

OPTIMASI *SUPPORT VECTOR MACHINE* MENGGUNAKAN SELEKSI FITUR *RANDOM FOREST* DAN *HYPERPARAMETER GRIDSEARCHCV* UNTUK KLASIFIKASI *RAISIN DATASET*

Gregian Bayu Anugrah¹, Toni Arifin²

1) Sistem Informasi, Fakultas Teknologi Informasi, Universitas Adhirajasa Reswara Sanjaya, Indonesia

2) Teknik Informatika, Fakultas Teknologi Informasi, Universitas Adhirajasa Reswara Sanjaya, Indonesia

Article Info

Article history:

Received: 29 Juni 2025

Revised: 04 Juli 2025

Accepted: 07 Juli 2025

ABSTRACT

Abstrak

Raisin (kismis) merupakan buah kering yang memiliki nilai kandungan gizi yang tinggi dan kaya akan serat. Seiring dengan meningkatnya produksi global, kebutuhan akan sistem klasifikasi otomatis yang akurat menjadi semakin krusial untuk efisiensi industri. Proses klasifikasi manual dinilai tidak efisien, memakan waktu, serta rawan kesalahan manusia, sehingga diperlukan pendekatan berbasis kecerdasan buatan. Penelitian ini bertujuan untuk mengoptimalkan model klasifikasi Support Vector Machine (SVM) dengan penerapan seleksi fitur menggunakan algoritma Random Forest dan optimasi hyperparameter menggunakan GridSearchCV. Dataset yang digunakan adalah Raisin Dataset dari UCI Machine Learning Repository yang terdiri dari 900 sampel dua jenis raisin, yaitu Kecimen dan Besni, masing-masing memiliki tujuh fitur numerik morfologis. Hasil evaluasi menunjukkan bahwa model SVM yang telah dioptimasi mampu mencapai akurasi sebesar 90.00% dan nilai AUC 0.9388. Temuan ini menunjukkan bahwa kombinasi seleksi fitur dan optimasi hyperparameter dapat meningkatkan kinerja model klasifikasi. Penelitian ini juga memiliki perbedaan dari studi sebelumnya dalam hal objek, waktu pelaksanaan, literatur, pendekatan teori, serta hasil akhir yang diperoleh. Pendekatan ini memperkuat pemanfaatan machine learning dalam sektor agroindustri dan membuka peluang pengembangan sistem klasifikasi pangan yang lebih efisien dan presisi.

Kata Kunci: Support Vector Machine, GridSearchCV, Random Forest, Raisin Dataset, Machine Learning

Abstract

Raisin is an high nutritious and high-fiber dried fruit. An accurate and automated classification system is becoming more and more necessary for industrial efficiency as global production rises. An approach based on artificial intelligence is necessary because manual classification is thought to be ineffective, time-consuming, and prone to human error. This study uses the Random Forest algorithm for feature selection and GridSearchCV for hyperparameter optimization in order to optimize the Support Vector Machine (SVM) classification model. The Raisin Dataset, which includes 900 samples of the Kecimen and Besni varieties of raisins, each with seven morphological numerical features, is taken from the UCI Machine Learning Repository. According to the evaluation results, the optimized SVM model obtained an AUC score of 0.9388 and an accuracy of 90.00%. These results show that feature selection and hyperparameter optimization together can greatly enhance the classification model's performance. This strategy improves the use of machine learning in the agro-industrial sector and creates chances to create food classification systems that are more accurate and efficient. The difference in this study lies in the object and time of research, the literature and theory used and the results of the research.

Keywords: Support Vector Machine, GridSearchCV, Random Forest, Raisin Dataset, Machine Learning

Djtechno: Jurnal Teknologi Informasi oleh Universitas Dharmawangsa Artikel ini bersifat open access yang didistribusikan di bawah syarat dan ketentuan dengan Lisensi



Corresponding Author:E-mail : ryanbayuanugrah@gmail.com

1. PENDAHULUAN

Raisin (kismis) merupakan buah kering yang memiliki nilai gizi tinggi dan kaya akan serat. Sebagai sumber karbohidrat, *raisin* mengandung sejumlah senyawa penting seperti antioksidan, kalium, zat besi, dan serat yang bermanfaat bagi kesehatan tubuh [1]. Konsumsi raisin secara rutin diketahui dapat membantu meningkatkan fungsi tubuh dan mencegah berbagai penyakit kronis, seperti penyakit kardiovaskular, gangguan pencernaan, karies gigi, hingga diabetes melitus tipe 2 [2].

Secara global, terdapat berbagai jenis *raisin* dengan karakteristik rasa, nutrisi, dan nilai ekonomi yang berbeda. Penilaian terhadap kualitas dan jenis *raisin* menjadi tantangan tersendiri, terutama karena meningkatnya volume produksi dunia yang mencapai 1,22 juta ton pada tahun 2017 [3] dan sekitar 1,3 juta ton pada periode 2022–2023. Turki tercatat sebagai produsen utama dengan kontribusi 24%, diikuti oleh Tiongkok dan Iran masing-masing 14%, Amerika Serikat (13%), serta India (11%) [4]. Seiring dengan meningkatnya jumlah produksi, proses klasifikasi *raisin* berdasarkan jenis dan kualitas menggunakan metode tradisional menjadi semakin mahal dan memakan waktu. Hal ini disebabkan oleh besarnya jumlah produksi serta tingginya biaya tenaga kerja [5]. Selain itu, proses manual yang dilakukan melalui teknik konvensional cenderung kurang stabil dan memiliki efektivitas yang terbatas. Faktor seperti kelelahan, kurangnya konsentrasi, dan perbedaan subjektivitas antar individu dalam proses klasifikasi konvensional turut menyulitkan pencapaian standar yang konsisten [6].

