

Klasifikasi Kondisi Pergerakan Palet Menggunakan Metode K-Nearest Neighbors

Winona Kaulika Putri Arda¹, Fauzi Imaduddin Adhim², Faiza Alif Fakhрина^{*3}

Email: ¹wilonakpa@gmail.com, ²adhim.fi@its.ac.id, ³alif.fakhрина@its.ac.id

^{*}Faiza Alif Fakhрина

Abstrak

Tingginya permintaan produk menuntut proses distribusi berjalan efisien dan minim hambatan. Palet merupakan material penting yang berperan dalam mendukung proses tersebut. Namun kondisi operasional yang tidak stabil sering menyebabkan posisi palet tidak sesuai dengan jalur konveyor sehingga terjadi gangguan pergerakan palet yang berdampak pada penurunan efisiensi dan peningkatan waktu henti produksi. Maka dari itu diperlukan adanya penelitian dengan mengembangkan sistem *monitoring* dan klasifikasi kondisi pergerakan palet menggunakan metode K-Nearest Neighbors (KNN). Sistem menggunakan sensor ultrasonik HC-SR04 untuk mengukur jarak sisi kiri dan kanan palet serta sensor proximity E18-D80NK untuk mendeteksi keberadaan palet. Data dikirim ke mikrokontroler ESP32, disimpan dalam database dan diproses menggunakan algoritma KNN untuk mengklasifikasikan kondisi palet menjadi tiga kelas yaitu normal, miring dan berhenti. Hasil pembacaan sensor, klasifikasi, dan peringatan akan ditampilkan melalui dashboard pemantauan. Hasil pengujian menunjukkan bahwa model KNN dengan $k=3$ dengan proporsi pembagian dataset 80:20 mampu mengklasifikasikan kondisi pergerakan palet dengan akurasi sebesar 97,68%, precision 97,21%, recall 97,07% dan F1-score 97,14%. Sistem ini juga mampu menampilkan hasil pembacaan sensor, klasifikasi dan memberikan peringatan apabila terdapat kondisi palet yang tidak normal pada dashboard pemantauan.

Kata kunci: konveyor, knn, palet, sistem pemantauan

Abstract

The high demand for products requires the distribution process to operate efficiently with minimal disruptions. Pallets are essential materials that play an important role in supporting this process. However, unstable operational conditions often cause the pallet position to deviate from the conveyor path, resulting in pallet movement disturbances that impact decreased efficiency and increased production downtime. Therefore, research is needed to develop a monitoring and classification system for pallet movement conditions using the K-Nearest Neighbors (KNN) method. The system utilizes HC-SR04 ultrasonic sensors to measure the distance on the left and right sides of the pallet, as well as an E18-D80NK proximity sensor to detect the presence of pallets. The data are transmitted to an ESP32 microcontroller, stored in a database, and processed using the KNN algorithm to classify pallet conditions into three categories: normal, tilted, and stopped. The sensor readings, classification results, and warning notifications are displayed through a monitoring dashboard. The testing results show that the KNN model with $k=3$ and an 80:20 dataset split ratio is capable of classifying pallet movement conditions with an accuracy of 97.68%, precision of 97.21%, recall of 97.07%, and an F1-score of 97.14%. The system is also able to display sensor readings, classification results, and provide warnings when abnormal pallet conditions are detected on the monitoring dashboard.

Keywords: conveyor, knn, pallet, monitoring system

I. PENDAHULUAN

Dalam industri, gudang berperan penting sebagai tempat penyimpanan, pengelolaan dan distribusi produk. Efisiensi operasional gudang sangat bergantung pada sistem penanganan produk yang baik. Sebelum produk memasuki area gudang penyimpanan, produk terlebih dahulu melalui proses *palletizing*. Pallet merupakan salah satu material penting yang terdapat pada proses *palletizing*. Proses ini penting dilakukan tidak hanya untuk memastikan produk tersusun dengan baik namun juga untuk memudahkan dalam penanganan produk, mempercepat proses distribusi dan efisiensi operasional pergudangan. Semakin bertambahnya permintaan dan efisiensi produk, otomatisasi proses *palletizing* dapat menjadi sebuah solusi. Integrasi sistem *palletizing* otomatis memungkinkan pemantauan dan pengendalian proses secara real-time sehingga dapat meningkatkan transparansi dalam proses. Otomatisasi proses *palletizing* meliputi distribusi pallet secara otomatis menjadikan proses ini memungkinkan terdapat kendala pada orientasi posisi dari distribusi pallet. Kendala dalam orientasi posisi pallet dapat menyebabkan gangguan operasional yang signifikan seperti meningkatnya waktu henti tambahan serta adanya tambahan biaya perawatan [1]. Untuk

