

Sistem Klasifikasi Kualitas Air dalam Akuakultur Budidaya Ikan Lele dengan Algoritma PCA dan KNN

Fadhli Aulady¹, Dahnia Syauqy², Rekyan Regasari Mardi Putri³

Program Studi Teknik Komputer, Fakultas Ilmu Komputer, Universitas Brawijaya
Email: ¹fadhlideede3@gmail.com, ²dahnial87@ub.ac.id, ³rekyan.rmp@ub.ac.id

Abstrak

Akuakultur merupakan salah satu metode budidaya organisme air seperti ikan, tanaman air, dan lain lain untuk memenuhi kebutuhan pangan manusia. Meskipun akuakultur sangat menjanjikan dan memiliki banyak manfaat, sistem ini membutuhkan pengelolaan kualitas air yang efektif. Untuk menjaga agar kualitas air kolam ikan tetap baik tentunya diperlukan pengecekan beberapa parameter air secara rutin oleh pembudidaya. Sehingga, dapat dikembangkan sistem klasifikasi kualitas air pada sistem akuakultur yang dapat dimonitoring kapan saja dan dimana saja. Dalam melakukan klasifikasi, sistem ini menggunakan metode KNN. Namun, KNN memiliki keterbatasan, seperti kebutuhan memori dan kompleksitas waktu. Dari permasalahan tersebut maka dibuatlah sebuah sistem klasifikasi menggunakan menggunakan mikrokontroler ESP-32, sensor DS18B20, sensor pH-4502C, sensor TDS, dan sensor Turbidity dengan menggunakan algoritma KNN dan PCA. Penelitian ini bertujuan untuk membuat sistem monitoring kualitas air serta mengetahui dampak penggunaan PCA dalam mengatasi kekurangan KNN. Pengujian penelitian dilakukan dengan menguji akurasi KNN serta membandingkan penggunaan memori dan kecepatan komputasi KNN dengan menggunakan metode PCA dan tanpa menggunakan metode PCA. Hasil yang diperoleh dari pengujian ialah akurasi klasifikasi PCA+KNN sebesar 80%, memori yang digunakan 240 bytes lebih hemat saat menggunakan PCA, dan komputasi lebih cepat dengan rata-rata waktu 17.2 μ s dengan menggunakan PCA.

Kata kunci: *Akuakultur, monitoring, KNN, PCA.*

Abstract

Aquaculture is one of the methods of cultivating aquatic organisms such as fish, aquatic plants, and others to meet the human food needs. Although aquaculture is very promising and has many benefits, this system requires effective water quality management. To maintain the good water quality of fish ponds, regular checking of several water parameters by cultivators is necessary. Thus, a water quality classification system in aquaculture systems that can be monitored anytime and anywhere can be developed. In performing the classification, this system uses the K-Nearest Neighbors (KNN) method. However, KNN has limitations, such as memory requirements and time complexity. To address these issues, a classification system was created using the ESP-32 microcontroller, DS18B20 sensor, pH-4502C sensor, TDS sensor, and Turbidity sensor with the KNN and PCA algorithms. This study aims to create a water quality monitoring system and understand the impact of using PCA to address the limitations of KNN. The research was conducted by testing the accuracy of KNN and comparing the memory usage and computation speed of KNN with and without the PCA method. The results obtained from the testing show that the accuracy of the PCA+KNN classification is 80%, and memory usage is 240 bytes less when using PCA, with faster computation at an average time of 17.2 μ s when using PCA.

Keywords: *Aquaculture, monitoring, KNN, PCA.*

1. PENDAHULUAN

Indonesia adalah negara dengan populasi lebih dari 270 juta jiwa, dan produksi pangan telah menjadi masalah yang semakin mendesak

seiring dengan pertumbuhan penduduk yang terus meningkat. Dengan meningkatnya kerawanan pangan, ada kebutuhan untuk mengembangkan metode pertanian atau peternakan inovatif. Salah satu solusi untuk

permasalahan tersebut ialah sistem akuakultur karena memiliki potensi untuk menyediakan nutrisi penting yang berkelanjutan bagi penduduk Indonesia.

Akuakultur merupakan budidaya organisme air seperti ikan, tanaman air, dan lain lain untuk memenuhi kebutuhan pangan manusia. Budidaya organisme akuatik termasuk ikan, moluska, krustasea, dan tanaman air. Peningkatan efisiensi dan regulasi yang tepat menjadi penting dalam akuakultur, pemantauan kualitas air harus menjadi sumber informasi pertama untuk akuakultur yang tepat (daoliang li & Shuangyin liu, 2019).

Keberhasilan akuakultur membutuhkan pengelolaan kualitas air yang efektif. Salah satu cara yang dapat dilakukan untuk memaksimalkan produksi adalah dengan menjaga kualitas air yang digunakan dalam membudidaya ikan lele tersebut (Fadillah A, 2019). Beberapa parameter yang dapat mempengaruhi kualitas sebuah air contohnya adalah pH, kekeruhan, suhu, kadar zat terlarut, kadar oksigen, dsb. Masing-masing variabel perlu diperhatikan untuk menjaga kualitas agar kualitas air tetap bagus. Untuk menjaga agar kualitas air kolam ikan tetap baik tentunya diperlukan pengecekan air secara rutin oleh pembudidaya dan menggantinya dengan air yang baru. Namun hal ini akan melelahkan para pembudidaya, serta membahayakan ikan yang dibudidaya jika mereka lupa memperhatikan kualitas air.

