

LAPORAN
PROJEK KEGIATAN KEWIRAUSAHAAN



Oleh:

Muhammad Rizqulloh Akbar	NIM 21538141013
Muhammad Muzaki	NIM 21538141024
Didik Aswin Wijaya	NIM 21538144001

Dosen Pengampu:

Prof. Ir. Moh. Khairudin, M.T., Ph.D.

DEPARTEMEN PENDIDIKAN TEKNIK ELEKTRO
FAKULTAS TEKNIK UNIVERSITAS NEGERI YOGYAKARTA

2023

A. Ringkasan/abstract

Dalam salah satu komponen UAV yaitu ESC dapat diukur/menetukan Kesehatan menggunakan, K-Nearest Neighbors (KNN) dan Fuzzy Logic menjadi fokus utama yang dilakukan eksplorasi sebagai metode penentu dalam konteks penilaian spesifik. KNN, sebagai algoritma berbasis instansi, menggunakan data pelatihan dan data uji untuk memproyeksikan kategori suatu entitas dengan menentukan tetangga terdekat berdasarkan metrik jarak. Di sisi lain, Fuzzy Logic metode sukamoto, dengan konsep keanggotaan sebagian dan variabel fuzzy, digunakan untuk mengatasi ketidakpastian dalam mengevaluasi dan mengklasifikasikan kondisi tertentu. Dengan memberi perbandingan dan juga implementasi kedua metode, Pengembangan sistem ini memberikan wawasan mendalam tentang efektivitas dan aplikasi praktis KNN dan Fuzzy Logic dalam pengambilan keputusan spesifik, khususnya dalam domain penentuan kesehatan ESC.

B. Latar Belakang dan Kajian Teori

Latar Belakang

Elektronik Speed Controller (ESC) merupakan sebuah perangkat elektronik yang memiliki peran penting dalam kendali kecepatan motor listrik. Perangkat ini sering digunakan dalam berbagai aplikasi, terutama pada kendaraan remote-control seperti mobil RC, pesawat terbang RC, dan juga dalam industri robotika.

Penggunaan ESC telah mengalami perkembangan yang signifikan seiring dengan kemajuan teknologi dalam bidang elektronika dan kontrol motor. Sejak diperkenalkan pertama kali, ESC telah mengalami peningkatan kinerja yang cukup mencolok, mulai dari kecepatan respons hingga efisiensi energi yang lebih baik.

Awalnya, ESC digunakan secara luas dalam industri model remote-control untuk mengatur kecepatan motor brushless dan motor brushed. Namun, dengan perkembangan teknologi, aplikasi ESC telah meluas ke berbagai sektor, termasuk kendaraan listrik, drone, dan industri otomotif.

Peran ESC menjadi semakin penting seiring dengan pergeseran masyarakat menuju teknologi yang lebih ramah lingkungan. Kemampuannya untuk mengatur kecepatan motor secara efisien tidak hanya mengoptimalkan kinerja kendaraan atau perangkat, tetapi juga membantu mengurangi konsumsi energi, memperpanjang umur motor, dan mengurangi dampak negatif terhadap lingkungan.

Dalam konteks ini, pemahaman mendalam tentang prinsip kerja ESC, penggunaan yang tepat, dan pengembangan terbaru dalam teknologi ESC menjadi krusial untuk memahami bagaimana perangkat ini berkontribusi pada perkembangan teknologi kendali motor yang lebih canggih dan efisien.

Kajian Teori

1. Electronic Speed Controller

Pengontrol kecepatan adalah suatu perangkat elektronik yang digunakan untuk mengontrol kecepatan motor listrik atau perangkat elektronik lainnya yang memerlukan pengontrol kecepatan. Alat ini sangat berguna dalam berbagai aplikasi seperti industri, kendaraan listrik, drone, mesin industri dan peralatan rumah tangga. Ada beberapa jenis pengontrol kecepatan dan kita menggunakan yang ESC karena kita membuat alat kesahatan ESC dengan putaran motor. dan ini ada 3 jenis pengontrol kecepatan (PWM, VFD,ESC):

Modulasi Lebar Pulsa (PWM): Ini adalah metode yang umum digunakan untuk mengontrol kecepatan motor atau perangkat lain dengan menyesuaikan lebar pulsa sinyal listrik yang diberikan ke perangkat. Kontrol kecepatan dapat dicapai dengan mengubah lebar pulsa.

