

KLASIFIKASI SAPI MADURA BERDASARKAN UKURAN TUBUH MENGGUNAKAN METODE *SUPPORT VECTOR MACHINE* DAN *K-NEAREST NEIGHBORS*

Rendi Agus Sucipto^{1*}, Bain Khusnul Khotimah²

^{1,2}Program Studi Teknik Informatika, Universitas Trunojoyo Madura

Received: 18 Desember 2024

Accepted: 14 Januari 2025

Published: 20 Januari 2025

Keywords:

Klasifikasi, KNN, sapi Madura, SVM.

Correspondent Email:

rendisucipto07@gmail.com

Abstrak. Sapi Madura memiliki peran penting dalam ekonomi dan budaya masyarakat, baik sebagai sapi potong maupun untuk tradisi kerapan dan sonok, yang memerlukan perawatan yang spesifik sejak dini. Metode gabungan *Support Vector Machine* (SVM) dan *K-Nearest Neighbors* (KNN) digunakan dalam penelitian ini untuk mengklasifikasikan sapi Madura berdasarkan ukuran tubuh. Dataset terdiri dari 450 data sapi yang didapat dari Dinas Ketahanan Pangan dan Pertanian Kabupaten Pamekasan, terbagi ke dalam tiga kelas, yakni kerapan, sonok, dan potong, dengan 150 sampel per kelas. Atribut yang diukur meliputi berat badan, tinggi pundak, lingkaran dada, tinggi pinggul, dan panjang badan. SVM digunakan untuk menangani data berdimensi tinggi, sedangkan KNN mengatasi kelemahan SVM dalam mengklasifikasikan data ambigu. *Hyperparameter tuning* dilakukan menggunakan *grid search* untuk meningkatkan kinerja model. Hasil penelitian menunjukkan akurasi mencapai 88,89%, *precision* 88,17%, *recall* 88,86%, dan *F1-score* 88,27%. Penggunaan SVM-KNN membuktikan adanya peningkatan performa dibandingkan hanya menggunakan metode SVM, sehingga dianggap lebih efektif dalam menentukan jenis sapi dan pemeliharaan yang sesuai.

Abstract. Madura cattle are important to the local community's economics and culture, serving as beef cattle and for traditional practices such as kerapan and sonok. These differing roles require specific care approaches from an early stage. This study classifies Madura cattle based on body measurements using *Support Vector Machine* (SVM) and *K-Nearest Neighbors* (KNN). The dataset comprises 450 cattle records obtained from the Dinas Ketahanan Pangan dan Peternakan Kabupaten Pamekasan, divided into three classes, kerapan, sonok, and beef cattle, with 150 samples per class. The measured attributes include body weight, shoulder height, hip height, body length, and chest girth. SVM was employed to handle high-dimensional data, while KNN addressed SVM's weakness in classifying ambiguous data. *Hyperparameter tuning* was performed using *grid search* to optimize model performance. The results showed an 88.89% accuracy, 88.17% precision, 88.86% recall, and 88.27% F1-score. The use of the SVM-KNN hybrid method demonstrated improved performance compared to SVM alone, proving to be more effective in determining cattle types and their appropriate care strategies.

1. PENDAHULUAN

Salah satu sumber utama daging di Indonesia adalah sapi. Sapi Madura adalah salah satu jenis sapi asli Indonesia yang harus

dilestarikan [1]. sapi Madura memiliki keunggulan dalam hal adaptasi terhadap iklim panas dan kondisi lingkungan dengan pakan berkualitas rendah, kebutuhan pakan yang lebih

sedikit, tahan terhadap serangan parasit, serta memiliki daging berkualitas baik[1]. Selain dimanfaatkan sebagai sapi potong dan tenaga kerja, masyarakat Madura juga menggunakan sapi Madura sebagai bagian dari tradisi sosial budaya, yaitu melalui kerapan sapi dan sapi sonok [2].

Adanya perbedaan pemanfaatan sapi Madura dalam tradisi kerapan dan sonok menimbulkan perlunya penanganan dan perawatan yang berbeda sejak usia muda. Penting bagi peternak untuk memiliki kemampuan dalam mengenali jenis bibit sapi agar dapat menentukan perawatan dan pemeliharaan yang sesuai [3]. Meskipun perbedaan jenis kelamin dapat memberikan petunjuk, dibutuhkan indikator tambahan yang dapat diukur secara objektif untuk mendukung penentuan jenis sapi [2]. Salah satu indikator penting yang mempengaruhi penampilan sapi, baik untuk kerapan maupun sonok, adalah ukuran tubuh sapi [1]. Klasifikasi berdasarkan ukuran tubuh dapat menjadi indikator awal menentukan kecocokan untuk kerapan atau sonok, atau cukup sebagai sapi potong saja. Sehingga memberikan informasi tambahan yang berguna untuk mengoptimalkan perawatan sesuai fungsi dan peran sapi sebagai sapi potong atau dapat diikuti ke dalam kedua tradisi tersebut, serta berpotensi meningkatkan performa dan kesehatan sapi secara keseluruhan [4].

Perkembangan teknologi komputer dan pembelajaran mesin menjadi sangat relevan. Teknologi telah memungkinkan penggunaan metode klasifikasi yang lebih efisien dan akurat, sehingga dapat membantu peternak dalam mengidentifikasi dan mengelompokkan jenis sapi Madura berdasarkan ukuran tubuh dan karakteristik lainnya. Proses klasifikasi merupakan tugas penting dalam pembelajaran mesin, di mana data dikelompokkan ke dalam kategori tertentu berdasarkan pola yang teridentifikasi [5].