Perkembangan teknologi membawa kemajuan signifikan dalam berbagai bidang, termasuk dalam pengolahan data melalui kecerdasan buatan. Salah satu teknologi yang banyak dimanfaatkan adalah *machine learning* (ML), yang memiliki kemampuan untuk mempelajari pola data dan membuat prediksi akurat [7][8]. *Machine learning* telah banyak diterapkan dalam klasifikasi berbagai objek, seperti benih, butir beras, hingga raisin [9]. Dalam konteks industri pangan, pemanfaatan machine learning didukung oleh

kemajuan teknologi big data yang mampu menangani data dalam jumlah besar dan kompleks secara efisien [10].

Machine learning terdiri dari dua pendekatan utama, yaitu *supervised* dan *unsupervised learning*, yang dibedakan berdasarkan keberadaan label pada data [11][12][13]. Pada penelitian klasifikasi, pendekatan supervised learning umumnya digunakan, dengan algoritma seperti *Decision Tree*, *Random Forest*, dan *Support Vector Machine* [14]. *Random Forest* merupakan metode *ensemble learning* yang menggabungkan banyak pohon keputusan untuk menghasilkan prediksi yang stabil dan akurat [15][16], sedangkan *Support Vector Machine* (SVM) dikenal unggul dalam memisahkan data linier maupun non-linier dan telah banyak digunakan dalam berbagai aplikasi seperti pengenalan tulisan tangan, deteksi penyusupan, pendeteksian wajah, klasifikasi email, klasifikasi gen, dan halaman web [17][18][19].

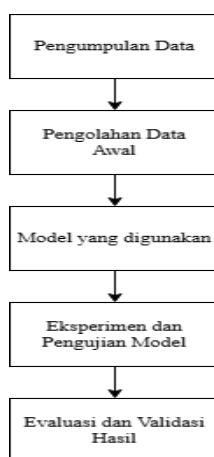
Kinerja algoritma *machine learning* sangat bergantung pada pemilihan *parameter* yang tepat. Dalam hal ini, *hyperparameter* menjadi faktor penting yang perlu ditentukan sebelum proses pelatihan model dilakukan [20]. Untuk mengatasi kelemahan algoritma yang sensitif terhadap data tidak seimbang atau *missing values*, diperlukan tahapan data *preprocessing* serta optimasi *hyperparameter* [21][22]. Salah satu teknik optimasi yang umum digunakan adalah *GridSearchCV*, yang bekerja dengan mencari kombinasi *hyperparameter* terbaik melalui validasi silang untuk menghindari *overfitting* [23][24].

Penelitian sebelumnya menunjukkan bahwa kombinasi SVM dan *Genetic Algorithm* (GA) dapat menghasilkan akurasi klasifikasi *raisin* yang cukup tinggi, yaitu sebesar 87,67% [25]. Namun, pendekatan tersebut masih terbatas pada satu metode seleksi fitur dan belum mengeksplorasi efektivitas kombinasi metode lainnya. Oleh karena itu, penelitian ini mengusulkan pendekatan baru dengan menggabungkan algoritma *Support Vector Machine* (SVM), *Random Forest* untuk seleksi fitur, serta *GridSearchCV* untuk optimasi *hyperparameter*.

Penelitian ini diharapkan mampu menghasilkan model klasifikasi yang lebih optimal karena *Random Forest* mampu menilai pentingnya setiap fitur secara efisien, dan *GridSearchCV* dapat membantu menemukan kombinasi parameter terbaik secara sistematis.

2. METODE PENELITIAN

Penelitian merupakan suatu proses yang terstruktur dan sistematis yang bertujuan untuk menemukan jawaban dan pemahaman terhadap suatu permasalahan atau pertanyaan. Dalam proses ini, peneliti melakukan serangkaian kegiatan seperti pengumpulan data, observasi, analisis, pencarian solusi, serta pengujian hipotesis [26]. Metodologi penelitian mencakup langkah-langkah sistematis mulai dari perumusan masalah, kajian literatur, perumusan hipotesis, identifikasi variabel, penentuan populasi dan sampel, hingga teknik pengumpulan dan analisis data [27]. Menurut Dawson metode penelitian yang paling umum digunakan ada empat yaitu penelitian tindakan, eksperimen, studi kasus, dan survei [28]. Penelitian ini menggunakan metode penelitian eksperimen, berikut tahapan atau langkah-langkah dalam penelitian ini, agar penelitian berjalan sesuai prosedur.



Gambar 1 Tahapan Penelitian

2.1 Pengumpulan Data

Penelitian ini menggunakan *Raisin Dataset* yang didapat melalui *UCI Machine Learning Repository*, yang dapat diakses dengan mudah karena bersifat terbuka. Dataset tersebut didonasikan tahun 2021 berdasarkan studi dari [1]. *Raisin Dataset* memiliki tujuh fitur bertipe numerik, yaitu: *area*, *perimeter*, *major_axis*, *minor_axis*, *eccentricity*, *convex_area*, dan *extent*. Dataset ini memiliki klasifikasi yang seimbang dan terbagi ke dalam dua kelompok kelas, dengan total sebanyak 900 data, yang terdiri dari 450 data kelas Kecimen dan 450 data kelas Besni [2]. Deskripsi atribut dari *Raisin Dataset* dapat dilihat pada Tabel 1.