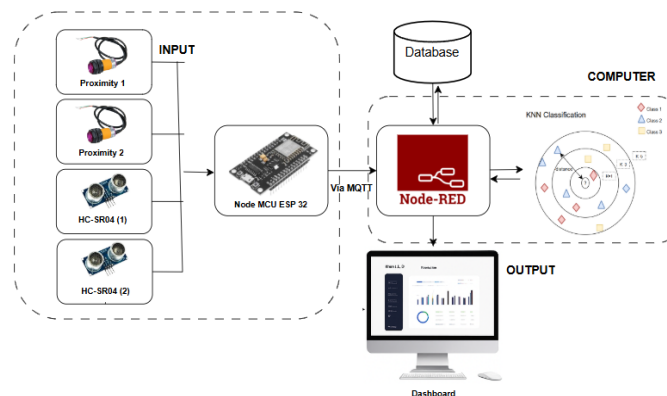
mengatasi hal tersebut, dibutuhkan sebuah sistem yang mampu mendeteksi dan memantau kondisi pergerakan distribusi pallet secara real-time saat proses *palletizing* berlangsung.

Sistem pemantauan ini diintegrasikan melalui pemasangan sensor pada area konveyor untuk membaca parameter tertentu selama proses *palletizing* berlangsung. Data yang dikumpulkan dari sensor kemudian digunakan sebagai input dalam sistem klasifikasi kondisi pergerakan palet secara real-time. Salah satu metode klasifikasi yang dapat diterapkan dalam sistem ini adalah K-Nearest Neighbors (KNN), algoritma *supervised learning* yang bekerja dengan menghitung jarak antara data uji dengan data pelatihan, lalu menentukan kelas berdasarkan mayoritas dari tetangga terdekatnya [2]. Dalam konteks ini, KNN digunakan untuk mengklasifikasikan data hasil pembacaan sensor menjadi beberapa kondisi pergerakan palet, seperti normal, miring, atau berhenti, yang merepresentasikan kestabilan palet di atas konveyor saat proses distribusi berlangsung. Terdapat tiga faktor yang memengaruhi performa algoritma ini, antara lain kualitas data pelatihan, metode pengukuran jarak, dan pemilihan nilai parameter K yang optimal [3].

Berbagai penelitian sebelumnya telah mengkaji penerapan algoritma *machine learning* dalam sistem konveyor, seperti klasifikasi produk, deteksi objek, dan pemantauan kondisi alat. Seperti [4] menggunakan metode polar KNN untuk mendeteksi kerusakan roller pada konveyor menggunakan sinyal audio, sementara [4] menerapkan KNN dalam pemantauan *real-time* terhadap kerusakan sabuk konveyor. Sebagian besar pendekatan tersebut belum mempertimbangkan klasifikasi untuk orientasi pallet secara real-time berdasarkan data pembacaan sensor saat proses *palletizing* berlangsung. Ketidakesesuaian orientasi pallet dapat berdampak terhadap efisiensi proses dan menyebabkan downtime operasional yang tidak diinginkan. Oleh karena itu, penelitian ini bertujuan untuk merancang sistem klasifikasi kondisi pergerakan palet menggunakan algoritma KNN yang dapat memberikan kondisi terkait orientasi palet saat proses distribusi berlangsung. Dengan adanya sistem ini, diharapkan proses identifikasi gangguan orientasi palet dapat dilakukan secara otomatis untuk mengurangi potensi waktu henti dan meningkatkan efisiensi operasional dalam proses *palletizing* produk.

II. METODE PENELITIAN

2.1. Sistem Arsitektur



Gambar 1. Arsitektur Sistem

Dalam *input* terdiri dari empat komponen diantaranya dua sensor *proximity* E18D80NK yang digunakan sebagai *trigger* apabila terdapat sebuah palet melintasi konveyor serta dua sensor ultrasonik HC-SR04 untuk menghitung jarak pada sisi yang berbeda serta waktu tempuh selama palet melintasi konveyor. Pembacaan sensor diteruskan ke mikrokontroler ESP32 untuk kemudian dikirimkan dalam *database*. Dalam bagian proses data yang telah dikumpulkan selanjutnya akan diolah menggunakan metode KNN untuk membuat model klasifikasi tiga kelas yaitu normal, miring, dan berhenti. Bagian *output* sistem ini terdiri dari tampilan *dashboard* pemantauan dimana hasil pembacaan sensor, klasifikasi kondisi pergerakan palet dan grafik waktu tempuh akan ditampilkan.