Penggerak Frekuensi Variabel (VFD): VFD yang digunakan pada motor listrik tiga fase mengontrol kecepatan dengan mengatur frekuensi dan tegangan yang disuplai ke motor. Hal ini memungkinkan pengguna untuk mengontrol kecepatan motor secara tepat.

Pengontrol Kecepatan Elektronik Tanpa Sikat (ESC): Biasa digunakan pada kendaraan listrik, drone, dan model yang dikendalikan radio. ESC ini dirancang khusus untuk motor tanpa sikat dan dapat mengontrol putaran motor secara tepat menggunakan sinyal dari pengontrol eksternal seperti remote control.

Setiap pengontrol kecepatan memiliki cara kerja yang berbeda-beda, namun tujuannya sama, yaitu menyesuaikan kecepatan perangkat yang terhubung untuk memenuhi kebutuhan spesifik aplikasi. Keuntungan utama menggunakan pengontrol kecepatan adalah kemampuan untuk mengontrol kecepatan secara tepat, efisiensi energi yang lebih baik, dan kemampuan untuk menyesuaikan kinerja perangkat sesuai kebutuhan.

2. Sensor Arus

Sensor arus adalah perangkat yang digunakan untuk mendeteksi atau mengukur arus listrik yang mengalir dalam suatu rangkaian listrik. Sensor arus membantu dalam memantau dan mengukur arus secara tepat untuk berbagai tujuan, seperti keamanan, pengukuran, pengendalian, dan pemantauan performa sistem listrik.

Beberapa jenis sensor arus meliputi:

Sensor Arus Pemutus: Sensor ini digunakan untuk mendeteksi arus lebih (overcurrent) dalam rangkaian listrik. Ketika arus melebihi batas yang telah ditetapkan, sensor ini akan memutus aliran listrik untuk melindungi perangkat atau sistem dari kerusakan.

Transformator Arus (Current Transformer - CT): Ini adalah sensor yang bekerja dengan prinsip transformasi arus. CT digunakan untuk mengukur arus tinggi dengan mengubahnya menjadi arus yang lebih rendah sehingga dapat diukur oleh alat pengukuran standar. Biasanya digunakan dalam sistem tenaga listrik untuk mengukur arus yang melewati kabel atau konduktor besar.

Hall Effect Sensor: Sensor ini memanfaatkan efek Hall untuk mengukur arus. Ketika arus mengalir melalui konduktor, sensor Hall akan menghasilkan tegangan keluaran yang proporsional terhadap arus yang mengalir tersebut.

Shunt Resistor: Meskipun bukan sensor dalam arti tradisional, shunt resistor sering digunakan untuk mengukur arus. Resistor ini memiliki resistansi rendah dan ditempatkan dalam jalur arus sehingga tegangan yang dihasilkan dapat diukur untuk menghitung arus yang mengalir.

Sensor arus memainkan peran penting dalam berbagai aplikasi, mulai dari industri, sistem tenaga listrik, kendaraan listrik, hingga perangkat elektronik yang memerlukan pengukuran arus untuk pengendalian atau pemantauan yang akurat. Dengan menggunakan sensor arus yang tepat, kita dapat mengontrol dan memantau aliran arus dengan lebih efisien serta memastikan keamanan sistem listrik.

3. Sensor Tegangan

Sensor tegangan adalah perangkat yang digunakan untuk mengukur atau mendeteksi tingkat tegangan dalam suatu rangkaian listrik. Sensor ini dirancang untuk mengonversi tegangan listrik menjadi sinyal yang dapat diukur atau diproses oleh perangkat elektronik lainnya.

Sensor tegangan bekerja berdasarkan prinsip memantau atau mendeteksi level tegangan suatu rangkaian listrik dan berdasarkan pengukuran tersebut, memberikan keluaran yang sesuai. Pengoperasian sensor tegangan bervariasi berdasarkan jenisnya, namun beberapa prinsip dasar yang sering digunakan:

Pembagi tegangan: Ini adalah rangkaian sederhana yang menggunakan resistor untuk membagi tegangan di seluruh rangkaian. Tegangan diukur pada titik-titik tertentu pada pembagi tegangan untuk memberikan perkiraan tegangan masukan total.

Sensor tegangan AC (arus bolak-balik): Sensor ini sering kali menggunakan trafo atau perangkat lain yang mampu mengubah tegangan AC ke tingkat yang dapat diukur. Trafo ini mempunyai lilitan yang dapat menaikkan atau menurunkan tegangan masukan sesuai kebutuhan. Sensor

Tegangan DC (Arus Searah): Pada tegangan DC, sensor tegangan dapat menggunakan rangkaian pembagi tegangan atau sensor khusus yang dapat mengubah tegangan menjadi sinyal yang dapat diukur.