Tabel 1 Deskripsi Raisin Dataset

No	Nama Variabel	Deskripsi
1	<i>Area</i>	Memberikan jumlah piksel di dalam batas-batas kismis.

2	<i>MajorAxisLength</i>	Memberikan panjang sumbu utama, yang merupakan garis terpanjang yang dapat digambar pada kismis.
3	<i>MinorAxisLength</i>	Memberikan panjang sumbu kecil, yang merupakan garis terpendek yang dapat digambar pada kismis.
4	<i>Eccentricity</i>	Ini memberikan ukuran eksentrisitas elips, yang memiliki momen yang sama dengan kismis.
5	<i>ConvexArea</i>	Memberikan jumlah piksel cangkang cembung terkecil dari wilayah yang dibentuk oleh kismis.
6	<i>Extent</i>	Memberikan rasio wilayah yang dibentuk oleh kismis terhadap total piksel dalam kotak pembatas
7	<i>Perimeter</i>	Mengukur lingkungan dengan menghitung jarak antara batas kismis dan piksel di sekitarnya
8	<i>Class</i>	Jenis kismis: Kecimen dan Besni.

2.2 Pengolahan Data Awal

Tahap *preprocessing* data dalam penelitian ini meliputi beberapa langkah penting, dalam penelitian ini tahapan data *preprocessing* yang diterapkan adalah sebagai berikut:

a. Pembersihan Data

Tahap awal pengolahan data adalah mengidentifikasi dan menghapus data yang bersifat duplikat serta memeriksa apakah terdapat data yang hilang (*missing values*). Duplikasi data dapat menyebabkan bias pada hasil pelatihan, sementara nilai kosong dapat mengganggu kinerja model, maka data duplikat dan data kosong dihapus agar data yang digunakan bersifat valid dan akurat. Proses dan hasil pembersihan data ditunjukkan pada Gambar 2.

```
[6] # Cek jumlah data sebelum dan sesudah cleaning
print(f"Jumlah data sebelum cleaning: {df.shape[0]}")
df = df.drop_duplicates().dropna()
print(f"Jumlah data setelah cleaning: {df.shape[0]}")
```

Jumlah data sebelum cleaning: 900
Jumlah data setelah cleaning: 900

Gambar 2 Pembersihan Data

b. Transformasi Label Target

Variabel target pada dataset memiliki nilai dalam bentuk kategorik, yaitu nama jenis *raisin*. Agar dapat digunakan dalam algoritma *Support Vector Machine* (SVM), label kategorik tersebut diubah menjadi bentuk numerik. Proses ini dilakukan menggunakan metode *encoding* dengan mengonversi masing-masing kategori menjadi angka yang merepresentasikan kelas tertentu. Hal ini penting untuk memastikan bahwa model dapat memproses label dengan tepat selama pelatihan dan pengujian. Proses mentransformasi label target dari bentuk kategorik ke numerik, seperti yang ditampilkan Gambar 3, dan hasil sebelum transformasi label ditunjukkan pada Gambar 4 dan hasil sesudah transformasi label ditunjukkan pada Gambar 5.

```
# Encode label (assuming 'Class' is the target variable)
print(["Unique classes before encoding:", df['Class'].unique()])
df['Class'] = df['Class'].astype('category').cat.codes
print("berdasarkan alfabet, maka besni:0 kecimen:1", df['Class'].unique())
```

Gambar 3 Proses Transformasi Label

	Area	MajorAxisLength	MinorAxisLength	Eccentricity	ConvexArea	Extent	Perimeter	Class
0	87524	442.246011	253.291155	0.819738	90546	0.758651	1184.040	Kecimen
1	75166	406.690687	243.032436	0.801805	78789	0.684130	1121.786	Kecimen
2	90856	442.267048	266.328318	0.798354	93717	0.637613	1208.575	Kecimen
3	45928	286.540559	208.760042	0.684989	47336	0.699599	844.162	Kecimen
4	79408	352.190770	290.827533	0.564011	81463	0.792772	1073.251	Kecimen

Gambar 4 Label Sebelum Transformasi

	Area	MajorAxisLength	MinorAxisLength	Eccentricity	ConvexArea	Extent	Perimeter	Class
0	87524	442.246011	253.291155	0.819738	90546	0.758651	1184.040	1
1	75166	406.690687	243.032436	0.801805	78789	0.684130	1121.786	1
2	90856	442.267048	266.328318	0.798354	93717	0.637613	1208.575	1
3	45928	286.540559	208.760042	0.684989	47336	0.699599	844.162	1
4	79408	352.190770	290.827533	0.564011	81463	0.792772	1073.251	1

Gambar 5 Label Setelah Transformasi

c. Seleksi Fitur

Seleksi fitur bertujuan memilih atribut-atribut paling relevan yang berperan penting dalam membedakan kelas, sehingga dapat meningkatkan kinerja model dan mengurangi kompleksitas perhitungan. Dalam penelitian ini, digunakan algoritma *Random Forest Classifier* untuk menghitung nilai *feature importance* dari setiap fitur. Berdasarkan hasil tersebut, dipilih enam fitur teratas dengan nilai kepentingan tertinggi. Fitur-fitur terpilih ini kemudian digunakan pada tahap pelatihan dan pengujian model guna menyederhanakan struktur model, mempercepat proses komputasi, serta membantu mencegah *overfitting* dengan hanya mempertahankan fitur yang paling berkontribusi terhadap klasifikasi. Proses seleksi fitur menggunakan *Random Forest*, dengan mengambil 6 fitur terpenting dari dataset, seperti yang ditampilkan pada Gambar 6, dan hasil seleksi fitur ditampilkan pada Gambar 7.

```
[9] # Menampilkan semua fitur sebelum seleksi
print("Semua fitur sebelum seleksi:")
print(list(df.columns))
```

```
Semua fitur sebelum seleksi:
['Area', 'MajorAxisLength', 'MinorAxisLength', 'Eccentricity', 'ConvexArea', 'Extent', 'Perimeter', 'Class']
```