Metode klasifikasi yang kerap digunakan dalam berbagai penelitian adalah *Support Vector Machine* (SVM) [6]. SVM merupakan salah satu model *supervised learning* yang sering digunakan dalam mengatasi masalah klasifikasi data karena kemampuannya dalam menangani data kompleks dan berdimensi tinggi [7]. SVM juga dapat mengeksekusi program lebih cepat [8]

dan efisien dalam penggunaan memori melalui teknik pemetaan kernel ke dalam ruang fitur berdimensi tinggi [7]. Dibandingkan dengan metode klasifikasi lain, SVM merupakan salah satu metode terbaik [8] dengan mencapai tingkat akurasi yang sangat tinggi [7]. Namun, metode SVM memiliki kelemahan dalam hal efisiensi komputasi ketika diaplikasikan pada dataset yang sangat besar. Selain itu, jika data terlalu terlalu dekat dengan *hyperplane*, terdapat kemungkinan terjadinya kesalahan dalam pengklasifikasian [9]. Di sisi lain, KNN menawarkan pendekatan lebih sederhana dan mudah diimplementasikan dengan menentukan kelas suatu titik data berdasarkan kesamaan tetangga terdekatnya [10]. Sehingga menggabungkan KNN dengan SVM dapat melengkapi kekurangan masing – masing metode dalam upaya meningkatkan akurasi.

Penelitian oleh Akbar dkk. pada tahun 2023 mengidentifikasi gagal ginjal kronis dengan mengkombinasikan metode SVM dan KNN. Dalam studi ini, mereka membandingkan akurasi metode gabungan SVM-KNN dengan masing – masing metode SVM dan KNN, serta melakukan berbagai percobaan pada parameter seperti *cost*, *tolerance*, *bias*, *k*, dan *miu*. Hasilnya, akurasi tertinggi diperoleh dari SVM-KNN, mencapai 94,25%, lebih baik daripada SVM yang mendapatkan 94,09% dan KNN dengan 91,73% [9]. Selain itu, pada tahun 2023, Ramadhan dkk. juga menggunakan kombinasi SVM-KNN untuk memilih pengguna SAKTI menurut kinerja pegawai di kementerian XYZ. *vector* yang paling dekat dengan *hyperplane* terbaik diidentifikasi menggunakan SVM, yang kemudian dimasukkan ke dalam algoritma KNN. Hasil kombinasi SVM-KNN mencapai akurasi 94%, lebih tinggi dibandingkan dengan akurasi SVM yang hanya 91% [11].

Berdasarkan penelitian terdahulu, kombinasi metode SVM-KNN menghasilkan performa yang bagus dalam meningkatkan akurasi dan efisiensi klasifikasi. SVM digunakan sebagai pengklasifikasi utama untuk menentukan *hyperplane* optimal. Data yang terletak dekat dengan *hyperplane* ini, yang memiliki nilai fungsi keputusan $f(x)$ di antara -1 dan 1 dianggap sebagai data yang ambigu dan memerlukan validasi lebih lanjut. Data ambigu ini disebut *support vectors* yang kemudian dijadikan sebagai sekumpulan data baru dan diklasifikasi ulang menggunakan algoritma

KNN. Hasil kombinasi SVM-KNN dianggap layak digunakan dalam berbagai data sehingga disarankan untuk melakukan penelitian lebih lanjut dengan mencoba berbagai variasi parameter guna mengevaluasi kinerja yang memberikan dampak terbaik.

Dengan mempertimbangkan pentingnya klasifikasi sapi Madura dalam mendukung keputusan perawatan dan pemeliharaan yang tepat, serta mengingat potensi teknologi pembelajaran mesin yang mendukung, penelitian ini bertujuan untuk mengklasifikasikan sapi Madura dengan menggunakan metode gabungan SVM dan KNN. Dengan memfokuskan KNN pada data yang sulit diklasifikasi oleh SVM, penelitian ini diharapkan mampu memperoleh model klasifikasi yang akurat dan dapat dipercaya, yang tidak hanya memberikan informasi yang lebih baik kepada peternak mengenai pemilihan bibit dan perawatan sapi, tetapi juga dapat berkontribusi pada pengembangan strategi pemeliharaan yang lebih efisien di sektor peternakan sapi Madura. Diharapkan penelitian ini dapat membantu peternak, peneliti, dan pemangku kepentingan dalam meningkatkan kesejahteraan dan produktivitas sapi Madura.

2. TINJAUAN PUSTAKA

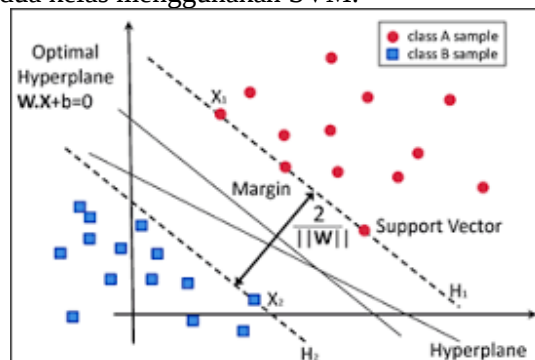
2.1 Sapi Madura

Sapi Madura merupakan salah satu jenis asli Indonesia yang banyak ditanakkan di Pulau Madura yang memiliki keunggulan dalam ketahanan terhadap cuaca ekstrem, pakan berkualitas rendah, serta serangan caplak [2]. Selain dimanfaatkan sebagai sapi potong, sapi Madura juga memiliki nilai sosial budaya, yakni digunakan dalam tradisi kerapan sapi dan kontes sapi sonok.

