

Sistem Klasifikasi Kualitas Udara dengan Integrasi Sensor menggunakan Metode *K-Nearest Neighbor*

Athif Tafrihan Achyar^{#1}, Rahmi Hidayati^{#2}, Kartika Sari^{#3}

[#]Program Studi Rekayasa Sistem Komputer, Universitas Tanjungpura

Jl. Prof. Dr. H. Hadari Nawawi, Pontianak 78124

¹h1051201010@student.untan.ac.id

²rahmihidayati@siskom.untan.ac.id

³Kartika.sari@siskom.untan.ac.id

Abstrak— Kualitas udara yang buruk dapat berdampak negatif pada sistem respirasi manusia dan dampak kesehatan lainnya, sehingga pemantauan kualitas udara berperan penting terhadap permasalahan ini. Penelitian ini mengembangkan sistem klasifikasi kualitas udara secara *real-time* menggunakan metode *K-Nearest Neighbor* (KNN) dan data dari 4 sensor: GP2Y1010AU0F (PM_{2.5}), MQ-135 (CO₂), MQ-131 (O₃), dan MQ-7 (CO). Sistem terdiri dari sensor yang terintegrasi dengan ESP32, penyimpanan *cloud* menggunakan Firebase, dan antarmuka web untuk pemrosesan serta visualisasi data. ESP32 berfungsi mengumpulkan data dari sensor dan mengirimkannya ke Firebase, yang kemudian diakses oleh aplikasi web untuk klasifikasi menggunakan algoritma KNN. Klasifikasi dilakukan dalam 4 kategori: baik, sedang, buruk, dan sangat buruk, dengan hasil ditampilkan di antarmuka web untuk pemantauan selama 24 jam terakhir. Pengujian menggunakan *confusion matrix* dengan 900 data latih dan 600 data uji menunjukkan tingkat akurasi sebesar 98,67%. Selain itu, pengujian *crossvalidation* dengan k=3 menghasilkan akurasi sebesar 99,39%.

Kata kunci— Kualitas Udara, *K-Nearest Neighbor*, *Cross Validation*, *Confusion Matrix*, *Real-time*

I. PENDAHULUAN

Udara adalah elemen vital bagi kelangsungan hidup. Udara yang bersih adalah udara yang tidak terkontaminasi oleh zat-zat atau gas berbahaya seperti partikel debu, karbon dioksida (CO₂), nitrogen dioksida (NO₂), serta berbagai gas lain yang dapat menimbulkan dampak negatif. Sangat disayangkan, meskipun udara berperan penting untuk kehidupan terutama sebagai penunjang sistem pernapasan manusia, namun seringkali tidak terdeteksi udara yang masuk ke tubuh telah tercemar zat kimia atau gas beracun dalam kadar berlebihan [1].

Berdasarkan kasus yang pernah terjadi pada masa pandemi COVID-19, banyak daerah yang ada di Indonesia melakukan program Pembatasan Sosial Berskala Besar (PSBB) serta pemerintah memaksakan mulai dari pegawai pekerja hingga pelajar sekolah untuk melakukan kegiatan *work from home* (WFH) dan *online learning*, kondisi ini

menyebabkan berkurangnya aktivitas transportasi di jalan raya, yang pada akhirnya memengaruhi kualitas udara di sejumlah kota di Indonesia, terutama di kota-kota besar. Penurunan aktivitas kendaraan bermotor di jalan raya, sebagai sumber utama polusi udara, menyebabkan berkurangnya dampak pada kualitas udara [2].

Pemantauan kualitas udara dapat melibatkan pengukuran berbagai parameter seperti karbon monoksida (CO), karbon dioksida (CO₂), ozon (O₃), dan partikel halus (PM_{2.5}). Setiap parameter ini memiliki ambang batas dan tingkat bahaya yang berbeda berdasarkan standar kesehatan internasional dan nasional. Klasifikasi kualitas udara dapat dikategorikan menjadi beberapa tingkat: baik, sedang, buruk, dan sangat buruk, dengan mempertimbangkan konsentrasi masing-masing parameter tersebut. Selain itu dengan adanya penerapan dan integrasi algoritma *machine learning* menjadi solusi penting dalam meningkatkan akurasi sistem pemantauan kualitas udara. Dengan mempertimbangkan berbagai standar pengukuran kualitas udara, baik nasional maupun internasional, sistem klasifikasi berbasis *machine learning* dapat memberikan penilaian yang lebih luas terhadap kondisi lingkungan setempat.

Penelitian mengenai penerapan metode KNN untuk memecahkan suatu masalah telah sebelumnya dilakukan oleh sejumlah peneliti., salah satunya yaitu Cholil dkk (2021) dengan judul “Implementasi Algoritma Klasifikasi K-Nearest Neighbor (KNN) Untuk Klasifikasi Seleksi Penerima Beasiswa”. Sistem yang dibuat menggunakan penerapan metode KNN untuk mengklasifikasikan data penerima calon beasiswa baru dengan beberapa atribut seperti nilai siswa, prestasi akademik siswa, prestasi non-akademik siswa, dan penghasilan orang tua dengan cara menjadikan data penerima beasiswa sebelumnya sebagai data *training*. Hasil uji penelitian tersebut menggunakan *confusion matrix* menunjukkan akurasi rata-rata sebesar 90,5% [3].