Penelitian ini akan berfokus pada algoritma KNN (K-Nearest Neighbor) untuk melakukan klasifikasinya. KNN merupakan algoritma klasifikasi yang sering digunakan karena mudah dan ringan. KNN merupakan algoritma klasifikasi yang sering digunakan karena mudah dan ringan. KNN dipilih karena kemampuannya untuk menangani data dengan pola non-linear dan dapat mengklasifikasikan data ke dalam beberapa kelas tanpa perlu mengatur batasan khusus. Klasifikasi menggunakan batasan hasilnya tidak selalu optimal karena batasan tersebut hanya merupakan nilai aman pada setiap kondisi, dimana nilai parameter optimal akan berbeda sesuai lingkungan, lokasi, cuaca, dan lainnya. Selain itu, klasifikasi menggunakan batasan tidak dapat mempertimbangkan efek sinergis dan antagonis dari parameter yang berbeda.

Algoritma KNN memiliki keterbatasan, seperti kebutuhan memori dan kompleksitas waktu, karena sepenuhnya tergantung pada

setiap contoh dalam set pelatihan (Hassanat et al, 2014). Algoritma KNN dalam implementasinya akan mengambil beberapa penyimpanan pada memori untuk menyimpan data-data yang akan digunakan. Selain itu algoritma KNN yang bekerja dengan menghitung jarak data uji dengan seluruh dataset akan membuat komputasi lebih lama jika dimensi atau fitur pada dataset banyak. Untuk mengatasi masalah tersebut maka perlu dilakukan pengurangan fitur pada dataset yaitu menggunakan metode PCA (Principal Component Analysis).

Algoritma PCA merupakan algoritma yang digunakan untuk mereduksi fitur atau dimensi suatu data menjadi lebih sedikit. Algoritma PCA akan mereduksi fitur pada data namun tetap mempertahankan informasi-informasi penting yang ada pada data. PCA dapat digunakan bersamaan dengan algoritma KNN untuk mengatasi beberapa masalah seperti kompleksitas komputasi dan kebutuhan memori. Selain itu, penerapan metode PCA pada tahap pre-processing telah terbukti efektif dalam meningkatkan waktu komputasi dan akurasi (Hediyati & Suartana, 2021). PCA nantinya akan digunakan untuk membantu algoritma KNN dalam melakukan klasifikasi dengan cara mengurangi dimensi atau fitur yang ada pada dataset sehingga akan mengurangi kompleksitas komputasi dan memori yang digunakan.

Dari beberapa permasalahan tersebut, dapat dikembangkan sistem kualitas air pada sistem akuakultur dengan menggunakan sensor pH, sensor TDS, sensor kekeruhan air, dan sensor suhu serta menggunakan ESP-32 sebagai unit pemrosesan. ESP-32 adalah mikrokontroler yang memiliki memori dan daya pemrosesan terbatas sehingga PCA dapat membantu mengurangi dimensi data dan hanya menyimpan fitur terpenting. Data-data yang sudah diperoleh dari semua sensor akan diproses melalui algoritma PCA (Principal Component Analysis) dan algoritma KNN (K-Nearest Neighbor). ESP-32 akan melakukan klasifikasi data dan mengirimkannya ke platform blynk untuk memantau parameter kualitas air secara daring dan real-time. Hal ini dapat membantu pembudidaya untuk dengan cepat mendeteksi perubahan kualitas air dan mengambil tindakan yang tepat untuk mempertahankan kondisi optimal bagi pertumbuhan dan kesehatan ikan.

Penelitian ini bertujuan untuk mengetahui performa dan akurasi dalam klasifikasi kualitas air menggunakan metode KNN. Penelitian ini juga berfokus pada dampak dari penerapan

metode PCA dalam mengatasi kekurangan-kekurangan KKN seperti penggunaan memori dan kompleksitas komputasi. Manfaat dari penelitian ini adalah dapat memotivasi, serta mempermudah masyarakat dalam melakukan kegiatan akuakultur.

2. DASAR TEORI

2.1 Kolam Budidaya Ikan Sistem Akuakultur

Budidaya ikan menggunakan sistem akuakultur merupakan kolam yang membudidayakan ikan ataupun biota air lainnya dengan tujuan untuk meningkatkan produksi pangan. Seiring waktu berjalan akan muncul zat amonia dari kotoran ikan dan sisa pangan pada kolam, zat amonia ini merupakan zat yang beracun bagi ikan yang ada di kolam tersebut. Jika terdapat zat amonia dalam jumlah besar pada kolam ikan, aka air tersebut disebut “sedang mengalami pengotoran (Pollution)” (Metcalf dan Eddy, 1991).