Sensor tegangan isolasi galvanik: Pada sensor ini, bagian sensitif tegangan diisolasi secara galvanis (tanpa sambungan langsung) dari sumber tegangan untuk mencegah bahaya atau kerusakan pada perangkat. Pengukuran dilakukan dengan isolasi sumber tegangan yang aman.

Pada dasarnya sensor tegangan bekerja dengan cara mendeteksi perubahan tegangan pada suatu rangkaian listrik dan menghasilkan sinyal keluaran yang sesuai, biasanya berupa tegangan atau arus yang dapat diukur. Hasilnya kemudian dapat dibaca oleh perangkat lain seperti mikrokontroler atau meter untuk dianalisis atau diatur lebih lanjut. Penting untuk diingat bahwa kinerja sensor tegangan bervariasi berdasarkan aplikasi dan jenisnya. Beberapa sensor memerlukan penyesuaian atau pengaturan khusus untuk memberikan hasil yang akurat untuk kebutuhan aplikasi spesifiknya.

4. LM35

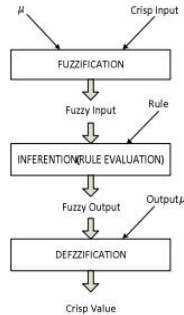
Sensor suhu LM35 merupakan salah satu jenis sensor suhu yang populer dan sering digunakan dalam berbagai aplikasi. Sensor ini dirancang khusus untuk mengukur suhu pada skala Celsius dengan akurasi tinggi. LM35 bekerja berdasarkan prinsip bahwa tegangan berubah sebanding dengan suhu. Ketika suhu meningkat, tegangan keluaran sensor juga meningkat secara linier. Setiap perubahan suhu 1 derajat Celcius menyebabkan perubahan tegangan keluaran sensor sebesar 10 milivolt (mV). Sensor suhu LM35 adalah sensor suhu presisi yang menghasilkan tegangan output sebanding dengan suhu dalam skala Celcius. Cara kerjanya didasarkan pada prinsip bahwa tegangan output dari sensor ini berubah seiring perubahan suhu.

5. Fuzzy Logic

Logika Fuzzy adalah metodologi sistem kontrol pemecahan masalah yang cocok diterapkan pada sistem, mulai dari sistem yang sederhana sampai sistem yang rumit atau kompleks. Logika Fuzzy dapat diterapkan dalam berbagai bidang, diantaranya yaitu pada sistem diagnosis penyakit (dalam bidang kedokteran); pemodelan sistem pemasaran, riset operasi (dalam bidang ekonomi); kendali kualitas air, prediksi adanya gempa bumi dan lain- lain. Logika Fuzzy adalah suatu cara yang tepat untuk memetakan suatu ruang input ke dalam suatu ruang output. Selain itu Logika Fuzzy juga dapat diartikan suatu cara yang tepat untuk memetakan suatu ruang input ke dalam suatu ruang output. Gambaran mengenai pemetaan ruang input ke output dapat dilihat pada gambar berikut :



Gambar 1. Pemetaan Input dan Output Logika fuzzy



Gambar 2. Cara Kerja Logika fuzzy

Di bawah ini adalah uraian cara kerja logika fuzzy, langkah-langkahnya yaitu sebagai berikut:

- Fuzzyfikasi

Fase pertama dari perhitungan fuzzyfikasi, yaitu mengubah masukan-masukan yang nilai kebenarannya bersifat pasti ke dalam bentuk fuzzy input yang berupa tingkat keanggotaan/tingkat kebenaran. Dengan demikian, tahap ini mengambil nilai-nilai crisp dan menentukan derajat di mana nilai-nilai tersebut menjadi anggota dari setiap himpunan fuzzy yang sesuai.

- Inferensi

Melakukan penalaran menggunakan fuzzy input dan fuzzy rules yang telah ditentukan sehingga menghasilkan fuzzy output. Secara sintaks, suatu fuzzy rule (aturan fuzzy) dituliskan sebagai berikut: IF antecedent THEN consequent

- Defuzzifikasi

Mengubah fuzzy output menjadi nilai tegas berdasarkan fungsi keanggotaan yang telah ditentukan. Defuzzifikasi merupakan metode yang penting dalam pemodelan sistem fuzzy.