Kerapan sapi merupakan tradisi Madura yang mempertandingkan seberapa cepat sepasang sapi berlari [12]. Sapi kerapan melibatkan sapi pejantan unggul yang cepat dan gesit, serta memiliki ukuran tubuh yang sedikit berbeda dibandingkan sapi jantan Madura lainnya [2]. Berbeda dengan kerapan sapi, yang dipertandingkan dalam kontes sapi sonok adalah kecantikan sapi saat berpakaian dan berjalan [13]. Sapi sonok menonjolkan keindahan fisik, warna bulu, dan keterampilan sapi betina sebagai bentuk kesenian tradisional Madura [2].

2.2 Support Vector Machine

Support Vector Machine (SVM) merupakan metode yang handal dalam menangani masalah klasifikasi data [14]. SVM termasuk dalam kategori *supervised learning* yang berfungsi mengklasifikasikan data baru ke dalam ruang berdimensi N dengan memanfaatkan *hyperplane* sebagai pemisah [11]. Metode ini tidak hanya efektif untuk data yang terpisah secara linier, tetapi juga mampu menangani data yang bersifat non-linier. Tujuan utama dari SVM yakni untuk menentukan *hyperplane* ideal yang memisahkan dua kelas dengan cara memaksimalkan margin antar kelas [11]. Margin mengacu pada jarak *hyperplane* dengan titik data terdekat (*support vector*) dari masing – masing kelas [15]. Dalam menangani data multi kelas, beberapa SVM biner digunakan dan hasilnya dikombinasikan untuk mengidentifikasi kelas atau target akhir [16]. Berikut merupakan gambar pemisahan dua kelas menggunakan SVM.



Gambar 1. Support Vector Machine

Gambar 1 memperlihatkan pola data yang terbagi menjadi dua kelas. Simbol lingkaran merah mewakili kelas A atau $+1$, sedangkan simbol kotak biru mewakili kelas B atau -1 . Kedua kelas ini dipisahkan oleh sebuah garis solid yang disebut *hyperplane*, dengan jarak antar kelas diukur sebagai margin [17]. Data – data terdekat yang terletak tepat pada garis putus – putus (H_1 dan H_2) dikenal sebagai *support vector*. Garis H_1 bersinggungan dengan data dari kelas $+1$, sedangkan garis H_2 bersinggungan dengan data dari kelas -1 [17]. Berikut merupakan persamaan untuk H_1 , H_2 , dan *optimal hyperplane* [17].

$$w \cdot X_i + b \geq +1 \quad \text{untuk } y_i = 1 \quad (H_1) \quad (1)$$

$$w \cdot X_i + b \leq -1 \quad \text{untuk } y_i = -1 \quad (H_2) \quad (2)$$

$$w \cdot X + b = 0 \quad \text{untuk hyperplane} \quad (3)$$

Keterangan :

- X_i = nilai atribut data ke - i
 w = nilai bobot vektor
 b = nilai bias
 y_i = kelas data ke - i

Vijayakumar dan Wu (1999) dalam penelitian [18] memperkenalkan metode pendekatan *sequential learning* untuk mengidentifikasi garis *hyperplane* optimal yang dapat memisahkan dua kelas data. Berikut merupakan langkah - langkah SVM dengan menggunakan pendekatan *sequential learning* [18].

1. Menginisialisasi parameter awal untuk SVM yang mencakup nilai C , nilai awal α_0 , ε (epsilon) λ (lambda), γ (gamma), dan iterasi maksimum.
2. Menghitung nilai kernel berdasarkan persamaan kernel yang digunakan.
3. Melakukan perhitungan *Matriks Hessian* dengan memanfaatkan persamaan berikut.

$$D_{i,j} = y_i y_j (K(x_i, x_j) + \lambda^2) \quad (4)$$

4. Mencari nilai *support vector*. Dapat dilakukan menggunakan persamaan - persamaan berikut.

- a. Menghitung nilai *error*

$$E_i = \sum_{j=1}^n \alpha_j D_{ij} \quad (5)$$

- b. Menghitung *delta apha*

$$\delta \alpha_i = \min \{ \max[\gamma(1 - E_i), -\alpha_i], C - \alpha_i \} \quad (6)$$

- c. Menghitung *alpha baru*

$$\alpha_i = \alpha_i + \delta \alpha_i \quad (7)$$

5. Mengulangi langkah 4 hingga mencapai iterasi maksimum atau hingga $\max(|\delta \alpha_i|) < \varepsilon$, yakni ketika konvergensi tercapai atau perubahan sudah tidak signifikan.
6. Setelah memperoleh nilai *support vector*, berikutnya menghitung nilai bias dengan persamaan berikut.

$$b = -\frac{1}{2} \left(\sum_{i=0}^n \alpha_i y_i K(x_i, x^-) + \sum_{i=0}^n \alpha_i y_i K(x_i, x^+) \right) \quad (8)$$

7. Menghitung nilai fungsi $f(x)$ menggunakan persamaan berikut.

$$f(x) = \sum_{i=0}^n \alpha_i y_i K(x_i, x) + b \quad (9)$$

8. Menentukan kelas. Jika $f(x) > 0$, maka akan dilabeli +1, dan jika $f(x) < 0$ maka diberi label -1.

2.3 K-Nearest Neighbor

K - Nearest Neighbors (KNN) merupakan salah satu metode *supervised learning* yang dapat digunakan untuk berbagai tujuan, seperti klasifikasi dan penambangan data [14]. Metode klasifikasi ini menggunakan perkiraan statistik untuk menemukan solusi yang efisien [9]. Algoritma ini menentukan kelas suatu titik data dengan mempertimbangkan kelas dari K tetangga terdekatnya [19]. Jarak antara kedua titik untuk mendapatkan titik terdekat dapat dihitung dengan berbagai pendekatan pengukuran jarak yang tersedia, seperti *Euclidean*, *Minkowski*, atau *Manhattan* [20].