2.2 MinMax Normalization

Min-Max normalization merupakan teknik normalisasi yang digunakan untuk mengubah data dalam database mempunyai nilai dengan kisaran serupa atau rentang yang sama. *Min-Max normalization* adalah teknik yang menskalakan fitur yang bergantung pada bobot maksimum fitur dan bobot minimum fitur vektor numerik (Syahputra et al, 2020). Karena teknik ini menskalakan fitur berdasarkan bobot maksimum dan minimum fitur, maka rentang nilai data yang akan dikonversi akan berada di rentang nilai 0 sampai nilai 1. Persamaan 2.1 merupakan persamaan min-max normalization:

$$X_{scaled} = \frac{(X - X_{min})}{(X_{Max} - X_{Min})} \tag{1}$$

Dimana,

- X_{scaled} = Data hasil konversi
- X = Data yang dikonversi
- X_{min} = Data minimum database database
- X_{max} = Data maksimum dalam database

2.3 Principal Component Analysis (PCA)

Principal Component Analysis (PCA) merupakan salah satu teknik reduksi dimensi pada pembelajaran mesin. PCA mengurangi dimensi dengan membuat variabel-variabel baru namun tetap mempertahankan informasi

sebanyak mungkin yang terkandung dalam data tersebut. Dengan mengurangi dimensi data, data akan lebih mudah dipahami, visualisasi, dianalisis, serta dapat menghilangkan variabel yang tidak penting. Dengan menggunakan metode PCA maka akan mengurangi kompleksitas dimensi, kompleksitas komputasi, dan juga persyaratan penyimpanan. Metode PCA bekerja dengan cara menghitung covariance matrix dari data, dan kemudian mencari eigen vectors, dan eigen values (Nugraha & Wiguna, 2020). Daya tarik utama dari PCA adalah mereka mengambil variabel-variabel yang banyak dan menyaringnya menjadi jumlah variabel yang lebih kecil dan lebih mudah dikelola sambil tetap mempertahankan sebagian besar informasi yang terkandung dalam variabel asli (beattie & Esmonde-White,2022). Langkah-langkah untuk mengurangi dimensi menggunakan algoritma PCA, menurut Adiwijaya (2018), adalah sebagai berikut:.

1. Pertama biarkan X menjadi matrix input untuk PCA. Dimana X merupakan data latih yang terdiri dari n-vektor dengan dimensi data m.
2. Hitung nilai mean atau rata-rata data dari setiap dimensi (X) menggunakan persamaan 2:

$$\bar{X} = \frac{1}{n} \sum_{i=1}^n X_i \tag{2}$$

Dimana,

- n = Jumlah sampel data
- X_i = Data sampel

3. Setelah itu hitung matrix kovarians untuk mengidentifikasi korelasi menggunakan persamaan 3:

$$C_X = \frac{1}{n - 1} \sum_{i=1}^n (X - \bar{X})(X_i - \bar{X})^T \tag{3}$$

Dimana,

- n = Jumlah sampel data
- X_i = Data sampel
- \bar{X} = Mean data
- C_X = Matriks kovarians

4. Hitung nilai eigen (eigen value) dan eigen vektor dari matriks kovarians yang sudah didapat menggunakan persamaan 4:

$$C_X v_m = \lambda_m v_m \tag{4}$$

Dimana,

C_x = Matriks kovarians
 v_m = *Eigen* Vektor
 λ_m = *Eigen* value

5. Langkah selanjutnya adalah mengurutkan eigen value dari yang terbesar ke terkecil.
6. Komponen utama (PC) adalah kumpulan eigen vektor sesuai dengan eigen value yang sudah diurutkan pada langkah 5.
7. Dimensi PC akan dikurangi berdasarkan jumlah K eigen value yang dipilih
8. Data yang sampel atau data uji dapat dikonversi dengan menggunakan menggunakan persamaan dibawah ini

$$d = \left((V_m)^T \left(\sum_{i=1}^n (X - \bar{X})(X_i - \bar{X})^T \right) \right)^T \quad (5)$$

Dimana,

d = Data yang ingin dikonversi
 n = Jumlah sampel data
 V_m = *Eigen vector*
 \bar{X} = Mean data
 X_i = Data sampel

Penelitian ini mengambil jumlah K *eigen value* dengan menggunakan persamaan 6. Persamaan ini akan mencari total varians yang diambil dari data aslinya, sehingga akan terlihat berapa persentase varians yang terkandung data setelah dilakukan konversi PCA.

$$VP = \frac{\sum_{i=1}^k \lambda_i}{\sum_{j=1}^n \lambda_j} * 100\% \quad (6)$$

Dimana,

λ_i = *Eigen Value*
 k = Jumlah eigen value yang digunakan
 n = Jumlah seluruh eigen value

2.4 KNN (K-Nearest Neighbor)

K-nearest neighbor (KNN) merupakan sebuah algoritma yang digunakan dalam pembelajaran mesin dalam melakukan klasifikasi maupun regresi. KNN umumnya digunakan karena kemudahan interpretasi dan waktu komputasi yang relatif rendah (Moldagulova & Sulaiman, 2017). Algoritma KNN bekerja dengan cara mencari K tetangga terdekat dari data yang ingin diprediksi dan menggunakan nilai tetangga tersebut untuk memprediksi kelas. KNN sering digunakan dalam berbagai aplikasi, termasuk pengenalan pola, pengolahan citra, dsb.