- Metode Tsukamoto

Metode Tsukamoto adalah perluasan dari penalaran monoton. Pada metode Tsukamoto, setiap konsekuen pada aturan yang berbentuk IF-THEN harus direpresentasikan dengan suatu himpunan fuzzy dengan fungsi keanggotaan yang monoton. Sebagai hasilnya,

output hasil inferensi dari tiap-tiap aturan diberikan secara tegas (crisp) berdasarkan α -predikat (fire strength). Hasil akhirnya diperoleh dengan menggunakan rata-rata terbobot.

6. KNN

K-Nearest Neighbors (KNN) adalah algoritma klasifikasi non-parametrik yang didasarkan pada konsep jarak. KNN bekerja dengan cara mencari k tetangga terdekat dari data baru yang akan diklasifikasikan. Data baru kemudian akan diklasifikasikan ke dalam kelas yang sama dengan mayoritas tetangga terdekatnya. KNN didasarkan pada konsep jarak. Jarak antara dua data dapat dihitung dengan berbagai cara, seperti jarak Euclidean, jarak Manhattan, dan jarak Minkowski. Jarak Euclidean adalah jarak yang paling umum digunakan dalam KNN. Jarak Euclidean dihitung dengan menggunakan rumus berikut:

$$d(x, y) = \sqrt{(x_1 - y_1)^2 + (x_2 - y_2)^2 + \dots + (x_n - y_n)^2}$$

Fleksibel dalam pemilihan fitur KNN tidak memerlukan pemilihan fitur yang kompleks. KNN dapat digunakan dengan berbagai fitur, baik fitur numerik maupun kategorikal. Hal ini karena KNN menggunakan jarak Euclidean untuk mengukur kedekatan antara dua data.

C. Metode

1. Spesifikasi hardware yang dikembangkan (foto[atas, bawah, kiri, kanan], ukuran dan spesifikasi hardware. Sensor yang digunakan dan software yang digunakan.

➤ Spesifikasi Sistem Electronic Speed Controller:

Keterangan	
Max Temp	100°C
Max Voltage	16 V
Max Current	50 A
Input	5 V
Dimension	3,5 x 7,5 x 2,5
Weight	65 g
Smart decision	Fuzzy logic (A

Berikut adalah tabel spesifikasi dari alat yang dibuat dengan maksimal temperature mencapai 100 derajat celcius, maksimal voltage 16 volt, masimal current 50 A, input 5 V, dimension 3,5 x 7,5 x 2,5, weigghy 65 g, smart decision fuzzy logic dan KNN.

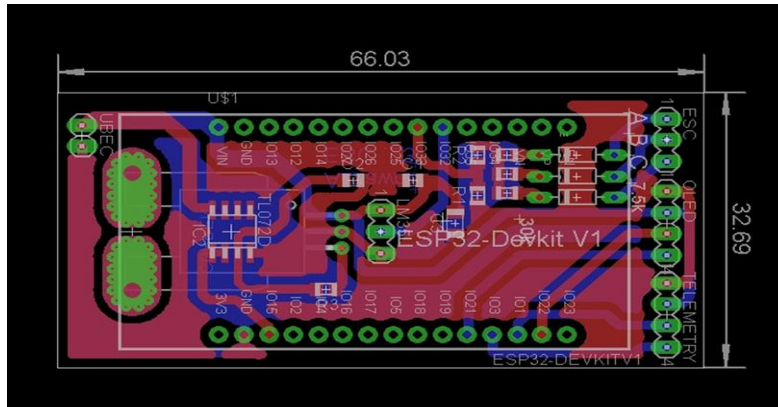
➤ Software yang digunakan

- Arduino Ide

Arduini Ide adalah software yang digunakan untuk pengembangan alat yang dibuat untuk memprogram sensor dan hasil fuzzy logic yang di dapat untuk menentukan kesehatan electronic speed controller dengan Bahasa yang digunakain yaitu Bahasa C.

- EAGLE

Eagle adalah software untuk merangkai atau mendesain PCB layout, desain pcb yang dilakukan semaksimal mungkin untuk pengembangan sistem ini sangat kompleks dan berukuran kecil agar ukuran nya lebih fleksibel.