2.2. Diagram Alir Sistem

Diagram alir sistem dimulai saat forklift meletakkan palet di atas dispenser palet. Selanjutnya, palet akan berjalan menuju area lifting sehingga sensor *proximity* mendeteksi keberadaan palet dan sensor ultrasonik menghitung jarak dari sisi kanan dan kiri palet. Data yang dihasilkan oleh pembacaan sensor akan diteruskan masuk ke dalam *database* untuk kemudian dilakukan tahap *preprocessing* data dan

klasifikasi menggunakan metode KNN. Setelah dilakukan proses klasifikasi didapatkan 3 kondisi pergerakan palet. Saat kondisi pergerakan palet normal, sensor akan terus melakukan pembacaan jarak dan waktu tempuh, namun jika kondisi pergerakan palet tidak normal maka terdapat notifikasi peringatan pada *dashboard* monitoring untuk dilanjutkan perbaikan posisi palet.

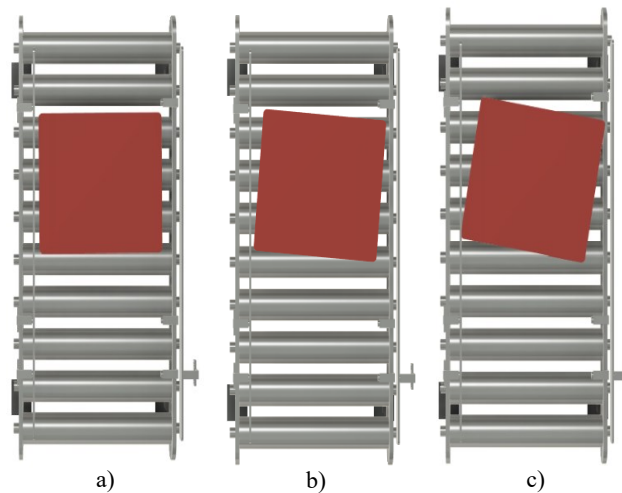
2.3. Pengumpulan Data

Pada penelitian ini, pengambilan data dilakukan dengan menggunakan empat sensor diantaranya dua sensor *proximity* dan dua sensor ultrasonik. Hasil dari pembacaan sensor tersebut dijelaskan pada Tabel 1. menghasilkan lima variabel yang kemudian disimpan dalam *database* MySQL dengan jumlah data 3,500 data. Namun dalam proses permodelan algoritma KNN hanya dua fitur yang diambil sebagai *input* utama yaitu jarak 1 yang merupakan jarak sisi kiri palet terhadap batas konveyor dan jarak 2 yang merupakan jarak dari sisi kanan palet terhadap batas konveyor. Data yang telah dikumpulkan selanjutnya akan melalui tahap *preprocessing* untuk memastikan tidak ada data duplikat, data yang hilang ataupun fitur yang tidak sesuai dengan tipe data.

Tabel 1. Dataset Sebelum Dilakukan *Preprocessing*

Id	Proximity 1	Proximity 2	Jarak 1 (cm)	Jarak 2 (cm)	Waktu Tempuh Palet (Detik)
1	0	1	20,5 cm	20,1 cm	5 detik
2	0	0	54,8 cm	78,9 cm	0 detik
3	1	1	115,8 cm	69,9 cm	6,3 detik
4	1	1	50,5 cm	43,8 cm	9,3 detik
:	:	:	:	:	:
3.456	0	1	10,7 cm	144,6 cm	0 detik
3.457	0	1	19,4 cm	138,4 cm	8,2 detik
3.458	1	1	11 cm	11 cm	0 detik
3.456	0	1	149,3 cm	149,3 cm	4,9 detik

Dalam proses palabelan kelas, terdapat tiga kategori label kondisi palet yang akan digunakan yaitu normal, miring dan berhenti seperti yang ditampilkan pada Gambar 3.10. Palet dikategorikan normal apabila nilai pembacaan jarak 1 dan jarak 2 sama atau memiliki selisih kurang dari atau sama dengan 20 cm. Palet dikategorikan miring apabila selisih antara jarak 1 dan jarak 2 lebih dari 20 cm namun kurang dari atau sama dengan 50 cm. Sedangkan palet dikategorikan berhenti apabila selisih dari kedua jarak tersebut lebih dari 50 cm namun kurang dari atau sama dengan 70 cm.