K-Nearest Neighbor merupakan algoritma supervised learning yang hasil klasifikasinya atau regresinya diambil dari mengelompokkan data baru dengan K tetangga terdekat (Hidayat & Perdana, 2017). Perhitungan K tetangga terdekat dapat dilakukan dengan menggunakan perhitungan jarak seperti jarak euclidean distance, manhattan distance, haversine formula, dsb. Dalam algoritma KNN, perhitungan jarak umumnya menggunakan euclidean distance sehingga untuk rumus dari KNN tersebut menggunakan rumus euclidean distance. Rumus algoritma KNN dapat dilihat pada persamaan 7 (Nurjanah et al, 2020).

$$d(x_i, x_j) = \sqrt{\sum_{r=1}^n (a_r(x_i) - a_r(x_j))^2} \quad (7)$$

Dimana,

$d(x_i, x_j)$ = Jarak Euclidean
 (x_i) = Data peroleh ke -i
 (x_j) = Data peroleh ke-j
 (a_r) = Data ke-r di dataset
 I_j = 1,2,3...n
 N = Jumlah data training

3. METODOLOGI PENELITIAN

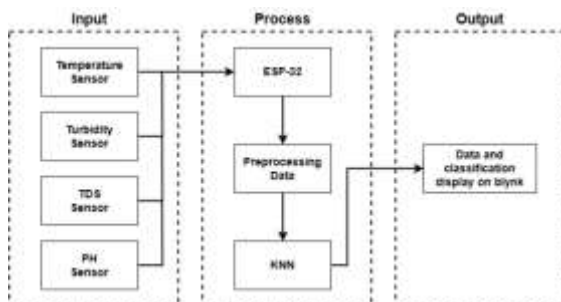
Tipe penelitian ini merupakan penelitian implementatif sehingga penelitian ini bertujuan untuk membuat produk atau artefak sebagai solusi terhadap permasalahan. Penelitian ini berupa pengembangan lanjut (enhancement) yaitu membuat sebuah produk yang dikembangkan dari sistem yang sudah ada. Penelitian ini didasari pada tahapan yang sistematis dalam melakukan pengembangan lanjut dari alat yang telah dibuat sebelumnya. Langkah pertama yang perlu dilakukan adalah studi literatur. Studi literatur berfungsi untuk pemahaman dasar teori yang berhubungan dalam proses perancangan dan implementasi penelitian. Langkah selanjutnya adalah analisis kebutuhan untuk mengetahui proses yang dilakukan oleh sistem. Jika tahap analisis kebutuhan sudah selesai, maka langkah selanjutnya adalah perancangan software dan hardware. Langkah selanjutnya adalah mengimplementasikan rancangan sistem yang sudah dibuat menjadi alat. Jika sudah diimplementasi, maka alat perlu dilakukan analisis agar dapat dilihat performa kinerja alat tersebut. Apabila hasil analisis sudah sesuai dengan yang diharapkan, maka langkah selanjutnya adalah membuat kesimpulan dan

saran. Jika alat belum menghasilkan hasil yang diharapkan maka dilakukan perancangan ulang.

4. PERANCANGAN & IMPLEMENTASI

4.1 Perancangan Sistem

Gambaran perancangan sistem secara keseluruhan ditunjukkan pada Gambar 1 dalam bentuk blok diagram.

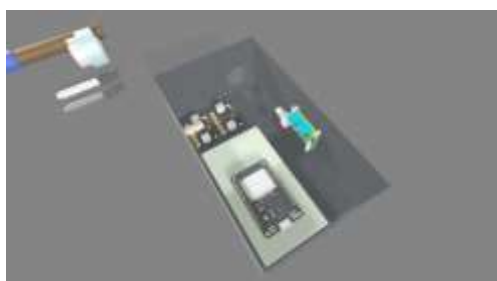


Gambar 1. Blok Diagram Sistem

Pada proses perancangan sistem klasifikasi kualitas air pada kolam ikan lele akan dibagi menjadi 3 bagian yaitu input, proses, dan output. Input sistem terdiri dari sensor suhu, sensor kekeruhan, sensor TDS, dan sensor pH. Nilai sensor akan diproses pada ESP-32 menggunakan *Minmax scaling* dan metode PCA, setelah dilakukan PCA maka data akan diklasifikasi menggunakan metode KNN. Data dan hasil klasifikasi akan di tampilkan pada platform *blynk* sebagai keluarannya.

4.2 Perancangan Fisik Sistem

Perancangan ilustrasi 3 dimensi menggambarkan keseluruhan desain untuk membantu meletakkan perangkat keras yang digunakan dalam sebuah wadah. Ilustrasi alat ditampilkan pada Gambar 2.



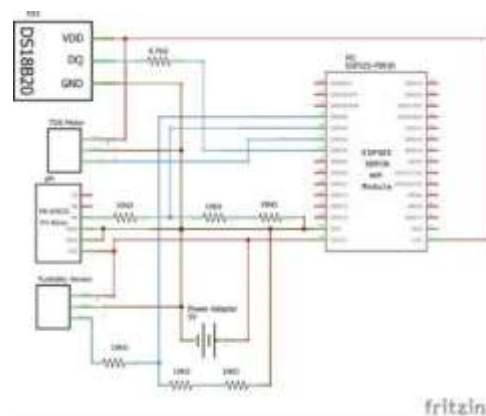
Gambar 2. Ilustrasi 3 Dimensi Sistem

Pada gambar 2 terdapat sebuah wadah kotak berwarna hitam yang akan digunakan sebagai wadah untuk penempatan mikrokontroler dan modul-modul. Wadah ini terbuat dari plastik dan akan dibuat 2 bolongan

untuk menyambungkan sensor dengan modul-modul yang ada di dalam kotak dan juga untuk menyambungkan adapter DC sebagai sumber daya. Mikrokontroler ESP-32 akan disambungkan pada PCB hijau dan untuk penempatan modul dan PCB digunakan mur dan baut nylon agar modul tidak berserakan.