Penelitian selanjutnya dilakukan oleh Purbakawaca dan Fauzan (2022) dengan judul “Rancang Bangun Sistem Pemantauan Kualitas Udara Dalam Ruangan Berbiaya

Rendah Berbasis IoT". Penelitian ini dirancang untuk menciptakan sistem yang dapat memantau kualitas udara di dalam ruangan, sehingga membantu penghuni rumah untuk mengakses dan memonitor tingkat polutan serta kenyamanan suhu di lingkungan dalam ruangan. Pengujian menunjukkan bahwa sistem pada penelitian tersebut memiliki *packet loss* dengan kategori sangat bagus [4].

Penelitian selanjutnya dilakukan oleh Putra dan Rismawan (2023) dengan judul "Klasifikasi Kualitas Udara Berdasarkan Indeks Standar Pencemaran Udara (ISPU) Menggunakan Metode *Fuzzy Tsukamoto*". Penelitian ini mengembangkan sebuah sistem yang mampu mengklasifikasikan kualitas udara berdasarkan (ISPU) dengan mempertimbangkan 5 parameter pengukuran yang memengaruhi kualitas udara dan menggunakan metode fuzzy Tsukamoto. Hasil dari penelitian ini diperoleh bahwa pengujian sistem dengan membandingkan 2 data yang berbeda yaitu data DLH kota Pontianak dan pengujian *confusion matrix* penelitian menunjukkan tingkat akurasi sebesar 97% [5].

Namun, dari studi-studi tersebut, belum banyak penelitian yang secara langsung mengintegrasikan sensor lingkungan secara real-time dengan pendekatan klasifikasi berbasis KNN, terutama pada sistem yang mendistribusikan data melalui cloud dan menyajikannya secara langsung pada *client-side web interface*. Di sinilah letak kebaruan penelitian ini.

Kelebihan metode K-Nearest Neighbor (KNN) adalah kemampuannya dalam menangani data multidimensi tanpa memerlukan asumsi distribusi data tertentu. KNN bersifat non-parametrik, mudah diimplementasikan, dan efektif untuk klasifikasi dengan jumlah data moderat. KNN juga memberikan fleksibilitas dalam menyesuaikan jumlah *neighbors* untuk mencapai akurasi optimal, sehingga cocok digunakan pada sistem klasifikasi berbasis sensor yang dinamis dan bersifat real-time.

Berdasarkan masalah yang telah dipaparkan, pada penelitian ini membahas "Sistem Klasifikasi Kualitas Udara Dengan Integrasi Sensor Menggunakan Metode *K-Nearest Neighbor*". Data masukan yang digunakan dalam penelitian ini berupa data karbon dioksida (CO₂), karbon monoksida (CO), PM_{2.5}, O₃, suhu dan kelembaban. Oleh karena itu, penelitian ini bertujuan untuk mengembangkan sebuah sistem pemantauan secara *real-time* dengan klasifikasi menggunakan metode *K-Nearest Neighbor* di mana data sensor akan diproses untuk dikonversi nilainya pada mikrokontroler NodeMCU ESP32 yang terkoneksi dengan jaringan wifi, selanjutnya diteruskan ke penyimpanan *cloud server*, setelah itu akan diklasifikasikan dan data hasil akan ditampilkan pada antarmuka aplikasi *website*. Hal ini dapat membantu pemrosesan data lanjutan dapat diimplementasikan secara lebih baik pada *website-client side* yang tidak terpengaruh terhadap latensi.

II. LANDASAN TEORI

A. Sensor MQ-135

Sensor MQ-135 adalah komponen penting dalam sistem deteksi gas dan kualitas udara, pada penelitian ini dirancang untuk mendeteksi jenis gas berbahaya dan salah satunya yaitu karbon dioksida (CO₂). Sensor ini dilengkapi dengan pin output digital dan analog sehingga memberikan fleksibilitas dalam penggunaannya [6].

B. Sensor MQ-7

Sensor MQ-7 pada penelitian ini dirancang untuk mendeteksi konsentrasi gas karbon monoksida (CO). Keunggulan sensor ini terletak pada sensitivitasnya yang tinggi terhadap CO serta stabilitasnya dalam penggunaan jangka panjang [7].

C. Sensor MQ-131

Sensor MQ-131 dalam penelitian ini dikembangkan secara khusus untuk mendeteksi dan mengukur kadar ozon (O₃) di udara. Sensor ini bekerja dengan prinsip perubahan resistansi, di mana resistansi akan berubah sesuai dengan tingkat konsentrasi ozon yang terdeteksi [8].

D. Sensor GP2Y1010AU0F

Sensor dalam penelitian ini berfungsi sebagai alat pendekripsi partikel halus PM_{2.5}. Cara kerja sensor ini adalah dengan mendekripsi debu atau partikel, lalu memantulkan cahaya ke bagian penerima. Ketika debu melewati seluruh permukaan, photodiode akan mengubahnya menjadi sinyal tegangan [9].

E. DHT22

Sensor DHT22 dalam penelitian ini dimanfaatkan untuk mengukur dua aspek lingkungan secara bersamaan, yaitu suhu dan tingkat kelembaban udara. DHT22 memiliki keunggulan seperti keluaran sinyal digital yang diolah oleh MCU 8-bit, tingkat akurasi dan presisi yang lebih tinggi dibandingkan DHT11 dan kemampuan transmisi sinyal output yang lebih jauh daripada DHT11 [10].