Gambar 1. Kondisi Palet a) Normal, b) Miring, c) Berhenti

2.4. Data Preprocessing

Pada tahap ini merupakan tahap awal sebelum pemodelan dilakukan, dengan tujuan untuk mempersiapkan data agar dapat dianalisis secara optimal oleh algoritma seperti pada data Tabel 2. Tahap

ini meliputi pembersihan data untuk menghilangkan duplikasi, memperbaiki struktur data serta menamangani *missing value*. Selanjutnya dilakukan normalisasi menggunakan metode *z-score* untuk menetralkan skala antar fitur. Proses ini diikuti dengan *label encoding* untuk mengubah label kondisi kategorikal menjadi numerik. Terakhir, dilakukan pembagian dataset ke dalam tiga skema proporsi data latih dan data uji, yaitu 90:10, 80:20 dan 70:30 guna mengevaluasi performa model secara lebih menyeluruh dan memastikan kestabilan klasifikasi pada kondisi distribusi data yang berbeda.

Tabel 2. Fitur Input Setelah Dilakukan *Preprocessing*

No	Jarak 1 (cm)	Jarak 2 (cm)	Kondisi
1	118,84 cm	137,5 cm	Normal
2	137,34 cm	107,72 cm	Miring
3	54,11 cm	60,47 cm	Normal
4	190,53 cm	139,62 cm	Berhenti
:	:	:	:
3.455	81,4 cm	164,2 cm	Berhenti
3.456	180,4 cm	180,4 cm	Normal
3.457	114,7 cm	134,5 cm	Normal
3.458	133,6 cm	133,6 cm	Normal

Setelah dilakukan tahap *preprocessing*, dataset dibagi menjadi data latih dan data uji dengan tiga variasi proporsi. Tujuan dari variasi ini adalah untuk mengevaluasi performa model pada konfigurasi pembagian data yang berbeda. Jumlah pembagian dataset dapat dilihat pada Tabel 3 di bawah ini.

Tabel 3. Confusion Matrix

No	Pembagian Dataset	Testing	Training	Total Dataset
1	90:10	3111	346	
2	80:20	2765	692	3.458
3	70:30	2419	1038	

2.4. Algoritma K-Nearest Neighbor (KNN)

Algoritma KNN mengklasifikasikan titik data berdasarkan kedekatannya dengan titik-titik lain menggunakan metrik jarak tertentu. Pada penelitian ini, digunakan jarak *Euclidean Distance* untuk mengukur jarak garis lurus antara dua titik data. Secara umum, rumus *Euclidean Distance* antara dua vektor berdimensi -n dinyatakan dalam persamaan (1)

$$d(x, y) = \|x - y\| = \sqrt{\sum_{i=1}^n (x_i - y_i)^2} \quad [1]$$

Algoritma menghitung jarak antara titik uji dengan seluruh titik pada data latih, kemudian memilih K tetangga terdekat. Kelas mayoritas dari tetangga tersebut akan menjadi prediksi kelas untuk data uji. Pemilihan nilai K sangat penting, nilai K yang terlalu kecil dapat menyebabkan *overfitting* karena sensitif terhadap *noise*, sedangkan nilai K yang terlalu besar dapat menyebabkan generalisasi berlebih. Setelah model dilatih, data uji yang tidak melibatkan dalam proses pelatihan diklasifikasikan menggunakan pendekatan ini, kemudian label hasil prediksi dibandingkan dengan label aktual untuk mengevaluasi tingkat akurasi model.

2.5. Logika IF-ELSE

Sebelum implementasi sistem klasifikasi menggunakan algoritma KNN, dilakukan perancangan sistem klasifikasi berbasis urutan menggunakan logika *IF-ELSE* sebagai metode pembandingan. Pendekatan ini menggunakan data dari dua sensor ultrasonik yang dipasang pada sisi kiri dan kanan konveyor untuk mengukur posisi lateral palet. Logika klasifikasi dirancang berdasarkan selisih jarak antara kedua sisi palet dengan asumsi bahwa semakin besar selisih jarak yang terdeteksi maka semakin besar kemungkinan palet mengalami deviasi atau gangguan posisi jalur normal. Pendekatan berbasis logika *IF-ELSE* tidak memerlukan proses pelatihan model, melainkan hanya mengandalkan aturan eksplisit yang ditetapkan berdasarkan *threshold*.