Berikut hasil desind PCB yang telah dibuat menggunakan software EAGLE

- Anaconda

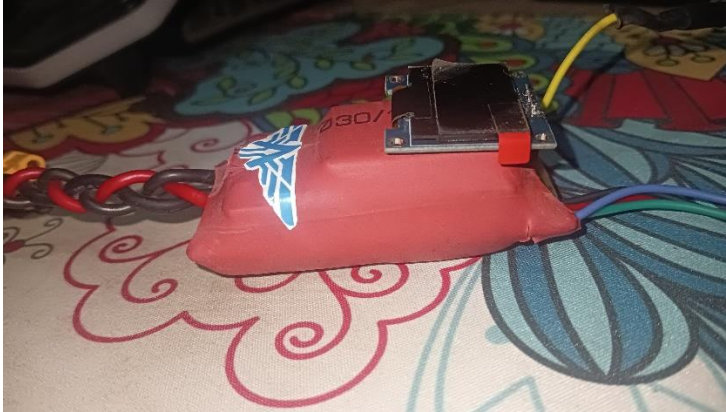
Anaconda adalah salah satu system software yang digunkan untuk memogram program software untuk membuat KNN untuk memprediksi keakurasian dari hasil KNN yang telah dibuat.

- Sensor Dalam System

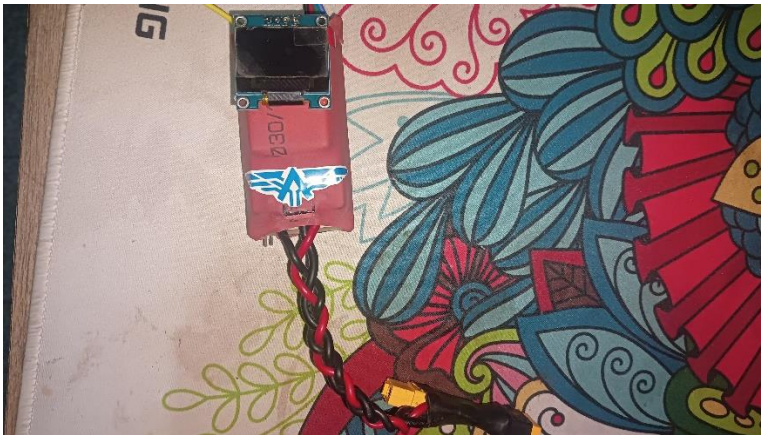
Sensor yang digunakan yaitu ada 3 sensor yang dmana terdiri dari sensor suhu, sensor arus, sensor kelembaban. Dengan pembuatan sensor arus dan sensor tegangan membuat sendiri tujuannya untuk bisa memperoleh maksimal dari sensor arus dan tegangan yaitu mencapai maksimal arus 50 A sedangkan sensor tegangan 16 volt seingga mudah diterapkan ke electronic speed controller yang Arus dan tegangan dipakai sangat itnggi.

- Gambar system

Posisi samping:



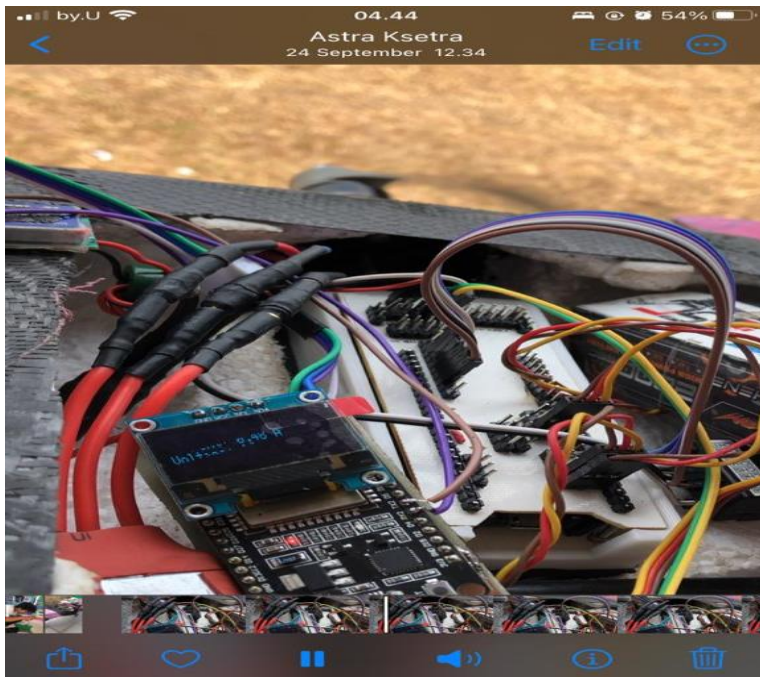
Posisi depan:



Posisi belakang:



Diterapkan pada pesawat:



Pesawat yang digunakan:



2. Fuzzy, jelaskan secara detail proses fuzzyfikasi, membership function, rules, dan defuzifikasi dari proyek yang dikembangkan termasuk tuliskan persamaannya.