4.3 Perancangan Perangkat Keras

Perancangan serta konfigurasi pin dari setiap sensor-sensor yang dihubungkan ke ESP-32 ditunjukkan pada skematik diagram pada gambar 3 dibawah ini.



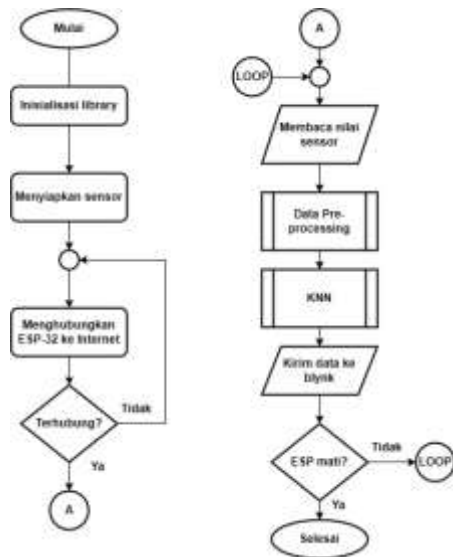
Gambar 3. Diagram Skematik Perangkat Keras

Sensor pH dan sensor kekeruhan mengeluarkan keluaran bernilai 5V, karena ESP-32 tidak bisa menerima masukan lebih dari 3.3V maka perlu ditambah voltage divider pada sensor pH dan sensor kekeruhan untuk mengubah output dari 5V ke 3.3V. Keluaran dari sistem ini adalah berupa data atau nilai dari setiap sensor dan juga klasifikasi yang akan ditampilkan pada platform *blynk*. Data-data sensor yang sudah diperoleh dan dikelola oleh ESP-32 akan dikirimkan ke platform *blynk* menggunakan jaringan internet.

4.4 Perancangan Perangkat Lunak

Diagram alir sistem secara keseluruhan dapat dilihat pada Gambar 4. Sistem diawali dengan memasukkan sensor yang sudah dihubungkan ke ESP-32 dan menyambungkan ESP-32 ke jaringan internet. ESP-32 yang sudah terhubung jaringan internet akan membaca data dari seluruh sensor. ESP-32 akan mengelola data yang sudah didapat dan melakukan klasifikasi menggunakan algoritma KNN. Jika sudah diklasifikasi maka ESP-32 akan mengirimkan datanya ke aplikasi *blynk* melalui jaringan internet. Setelah data terkirim maka aplikasi *blynk* akan menampilkan data sensor beserta

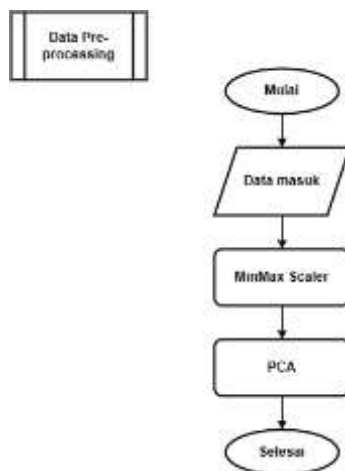
klasifikasinya.



Gambar 4. Flowchart Sistem Klasifikasi Kualitas Air

4.4.1 Data Preprocessing

Data preprocessing pada sistem menggunakan *MinMax Scaler* yang selanjutnya akan dilakukan reduksi dimensi menggunakan metode PCA.



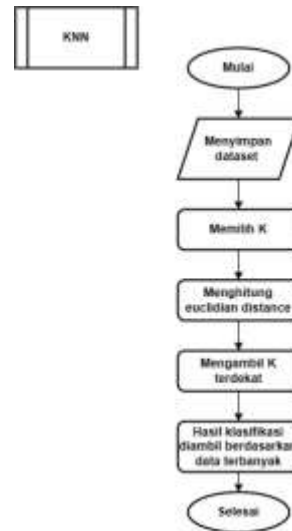
Gambar 5. Flowchart Data-preprocessing

Pada Gambar 5 terlihat bahwa data yang sudah masuk akan di normalisasi terlebih dahulu. Normalisasi digunakan untuk menyamakan rentang nilai yang ada. Metode normalisasi yang digunakan merupakan *MinMax Scaler* yang mengubah nilai menjadi nilai 0 hingga 1. Data yang sudah di normalisasi akan direduksi dimensinya menggunakan metode PCA, hasil dari PCA ini nantinya akan digunakan sebagai klasifikasi menggunakan metode KNN.

4.4.2 Klasifikasi KNN

Metode KNN akan dijalankan jika data

sudah diproses melalui *MinMax Scaler* and metode PCA. Alur program klasifikasi KNN akan ditampilkan pada Gambar 6.



Gambar 6. Flowchart Metode KNN

Pada gambar 7, dapat terlihat bahwa KNN bermula dengan menyimpan semua dataset yang dipakai. Perlu adanya dataset karena KNN melakukan klasifikasi dengan menghitung jarak data uji dengan seluruh dataset yang ada. Setelah program menyimpan dataset maka perlu memilih K yang ingin dipakai dan disarankan untuk menggunakan K dengan nilai ganjil. Data uji akan diukur jaraknya terhadap setiap dataset yang ada. Setelah mengetahui setiap jarak, maka diurutkan jarak terpendek ke jarak terjauh. Program akan mengambil K data dengan jarak terpendek dan hasil klasifikasi akan diambil berdasarkan data terbanyak..