Gambar 6 Fitur Sebelum Di Seleksi

```
[27] X = df.drop(columns=['class'])
     y = df['class']
     # Seleksi fitur sebelum split
     num_features = 6
     rf = RandomForestClassifier(n_estimators=100, random_state=42)
     rf.fit(X, y)

     # Ambil fitur dengan importance tertinggi
     feature_importances = rf.feature_importances_
     selected_indices = np.argsort(feature_importances)[-num_features:]
     X_selected = X.iloc[:, selected_indices]
     selected_features = X.columns[selected_indices]

     print("\nSelected features:", list(selected_features))
```

Selected features: ['MinorAxisLength', 'Eccentricity', 'Area', 'ConvexArea', 'MajorAxisLength', 'Perimeter']

Gambar 7 Fitur Setelah Di Seleksi

d. Data Splitting

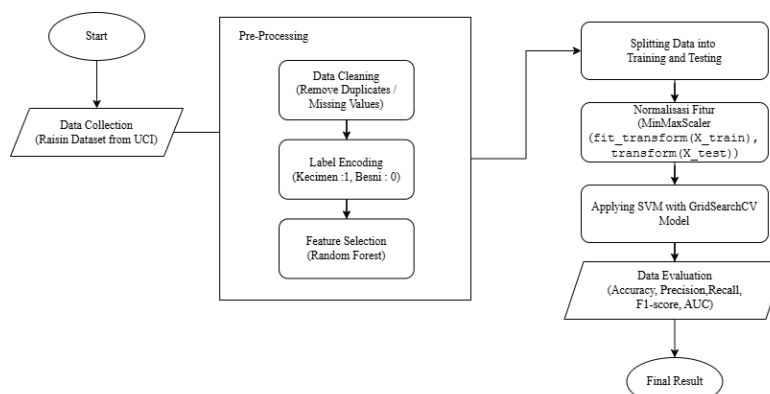
Data dibagi menjadi data *training* dan *testing* dengan rasio 90:10 secara stratifikasi untuk menjaga proporsi kelas. Validasi model dilakukan menggunakan *GridSearchCV* dengan *10-fold cross-validation* guna menghindari *overfitting* dan bias, serta menghasilkan evaluasi model yang lebih akurat.

e. Normalisasi Fitur

Proses normalisasi dalam penelitian ini menggunakan pendekatan *Min-Max Scaling*, yaitu metode yang bertujuan untuk mengubah nilai setiap fitur ke dalam skala antara 0 hingga 1. Normalisasi dilakukan agar semua fitur berada dalam rentang yang seragam, sehingga tidak ada fitur yang mendominasi algoritma pembelajaran karena memiliki skala nilai yang lebih besar. Proses ini penting khususnya untuk algoritma yang sensitif terhadap perbedaan skala antar fitur, seperti *Support Vector Machine* (SVM).

2.3 Model Yang Digunakan

Model atau metode yang diusulkan dalam penelitian ini adalah algoritma *Support Vector Machine* (SVM) dengan menggunakan teknik *Hyperparameter GridSearchCV*. Alur model yang digunakan ditunjukkan pada Gambar 8.



Gambar 8 Alur Model

Sumber: Penulis

Penelitian ini menggunakan algoritma *Support Vector Machine* (SVM) untuk klasifikasi jenis *raisin* berdasarkan fitur morfologi yang telah dipilih. Setelah melalui tahap normalisasi dengan *Min-Max Scaling*, model SVM diterapkan pada data latih dan diuji terhadap data uji. Untuk meningkatkan performa model, dilakukan tuning *hyperparameter* menggunakan *GridSearchCV* dengan validasi silang 10-fold guna mencari kombinasi *parameter* terbaik dan mencegah *overfitting*. Evaluasi model dilakukan menggunakan metrik *Accuracy*, *Precision*, *Recall*, *F1-Score*, dan *AUC*, untuk mengukur kinerja dan kemampuan generalisasi model terhadap data yang belum pernah dilihat sebelumnya.

2.4 Eksperimen dan Pengujian Model

Eksperimen dan pengujian model dilakukan melalui beberapa tahapan, eksperimen menggunakan bantuan *Google Colab*. Dimulai dengan mengimpor dataset dan pustaka yang dibutuhkan di *Google Colab*, diikuti preprocessing seperti penghapusan duplikat, data kosong, dan transformasi label target. Selanjutnya, dilakukan seleksi enam fitur terbaik menggunakan algoritma *Random Forest*, pembagian data menjadi *training* dan *testing*, serta normalisasi fitur dengan *Min-Max Scaling*. Model kemudian dilatih menggunakan algoritma SVM dengan optimasi *hyperparameter* melalui *GridSearchCV* dan validasi silang 10-fold. Adapun *parameter* yang akan dioptimasi menggunakan *GridSearchCV* ditampilkan pada Gambar 9:

```
[ ] param_grid = {
    'C': [10, 20, 100],
    'kernel': ['linear', 'sigmoid'],
    'gamma': ['scale', 'auto', 0.01, 0.1, 1],
    'coef0': [0.0, 0.1, 0.9],
}
```

Gambar 9 Parameter SVM

Fungsi utama dari *GridSearchCV* adalah untuk melakukan pencarian *parameter* secara sistematis berdasarkan kombinasi nilai-nilai dalam ruang pencarian serta mengevaluasi berbagai kombinasi *hyperparameter* yang telah ditentukan pada model SVM, kemudian memilih kombinasi *parameter* terbaik berdasarkan performa evaluasi tertinggi. Terakhir, evaluasi kinerja dilakukan menggunakan metrik *Accuracy*, *AUC*, *Precision*, *Recall*, *F1-Score* dan *AUC*.