F. Arduino Uno

Arduino Uno pada penelitian ini digunakan sebagai mikrokontroler utama dan sebagai pembaca dari nilai sensor-sensor yang digunakan. Selain itu, Arduino ini digunakan untuk mengirim data sensor ke mikrokontroler ESP32 secara serial. Arduino Uno R3 ini merupakan versi ketiga dari Arduino Uno yang diluncur pada tahun 2011. Mikrokontroler ini menggunakan chip Atmega328 8-bit keluaran Atmel sebagai komponen utamanya [11].

G. ESP32

NodeMCU ESP32 adalah sebuah sistem yang beroperasi dengan konsumsi daya rendah, yang tergolong dalam keluarga *chip* (SoC), yang memiliki kemampuan Wi-Fi dan Bluetooth dalam dua mode [12]. Pada penelitian ini ESP32 digunakan sebagai gateway IoT sistem agar data yang sebelumnya dibaca oleh Arduino bisa terima dan akan meneruskan data ke *cloud server*.

H. K-Nearest Neighbor

Algoritma *K-Nearest Neighbor* (KNN) diterapkan untuk tugas klasifikasi dan regresi, dengan input berupa sejumlah titik data terdekat (*k*) dari suatu dataset. Tujuan dari klasifikasi KNN adalah menentukan keanggotaan kelas suatu objek. Objek tersebut diklasifikasikan berdasarkan mayoritas kelas dari *k* tetangga terdekatnya, di mana objek akan dimasukkan ke dalam kelas yang paling dominan di antara tetangga tersebut. Algoritma K-NN memanfaatkan konsep kedekatan untuk melakukan klasifikasi atau prediksi terhadap pengelompokan titik data, dengan asumsi bahwa titik-titik yang memiliki kemiripan cenderung berada berdekatan satu sama lain. Kata “dekat” didefinisikan dalam bentuk mterik jarak yang disebut dengan *Euclidean Distance*. Dalam bahasa sederhana, algoritma K-NN bertujuan untuk mengklasifikasikan objek baru berdasarkan atribut dan sampel dari data latih yang diambil dari *k* tetangga terdekatnya (*nearest neighbors*), di mana *k* merupakan banyaknya tetangga terdekat.

Berdasarkan langkah dasar K-NN, pengukuran tetangga terdekat dengan data baru adalah menggunakan persamaan *euclidean Distance*. Pada implementasinya, data dapat berada dalam 1 dimensi atau lebih. Berikut persamaannya.

$$d(x_1, x_2) = \sqrt{\sum_{i=0}^n (x_{1i} - x_{2i})^2} \quad (1)$$

Dimana X_1 adalah nilai dari data latih dan X_2 adalah nilai dari data uji atau bisa disebut *Euclidean vectors*. Setelah menghitung nilai jarak *euclidean* dari setiap tetangga, langkah selanjutnya adalah mengurutkan nilai-nilai tersebut dari yang terkecil hingga terbesar. Kemudian, kategori yang paling dominan dari *k* tetangga terdekat dipilih berdasarkan nilai *k* yang telah ditetapkan sebelumnya [13].

I. Cross validation

Cross validation atau yang sering disebut sebagai estimasi rotasi, merupakan metode untuk menilai seberapa baik hasil analisis dapat diterapkan pada data yang independen. Teknik ini bertujuan untuk memprediksi performa model dan mengestimasi tingkat akurasi model saat diaplikasikan. Salah satu pendekatan dalam *cross validation* adalah *k-fold cross validation*, di mana data dibagi menjadi *K* bagian dengan ukuran yang setara. Penerapan *k-fold=5 cross validation* berguna untuk meminimalisir bias yang mungkin terjadi pada data [14].

J. Confusion Matrix

Confusion matrix merupakan sebuah tabel yang dipakai untuk mengukur performa model klasifikasi dengan cara membandingkan prediksi yang dihasilkan model terhadap label sebenarnya. Tabel ini terdiri dari empat komponen utama: *True Positive* (TP) dan *True Negative* (TN), yang mewakili prediksi yang benar, serta *False Positive* (FP) dan *False Negative* (FN), yang mewakili prediksi yang salah. Komponen ini memberikan gambaran detail tentang bagaimana model mengklasifikasikan data benar maupun salah [15]. Pada *confusion matrix* ini, akurasi digunakan

untuk mengukur persentase hasil yang benar dari keseluruhan hasil yang diperoleh. Presisi digunakan untuk mengukur persentase dari seluruh kelas positif yang benar-benar positif. *Recall* mengukur persentasi kelas positif yang berhasil di prediksi dengan benar. *Confusion matrix* untuk 4 kelas bisa dilihat pada Tabel 1.

TABEL I
CONFUSION MATRIX UNTUK 4 KELAS

	Predict A	Predict B	Predict C	Predict D
Actual A	TP			
Actual B		TP		
Actual C			TP	
Actual D				TP

Pengukuran *confusion matrix* menggunakan persamaan berikut.

$$\text{Accuracy (\%)} = \frac{(TP)}{(Total\ Data)} \times 100 \quad (2)$$

$$\text{Precision (\%)} = \frac{TP}{(TP+FP)} \times 100 \quad (3)$$

$$\text{Avg Precision (\%)} = \frac{\text{Precision}(A) + \dots + \text{Precision}(n)}{\text{The number of classes}} \times 100 \quad (4)$$

$$\text{Recall (\%)} = \frac{TP}{(TP+FN)} \times 100 \quad (5)$$

$$\text{Avg Recall (\%)} = \frac{\text{Recall}(A) + \dots + \text{Recall}(n)}{\text{The number of classes}} \times 100 \quad (6)$$