4.5 Implementasi Sistem

Setelah Pada bagian ini akan memperlihatkan implementasi perangkat keras yang sudah dirancang sebelumnya. Berikut merupakan gambar implementasi perangkat yang sudah dibuat.



Gambar 8. Implementasi Sistem

Agar sistem dapat berjalan sesuai fungsinya

maka perlu pengunggahan kode program sesuai flowchart yang sudah dibuat. Kode program akan ditulis menggunakan arduino IDE dan diunggah ke ESP-32 menggunakan kabel USB.

5. PENGUJIAN & ANALISIS

5.1 Pengujian Pembacaan Data Sensor

Pengujian bertujuan untuk menguji sensor yang diimplementasikan pada alat berfungsi atau tidak. Metode yang digunakan adalah memasukkan ke lima sensor kedalam dua air yang berbeda lalu melihat hasil pada serial monitor. Pengujian dilakukan sebanyak lima kali dengan menggunakan dua air yang berbeda. Hasil pengujian yang diperoleh disajikan pada Tabel 1 dan Tabel 2.

Tabel 1. Hasil Pengujian Air Pada Ember A

No	Suhu	TDS	pH	Kekeruhan
1	26.06	224.05	7.94	2.17
2	25.87	222.11	7.66	2.09
3	25.51	219.91	6.83	2.10
4	25.31	215.56	6.80	2.11
5	26.25	226.24	6.52	2.11

Tabel 2. Hasil Pengujian Air Pada Ember B

No	Suhu	TDS	pH	Kekeruhan
1	37.06	167.68	6.81	2.26
2	37.13	166.69	6.57	2.25
3	37.06	167.02	6.28	2.26
4	37.13	166.35	6.43	2.24
5	37.13	168.01	6.52	2.26

Dari tabel 1. dan 2 terlihat bahwa keempat sensor sudah membaca data air dengan baik. Terdapat perbedaan dari kedua air tersebut ialah kekeruhan, suhu, dan TDS. Untuk air pada ember B diisi dengan air hangat sedangkan ember A diisi dengan air dingin sehingga terlihat ada perbedaan suhu antara kedua data. Selain itu diberi beberapa cairan tambahan seperti sabun pada ember A untuk membuatnya lebih keruh sehingga terlihat ada perbedaan kekeruhan dan juga zat padat terlarut didalam air tersebut.

5.2 Pengujian Nilai Error Sensor

Pengujian ini bertujuan untuk mengetahui nilai error dari masing-masing sensor. Pengujian dikdilakukan dengan membandingkan hasil pengukuran sensor dengan hasil pengukuran alat ukur konvensional menggunakan rumus seperti di bawah ini.

$$Error = \frac{|Nilai\ pengujian - Nilai\ sebenarnya|}{Nilai\ sebenarnya} \cdot 100\% \quad (8)$$

Dikarenakan keterbatasan alat ukur konvensional, maka pengujian ini hanya

dilakukan pada sensor pH, padatan terlarut (TDS), dan suhu. Pengujian ini diambil menggunakan 2 air yang berbeda dan diambil saat pengambilan dataset dilapangan dengan mencatat nilai yang sudah didapat oleh sensor dan nilai yang didapat dari alat ukur konvensional. Hasil pengujian yang diperoleh akan disajikan pada Tabel 3.

Tabel 3. Hasil Uji Nilai Error Pada Sensor

No	Error(%)		
	Suhu	TDS	pH
1	8.7	6.8	10.3
2	7.9	3	12.5
3	5	2.8	6.1
4	7	5.9	7.1
5	1.6	2.5	3.2
6	4.9	2.8	4.3
7	2.6	0.44	1.7
8	3.2	10.6	1.6
9	6.4	6.5	17
10	3.4	5.3	32.7
Rata-rata	5.1	4.7	9.7

Dapat dilihat bahwa nilai error dari ketiga sensor yang sudah dibandingkan dengan nilai yang didapat melalui alat ukur konvensionalnya menghasilkan nilai rata-rata error sebesar 5.1% untuk sensor suhu, 4.7% untuk sensor TDS, dan 9.7% untuk sensor pH. Untuk nilai error dari ketiga sensor tersebut sudah bernilai cukup kecil sehingga sensor sudah dapat melakukan pengukuran yang sesuai.

5.3 Pengujian Nilai K dan Akurasi Pada KKN

Pengujian ini dilakukan untuk menguji nilai K pada algoritma KNN. Pemilihan nilai K pada KNN sangat mempengaruhi hasil dari klasifikasinya maka diperlukan pemilihan dan pengujian nilai K yang sesuai. Pengujian ini akan menguji nilai K dari nilai 3, 5, dan 7 dengan menguji menggunakan lima data uji yang sudah dikumpulkan.