- **Fuzzyfikasi:**

```
void fuzzifikasi () {  
    arusRendah();  
    arusSedang();  
    arusTinggi();  
    teganganRendah();  
    teganganSedang();  
    teganganTinggi();  
    suhuDingin();  
    suhuNormal();  
    suhuPanas();  
}
```

Fuzzyfikasi dari electronic speed controller yang dibuat yaitu terdiri dari 3 input dan 1 output. Dari ketiga input tersebut terdiri dari arus, tegangan, dan suhu pada ESC. Untuk outputnya adalah Kesehatan esc itu sendiri.

- **Membership Function**

- Arus

```
unsigned char arusRendah(){  
    if (current <= 0) {ARendah=1;}  
    else if (current >= 0 && current <= 20) {ARendah=(45-current)/20;}  
    else if (current >=20){ARendah = 0;}  
    return ARendah;  
}  
unsigned char arusSedang(){  
    if (current < 15) {ASedang=0;}  
    else if (current > 15 && current <= 22.5) {ASedang=(current-15)/7.5;}  
    else if (current >= 22.5 && current < 30) {ASedang=(30-arus)/7.5;}  
    else if (current >= 30){ASedang = 0;}  
    return ASedang;  
}  
unsigned char arusTinggi(){  
    if (current <= 25) {ATinggi=0;}  
    else if (current >= 25 && current <= 40) {ATinggi=(current-25)/15;}  
    else if (current >=40){ATinggi = 1;}  
    return ATinggi;  
}
```

Untuk membership function arus terdiri dari himpunan rendah, sedang, tinggi. Rendah kurang dari sama dengan 20 A, sedang antara 15 A dan 30 A, dan tinggi lebih dari sama dengan 25 A hingga batas atas 40 A

- Tegangan.

```
unsigned char teganganRendah(){
    if (hasil <= 0) {TRendah=1;}
    else if (hasil >= 0 && hasil <= 4) {TRendah=(4-hasil)/4;}
    else if (hasil >=4){TRendah = 0;}
    return TRendah;
}
unsigned char teganganSedang(){
    if (hasil < 2.5) {TSedang=0;}
    else if (hasil > 2.5 && hasil <= 5.25) {TSedang=(hasil-2.5)/2.75;}
    else if (hasil >= 5.25 && hasil < 8) {TSedang=(8-hasil)/2.75;}
    else if (hasil >= 8){TSedang = 0;}
    return TSedang;
}
unsigned char teganganTinggi(){
    if (hasil <= 6) {TTinggi=0;}
    else if (hasil >= 6 && hasil <= 12) {TTinggi=(hasil-6)/6;}
    else if (hasil >=12){TTinggi = 1;}
    return TTinggi;
}
```

Untuk membership fuction tegangan terdiri dari himpunan rendah, sedang, tinggi. Rendah kurang dari sama dengan 4 V, sedang antara 2.5 V dan 8 V, dan tinggi lebih dari sama dengan 6 V hingga batas atas 12 V

- Suhu

```
unsigned char suhuDingin(){
    if (temp <= 25){Dingin =1;}
    else if (temp >=25 && temp <=32.5){Dingin=(32.5-temp)/7.5;}
    else if (temp >= 32.5){Dingin =0;}
    return Dingin;
}
unsigned char suhuNormal(){
    if (temp <= 30){Normal =0;}
    else if (temp >=30 && temp <=35){Normal=(temp-30)/5;}
    else if (temp >=35 && temp <=40){Normal=(40-temp)/5;}
    else if (temp >= 40){Normal =0;}
    return Normal;
}
unsigned char suhuPanas (){
    if (temp <= 37.5 ){Panas =0;}
    else if (temp >=37.5 && temp <=50){Panas=(temp-37.5)/12.5;}
    else if (temp >= 50){Panas =1;}
    return Panas;
}
```

Untuk membership fuction suhu terdiri dari himpunan dingin, normal, panas. dingin kurang dari sama dengan 32,5 °C, normal antara 30 °C dan 40 °C, dan panas lebih dari sama dengan 37,5 °C hingga batas atas 50 °C