Keterangan :

TP : Data yang sebenarnya positif dan diprediksi sebagai positif

FP : Data yang sebenarnya negatif tetapi diprediksi sebagai positif

FN : Data yang sebenarnya positif tetapi diprediksi sebagai negatif

K. Kualitas Udara

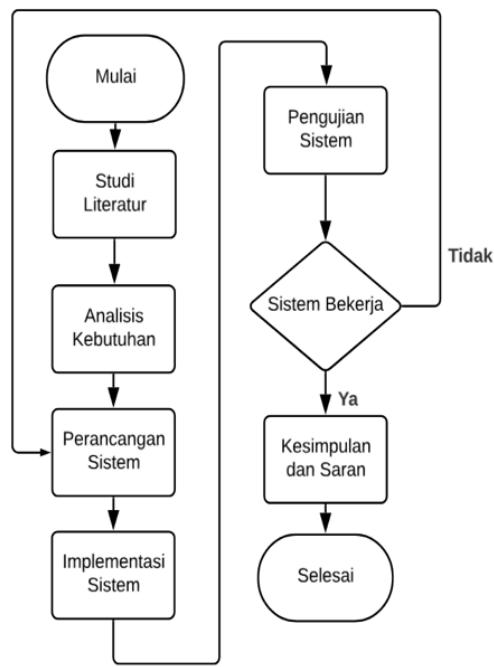
Kualitas udara merupakan aspek penting yang mencerminkan kesehatan lingkungan di sekitar kita. Kualitas udara dinilai berdasarkan konsentrasi polutan yang dibandingkan dengan batas baku mutu udara ambien, yang merupakan batas kadar polutan dalam udara yg aman bagi kesehatan manusia [16,17]. Kondisi udara yang kita hirup setiap hari dipengaruhi oleh berbagai komponen kimia, baik yang terbentuk secara alami maupun sebagai hasil aktivitas manusia. Keberadaan gas-gas seperti karbon monoksida yang umumnya dihasilkan dari pembakaran tidak sempurna, nitrogen dioksida yang banyak berasal dari emisi kendaraan bermotor, serta sulfur dioksida yang sering dikaitkan dengan aktivitas industri, menjadi parameter penting dalam menentukan tingkat kesehatan udara yang kita hirup. Adapun baku mutu udara ambien yang digunakan pada penelitian ini berpacu berdasarkan Indeks Standar Pencemar Udara (ISPU) dan dapat dilihat pada Tabel 2.

TABEL II
BAKU MUTU UDARA AMBIEN

Kategori	CO (ppm)	CO ₂ (ppm)	PM _{2.5} ($\mu\text{g}/\text{m}^3$)	(ppm) O ₃
Baik	0 – 3.2	400 – 1000	0 – 15.5	0 – 0.110
Sedang	3.2 – 6.4	1000 – 2000	15.6 – 55.4	0.110 – 0.187
Buruk	6.4 – 12	2000 – 5000	55.5 – 150.4	0.187 – 0.373
Sangat Buruk	> 12	> 50000	150.5 – 250.4	> 0.373

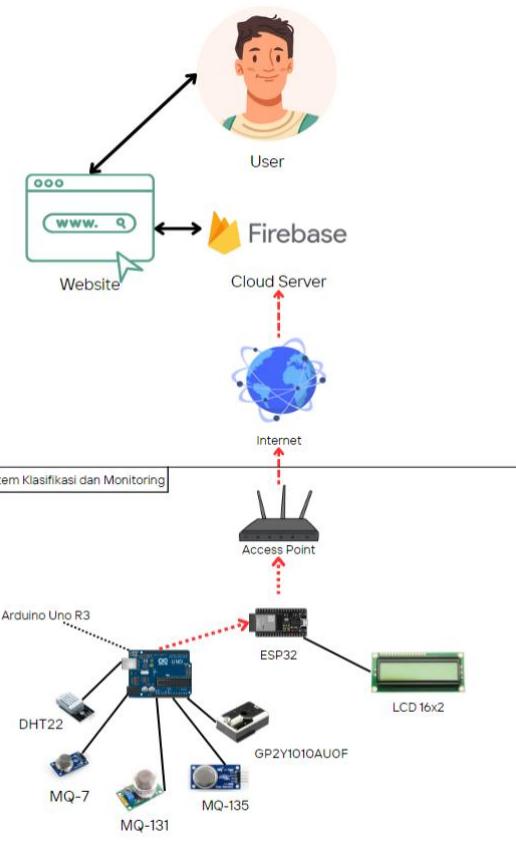
III. METODE PENELITIAN

Penelitian ini mengembangkan sistem pemantauan dan klasifikasi kualitas udara dengan mengimplementasikan metode *K-Nearest Neighbor* secara (*real-time*). Berikut merupakan alur atau tahapan yang dilakukan dalam proses membuat sistem bisa dicermati dalam Gambar 1.