Adapun tahapan algoritma KNN dijabarkan sebagai berikut [11].

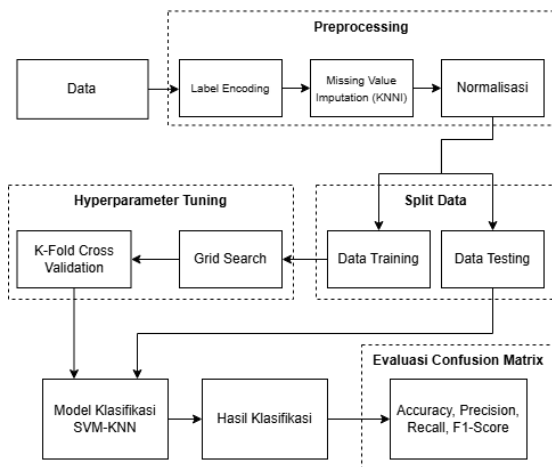
1. Menentukan jumlah tetangga terdekat yakni parameter K .
2. Menghitung jarak data baru dari tiap titik objek sampel data yang tersedia.
3. Mengurutkan objek - objek berdasarkan nilai jarak terkecil.
4. Kumpulkan kelas k *nearest neighbors*.
5. Menggunakan kelas k *nearest neighbors* mayoritas sehingga nilai kelas data uji baru dapat diprediksi.

Dalam metode KNN, penentuan parameter K (jumlah tetangga terdekat) harus dilakukan sebelum mengklasifikasikan data uji baru. Perlu adanya kecermatan dalam penentuan nilai K dengan mempertimbangkan jumlah data dan dimensi yang terbentuk [11]. Semakin besar jumlah data yang tersedia, sebaiknya semakin rendah nilai K . Namun, jika dimensi data semakin besar, maka nilai K juga perlu ditingkatkan [11]. Meskipun perhitungan KNN sangat kuat dan mudah diimplementasikan, metode ini juga memiliki kelemahan, terutama dalam menangani data yang berdimensi tinggi dan berskala besar, serta memerlukan waktu komputasi yang cukup tinggi [14].

3. METODE PENELITIAN

3.1 Arsitektur Sistem

Dalam proses perancangan sistem, terdapat beberapa tahapan yang akan dilakukan. Tahapan - tahapan tersebut meliputi pengumpulan data, *preprocessing* data, *split* data, *hyperparameter tuning*, pembuatan dan pelatihan model, serta evaluasi model. Tahapan ini ditunjukkan pada Gambar 2 berikut.



Gambar 2. Arsitektur Sistem

3.2 Pengumpulan Data

Data yang digunakan berupa ukuran tubuh sapi Madura jantan dan betina yang didapat dari Dinas Ketahanan Pangan dan Pertanian Kabupaten Pamekasan. Data berjumlah 450 dengan 5 atribut ukuran tubuh sapi dan 3 kelas sapi, 150 data sapi kerapan, 150 data sapi sonok, dan 150 data sapi potong. Bagian – bagian tubuh sapi yang diukur meliputi tinggi pinggul, lingkaran dada, berat badan, tinggi gumba, dan panjang badan [21].

3.3 Preprocessing Data

Sebelum proses klasifikasi, tahap *preprocessing* data digunakan dalam mengatasi data yang bermasalah karena dapat mengganggu proses klasifikasi [17]. Tahapan *preprocessing* data yang dilakukan adalah *label encoding*, penanganan *missing value* dan normalisasi data. *Label encoding* merupakan proses mengubah data kategorikal menjadi nilai numerik sehingga semua data memiliki tipe data yang sama dan dapat diproses oleh model klasifikasi [22]. Langkah berikutnya yang dilakukan yakni mengatasi data yang memiliki nilai hilang atau *missing value*. Nilai rata – rata untuk kelas dan atribut yang sama dapat digunakan untuk mengatasi masalah ini [17]. Setelah mengatasi *missing value*, dilakukan proses normalisasi data yakni mengubah rentang data sehingga memiliki skala yang seragam antara 0 hingga 1 [17]. Hal ini bertujuan untuk meningkatkan akurasi serta dapat mengurangi waktu proses komputasi [9].

3.4 Pembagian Data

Data dipisah menjadi dua bagian. Data latih digunakan dalam melatih model untuk memahami pola dalam data melalui *grid search*

dan *k-fold cross validation*. Sedangkan data uji digunakan dalam mengukur kinerja model yang telah melalui proses pelatihan. Pembagian data dilakukan dalam rasio 8:2, di mana 80% merupakan data latih dan 20% untuk data uji [11].