2.5 Evaluasi dan Validasi Validasi

Evaluasi model dilakukan menggunakan confusion matrix dan metrik seperti *Accuracy*, *Precision*, *Recall*, *F1-Score*, serta *AUC*. Validasi dilakukan dengan 10-fold cross-

validation untuk memastikan model dilatih dan diuji secara merata. Pemilihan kombinasi parameter terbaik dilakukan melalui *GridSearchCV*, lalu model dievaluasi kembali menggunakan metrik tersebut dalam bentuk *Classification Report* guna memperoleh gambaran menyeluruh terhadap kinerja model.

3. HASIL DAN PEMBAHASAN

Penerapan metode *Support Vector Machine* (SVM) menggunakan teknik seleksi fitur *Random Forest* dan optimasi *hyperparameter GridSearchCV* pada *Raisin Dataset* bertujuan untuk meningkatkan akurasi klasifikasi dalam membedakan dua jenis *raisin*, yaitu Kecimen dan Besni. Perbedaan teknik yang digunakan dengan penelitian sebelumnya adalah dalam hal kombinasi seleksi fitur dan optimasi *parameter* yang sistematis.

3.1 Hasil Penelitian

Hasil eksperimen akan menunjukkan sejauh mana pengaruh seleksi fitur terhadap penyederhanaan data dan peningkatan performa model, serta bagaimana penggunaan *GridSearchCV* mampu menemukan kombinasi *parameter* yang optimal dalam algoritma SVM. Akurasi yang optimal berhasil dicapai maupun tidak, seluruh hasil dari pengujian dan evaluasi model akan ditampilkan dan dianalisis secara rinci. Evaluasi dilakukan dengan menggunakan berbagai metrik seperti *accuracy*, *precision*, *recall*, *F1-score*, dan *AUC*.

a. Hasil eksperimen menggunakan metode *Support Vector Machine* (SVM), yang dievaluasi melalui *classification report* serta nilai *Area Under Curve* (AUC). Hasil dari evaluasi tersebut ditampilkan pada Gambar 10.

```
===== [1] SVM TANPA Seleksi Fitur & TANPA GridSearch =====
```

Accuracy: 0.8889				
AUC: 0.9200				
Classification Report:				
	precision	recall	f1-score	support
0	0.9730	0.8000	0.8780	45
1	0.8302	0.9778	0.8980	45
accuracy			0.8889	90
macro avg	0.9016	0.8889	0.8880	90
weighted avg	0.9016	0.8889	0.8880	90

Gambar 10 Hasil Metode SVM

Pada Gambar 10 menjelaskan hasil dari *Support Vector Machine* sebelum menggunakan seleksi fitur *Random Forest* dan *GridSearchCV* dengan hasil *Accuracy*

88.89%, *Precision* 90.16%, *Recall* 88.89%, dan *F1-Score* 88.80% berdasarkan *Macro Average* serta nilai *Area Under Curve* (AUC) sebesar 0.9200.

b. Hasil eksperimen menggunakan metode *Support Vector Machine* (SVM) dan Seleksi Fitur menggunakan *Random Forest*, yang dievaluasi melalui *classification report* serta nilai *Area Under Curve* (AUC). Hasil dari evaluasi tersebut ditampilkan pada Gambar 11.

```
===== [2] SVM + Seleksi Fitur Random Forest =====
```

Accuracy: 0.8889				
AUC: 0.9205				
Classification Report:				
	precision	recall	f1-score	support
0	0.9730	0.8000	0.8780	45
1	0.8302	0.9778	0.8980	45
accuracy			0.8889	90
macro avg	0.9016	0.8889	0.8880	90
weighted avg	0.9016	0.8889	0.8880	90

Gambar 11 Hasil Metode SVM dan seleksi Fitur RF

Pada Gambar 11 menjelaskan hasil dari *Support Vector Machine* dengan menggunakan seleksi fitur *Random Forest* dengan hasil *Accuracy* 88.89%, *Precision* 90.16%, *Recall* 88.89%, dan *F1-Score* 88.89% berdasarkan *Macro Average* serta nilai *Area Under Curve* (AUC) sebesar 0.9205.

c. Hasil eksperimen menggunakan metode *Support Vector Machine* (SVM) dan *GridSearchCV* tanpa Seleksi Fitur menggunakan *Random Forest*, yang dievaluasi melalui *classification report* serta nilai *Area Under Curve* (AUC). Hasil dari evaluasi tersebut ditampilkan pada Gambar 12.

```
===== [3] SVM + GridSearchCV (TANPA Seleksi Fitur) =====
```