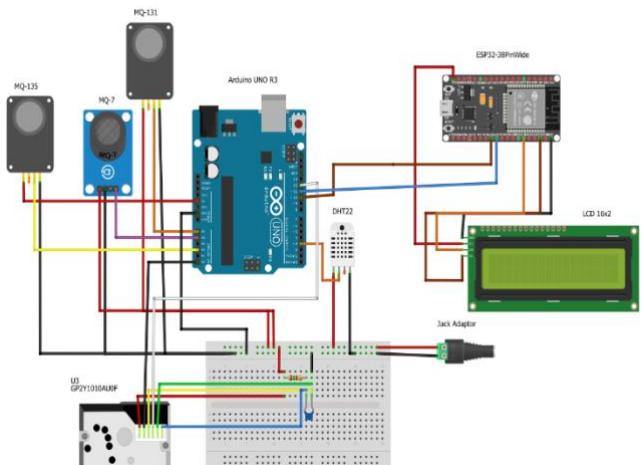
A. Perancangan Sistem

Perancangan sistem pada penelitian ini meliputi perancangan perangkat keras dan perangkat lunak. Sistem ini dirancang untuk memantau dan mengklasifikasikan kualitas udara secara real-time dengan mengintegrasikan berbagai sensor dan menggunakan metode KNN untuk klasifikasi. Gambar perancangan sistem klasifikasi dan pemantauan dapat dilihat pada Gambar 1.



Gambar. 2 Perancangan sistem klasifikasi dan pemantauan

1) *Perancangan Perangkat Keras*: Perancangan perangkat keras pada sistem klasifikasi kualitas udara dengan integrasi sensor melibatkan integrasi antara Arduino Uno R3 dan ESP32. Arduino Uno R3 berfungsi sebagai unit pengumpul data utama, terhubung dengan berbagai sensor seperti MQ-135 (CO₂), MQ-7 (CO), MQ-131 (O₃), GP2Y1010AUOF (PM_{2.5}), dan DHT22 (suhu dan kelembaban). Perancangan perangkat keras dapat dilihat pada Gambar 3.

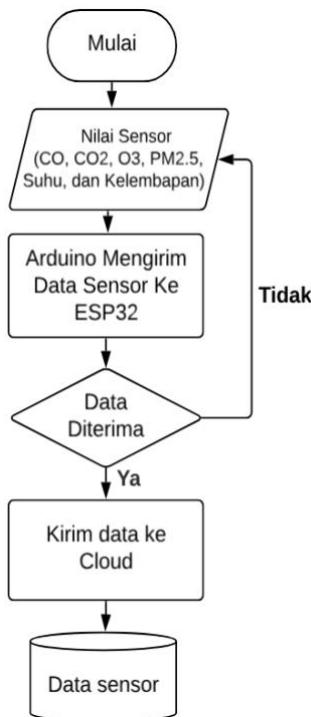


Gambar. 3 Perancangan perangkat keras

B. Perancangan Perangkat Lunak

Perangkat lunak yang dirancang adalah program pada Arduino Uno serta program pada ESP32 untuk menjalankan komponen-komponen perangkat keras agar berfungsi untuk membaca dan mendeteksi nilai sensor dan saling terhubung satu sama lain.

Sensor-sensor gas akan membaca nilai masing-masing konsentrasi gas. Selanjutnya nilai tersebut akan diproses oleh Arduino untuk mengirimkan data tersebut ke ESP32 secara serial. Setelah itu ESP32 akan mengirimkan data-data sensor tersebut ke *cloud server* yang digunakan. Diagram alir perancangan pembacaan nilai sensor pada Arduino Uno dan ESP32 dapat dilihat pada Gambar 4.



Gambar. 4 Diagram alir perancangan pembacaan nilai sensor pada arduino dan ESP32

Selanjutnya adalah proses bagaimana algoritma *K-Nearest Neighbor* (KNN) dirancang untuk mengklasifikasi kualitas udara menggunakan parameter CO, CO₂, O₃, dan PM_{2.5}. Jarak antar data dihitung dengan metode Euclidean, yang dikombinasikan dengan normalisasi untuk memastikan setiap parameter berkontribusi secara seimbang. Model ini menentukan klasifikasi berdasarkan mayoritas kelas dari k tetangga terdekat dalam data pelatihan. Untuk meningkatkan akurasi, digunakan *cross validation* dengan *stratified splitting*, memastikan distribusi kelas dalam data pelatihan dan pengujian tetap seimbang. Sistem ini dirancang untuk klasifikasi *real-time*, sehingga dapat menyesuaikan dengan perubahan data kualitas udara secara dinamis. Perancangan algoritma klasifikasi KNN pada sistem dapat dilihat pada Gambar 5.



Gambar. 5 Perancangan algoritma klasifikasi KNN

Diagram alir algoritma klasifikasi knn dimulai dari diterimanya data sensor (CO, CO₂, O₃, PM_{2.5}, PM₁₀, suhu, dan kelembapan) lalu diproses dengan menentukan k terbaik menggunakan cross-validation, diikuti oleh normalisasi dan perhitungan jarak *Euclidean* untuk menemukan k tetangga terdekat. Mayoritas kelas dari tetangga tersebut menentukan hasil klasifikasi. Status kualitas udara dievaluasi mulai dari "Baik," "Sedang," "Buruk," hingga "Sangat Buruk," lalu hasil ditampilkan dan proses selesai. Sistem ini membuat klasifikasi kualitas udara secara otomatis berdasarkan data sensor yang diterima.

IV. HASIL DAN PEMBAHASAN

Bagian ini menjelaskan hasil yang didapatkan dari penelitian meliputi hasil perancangan sistem, pengujian sistem dan pembahasan.