Tabel 4. Hasil Perbandingan Nilai K

Data uji	Kelas asli	Hasil Klasifikasi		
		K=3	K=5	K=7
Data 1	Baik	Baik	Baik	Baik
Data 2	Baik	Tidak	Baik	Baik
Data 3	Baik	Baik	Baik	Baik
Data 4	Tidak Baik	Baik	Baik	Tidak Baik
Data 5	Tidak Baik	Baik	Baik	Baik

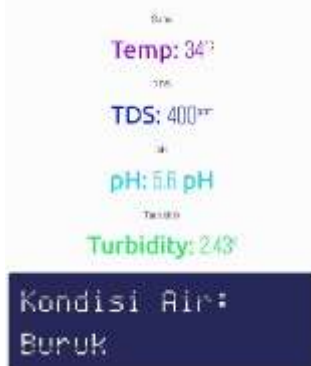
Dari tabel 4 dapat dilihat perbandingan hasil klasifikasi knn dengan kelas aslinya menggunakan K yang berbeda. Untuk K=3, terdapat dua klasifikasi yang benar dari lima data sehingga untuk K=3 memiliki akurasi sebesar 40%. Untuk K=5, terdapat tiga klasifikasi yang benar dari lima data sehingga untuk K=5 memiliki akurasi sebesar 60%. Terakhir untuk K=7, terdapat empat klasifikasi yang benar dari lima sehingga untuk K=7 memiliki akurasi sebesar 80%. Sehingga dari data tersebut maka sistem ini akan menggunakan K=7 dengan akurasi sebesar 80%.

5.4 Pengujian Pengiriman Data dari ESP-32 ke Blynk

Pada pengujian ini menguji apakah ESP-32 mampu mengirimkan data nilai sensor ke platform blynk. Pengujian ini dilakukan dengan mengirimkan data dummy (data buatan) melalui jaringan WiFi dan menampilkannya di platform blynk. Sistem ini ditampilkan hanya melalui platform blynk, sehingga perlu dilakukan pengujian untuk mengetahui apakah sistem dapat mengirimkan data melalui jaringan internet. Sistem akan diuji menggunakan sepuluh data dummy dan menghitung persentase keberhasilan dari pengiriman data.



Gambar 9. Mengamati Hasil Pada Serial Monitor



Gambar 10. Mengamati Nilai Pada Platform Blynk

Tabel 5. Hasil Pengiriman Data ESP-32 ke Blynk

Data	Status
Data 1	Berhasil terkirim
Data 2	Berhasil terkirim

Data 3	Berhasil terkirim
Data 4	Berhasil terkirim
Data 5	Berhasil terkirim
Data 6	Berhasil terkirim
Data 7	Berhasil terkirim
Data 8	Berhasil terkirim
Data 9	Berhasil terkirim
Data 10	Berhasil terkirim

Dapat dilihat pada gambar 9 dan 10 dimana data akan dikirimkan selesai data diolah melalui jaringan internet ke platform blynk. Data yang sudah diambil oleh sensor akan diolah terlebih dahulu lalu ESP-32 melalui jaringan internet akan mengirimkan nilai data beserta hasil klasifikasinya ke platform blynk, dapat terlihat bahwa data mulai dari data sensor dan juga hasil klasifikasi sukses terkirim dan sudah ditampilkan melalui platform blynk. Pengguna dapat mengakses platform blynk melalui ponsel ataupun laptop yang sudah terkoneksi jaringan internet. Dalam percobaan ini terdapat 10 data yang berhasil terkirim sehingga data akan 100% terkirim jika ada jaringan internet.

5.5 Pengujian Perbedaan Waktu Komputasi dan Penggunaan Memori

Pengujian ini merupakan pengujian perbedaan waktu dan penggunaan memori antara menggunakan PCA dan tidak menggunakan PCA dalam metode KNN. Tujuan dari pengujian ini adalah untuk mengetahui apakah PCA dapat mempercepat komputasi dan mengurangi penggunaan memori pada penggunaan algoritma KNN. Hasil pengujian akan disajikan pada tabel 6 dan 7.

Tabel 6. Hasil Perbandingan Penggunaan Memori

No	Pre-processing	Memori (bytes)
1	PCA	240
2	Tanpa PCA	480

Tabel 7. Hasil Perbandingan Komputasi Waktu

No	Waktu (µs)		Selisih (µs)
	PCA	Tanpa PCA	
1	3893	3906	13
2	3885	3906	21
3	3891	3910	19
4	3893	3905	12
5	3885	3906	21
Rata-rata			17.2

Pada tabel 6 dapat dilihat bahwa penggunaan PCA dapat membantu mengurangi kebutuhan memori yang digunakan oleh mikrokontroler dimana saat sistem menggunakan PCA, memori yang digunakan adalah sebesar 240 bytes dan

saat sistem tidak menggunakan PCA, memori yang digunakan adalah sebesar 480 bytes. Pada tabel 7 dapat dilihat bahwa penggunaan PCA juga dapat membantu meringankan dan mempercepat komputasi dengan selisih rata-rata 17.2 μ s. PCA dapat meringankan komputasi. Perlu menggunakan dataset dengan fitur atau dimensi yang besar untuk melihat perbedaan yang signifikan pada komputasi KNN menggunakan PCA. Namun, menggunakan fitur atau dimensi yang sedikit dan menggunakan PCA sudah dapat mengurangi komputasi yang dilakukan oleh KNN.