Best Parameters: {'C': 100, 'coef0': 0.9, 'gamma': 'auto', 'kernel': 'sigmoid'}				
Training CV Accuracy: 0.8691				
Test Accuracy: 0.8889				
AUC: 0.9215				
Classification Report:				
	precision	recall	f1-score	support
0	0.9730	0.8000	0.8780	45
1	0.8302	0.9778	0.8980	45
accuracy			0.8889	90
macro avg	0.9016	0.8889	0.8880	90
weighted avg	0.9016	0.8889	0.8880	90

Gambar 12 Hasil Metode SVM dan seleksi Fitur RF

Pada Gambar 12 menjelaskan hasil dari *Support Vector Machine* (SVM) dengan menggunakan *GridSearchCV* dengan hasil *Accuracy* 88.89%, *Precision* 90.16%, *Recall* 88.89%, dan *F1-Score* 88.80% berdasarkan *Macro Average* serta nilai *Area Under Curve* (AUC) sebesar 0.9215.

d. Hasil eksperimen metode *Support Vector Machine* (SVM) dengan seleksi fitur *Random Forest* dan optimasi *GridSearchCV*, yang dievaluasi melalui *classification report*

serta nilai *Area Under Curve* (AUC). Hasil dari evaluasi tersebut ditampilkan pada Gambar 13.

```
==== [4] SVM + Seleksi Fitur RF + Optimasi GridSearchCV ====
Best Parameters: {'C': 100, 'coef0': 0.0, 'gamma': 'scale', 'kernel': 'linear'}
Training CV Accuracy: 0.8654320987654321

Test Set Performance:
Accuracy: 0.9000
AUC: 0.9388

Classification Report:
      precision    recall  f1-score   support

     0       0.9737     0.8222     0.8916         45
     1       0.8462     0.9778     0.9072         45

 accuracy          0.9099     0.9000     0.8994         90
 macro avg          0.9099     0.9000     0.8994         90
 weighted avg          0.9099     0.9000     0.8994         90
```

Gambar 13 Hasil SVM dengan Seleksi Fitur RF dan GridSearchCV

Pada Gambar 13 menjelaskan hasil dari *Support Vector Machine* (SVM) setelah menggunakan seleksi fitur *Random Forest* dan *GridSearchCV* dengan hasil *Accuracy* 90%, *Precision* 90.99%, *Recall* 90%, dan *F1-Score* 89.94% berdasarkan *Macro Average* serta nilai *Area Under Curve* (AUC) sebesar 0.9388.

3.2 Pembahasan

Berdasarkan eksperimen yang dilakukan sebelumnya, pada tahap ini menjelaskan hasil peningkatan nilai metrik dari pengujian klasifikasi metode *Support Vector Machine* (SVM) menggunakan seleksi fitur *Random Forest* dan optimasi *GridSearchCV*. Adapun peningkatan hasil pengujian ini didapat dari pemilihan parameter dari model menggunakan seleksi fitur *Random Forest* dan optimasi *GridSearchCV*.

Fitur terpilih yang digunakan dalam penelitian ini berdasarkan *feature importance* dari *Random Forest Classifier* pada penelitian ini ditunjukkan pada Gambar 14.

```
Selected features: ['MinorAxisLength', 'Eccentricity', 'Area', 'ConvexArea', 'MajorAxisLength', 'Perimeter']
```

Gambar 14 Fitur yang Digunakan

Adapun kombinasi parameter terbaik yang digunakan berdasarkan hasil optimasi menggunakan metode *GridSearchCV* dalam proses optimasi model pada penelitian ini ditampilkan pada Tabel 2.

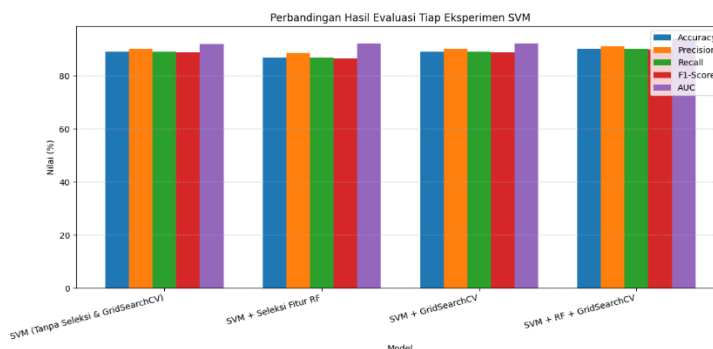
Tabel 2 Parameter yang digunakan

Parameter	Nilai
<i>C</i>	100
<i>Coef0</i>	0.0
<i>Gamma</i>	scale
<i>Kernel</i>	linear

Grafik perbandingan hasil evaluasi dari beberapa eksperimen klasifikasi menggunakan algoritma *Support Vector Machine* (SVM), baik sebelum maupun sesudah

dilakukan proses seleksi fitur dan optimasi *hyperparameter* menggunakan *GridSearchCV*.

Grafik ini menyajikan lima metrik evaluasi utama, yaitu *Accuracy*, *Precision*, *Recall*, *F1-Score*, dan *AUC*, yang dibandingkan secara visual antar model. Grafik perbandingan ditampilkan pada Gambar 14.



Gambar 14 Grafik Hasil Evaluasi Eksperimen

Dalam upaya mengevaluasi hasil penelitian ini, dilakukan perbandingan dengan penelitian sebelumnya. Tujuan dari perbandingan ini adalah untuk menilai sejauh mana efektivitas metode yang diterapkan dalam penelitian ini. Rincian hasil perbandingan dengan penelitian terdahulu disajikan pada Tabel 3.