A. Implementasi Sistem

Pada tahap ini, perangkat dikembangkan secara menyeluruh berdasarkan desain perangkat keras dan perangkat lunak yang telah dirancang sebelumnya. Sistem ini menggunakan Arduino Uno untuk membaca data dari lima sensor: MQ-135 (CO₂), MQ-7 (CO), MQ-131 (O₃), GP2Y1010AU0F (PM_{2.5}), dan DHT22 (suhu dan kelembaban). Data sensor dikirimkan ke ESP32 melalui komunikasi serial, kemudian diteruskan ke Firebase Realtime Database menggunakan koneksi Wi-Fi. Proses ini membantu pengiriman data secara *real-time* untuk pemantauan kualitas udara dan klasifikasi menggunakan

metode KNN. Hasil sistem pemantauan dan klasifikasi menggunakan Arduino Uno dan ESP32 dapat dilihat pada Gambar 6.



Gambar. 6 Hasil implementasi keseluruhan sistem

B. Antarmuka Halaman Dashboard Website

Halaman *dashboard* merupakan halaman yang dapat diakses oleh pengunjung untuk melihat informasi kualitas udara secara *real-time*. Antarmuka halaman *dashboard* dapat dilihat pada Gambar 7.



Gambar. 7 Antarmuka halaman *dashboard*

C. Implementasi Algoritma KNN

Pada implementasi algoritma K-Nearest Neighbor (KNN) dalam sistem klasifikasi kualitas udara, pendekatan ini dirancang untuk mengolah data lingkungan yang diperoleh dari sensor karbon monoksida (CO), karbon dioksida (CO₂), ozon (O₃), dan partikel debu (PM_{2.5}). Data ini digunakan untuk mengklasifikasikan kualitas udara ke dalam empat kategori, yaitu Baik, Sedang, Buruk, dan Sangat Buruk.

Pengambilan data dilakukan secara *real-time* melalui sensor yang terpasang pada sistem monitoring berbasis mikrokontroler, dengan interval pengambilan antara 2 hingga 5 menit. Proses ini dilaksanakan selama lebih dari dua minggu untuk memastikan data mencerminkan kondisi udara yang bervariasi. Dari proses tersebut, terkumpul 900 dataset yang kemudian dibagi menggunakan teknik stratified splitting untuk menjaga proporsi kelas antara data latih dan data uji. Komposisi data yang digunakan adalah 80% data latih (720 data) dan 20% data uji (180 data).

Dalam pemrosesan data, digunakan algoritma KNN dengan jarak Euclidean sebagai metrik kedekatan antara data uji dan data latih. Untuk menghindari overfitting dan bias pada model, dilakukan evaluasi kinerja melalui skema cross-validation dengan berbagai nilai *k*. Berdasarkan pengujian, nilai *k* = 3 memberikan hasil terbaik dengan akurasi klasifikasi mencapai 99,39% menggunakan 900 data latih. Pada implementasinya, pengujian sistem menggunakan metode KNN dilakukan pengambilan dan penyimpanan data dilakukan secara terstruktur dengan interval 2-5 menit untuk setiap kelompok data. Hasil klasifikasi data disajikan pada Tabel 4.

Hasil evaluasi cross-validation untuk beberapa nilai *k* disajikan pada Tabel 3.

TABEL III
HASIL EVALUASI CROSS VALIDATION

No	Nilai K	Jumlah Data Training	Akurasi
1	3	900	99.39%
2	5	900	99.08%
3	7	900	98.24%
4	9	900	98.47%
5	11	900	98.78%
6	13	900	98.70%

Pada implementasinya, pengujian sistem menggunakan metode KNN dilakukan pengambilan dan penyimpanan data dilakukan secara terstruktur dengan interval 2-5 menit untuk setiap kelompok data. Hasil klasifikasi data disajikan pada Tabel 4.

TABEL IV
HASIL KLASIFIKASI KNN

No	Waktu	CO (ppm)	CO ₂ (ppm)	O ₃ (ppm)	Debu (ug/m3)	Status
1	07/10/2024, 16.10.44	3.11	486.38	0.06	15.00	Baik
2	07/10/2024, 16.10.42	3.11	486.38	0.06	15.00	Baik
3	07/10/2024, 16.10.40	3.11	486.38	0.06	15.00	Baik
4	07/10/2024, 16.06.54	3.11	486.38	0.05	15.00	Sedang
5	07/10/2024, 15.58.47	3.54	495.57	0.05	26.00	Sedang
6	07/10/2024, 15.58.47	3.54	495.57	0.05	26.00	Sedang
7	07/10/2024, 15.58.46	3.19	495.57	0.05	26.00	Sedang
8	07/10/2024, 15.58.44	3.19	495.57	0.05	34.00	Sedang