2.6. Evaluasi Model Klasifikasi

Evaluasi performa model KNN dilakukan untuk mengukur kinerja model terhadap data pengujian. Seperti yang ditunjukkan pada Tabel 4, *confusion matrix* menggambarkan jumlah klasifikasi yang benar dan salah untuk setiap kategori dalam bentuk matriks. Akurasi performa model dinilai menggunakan beberapa metrik, antara lain akurasi, presisi, *recall* dan *F1-score*. Nilai *True Positif* (TP) menunjukkan jumlah data yang diklasifikasikan dengan benar sebagai kelas target, sedangkan *True Negative* (TN) menunjukkan jumlah data yang diklasifikasikan dengan benar sebagai bukan kelas target. Sementara itu, *False Positif* (FP) mengacu pada jumlah data yang salah diklasifikasikan sebagai kelas target, dan *False Negative* (FN) menunjukkan jumlah data yang salah diklasifikasikan sebagai bukan kelas target [17].

Tabel 4. Confusion Matrix

	Positif	Negatif
Positif	TP	FP
Negatif	FN	TN

Akurasi Model dihitung menggunakan rumus yang ditunjukkan pada Tabel 5. Akurasi didefinisikan sebagai proporsi data kelas positif prediksi yang benar terhadap seluruh data yang diuji. *Recall* didefinisikan sebagai proporsi data kelas positif yang berhasil diklasifikasikan dengan benar. Presisi merupakan proporsi prediksi positif yang benar berasal dari kelas positif. Sementara itu, *F1-score* merupakan rata-rata harmonik dari nilai presisi dan *recall* yang digunakan untuk memberikan keseimbangan antara keduanya dalam satu metrik evaluasi.

Tabel 5. Metrik Performa

Metriks	Persamaan
Akurasi	$\frac{TN + TP}{TN + FN + FP + TP}$
Presisi	$\frac{TP}{TP + FP}$
<i>Recall</i>	$\frac{TP}{TP + FN}$
<i>F1-Score</i>	$\frac{2}{\frac{1}{precision} + \frac{1}{recall}}$

III. HASIL DAN PEMBAHASAN

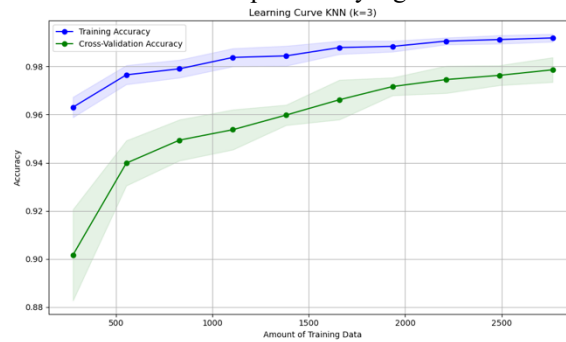
3.1. Pemasangan Sensor dan Akuisisi Data

Pengujian awal dilakukan untuk mengevaluasi akurasi sensor ultrasonik HC-SR04 yang digunakan dalam sistem. Hasil pengujian menunjukkan bahwa sensor memiliki akurasi yang baik dengan nilai *Mean Absolute Percentage Error* (MAPE) sebesar 1,02% untuk sensor kiri dan 0,73% untuk sensor kanan. Nilai ini menunjukkan bahwa penyimpangan pembacaan terhadap jarak referensi masih berada dalam batas toleransi sehingga sensor layak digunakan untuk akuisisi data posisi palet.