Tabel 3 Perbandingan Hasil Penelitian

Perbandingan Hasil Penelitian		
Model	Accuracy	AUC
SVM + GA (tanpa Hyperparameter)	87.67%	0.930
SVM + RF (Dengan GridSeachCV)	90.00%	0.9388

Penelitian yang dilakukan sebelumnya menerapkan algoritma *Support Vector Machine* (SVM) yang dioptimasi menggunakan metode *Genetic Algorithm* (GA) untuk seleksi fitur. Hasil terbaik dari penelitian tersebut dicapai oleh SVM dengan akurasi sebesar 87.67% dan *AUC* sebesar 0.930 [25]. Meskipun demikian, penelitian ini belum menerapkan seleksi fitur berbasis *Random Forest* maupun teknik optimasi *hyperparameter* secara sistematis seperti *GridSearchCV*.

Peningkatan akurasi sebesar 2.33% menunjukkan penggunaan teknik penelitian ini dapat meningkatkan kinerja model dalam mengklasifikasikan jenis *raisin*. Adapun hasil dari penelitian sebelumnya yang digunakan sebagai bahan perbandingan adalah hasil dari SVM dan GA. Perbandingan ini bertujuan untuk melihat sejauh mana penerapan teknik seleksi fitur *Random Forest* dan *Hyperparameter Optimization* dapat meningkatkan performa model SVM.

4. SIMPULAN

Berdasarkan hasil penelitian dan analisis yang telah dilakukan, dapat disimpulkan bahwa model klasifikasi berbasis algoritma *Support Vector Machine* (SVM) berhasil dikembangkan untuk membedakan dua jenis kismis, yaitu Kecimen dan Besni. Hasil klasifikasi SVM melalui seleksi fitur menggunakan *Random Forest* dan optimasi *hyperparameter* dengan *GridSearchCV* meningkatkan performa model. Model yang dihasilkan mampu mencapai akurasi 90%, *precision* 90.99%, *recall* 90%, *F1-score* 89.94%, dan *AUC* sebesar 0.9388, menunjukkan kemampuan klasifikasi yang kuat dan seimbang.

Untuk penelitian selanjutnya, terdapat beberapa langkah yang dapat dilakukan guna meningkatkan akurasi dan performa klasifikasi jenis *raisin*. Salah satunya dapat dilakukan pengujian dengan variasi jumlah fitur terpilih guna melihat pengaruhnya terhadap performa model. Selain itu penerapan teknik seleksi fitur alternatif seperti *Recursive Feature Elimination* (RFE) atau metode *Forward Selection*, dan menerapkan optimasi lanjutan lain seperti *Bayesian Optimization* atau *RandomizedSearchCV* juga dapat menjadi pilihan untuk meningkatkan performa model.

REFERENCES

- [1] İ. Cinar, M. KOKLU, and Ş. TAŞDEMİR, "Kuru Üzüm Tanelerinin Makine Görüşü ve Yapay Zeka Yöntemleri Kullanılarak Sınıflandırılması," *Gazi J. Eng. Sci.*, vol. 6, no. 3, pp. 200–209, 2020, doi: 10.30855/gmbd.2020.03.03.
- [2] A. Bisri and M. Man, "Machine Learning Algorithms Based on Sampling Techniques for Raisin Grains Classification," *Int. J. Informatics Vis.*, vol. 7, no. 1, pp. 7–14, 2023, doi: 10.30630/joiv.7.1.970.
- [3] Y. Zhang, Y. Yang, C. Ma, and L. Jiang, "Identification of multiple raisins by feature fusion combined with NIR spectroscopy," *PLoS One*, vol. 17, no. 7 July, pp. 1–13, 2022, doi: 10.1371/journal.pone.0268979.
- [4] H. Incekara, "Classification of Raisin Grains Using Different Artificial Neural Network Methods," *Intell. Methods Eng. Sci.*, Sep. 2024, doi: 10.58190/imiens.2024.104.
- [5] E. K. Yılmaz, T. Oğuz, and K. Adem, "A CNN-Based Hybrid Approach to Classification of Raisin Grains," *1st Int. Conf. Front. Acad.*, no. February, pp. 18–21, 2023, [Online]. Available: <https://as-proceeding.com/index.php/icfar/article/view/157/108>
- [6] A. Tulchhia and M. Rathore, "RANDOM FOREST MODEL FOR CLASSIFICATION OF RAISINS USING," *OORJA*, vol. 19, no. 1, pp. 79–86, 2021.
- [7] A. Polowczyk and A. Polowczyk, "The Effectiveness of PCA in KNN, Gaussian Naive Bayes Classifier and SVM for Raisin Dataset," *CEUR Workshop Proc.*, vol. 3695, pp. 9–16, 2023.
- [8] M. Dirik, "Improving Raisin Grains Classification with a Hybrid PSO-NN Approach," *All Sci. Proc.*, 2023, [Online]. Available: <http://as-proceeding.com/:Konya,Turkeyhttps://as-proceeding.com/index.php/iccar>
- [9] S. J. Sammut *et al.*, "Multi-omic machine learning predictor of breast cancer therapy response," *Nature*, vol. 601, no. 7894, pp. 623–629, 2022, doi: 10.1038/s41586-021-04278-5.
- [10] M. N. Raihen and S. Akter, "Prediction modeling using deep learning for the classification of grape-type dried fruits," *Int. J. Math. Comput. Eng.*, vol. 2, no. 1, pp. 1–12, 2024, doi: 10.2478/ijmce-2024-0001.
- [11] Q. H. Su and K. N. Chiang, "Predicting Wafer-Level Package Reliability Life Using Mixed Supervised and Unsupervised Machine Learning Algorithms," *Materials (Basel)*, vol. 15, no. 11, 2022, doi: 10.3390/ma15113897.
- [12] T. Arifin, I. W. P. Agung, E. Junianto, R. Rachman, I. R. Wibowo, and D. D. Agustin, "Breast cancer