- Output

```
// Menentukan kesehatan ESC berdasarkan nilai out
// Serial.print(", ");
Serial.print("Kesehatan ESC: ");
if (out <= 55) {
    Serial.print("Buruk");
} else if (out > 55 && out < 80) {
    Serial.print("Normal");
} else if (out >= 80) {
    Serial.println("Baik");
}
```

Untuk membership Kesehatan ESC terdiri dari himpunan buruk, normal, baik. Buruk bernilai kurang dari sama dengan 55 %, normal antara 50 % dan 85 %, dan baik lebih dari sama dengan 80 % hingga batas atas 100 %

- Rules/Aturan

```
void rules() {
    Atotal = 0;
```

```

Ztotal = 0;
int index = 0;
in_a[0] = ARendah; in_a[1] = ASedang; in_a[2] = ATinggi;
in_b[0] = TRendah; in_b[1] = TSedang; in_b[2] = TTinggi;
in_c[0] = Dingin; in_c[1] = Normal; in_c[2] = Panas;
for (int i = 0; i < 3; i++) {
    for (int j = 0; j < 3; j++) {
        for (int x = 0; x < 3; x++) {
            A[index] = min(min(in_a[i], in_b[j]), in_c[x]);

            if (index < 3) {
                Z[index] = (80 + A[index] * (100 - 80)) * A[index]; // Baik
            } else if (index < 10) {
                Z[index] = ((50 + A[index] * (67.5 - 50)) * A[index]) + ((85 - A[index] * (85 - 67.5)) *
A[index]); // Normal
            } else if (index < 13) {
                Z[index] = (50 - (A[index] * (50 - 0))) * A[index]; // Buruk
            } else if (index < 20) {
                Z[index] = ((50 + A[index] * (67.5 - 50)) * A[index]) + ((85 - A[index] * (85 - 67.5)) *
A[index]); // Normal
            } else if (index < 26) {
                Z[index] = (50 - (A[index] * (50 - 0))) * A[index]; // Buruk
            } else if (index == 26) {
                Z[index] = ((50 + A[index] * (67.5 - 50)) * A[index]) + ((85 - A[index] * (85 - 67.5)) *
A[index]); // Normal
            } else if (index == 27) {
                Z[index] = (50 - (A[index] * (50 - 0))) * A[index]; // Buruk
            } if (index > 3 && index < 10 || index > 12 && index < 20 || index == 26) {
                A[index] = A[index]*2;

            }

            Serial.print("A ");
            Serial.print(index);

```

```

Serial.print(" = ");
Serial.println(A[index]);

Atotal += A[index];
Ztotal += Z[index];
index++;

```

Rules yang dihasilkan dari masing masing membership function yaitu 27 rules.

-Defuzzyfikasi

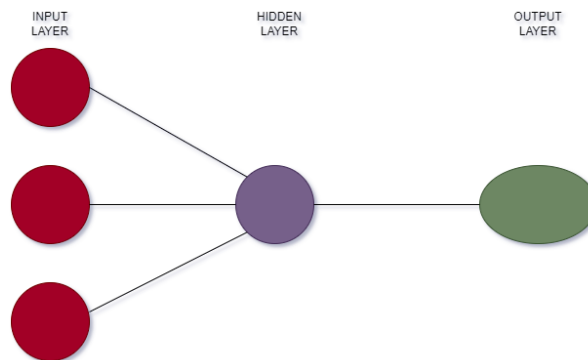
Rumus persamaan defuzzyfikasi sebagai berikut :

=IF(I11="Buruk";(50-(J11*(50-0)))*J11;IF(I11="Normal";((50+J11*(67,5-50))*J11)+((85-J11*(85-67,5))*J11);IF(I11="Baik";((80+J11*(100-80))*J11);0)))

Dari rules diatas dihasilkannya rumus persamaan defuzzyfikasi yaitu jika output = buruk maka akan menghitung $(50-(\text{Alpha}*(50-0)))*\text{Alpha}$, output = normal $(50-(\text{Alpha}*(50-0)))*\text{Alpha}$, dan output = baik $((80+\text{Alpha}*(100-80))*\text{Alpha};0)))$

3. NN, jelaskan secara detail metode NN yang digunakan

- a. proses jaringan input, hidden, dan output layer. Berapa jumlah neuron tiap layer input, hidden, dan output.