3.2. Pengujian Algoritma K-Nearest Neighbors

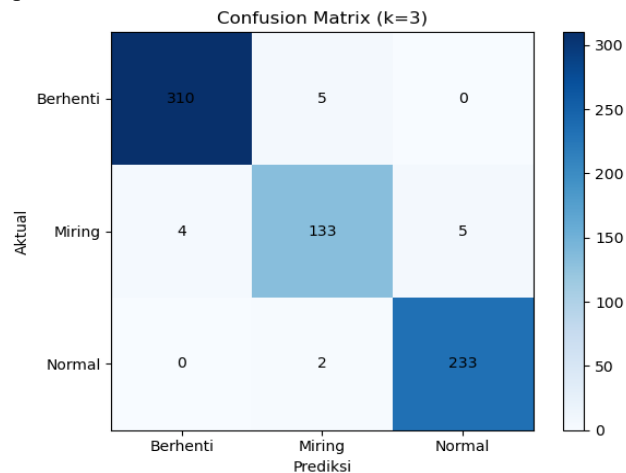
Pengujian terhadap algoritma KNN dilakukan dengan tiga variasi proporsi pembagian data latih dan data uji, yaitu 90:10, 80:20 dan 70:30. Nilai parameter K ditentukan melalui dua tahap yaitu pengujian langsung terhadap data latih dan validasi menggunakan metode *5-fold-cross validation* dengan variasi nilai

$k=1,3,5$ dan 7 . Berdasarkan hasil evaluasi dan learning curve yang ditampilkan pada Gambar 5, nilai $k=3$ dipilih sebagai nilai optimal karena memberikan performa yang stabil dan terhindari dari *overfitting*.



Gambar 2. Visualisasi *Learning Curve* Parameter K

Model KNN dengan $k=3$ dan proporsi data 80:20 menunjukkan akurasi sebesar 97,68%. Nilai metrik yang lainnya yaitu presisi, *recall* dan *F1-score* juga menunjukkan hasil di atas 94% untuk seluruh kelas, yang menandakan bahwa model mampu melakukan klasifikasi secara akurat terhadap kondisi pergerakan palet seperti pada Gambar 6.



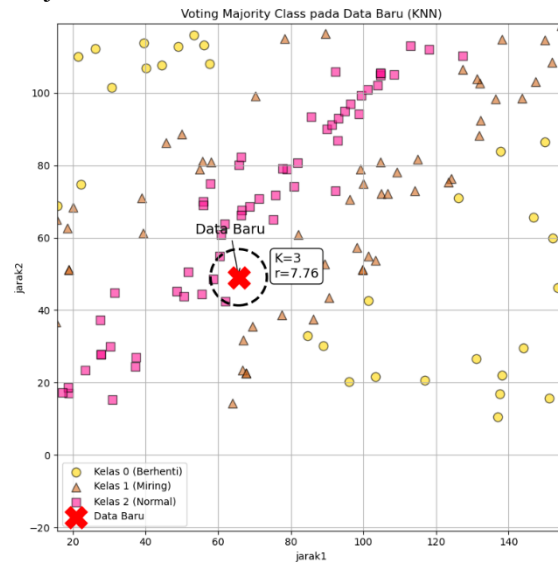
Gambar 3. Visualisasi *Confusion Matrix*

Tabel 6. Hasil Performa Model Dengan 3 Variasi Proporsi Dataset

Kelas	Metrik	90:10	80:20	70:30
Berhenti	Presisi	0,99363	0,98726	0,98517
	Recall	0,98734	0,98413	0,98517
	F1-Score	0,99048	0,98569	0,98517
Miring	Presisi	0,95714	0,95000	0,94811
	Recall	0,94366	0,93662	0,93925
	F1-Score	0,95035	0,94326	0,94366
Normal	Presisi	0,97479	0,97899	0,98305
	Recall	0,99145	0,99149	0,98864
	F1-Score	0,98305	0,98520	0,98584
Akurasi		0,97977	0,97688	0,97688

Dari hasil klasifikasi pada data uji, diperoleh nilai evaluasi performa model seperti yang ditampilkan pada Tabel 6. Nilai presisi, *recall* dan *F1-score* untuk ketiga kelas menunjukkan performa yang baik dan stabil dengan nilai tertinggi berada pada kelas normal dan berhenti.

Sebagai ilustrasi tambahan, Gambar 7 menunjukkan proses klasifikasi data baru oleh model KNN. Titik data baru diklasifikasikan ke dalam kelas normal berdasarkan mayoritas dari tiga tetangga terdekat di sekitar ruang fitur jarak 1 dan jarak 2.