3.5 Grid Search Hyperparameter Tuning

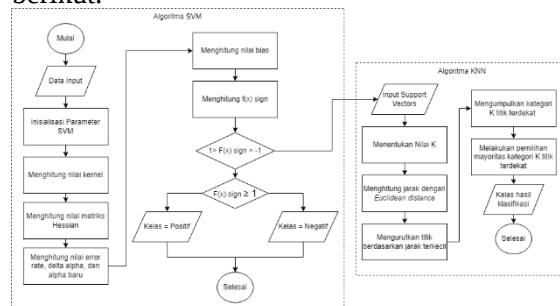
Grid Search merupakan metode untuk menemukan kombinasi *hyperparameter* yang paling optimal bagi suatu model [23]. *Hyperparameter* adalah parameter yang tidak dapat diperoleh langsung dari data *training* dan perlu ditetapkan sebelumnya [24]. Parameter yang akan ditentukan meliputi jenis kernel, nilai *gamma*, dan *C* untuk SVM dan jumlah tetangga terdekat (*K*) dan metode jarak antar data pada KNN.

3.6 K-Fold Cross Validation

Setiap nilai dari masing – masing parameter akan dikombinasikan oleh *Grid Search*, model tersebut kemudian divalidasi menggunakan *k-fold cross validation* untuk meninjau kinerjanya dan memutuskan kombinasi model terbaik. Data latih akan dibagi lagi dengan nilai *k* sebesar 5 menjadi data latih dan data validasi, sehingga data akan disebar menjadi lima bagian atau *fold*.

3.7 Klasifikasi SVM-KNN

Pada tahap ini akan dibangun model klasifikasi dengan menggabungkan algoritma metode SVM dan algoritma metode KNN. Algoritma metode SVM-KNN adalah sebagai berikut.



Gambar 3. Algoritma SVM-KNN

Penggabungan terjadi ketika nilai $F(x)$ yang dioutputkan oleh SVM berada di antara -1 hingga 1, dan akan dilanjutkan menggunakan metode KNN.

3.8 Evaluasi

Pada penelitian ini menggunakan *confusion matrix* untuk tahap evaluasi. Jumlah prediksi yang akurat, kesalahan prediksi, dan perbandingan hasil klasifikasi sistem dengan

kondisi yang sebenarnya disertakan dalam informasi data *Confusion Matrix*[17]. Perhitungan akurasi menggunakan empat istilah untuk menggabungkan nilai prediksi dan nilai aktual, yakni *True Positive* (TP), *False Positive* (FP), *True Negative* (TN), dan *False Negative* (FN). Evaluasi dilakukan untuk mengetahui nilai *accuracy*, *recall*, *precision*, dan *F1-score* yang dapat menunjukkan performa kinerja model yang telah dibangun. Berikut merupakan persamaan untuk menghitung keempat metrik tersebut [25].

$$Accuracy = \frac{(TP + TN)}{(TP + FP + FN + TN)} \quad (10)$$

$$Precision = \frac{TP}{(TP + FP)} \quad (11)$$

$$Recall = \frac{TP}{(TP + FN)} \quad (12)$$

$$F1 - Score = 2 * \frac{(Recall * Precision)}{(Recall + Precision)} \quad (13)$$

4. HASIL DAN PEMBAHASAN

4.1. Preprocessing Data

Setelah data dibaca oleh Python, dilakukan tahapan *preprocessing* data sebagai berikut.

1. Label Encoding

Pada tahap ini mengubah kategori kelas dalam data menjadi bentuk numerik, di mana setiap kelas diwakili oleh sebuah angka. Kelas 'Karapan' berubah menjadi 0, 'Potong' menjadi 1, dan 'Sonok' menjadi 2 seperti pada gambar berikut.

	Lingkar Dada	Panjang Badan	Tinggi Gumba	Tinggi Hip	Berat Badan	Kelas
8	126.838	97.224	115.064	NaN	113.597	0
16	175.677	102.487	98.862	NaN	111.092	0
21	182.030	91.897	109.762	NaN	127.522	0
33	146.140	99.312	112.007	NaN	109.734	0
39	188.181	99.981	104.904	NaN	125.933	0
...	...	...	...	...	...	...
436	149.433	124.848	120.664	123.117	NaN	1
437	168.910	120.388	129.921	NaN	272.102	1
440	156.581	119.884	NaN	119.735	246.090	1
446	157.324	114.484	NaN	125.291	287.680	1
447	149.954	110.997	NaN	116.577	294.085	1

Gambar 4. Data hasil *label encoding*

2. Penanganan Missing Value

Pada tahap ini dilakukan pengecekan data dengan nilai yang kosong (*missing value*). Dari pengecekan yang telah dilakukan, membuktikan bahwa adanya *missing value* pada data yang dibuktikan dengan "NaN" pada gambar, sehingga dilakukan penanganan dengan mengisi nilai yang kosong dengan nilai rata – rata tiap fitur pada masing – masing kelas.

	Lingkar Dada	Panjang Badan	Tinggi Gumba	Tinggi Hip	Berat Badan	Kelas
0	185.157	105.971	106.935000	104.472	98.354	0
1	137.561	93.726	118.225000	95.398	108.822	0
2	165.493	100.537	116.851000	90.401	97.938	0
3	187.329	94.834	104.060000	93.886	105.188	0
4	189.981	91.559	102.617000	102.787	114.142	0
...	...	...	...	...	...	...
445	151.885	114.653	125.906000	122.430	256.156	1
446	157.324	114.484	113.459866	125.291	287.680	1
447	149.954	110.997	113.459866	116.577	294.085	1
448	147.724	123.710	117.409000	121.502	229.084	1
449	156.902	112.527	116.855000	127.622	225.566	1

Gambar 5. Data hasil penanganan *missing value*

3. Normalisasi Data

Setelah menangani *missing value*, dilakukan normalisasi data untuk menyamakan rentang data dari 0 hingga 1 pada tiap fitur menggunakan *Min-max normalization*.