- identification using machine learning and hyperparameter optimization," *Indones. J. Electr. Eng. Comput. Sci.*, vol. 36, no. 3, pp. 1620–1630, 2024, doi: 10.11591/ijeecs.v36.i3.pp1620-1630.
- [13] W. U. Dewi, N. Khoirin, and U. Mustofa, "IMPLEMENTATION OF DECISION TREE AND SUPPORT VECTOR MACHINE ON RAISIN SEED CLASSIFICATION," *J. Progr. Stud. Pendidik. Mat.*, vol. 12, no. 1, pp. 1309–1321, 2023.
- [14] A. Mishra and A. Dasgupta, "Supervised and Unsupervised Machine Learning Algorithms for Forecasting the Fracture Location in Dissimilar Friction-Stir-Welded Joints," *Forecasting*, vol. 4, no. 4, pp. 787–797, 2022, doi: 10.3390/forecast4040043.
- [15] G. B. Anugrah, R. N. Z. Shafa, and A. Kurniawan, "Perbandingan Algoritma Machine Learning Untuk Klasifikasi Dry Bean Dataset," *J. Responsif Ris. Sains dan Inform.*, vol. 5, no. 2, pp. 407–415, 2023, doi: 10.51977/jti.v5i2.1292.
- [16] Q. Du and J. Zhai, "Application of artificial intelligence Sensors based on random forest algorithm in financial recognition models," *Meas. Sensors*, vol. 33, no. June, p. 101245, 2024, doi: 10.1016/j.measen.2024.101245.
- [17] M. R. Reddy, "Implementation of SVM machine learning Algorithm to predict lung And Breast Cancer," *Turkish J. Comput. Math. Educ.*, vol. 12, no. 12, pp. 3050–3060, 2021.
- [18] J. Wu and C. Hicks, "Breast cancer type classification using machine learning," *J. Pers. Med.*, vol. 11, no. 2, pp. 1–12, 2021, doi: 10.3390/jpm11020061.
- [19] A. Selvan M, "AN INTRUSION DETECTION SYSTEM MODEL FOR DETECTING KNOWN AND INNOVATIVE CYBER ATTACKS," *J. Sci. Technol. Res.*, vol. 1, no. 2, pp. 151–157, 2021.
- [20] E. Elgeldawi, A. Sayed, A. R. Galal, and A. M. Zaki, "Hyperparameter tuning for machine learning algorithms used for arabic sentiment analysis," *Informatics*, vol. 8, no. 4, pp. 1–21, 2021, doi: 10.3390/informatics8040079.
- [21] A. M. Rahmani *et al.*, "Machine learning (ML) in medicine: Review, applications, and challenges," *Mathematics*, vol. 9, no. 22, pp. 1–52, 2021, doi: 10.3390/math9222970.
- [22] M. Daviran, M. Shamekhi, R. Ghezelbash, and A. Maghsoudi, "Landslide susceptibility prediction using artificial neural networks, SVMs and random forest: hyperparameters tuning by genetic optimization algorithm," *Int. J. Environ. Sci. Technol.*, vol. 20, no. 1, pp. 259–276, 2023, doi: 10.1007/s13762-022-04491-3.
- [23] T. Yan, S. L. Shen, A. Zhou, and X. Chen, "Prediction of geological characteristics from shield operational parameters by integrating grid search and K-fold cross validation into stacking classification algorithm," *J. Rock Mech. Geotech. Eng.*, vol. 14, no. 4, pp. 1292–1303, 2022, doi: 10.1016/j.jrmge.2022.03.002.
- [24] T. Alshammari, "Using Artificial Neural Networks with GridSearchCV for Predicting Indoor Temperature in a Smart Home," *Eng. Technol. Appl. Sci. Res.*, vol. 14, no. 2, pp. 13437–13443, 2024, doi: 10.48084/etasr.7008.
- [25] Y. Ramdhani, D. F. Apra, and D. P. Alamsyah, "Feature selection optimization based on genetic algorithm for support vector classification varieties of raisin," *Indones. J. Electr. Eng. Comput. Sci.*, vol. 30, no. 1, pp. 192–199, 2023, doi: 10.11591/ijeecs.v30.i1.pp192-199.
- [26] A. Sofya *et al.*, "Metode Survey : Explanatory Survey dan Cross Sectional dalam Penelitian Kuantitatif Survey Methods : Explanatory Survey and Cross Sectional in Quantitative Research," *J. Pendidikan, Ilmu Sos. dan Pengabd. Kpd. Masy.*, vol. 4, no. 3, pp. 1695–1708, 2024.
- [27] M. Waruwu, S. Natijatul, P. R. Utami, and E. Yanti, "Metode Penelitian Kuantitatif : Konsep , Jenis , Tahapan dan Kelebihan," *J. Ilm. Profesi Pendidik.*, vol. 10, pp. 917–932, 2025.
- [28] R. Resmiati and T. Arifin, "Klasifikasi Pasien Kanker Payudara Menggunakan Metode Support Vector Machine dengan Backward Elimination," *Sistmasi*, vol. 10, no. 2, p. 381, 2021, doi: 10.32520/stmsi.v10i2.1238.