No	Waktu	CO (ppm)	CO ₂ (ppm)	O ₃ (ppm)	Debu (ug/m3)	Status
9	07/10/2024, 15.54.53	3.19	495.57	0.05	34.00	Sedang
10	07/10/2024, 15.54.51	3.19	495.57	0.05	34.00	Sedang
11	07/10/2024, 15.54.50	3.19	495.57	0.05	34.00	Sedang
12	07/10/2024, 15.54.49	3.19	495.57	0.05	34.00	Sedang
13	07/10/2024, 15.54.47	3.41	495.57	0.05	34.00	Sedang
14	07/10/2024, 15.54.45	3.11	495.57	0.05	15.00	Baik
15	07/10/2024, 15.50.55	3.11	495.57	0.05	15.00	Baik
16	07/10/2024, 15.50.52	3.11	495.57	0.05	14.00	Baik
17	07/10/2024, 15.50.51	3.11	495.57	0.05	14.00	Baik
18	07/10/2024, 15.50.50	3.11	495.57	0.05	14.00	Baik
19	07/10/2024, 15.50.48	3.11	495.57	0.05	14.00	Baik
20	07/10/2024, 15.50.46	3.11	495.57	0.05	2.00	Baik
21	07/10/2024, 15.46.54	3.11	495.57	0.05	2.00	Baik
22	07/10/2024, 15.46.53	3.11	498.68	0.05	2.00	Baik
23	07/10/2024, 15.46.52	3.11	498.68	0.05	2.00	Baik
24	07/10/2024, 15.46.51	3.11	498.68	0.05	2.00	Baik
25	07/10/2024, 15.46.50	3.07	498.68	0.05	2.00	Baik
26	07/10/2024, 15.46.48	3.17	498.68	0.05	17.00	Sedang
27	07/10/2024, 15.42.57	3.17	498.68	0.05	9.00	Baik
28	07/10/2024, 15.42.55	3.17	501.81	0.05	9.00	Baik
29	07/10/2024, 15.42.54	3.17	501.81	0.05	9.00	Baik
30	07/10/2024, 15.42.53	3.17	501.81	0.05	9.00	Baik

D. PENGUJIAN CONFUSION MATRIX

Hasil akurasi penggunaan metode KNN untuk kualitas udara ini mendapatkan 98,67% di mana sistem berhasil memprediksi data benar sebanyak 592 dari 600 data. Selanjutnya dilakukan pengujian confusion matrix dengan hasil pada Tabel 5.

TABEL V
HASIL PENGUJIAN CONFUSION MATRIX

Aktual	Prediksi				
	Baik	Sedang	Buruk	S. Buruk	Total
Baik	74	7	0	0	81
Sedang	1	217	0	0	218
Buruk	0	0	144	0	144
S.Buruk	0	0	0	157	157
Total	75	224	144	157	600

Perhitungan akurasi dengan confusion matrix terhadap metode KNN adalah sebagai berikut:

$$\text{Akurasi} = \frac{592}{600} \times 100\%$$

$$\text{Akurasi} = 98,67\%$$

Adapun perhitungan presisi dan recall untuk kelas baik sebagai berikut.

$$\text{Presisi} = \frac{74}{(74+1)} = 0.987$$

$$\text{Recall} = \frac{74}{(74+7)} = 0.9136$$

Adapun perhitungan presisi dan recall untuk kelas sedang sebagai berikut.

$$\text{Presisi} = \frac{217}{(217+17)} = 0.969$$

$$\text{Recall} = \frac{217}{(217+1)} = 0.995$$

Adapun perhitungan presisi dan recall untuk kelas buruk sebagai berikut.

$$\text{Presisi} = \frac{144}{(144+0)} = 1$$

$$\text{Recall} = \frac{144}{(144+0)} = 1$$

Adapun perhitungan presisi dan recall untuk kelas sangat buruk sebagai berikut.

$$\text{Presisi} = \frac{157}{(157+0)} = 1$$

$$\text{Recall} = \frac{157}{(157+0)} = 1$$

Adapun perhitungan rata-rata presisi dan recall dihitung seperti berikut.

$$\text{Presisi} = \frac{(0.987 + 0.969 + 1 + 1)}{4} \times 100\% = 98.9\%$$

$$\text{Recall} = \frac{(0.913 + 0.995 + 1 + 1)}{4} \times 100\% = 97.73\%$$

V. KESIMPULAN

Berdasarkan hasil analisis dan pengujian yang telah dilakukan terhadap sistem klasifikasi kualitas udara, dapat disimpulkan bahwa sistem yang dirancang mampu mengidentifikasi tingkat kualitas udara secara akurat menggunakan empat parameter utama, yaitu karbon dioksida (CO_2), karbon monoksida (CO), ozon (O_3), dan partikel halus ($\text{PM}_{2.5}$). Sistem ini berhasil dibangun dengan memanfaatkan integrasi sensor lingkungan yang terhubung dengan mikrokontroler Arduino Uno R3 sebagai unit kendali utama, ESP32 sebagai penghubung jaringan (gateway), serta Firebase sebagai media penyimpanan data secara real-time. Dengan menerapkan algoritma K-Nearest Neighbor (KNN), sistem menunjukkan performa klasifikasi yang sangat baik, dengan tingkat akurasi keseluruhan mencapai 98,67%. Selain itu, sistem mampu mengklasifikasikan kondisi udara dalam kategori "Buruk" dan "Sangat Buruk" dengan presisi sempurna sebesar 100%, sedangkan kategori "Sedang" mencapai presisi sebesar 96,88% dan kategori "Baik" mencapai presisi 98,67%. Hasil ini menunjukkan bahwa pendekatan yang digunakan tidak hanya efektif dari sisi teknis, tetapi juga memiliki potensi signifikan dalam mendukung sistem pemantauan kualitas udara yang cerdas dan responsif secara real-time.