	Lingkar Dada	Panjang Badan	Tinggi Gumba	Tinggi Hip	Berat Badan
0	0.558085	0.416705	0.296638	0.359382	0.033446
1	0.226181	0.169835	0.580985	0.131226	0.075385
2	0.420961	0.307151	0.546380	0.005582	0.031779
3	0.573231	0.192174	0.224229	0.093209	0.060826
4	0.591724	0.126147	0.187886	0.317015	0.096699
...	...	...	...	...	...
445	0.326067	0.591742	0.774436	0.810918	0.665670
446	0.363995	0.588335	0.460971	0.882854	0.791969
447	0.312602	0.518034	0.460971	0.663750	0.817631
448	0.297051	0.774339	0.560433	0.787584	0.557208
449	0.361052	0.548880	0.546480	0.941465	0.543113

Gambar 6. Data hasil normalisasi

4.2. Pembuatan Model

Pada tahap ini akan dibangun model klasifikasi dengan menggabungkan algoritma metode SVM dan algoritma metode KNN. Model dilatih menggunakan tiap kombinasi parameter yang telah dibentuk *Grid Search* dan mengukur akurasi tiap fold untuk memperoleh kombinasi parameter yang menghasilkan model terbaik. Berikut merupakan tabel nilai tiap *hyperparameter* pada penelitian ini.

Tabel 1. Nilai *hyperparameter*

Metode	Parameter	Nilai
SVM	Kernel	[linear, RBF, polynomial, sigmoid]
	C	[1, 0.1, 0.0001]
	Gamma	[0.1, 0.001, 0.0001]
KNN	Jarak	[Euclidean, Manhattan, Minkowski]
	K	[3, 5, 7]

Dari proses yang telah dilakukan dengan nilai *k-fold* = 5, diperoleh akurasi *fold* terbaik sebesar 93,05% pada kombinasi parameter sebagai berikut.

Tabel 2. Kombinasi *hyperparameter* terbaik

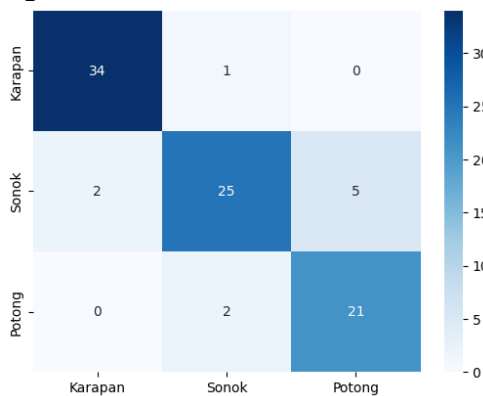
Metode	Parameter	Nilai	Akurasi
--------	-----------	-------	---------

SVM	Kernel	RBF	93,05%
	C	1	
	<i>Gamma</i>	0.1	
KNN	Jarak	7	
	K	Euclidean	

Parameter ini kemudian digunakan untuk membangun model optimal dan diuji dengan data uji, data yang belum pernah digunakan pada proses pelatihan dengan *K-fold Cross Validation* sebelumnya.

4.3. Evaluasi

Dari hasil pengujian model SVM-KNN menggunakan kombinasi parameter terbaik pada data uji menghasilkan *confusion matrix* sebagai berikut.



Gambar 7. Hasil *Confusion Matrix*

Pada gambar, sumbu Y (vertikal) menunjukkan kelas data aktual dan sumbu X menunjukkan kelas data hasil prediksi. Dari 35 data pada Karapan, 34 data diprediksi benar Karapan dan 1 data salah diprediksi sebagai Sonok. Kemudian pada 32 data kelas Sonok, berhasil diprediksi benar sebagai Sonok sebanyak 25 data, 2 data salah diprediksi sebagai Karapan dan 5 data salah diprediksi sebagai Potong. Sedangkan pada 23 data kelas Potong, 21 data diprediksi benar sebagai Potong tetapi 2 data salah diprediksi sebagai Sonok. Berdasarkan perhitungan *confusion matrix* ini, diperoleh nilai akurasi, *precision*, *recall*, dan *F1-score*. Untuk menunjukkan hasil evaluasi metode usulan SVM-KNN mengalami peningkatan, maka akan dibandingkan dengan hasil evaluasi metode SVM berdasarkan nilai akurasi, *recall*, *precision*, dan *F1-score* dengan menggunakan parameter SVM yang telah didapatkan dari kombinasi *hyperparameter* terbaik. Perbandingan kinerja metode ini ditampilkan dalam tabel berikut.

Tabel 3. Perbandingan hasil kinerja model

Kinerja	SVM	SVM-KNN (metode usulan)
Akurasi	76.67%	88.89%
<i>Precision</i>	77.39%	88.17%
<i>Recall</i>	77.12%	88.86%
<i>F1-Score</i>	76.55%	88.27%

Dari hasil pengujian, metode usulan SVM-KNN memperoleh performa yang baik dengan nilai akurasi 88.89%, 88.17% *precision*, 88.86% *recall*, dan 88.27% *F1-score*. Nilai pada keempat matriks evaluasi tersebut terbukti lebih baik dibandingkan metode SVM dengan 76.67% akurasi, 77.39% *precision*, 77.12% *recall*, dan 76.55% *F1-score*.

