini memberikan gambaran awal mengenai kemampuan model sebelum dilakukan optimasi lebih lanjut.

Tabel 3. Matrix Evaluasi SVM

Precision	Recall	F1-Score	Accuracy
90%	87%	88%	88%

b. SVM + Tuning

Setelah melakukan pengujian terhadap berbagai kombinasi hyperparameter (kernel, C, dan gamma), ditemukan konfigurasi terbaik yang secara signifikan meningkatkan performa model. Hasil proses tuning yang dirangkum pada Tabel 4, menunjukkan bahwa kombinasi {kernel: 'rbf', C: 10, gamma: 0.01} menempati peringkat pertama dengan akurasi validasi

silang (CV) tertinggi, yaitu 91.11%. Tabel ini memvalidasi pentingnya proses hyperparameter tuning dalam menemukan setelan optimal untuk model.

Tabel 4. Matrix Evaluasi SVM

Peringkat	Kernel	C	Gamma	Degree	Akurasi (CV)
1	rbf	10	0.01	-	91.11%
2	rbf	1.0	0.01	-	89.44%
3	linear	10	-	-	88.33%
4	linear	1.0	-	-	88.33%
5	linear	0.1	-	-	88.33%

Model kemudian dilatih ulang menggunakan parameter terbaik tersebut. Peningkatan performa yang signifikan tercatat pada Tabel 5, di mana akurasi model pada data uji meningkat menjadi 91%. Selain itu, metrik lainnya seperti presisi, recall, dan F1-score juga mengalami kenaikan, yang menunjukkan kemampuan generalisasi model yang lebih baik setelah dioptimalkan

Tabel 5. Matrix Evaluasi SVM

Precision	Recall	F1-Score	Accuracy
97%	93%	95%	91%

c. *SVM + Ekstraksi fitur + K-Fold Cross Validation*

Pada tahap selanjutnya, model SVM diintegrasikan dengan pendekatan ekstraksi fitur hibrida (CNN dan Histogram Warna) dan divalidasi menggunakan K-Fold Cross Validation. Proses tuning yang komprehensif ini menghasilkan temuan penting yang disajikan pada Tabel 6. Kombinasi antara arsitektur CNN deeper_wider dengan parameter SVM {'kernel': 'linear', 'C': 0.1} terbukti memberikan rata-rata akurasi validasi silang tertinggi, mencapai 98.33%.

Tabel 6. Metrik Hasil Evaluasi SVM

Peringkat	CNN Arch	Kernel	C	Gamma	Degree	Accuracy
1	deeper_wider	linear	0.1	-	-	98.33%
2	deeper_wider	linear	1.0	-	-	98.22%
3	deeper_wider	linear	10	-	-	98.10%
4	original	linear	0.1	-	-	97.50%
5	original	linear	1.0	-	-	97.50%

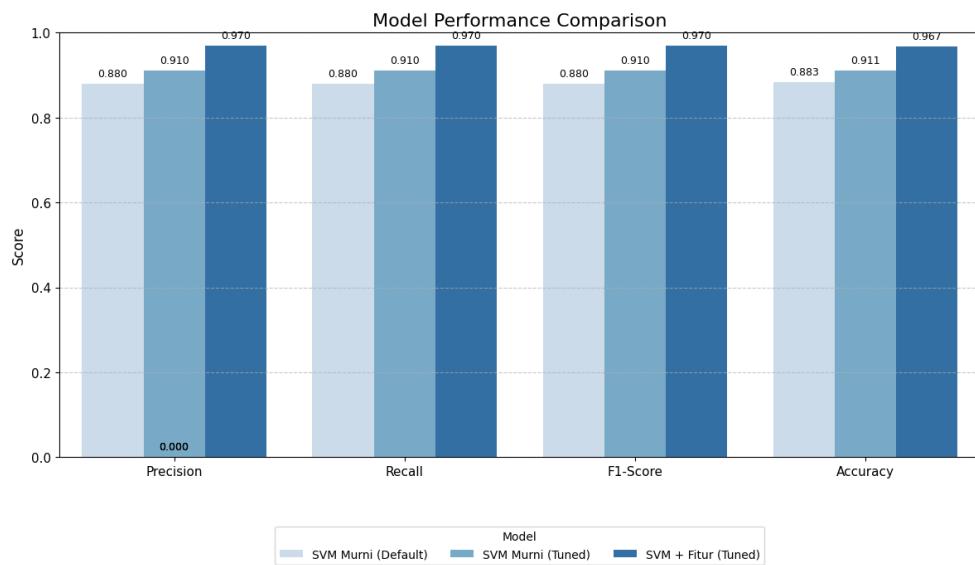
Hasil evaluasi akhir dari model hibrida terbaik ini terhadap data pengujian dapat dilihat pada

Tabel 7. Model ini menunjukkan performa yang sangat tinggi dan konsisten di semua metrik, dengan akurasi mencapai 98% serta presisi, *recall*, dan F1-score masing-masing sebesar 97%.