REFERENSI

- [1] A. Amalia, A. Zaidiah, and I. N. Isnainiyah, "PREDIKSI KUALITAS UDARA MENGGUNAKAN ALGORITMA K-NEAREST NEIGHBOR," *JIPI (Jurnal Ilmiah Penelitian dan Pembelajaran Informatika)*, vol. 7, no. 2, pp. 496–507, 2022, [Online]. Available: <https://data.jakarta.go.id/>.
- [2] I. I. Ridho and G. Mahalisa, "ANALISIS KLASIFIKASI DATASET INDEKS STANDAR PENCEMARAN UDARA (ISPU) DI MASA PANDEMI MENGGUNAKAN ALGORITMA SUPPORT VECTOR MACHINE (SVM)," *Technologia: Jurnal Ilmiah*, vol. 14, no. 1, pp. 38–41, May 2022, doi: 10.12928/biste.v4i1.6079.
- [3] S. R. Cholil, T. Handayani, R. Prathivi, and T. Ardianita, "Implementasi Algoritma Klasifikasi K-Nearest Neighbor (KNN) Untuk Klasifikasi Seleksi Penerima Beasiswa," *IJCIT (Indonesian Journal on Computer and Information Technology)*, vol. 6, no. 2, pp. 118–127, 2021.
- [4] R. Purbakawaca and S. A. Fauzan, "Rancang Bangun Sistem Pemantauan Kualitas Udara Dalam Ruangan Berbiaya Rendah Berbasis IoT," *Jurnal Talenta Sipil*, vol. 5, no. 1, p. 118, Feb. 2022, doi: 10.33087/talentasipil.v5i1.104.
- [5] A. E. Putra and T. Rismawan, "Klasifikasi Kualitas Udara Berdasarkan Indeks Standar Pencemaran Udara (ISPU) Menggunakan Metode Fuzzy Tsukamoto," *Coding: Jurnal Komputer dan Aplikasi*, vol. 11, no. 2, pp. 190–196, 2023.
- [6] I. Alexander Rombang, L. Bambang Setyawan, and G. Dewantoro, "Perancangan Prototipe Alat Deteksi Asap Rokok dengan Sistem Purifier Menggunakan Sensor MQ-135 dan MQ-2," *Ultima Computing: Jurnal Sistem Komputer*, vol. 21, no. 1, pp. 131–144, 2022.
- [7] A. A. Rosa, B. A. Simon, and K. S. Lieanto, "Sistem Pendekripsi Pencemaran Udara Portabel Menggunakan Sensor MQ-7 dan MQ-135," *Ultima Computing: Jurnal Sistem Komputer*, vol. 12, no. 1, pp. 23–28, 2020.
- [8] D. Prasetyo, Ibrahim, W. Nurrul Adzilla, and Y. Saragih, "Implementasi Pemantauan Kualitas Udara dengan Menggunakan MQ-7 dan MQ-131 Berbasis Internet of Things," *Journal of Electrical Technology*, vol. 6, no. 1, pp. 18–22, 2021.
- [9] D. S. Lasmana and E. Fitriani, "RANCANG BANGUN PROTOTYPE ROBOT PENGHISAP DEBU MENGGUNAKAN OPTICAL DUST SENSOR GP2Y10AU0F," *Bina Darma Conference on Engineering Science (BDCES)*, vol. 2, no. 1, pp. 20–29, 2020, [Online]. Available: <http://conference.binadarma.ac.id/index.php/BDCES>
- [10] Siswanto, I. Rojikin, and W. Gata, "Pemanfaatan Sensor Suhu DHT-22, Ultrasonik HC-SR04 Untuk Mengendalikan Kolam Dengan Notifikasi Email," *Jurnal RESTI (Rekayasa Sistem Dan Teknologi Informasi)*, vol. 3, no. 3, pp. 544–551, 2019.
- [11] R. Arrahman and C. Bella, "RANCANG BANGUN PINTU GERBANG OTOMATIS MENGGUNAKAN ARDUINO UNO R3," *Portaldatas.org*, vol. 2, no. 2, p. 1, 2022.
- [12] A. Sanaris and I. Suharjo, "Prototype Alat Kendali Otomatis Penjemur Pakaian Menggunakan NodeMCU ESP32 Dan Telegram Bot Berbasis Internet of Things (IOT) Prototype Automatic Drying Tool Using NodeMCU ESP32 and Telegram Bot Based on Internet of Things (IOT)," *Journal Of Information System And Artificial Intelligence*, vol. 1, no. 1, pp. 17–24, 2020.
- [13] D. Jollyta, Prihandoko, A. Hajjah, E. Haerani, and M. Siddik, *Algoritma Klasifikasi untuk Pemula Solusi Python dan RapidMiner*. Deepublish, 2023.
- [14] H. Azis, P. Purnawansyah, F. Fattah, and I. P. Putri, "Performa Klasifikasi K-NN dan Cross Validation Pada Data Pasien Pengidap Penyakit Jantung," *ILKOM Jurnal Ilmiah*, vol. 12, no. 2, pp. 81–86, Aug. 2020, doi: 10.33096/ilkom.v12i2.507.81-86.
- [15] L. Firdaus and T. Setiadi, "Perbandingan Algoritma Naive Bayes, Decision Tree, dan KNN untuk Klasifikasi Produk Populer Adidas US dengan Confusion Matrix," *Jurnal Sistem Komputer dan Informatika (JSON)*, vol. 5, no. 2, p. 185, Dec. 2023, doi: 10.30865/json.v5i2.6124.
- [16] H. Subagiyo et al., "Rancang Bangun Sensor Node untuk Pemantauan Parameter Kualitas Udara," *Jurnal Sains, Teknologi dan Industri*, vol. 18, no. 1, pp. 72–79, 2020.
- [17] R. Hidayati et al., "SISTEM PEMANTAUAN KUALITAS UDARA SECARA REAL-TIME MENGGUNAKAN ESP32 DAN TEKNOLOGI IOT," *Jurnal Teknologi Informatika*, vol. 5, no. 2, 2024, doi: 10.46576/djtechno.