Gambar 4. Visualisasi Pemilihan Kelas Mayoritas

3.3. Pengujian Logika IF-ELSE

Sebagai metode pembandingan, dilakukan pengujian menggunakan pendekatan logika *IF-ELSE* berdasarkan aturan selisih jarak antara sisi palet. Hasil pengujian terhadap 720 data uji menunjukkan bahwa metode ini menghasilkan akurasi 100% dengan waktu komputasi rata-rata 0,0578 detik per data. Meskipun menunjukkan akurasi 100%, metode ini kurang fleksibel terhadap data yang mengandung *noise* atau data yang berada di sekitar nilai threshold klasifikasi.

3.4. Pengujian Sistem

Pengujian ini bertujuan untuk memastikan bahwa sistem klasifikasi kondisi yang digunakan dalam sistem peringatan dini palet berfungsi dengan baik. Pengujian pengiriman sistem klasifikasi ini menggunakan Node-RED sebagai media pengolahan dan intraksi data. Visualisasi antar muka dirancang dengan menggunakan HTML tampilan antarmuka dapat dilihat pada Gambar 8 di bawah ini.



Gambar 5. Tampilan Dashboard Pemantauan

IV. KESIMPULAN

1. Pemilihan nilai k pada KNN dilakukan melalui dua tahap pengujian yaitu, *tuning* parameter pada data *training* dan validasi menggunakan metode *5-fold cross validation* dengan variasi nilai $k=1,3,5$ dan 7 . Meskipun nilai $k=1$ memberikan akurasi tertinggi, analisa terhadap *learning curve* menunjukkan adanya indikasi *overfitting* sehingga diputuskan untuk menggunakan nilai $k=3$ sebagai parameter paling stabil dan seimbang.

2. Setelah dilakukan pengujian terhadap tiga variasi pembagian *dataset* yaitu 90:10, 80:20 dan 70:30 untuk mengevaluasi pengaruh proporsi data terhadap performa model. Hasil pengujian menunjukkan bahwa pembagian *dataset* 80:20 memberikan performa paling stabil dimana data *training* cukup untuk pelatihan dan data *testing* cukup untuk evaluasi.
3. Evaluasi akhir model KNN dilakukan dengan menggunakan 20% data *testing* dengan parameter $k=3$. Model menunjukkan performa klasifikasi yang baik pada seluruh kelas. Pada kelas berhenti nilai *precision* 0,98726, *recall* 0,9841 dan *F1-score* 0,98569. Untuk kelas miring nilai *precision* 0,95000, *recall* 0,93662 dan *F1-score* 0,94326. Sedangkan untuk kelas normal nilai *precision* 0,97899, *recall* 0,99149 dan *F1-score* 0,98520. Hasil menunjukkan bahwa model dapat membedakan masing-masing kelas dengan baik dan menghasilkan nilai *F1-score* yang stabil diatas 0,94 untuk semua kelas.
4. Sebagai pembandingan dilakukan pengujian menggunakan metode *rule-based* dengan logika *IF-ELSE* berdasarkan *threshold* selisih jarak. Pengujian terhadap 720 data uji menunjukkan bahwa metode ini dapat menghasilkan akurasi 100% dengan nilai *precision*, *recall* dan *F1-score* 1,00 pada semua kelas. Namun metode ini sangat bergantung pada *threshold* tetap serta kurang fleksibel terhadap data yang mengandung *noise* atau variasi tinggi.
5. Setelah model klasifikasi diuji, sistem dilanjutkan ke tahap integrasi dengan *dashboard* pemantauan berbasis menggunakan Node-RED. Hasil menunjukkan bahwa sistem dapat menampilkan data sensor dengan benar dan baik.