6. KESIMPULAN & SARAN

Berdasarkan pengujian yang telah dilakukan, hasil yang diperoleh menunjukkan, sistem telah diuji menggunakan ketiga nilai K menggunakan 5 data uji menghasilkan nilai akurasi untuk K=3 sebesar 40%, sedangkan untuk K=5 memiliki akurasi sebesar 60%, dan yang terakhir untuk K=7 memiliki akurasi sebesar 80% dalam klasifikasinya. Sehingga pemilihan K dalam sistem ini menggunakan K=7 dengan akurasi klasifikasi sebesar 80%. Kemudian dampak PCA ialah dapat mengurangi kompleksitas komputasi dan juga penggunaan memori yang diperlukan dalam klasifikasi. Algoritma KNN akan bermasalah jika ada banyak dataset dan fitur yang digunakan dalam klasifikasi dikarenakan KNN mewajibkan mikrokontroler untuk menyimpan semua dataset yang ada dan semakin banyak fitur dan data maka semakin banyak komputasi yang perlu dilakukan oleh KNN. Dengan menggunakan PCA maka mikrokontroler dapat menyimpan dataset yang lebih sedikit dan mempercepat komputasi KNN.

Saran untuk pengembangan selanjutnya ialah menambah sensor seperti sensor dissolved oxygen dan menggunakan sensor pH yang lebih baik seperti dfrobot pH meter, memperbanyak dataset sehingga penggunaan PCA pada algoritma KNN dapat terasa, membuat pengiriman data dengan basis LAN (Local Area Network) dan membuat database sendiri untuk membantu pembudidaya di area pelosok yang memiliki jaringan internet tidak stabil atau tidak memiliki akses internet, dan menambah baterai atau sumber daya lainnya untuk sistem agar tidak terkendala ketika ada pemadaman listrik.

7. DAFTAR PUSTAKA

Li, D., & Liu, S. (2019). Water Quality

Monitoring and Management (pp. 303-328). Academic Press. <https://doi.org/10.1016/B978-0-12-811330-1.00012-0>

Metcalf dan Eddy. 1991. Wastewater Engineering Treatment Disposal Reuse. McGraw-Hill Book Co, Singapore.

Raihan, D. H. (2021). Sistem Monitoring dan Kontrol Air Kolam Ikan Lele Berbasis IoT (Internet of Things).

Putrawan, I. G. H., Rahardjo, P., & Agung, I. G. A. P. R. (2019). Sistem Monitoring Tingkat Kekeuhan Air dan Pemberi Pakan Otomatis pada Kolam Budidaya Ikan Koi Berbasis NodeMCU. *Majalah Ilmiah Teknologi Elektro*, 19(1), 1. <https://doi.org/10.24843/mite.2020.v19i01.p01>

Aditya Nugraha, D., & Sartika Wiguna, A. (2020). SELEKSI FITUR WARNA CITRA DIGITAL BIJI KOPI MENGGUNAKAN METODE PRINCIPAL COMPONENT ANALYSIS Digital Image Selection of Coffee Seed Using Component Analysis Method. In *Research : Journal of Computer* (Vol. 3, Issue 1).

Adiwijaya, Wisesty, U. N., Lisnawati, E., Aditsania, A., & Kusumo, D. S. (2018). Dimensionality reduction using Principal Component Analysis for cancer detection based on microarray data classification. *Journal of Computer Science*, 14(11), 1521–1530. <https://doi.org/10.3844/jcssp.2018.1521.1530>

Beattie, J. R., & Esmonde-White, F. W. L. (2021). Exploration of Principal Component Analysis: Deriving Principal Component Analysis Visually Using Spectra. In *Applied Spectroscopy* (Vol. 75, Issue 4, pp. 361–375). SAGE Publications Inc. <https://doi.org/10.1177/0003702820987847>

Hassanat, A., Ali Abbadi, M., Awad Altarawneh, G., Alhasanat, A. A., Basheer Hassanat, A., & Ali Alhasanat, A. (2014). Solving the Problem of the K Parameter in the KNN Classifier Using an Ensemble Learning Approach Revising some biometrics work View project Cement

Stabilized Rammed Earth View project Solving the Problem of the K Parameter in the KNN Classifier Using an Ensemble Learning Approach. In IJCSIS) International Journal of Computer Science and Information Security (Vol. 12, Issue 8).

<https://www.researchgate.net/publication/265168466>

Hediyati, D., & Suartana, I. M. (n.d.). Penerapan Principal Component Analysis (PCA) Untuk Reduksi Dimensi Pada Proses Clustering Data Produksi Pertanian Di Kabupaten Bojonegoro.

Setya Perdana, R. (2017). Implementasi Algoritma Modified K-Nearest Neighbor (MKNN) untuk Klasifikasi Penyakit Demam. <http://j-ptiik.ub.ac.id>

Syahputra, H., Basyar, L. K., & Tamba, A. A. S. (2020). Setiment Analysis of Public Opinion on the Go-Jek Indonesia Through Twitter Using Algorithm Support Vector Machine. *Journal of Physics: Conference Series*, 1462(1). <https://doi.org/10.1088/1742-6596/1462/1/012063>

Nurjanah, S., Siregar, A. M., & Kusumaningrum, D. S. (2020). PENERAPAN ALGORITMA K-NEARESTNEIGHBOR (KNN)UNTUK KLASIFIKASI PENCEMARAN UDARA DI KOTA JAKARTA. *Scientific Student Journal for Information, Technology and Science*, 1 no 2.

Moldagulova, A., & Sulaiman, R. B. (2017). Using KNN Algorithm for Classification of Textual Documents. *International Conference on Information Technology (ICIT)*. <https://doi.org/10.1109/ICITECH.2017.8079924>