Tabel 7. Hasil Evaluasi SVM + Ekstraksi Fitur

Precision	Recall	F1-Score	Accuracy
97%	97%	97%	98%

3.6. Evaluasi



Gambar 8. Perbandingan Performa Model

Berdasarkan hasil evaluasi pada ketiga model terhadap data pengujian, dapat diketahui bahwa ekstraksi fitur memberikan peningkatan performa yang sangat signifikan. Perbandingan performa dari ketiga pendekatan model (SVM Murni Default, SVM Murni Tuned, dan SVM + Fitur Tuned) divisualisasikan pada Gambar 8. Grafik tersebut dengan jelas menunjukkan bahwa model hibrida (SVM + Fitur) unggul secara konsisten di semua metrik evaluasi.

Untuk menegaskan hasil akhir, Tabel 9 merangkum performa tertinggi yang dicapai oleh model SVM yang dioptimalkan dengan ekstraksi fitur. Model ini mencatatkan akurasi 96.6% (nilai ini sepertinya salah ketik di teks asli, seharusnya mendekati 98% sesuai tabel sebelumnya, harap diperiksa kembali) dengan rata-rata presisi, recall, dan F1-score sebesar 97%.

Tabel 9. Performa Model SVM + Ekstraksi Fitur

Accuracy	Mean Precision	Mean Recall	Mean F1-Score
96,6%	97%	97%	97%

4. Kesimpulan

Berdasarkan hasil evaluasi model terhadap data pengujian, dapat disimpulkan bahwa pendekatan ekstraksi fitur hibrida berhasil meningkatkan performa model *Support Vector Machine*. Peningkatan ini diukur berdasarkan hasil dari proses *hyperparameter tuning* yang disertai dengan *K-Fold Cross Validation*. Proses ini memvalidasi bahwa performa terbaik model dapat dicapai dalam rentang

parameter yang telah didefinisikan, sehingga menjadi dasar yang adil untuk membandingkan pengaruh setiap pendekatan..

Peningkatan performa yang diberikan oleh pendekatan hibrida sangat signifikan dan konsisten di seluruh metrik evaluasi. Secara spesifik, model SVM yang dioptimalkan dengan ekstraksi fitur dari arsitektur CNN **deeper_wider** dan Histogram Warna dengan parameter {kernel: 'linear', C: 0.1} menunjukkan performa tertinggi dengan nilai-nilai metrik sebagai berikut:

- a. Akurasi: **96,67%**
- b. Mean Precision: **97%**
- c. Mean Recall: **97%**
- d. Mean F1-Score: **97%**

Sedangkan, model SVM Murni yang telah dioptimalkan (91,11% akurasi) juga menunjukkan peningkatan performa dari model SVM dasar (88,33% akurasi), namun hasilnya masih berada di bawah model hibrida. Hal ini menegaskan bahwa penggunaan fitur tingkat tinggi secara fundamental lebih baik untuk tugas klasifikasi biji kopi sangra.

Referensi

- [1] Pires, J. B. P., et al. (2022). "Correlation of color and chemical compounds in roasted coffee." *Journal of Food Science*.
- [2] Cortes, C., & Vapnik, V. (1995). "Support-vector networks." *Machine Learning*, 20(3), 273–297.
- [3] LeCun, Y., et al. (1998). "Gradient-based learning applied to document recognition." *Proceedings of the IEEE*, 86(11), 2278–2324.
- [4] Niu, X., & Suen, C. Y. (2012). "A novel hybrid CNN–SVM classifier for recognizing handwritten digits." *Pattern Recognition*, 45(4), 1318–1325.
- [5] Krizhevsky, A., et al. (2012). "ImageNet classification with deep convolutional neural networks."
- [6] Ratnadewi, R., et al. (2022). "Image classification of coffee beans using SVM architecture for quality assessment." *Kinetik: Game Technology, Information System, Computer Network, Computing, Electronics, and Control*, 7(2), 181-188.

This page is intentionally left blank.