REFERENSI

- [1] A. Yasin, J. Delaney, C.-T. Cheng, and T. Y. Pang, "The Design and Implementation of an IoT Sensor-Based Indoor Air Quality Monitoring System Using Off-the-Shelf Devices," *Appl. Sci.*, vol. 12, no. 19, p. 9450, Sep. 2022, doi: 10.3390/app12199450.
- [2] S. Gallagher and J. R. Heberger, "The effects of operator position, pallet orientation, and palletizing condition on low back loads in manual bag palletizing operations," *Int. J. Ind. Ergon.*, vol. 47, pp. 84–92, May 2015, doi: 10.1016/j.ergon.2015.03.005.
- [3] Q. Chen *et al.*, "State Monitoring and Fault Diagnosis of HVDC System via KNN Algorithm with Knowledge Graph: A Practical China Power Grid Case," *Sustainability*, vol. 15, no. 4, p. 3717, Feb. 2023, doi: 10.3390/su15043717.
- [4] J. Liu, S. Fu, F. Liu, and X. Cheng, "Intelligent fault diagnosis of belt conveyor rollers using a polar KNN algorithm with audio features," *Eng. Fail. Anal.*, vol. 168, p. 109101, Feb. 2025, doi: 10.1016/j.engfailanal.2024.109101.
- [5] X. Zeng and L. Koehl, "179 PUBLICATIONS 3,183 CITATIONS SEE PROFILE".
- [6] H. Li, X. Wang, Y. Feng, Y. Qi, and J. Tian, "Driving intelligent IoT monitoring and control through cloud computing and machine learning," *Appl. Comput. Eng.*, vol. 64, no. 1, pp. 76–81, May 2024, doi: 10.54254/2755-2721/64/20241358.
- [7] N. Hütten, M. Alves Gomes, F. Hölken, K. Andricevic, R. Meyes, and T. Meisen, "Deep Learning for Automated Visual Inspection in Manufacturing and Maintenance: A Survey of Open- Access Papers," *Appl. Syst. Innov.*, vol. 7, no. 1, p. 11, Jan. 2024, doi: 10.3390/asi7010011.
- [8] Z. Zhang, Q. Zhou, H. Jin, Q. Li, and Y. Dai, "A Real-Time Intelligent Valve Monitoring Approach through Cameras Based on Computer Vision Methods," *Sensors*, vol. 24, no. 16, p. 5337, Aug. 2024, doi: 10.3390/s24165337.
- [9] T. Nesti, S. Boddana, and B. Yaman, "Ultra-Sonic Sensor based Object Detection for Autonomous Vehicles," in *2023 IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops (CVPRW)*, Vancouver, BC, Canada: IEEE, Jun. 2023, pp. 210–218. doi: 10.1109/CVPRW59228.2023.00026.
- [10] J. Gao, "A comprehensive analysis of different types of proximity sensors in wearable electronic devices," *Appl. Comput. Eng.*, vol. 62, no. 1, pp. 42–47, May 2024, doi: 10.54254/2755-2721/62/20240376.
- [11] A. Karagoz and G. Dindis, "Object Recognition and Positioning with Neural Networks: Single Ultrasonic Sensor Scanning Approach," *Sensors*, vol. 25, no. 4, p. 1086, Feb. 2025, doi: 10.3390/s25041086.

-
- [12] Y. Ding, H. Kisner, T. Kong, and U. Thomas, "Using Machine Learning for Material Detection with Capacitive Proximity Sensors," in *2020 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, Las Vegas, NV, USA: IEEE, Oct. 2020, pp. 10424–10429. doi: 10.1109/IROS45743.2020.9341016.
 - [13] H. Wu, X. Wang, and Y. Li, "Placement Optimization of Flexible Proximity Sensors for Human-Robot Collaboration," *IEEE Robot. Autom. Lett.*, vol. 9, no. 6, pp. 5591–5598, Jun. 2024, doi: 10.1109/LRA.2024.3396105.
 - [14] H. Sukri, F. Adiputra, A. Basuki, A. A. Syamsuddin, and A. Aziz, "K-Nearest Neighbor Method Approach in Implementing Automatic Feeding Efficiency Based on the Internet of Things in Broiler Chickens," in *2023 IEEE 9th Information Technology International Seminar (ITIS)*, Batu Malang, Indonesia: IEEE, Oct. 2023, pp. 1–6. doi: 10.1109/ITIS59651.2023.10420176.
 - [15] S. Madi and A. R. Baba-Ali, "A new hybrid incremental learning system for an enhanced KNN algorithm (hoKNN)," *Evol. Syst.*, vol. 15, no. 3, pp. 1001–1019, Jun. 2024, doi: 10.1007/s12530-023-09531-y.
 - [16] A. L. Bowler, M. P. Pound, and N. J. Watson, "A review of ultrasonic sensing and machine learning methods to monitor industrial processes," *Ultrasonics*, vol. 124, p. 106776, Aug. 2022, doi: 10.1016/j.ultras.2022.106776.
 - [17] M. Fahmy Amin, "Confusion Matrix in Binary Classification Problems: A Step-by-Step Tutorial," *J. Eng. Res.*, vol. 6, no. 5, pp. 0–0, Dec. 2022, doi: 10.21608/erjeng.2022.274526.