5. KESIMPULAN

Hasil penelitian ini menunjukkan bahwa:

- Penggabungan metode SVM dengan KNN memperoleh hasil yang efektif dalam klasifikasi sapi Madura berdasarkan ukuran tubuh, dengan tingkat akurasi mencapai 88.89%, 88.17% *precision*, 88.86% *recall*, dan 88.27% *F1-Score*. Nilai – nilai evaluasi tersebut mengalami peningkatan yang signifikan ketika dibandingkan dengan hanya menggunakan metode SVM tunggal dengan 76.67% akurasi, 77.39% *precision*, 77.12% *recall*, dan 76.55% *F1-Score*.
- Penggunaan *grid search* dan *k-fold cross validation* juga sangat berpengaruh dalam mendapatkan kombinasi *hyperparameter* paling optimal. Akurasi *fold* terbaik mencapai 93.05% menggunakan kombinasi kernel RBF, C = 1, *gamma* = 0.1 pada parameter SVM serta nilai K = 7 dan metrik *Euclidean* pada parameter KNN.

UCAPAN TERIMA KASIH

Penulis menyampaikan rasa syukur kehadiran Allah SWT dan Nabi Muhammad SAW atas segala nikmat-Nya sehingga penulis dapat menyelesaikan penelitian ini dengan baik. Selain itu, Penulis ingin mengucapkan terima kasih kepada Dinas Ketahanan Pangan dan Pertanian Kabupaten Pamekasan yang telah menyediakan data untuk mendukung penelitian ini.

DAFTAR PUSTAKA

- [1] D. Herviyanto and Kuswati, "Karakteristik Sapi Madura Betina Yang Digunakan Dalam Kontes Sapi Sonok," *TERNAK TROPIKA Journal of Tropical Animal Production*, vol. 25, no. 1, pp. 65–72, Jun. 2024, doi: 10.21776/ub.jtapro.2024.025.01.8.
- [2] S. Mubarak Z, S. Maylinda, and G. Ciptadi, "Korelasi dan Regresi Bobot Badan dan Morfometrik Tubuh Sapi Madura Kerapan Pada Kelompok Umur Berbeda," *TERNAK TROPIKA Journal of Tropical Animal Production*, vol. 25, no. 1, pp. 51–58, Jun. 2024, doi: 10.21776/ub.jtapro.2024.025.01.6.
- [3] S. F. A. Wijaya, K. Usman, and S. Saidah, "Analisis Perbandingan K-Nearest Neighbor dan Support Vector Machine pada Klasifikasi Jenis Sapi dengan Metode Gray Level Coocurrence Matrix," *Jurnal Ilmu Komputer dan Informatika (JIKI)*, vol. 2, no. 2, pp. 93–102, Mar. 2022, doi: 10.54082/jiki.27.
- [4] Kuswati, W. A. Septian, T. Susilawati, and D. Herviyanto, "Karakteristik Morfometrik Sapi Madura Betina Tipe Sonok," *JITRO (Jurnal Ilmu dan Teknologi Peternakan Tropis)*, vol. 9, no. 1, pp. 149–157, Jan. 2022, doi: 10.33772/jitro.v9i1.10691.
- [5] H. N. Irmanda and R. Astriratma, "Klasifikasi Jenis Pantun dengan Metode Support Vector Machines (SVM)," *Jurnal RESTI (Rekayasa Sistem dan Teknologi Informasi)*, vol. 1, no. 5, pp. 915–922, 2020.
- [6] A. Nurkholis, D. Alita, and A. Munandar, "Comparison of Kernel Support Vector Machine Multi-Class in PPKM Sentiment Analysis on Twitter," *Jurnal RESTI (Rekayasa Sistem dan Teknologi Informasi)*, vol. 6, no. 2, pp. 227–233, Apr. 2022, doi: 10.29207/resti.v6i2.3906.
- [7] S. W. Iriananda, R. W. Budiawan, A. Y. Rahman, and Istiadi, "Optimasi Klasifikasi Sentimen Komentar Pengguna Game Bergerak Menggunakan Svm, Grid Search Dan Kombinasi N-Gram," *Jurnal Teknologi Informasi dan Ilmu Komputer*, vol. 11, no. 4, pp. 743–752, Aug. 2024, doi: 10.25126/jtiik.1148244.
- [8] D. P. H. Putri, N. P. N. P. Dewi, I. K. Purnamawan, and N. W. Marti, "Perbandingan Performansi Support Vector Machine (Svm) dan Backpropagation untuk Klasifikasi Studi Mahasiswa Undiksha," *JEPIN (Jurnal Edukasi dan Penelitian Informatika)*, vol. 9, no. 3, pp. 492–501, 2023.
- [9] A. T. Akbar, N. Yudistira, and A. Ridok, "Identifikasi Gagal Ginjal Kronis Dengan Mengimplementasikan Metode Support Vector Machine Beserta K-Nearest Neighbour (SVM-KNN)," *Jurnal Teknologi Informasi dan Ilmu Komputer (JTIK)*, vol. 10, no. 2, pp. 301–308, 2023, doi: 10.25126/jtiik.2023106059.
- [10] S. Wardani, Sawaluddin, and P. Sihombing, "Hybrid of Support Vector Machine Algorithm and K-Nearest Neighbor Algorithm to Optimize the Diagnosis of Eye Disease," in *MECnIT 2020 - International Conference on Mechanical, Electronics, Computer, and Industrial Technology*, Institute of Electrical and Electronics Engineers Inc., Jun. 2020, pp. 321–326. doi: 10.1109/MECnIT48290.2020.9166599.
- [11] S. Ramadhan, Kusrini, and Kusnawi, "Penerapan Kombinasi Algoritma SVM-KNN dalam seleksi User SAKTI berdasarkan Hasil Kinerja Pegawai pada Kementerian XYZ," *Jurnal Teknologi Informatika dan Komputer*, vol. 9, no. 2, pp. 630–644, Sep. 2023, doi: 10.37012/jtik.v9i2.1672.
- [12] M. B. Ainun, "Metafor Kerapan Sapi dalam Tata Kelola Perusahaan dari Sudut Pandang Teori Stewardship," *Pamator Journal*, vol. 14, no. 2, pp. 95–100, Nov. 2021, doi: 10.21107/pamator.v14i2.10801.
- [13] R. Kurniawan and L. A. Setiyaningsih, "The Development of Halal Tourism Destinations Model Based on Art Culture Performers of Sapi Sonok Madura in Pamekasan Regency," *Profetik Jurnal Komunikasi*, vol. 16, no. 2, pp. 400–418, Oct. 2023, doi: 10.14421/pjk.v16i2.2745.
- [14] Y. K. Zamil, S. A. Ali, and M. A. Naser, "Spam image email filtering using K-NN and SVM," *International Journal of Electrical and Computer Engineering*, vol. 9, no. 1, pp. 245–254, Feb. 2019, doi: 10.11591/ijece.v9i1.pp245-254.
- [15] S. L. B. Ginting and A. A. Permana, "Hybrid Classifier System: Support Vector Machines Dikombinasikan dengan K-Nearest Neighbors untuk Menentukan Kelayakan Nasabah Bank dalam Pengajuan Kredit," *Komputika : Jurnal Sistem Komputer*, vol. 7, no. 1, pp. 9–16, Apr. 2018, doi: 10.34010/komputika.v7i1.1402.
- [16] Irma, M. Muchtar, R. Adawiyah, and Sarimuddin, "Klasifikasi Tingkat Kematangan Cabai Merah Keriting Menggunakan SVM Multiclass Berdasarkan Ekstraksi Fitur Warna," *Jurnal Informatika dan Teknik Elektro Terapan*, vol. 12, no. 3, Aug. 2024, doi: 10.23960/jitet.v12i3.4430.
- [17] M. D. Purbolaksono, M. Irvan Tantowi, A. Imam Hidayat, and A. Adiwijaya, "Perbandingan Support Vector Machine dan Modified Balanced Random Forest dalam Deteksi Pasien Penyakit Diabetes," *Jurnal*

- RESTI (Rekayasa Sistem dan Teknologi Informasi)*, vol. 5, no. 2, pp. 393–399, Apr. 2021, doi: 10.29207/resti.v5i2.3008.
- [18] S. R. Yustihan, P. P. Adikara, and Indriati, “Analisis Sentimen berbasis Aspek terhadap Data Ulasan Rumah Makan menggunakan Metode Support Vector Machine (SVM),” *Jurnal Pengembangan Teknologi Informasi dan Ilmu Komputer*, vol. 5, no. 3, pp. 1017–1023, Mar. 2021, [Online]. Available: <http://j-ptiik.ub.ac.id>
- [19] G. H. Zghair and D. S. Al-Azzawi, “Comparing emotion classification: machine learning algorithms and hybrid model with support vector machines,” *IAES International Journal of Artificial Intelligence (IJ-AI)*, vol. 13, no. 3, pp. 3671–3685, Sep. 2024, doi: 10.11591/ijai.v13.i3.pp3671-3685.
- [20] L. A. Susanto, “Komparasi Model Support Vector Machine dan K-Nearest Neighbor Pada Analisis Sentimen Aplikasi POLRI SUPER APP,” *Jurnal Informatika dan Teknik Elektro Terapan*, vol. 12, no. 2, Apr. 2024, doi: 10.23960/jitet.v12i2.4152.
- [21] Z. Shamad, C. S. Widyandana, and V. M. A. Nurgiantiningsih, “Korelasi Ukuran Tubuh dengan Bobot Karkas Sapi Madura di Pamekasan,” *Jurnal Ilmu Peternakan Terapan*, vol. 6, no. 2, pp. 72–81, Mar. 2023, doi: 10.25047/jipt.v6i2.3307.
- [22] F. Hamami and I. A. Dahlan, “Klasifikasi Cuaca Provinsi DKI Jakarta Menggunakan Algoritma Random Forest Dengan Teknik Oversampling,” *Jurnal TEKNOINFO*, vol. 16, no. 1, pp. 87–92, 2022.
- [23] Sulistiana and M. A. Muslim, “Support Vector Machine (SVM) Optimization Using Grid Search and Unigram to Improve E-Commerce Review Accuracy,” *Journal of Soft Computing Exploration*, vol. 1, no. 1, pp. 8–15, Sep. 2020.
- [24] N. A. Yattoo, I. S. Ali, and I. Mirza, “Comparing Hyperparameter Optimized Support Vector Machine, Multi-layer Perceptron and Bagging Classifiers for Diabetes Mellitus Prediction,” *International Journal of Electrical and Computer Engineering*, vol. 14, no. 5, pp. 5834–5847, Oct. 2024, doi: 10.11591/ijece.v14i5.pp5834-5847.
- [25] A. D. Putra, D. Nurani, M. M. Dewi, A. N. Rahmi, and Supriatin, “Supervised Machine Learning Model untuk Prediksi Penyakit Hepatitis,” *Indonesian Journal of Computer Science*, vol. 13, no. 2, pp. 3329–3341, 2024, [Online]. Available: <https://archive.ics.uci.edu/ml/datasets/Hepatitis>