Untuk input layer terdiri dari 3 jenis lalu Ketika masuk ke hidden layer, di hidden layer terdapat 1 jenis dan untuk outputnya juga 1 jenis

- b. Jelaskan proses training dan update bobot dan bias
 - Proses Training
 - Pertama pemuatan data

```
df_train = pd.read_csv("KNN_AI.csv")
df_validate = pd.read_csv("KNN_VAL.csv")
```

Dua baris kode ini memuat data sheet pelatihan dan validasi ke dalam data frame menggunakan library pandas.

- Pemisahan data

```
X = df_train.drop('Kategori Kesehatan', axis=1)
Y = df_train['Kategori Kesehatan']
x_train,x_test,y_train,y_test = train_test_split(X, Y, test_size=0.
↪20,random_state=42)
```

Variab X dan Y menampung fitur dan variable target selanjutnya, dataset dipisahkan menjadi set pelatihan dan pengujian menggunakan 'train_test_split'. Set pelatihan mengandung 80 % data, sementara set pengujian berisi 20 %

- Inisialisasi dan pelatihan model KNN

```
knn_clf = KNeighborsClassifier(n_neighbors=5, metric='euclidean', p=2)
knn_clf.fit(x_train, y_train)
```

KNN classifier diinisialisasikan dengan parameter:

1. n_neighbors: jumlah tetangga yang dipertimbangkan (diatur kelima)
2. Metric jarak yang digunakan yaitu Euclidean
3. p: parameter daya untuk jarak minkowski diatur ke 2 untuk jarak Euclidean

Model kemudian dilatih pada set pelatihan menggunakan 'fit'

- Update bobot dan bias

KNN adalah sebuah algoritma non-parametrik dan tidak melibatkan bobot atau bias.

KNN membuat prediksi berdasarkan kelas mayoritas dari K tetangga terdekatnya. Tidak ada fase pelatihan eksplisit yang melibatkan penyesuaian bobot atau bias.

c. Jelaskan proses running projek nya

	precision	recall	f1-score	support
Baik	0.93	1.00	0.97	56
Buruk	0.64	0.66	0.65	65
Normal	0.91	0.86	0.88	113
Rusak	0.69	0.75	0.72	36
Sehat	1.00	0.88	0.93	24
accuracy			0.83	294
macro avg	0.83	0.83	0.83	294
weighted avg	0.83	0.83	0.83	294

Dari kedua dataset yaitu data pelatihan dan data uji yang sudah dimasukkan kedalam program akan menghasilkan prediksi dari KNN sesuai dengan gambar diatas adalah prediksinya mencapai 0,83 ini adalah angka yang cukup bagus dari hasil kedua data yang telah diambil sendiri. Mengapa hasil hanya menunjukkan angka 0,83? Karena dari kedua data tersebut data pelatihan tidak sebanyak data uji maka dari itu hasil prediksi KNN hanya menunjukkan 0,83%

D. Hasil dan Pembahasan

a. Jelaskan hasil Fuzzy, tampilkan table input, output, dan persamaan defuzzifikasinya

- **Fuzzyfikasi:**

fuzzyfikasi	membership		
Arus pd ESC	Tegangan ESC	Suhu ESC	Kesehatan ESC
Rendah	Rendah	Dingin	Baik
Sedang	Sedang	Normal	Normal
Tinggi	Tinggi	Panas	Buruk

Fuzzyfikasi dari electronic speed controller yang dibuat yaitu terdiri dari 3 input dan 1 output.

Dari ketiga input tersebut terdiri dari arus, tegangan, dan suhu pada ESC. Untuk outputnya adalah Kesehatan esc itu sendiri.

- **Membership Function**

- Arus

Variable Arus (A)		membership
Himpunan		
Rendah	≤ 20	1,0
Sedang	$15 < x < 30$	0,0
Tinggi	≥ 25	0,0

Untuk membership fuction arus terdiri dari himpunan rendah, sedang, tinggi. Rendah kurang dari sama dengan 20 A, sedang antara 15 A dan 30 A, dan tinggi lebih dari sama dengan 25 A hingga batas atas 40 A

- Tegangan

Variable Tegangan (V)		
Himpunan	Batas Awal	Batas Akhir
Rendah	≤ 4	0,0
Sedang	$2.5 < x < 8$	0,0
Tinggi	≥ 6	0,7

Untuk membership fuction tegangan terdiri dari himpunan rendah, sedang, tinggi.

Rendah kurang dari sama dengan 4 V, sedang antara 2.5 V dan 8 V, dan tinggi lebih dari sama dengan 6 V hingga batas atas 12 V

- Suhu

Variable suhu (°C)		
Himpunan	Batas Awal	Batas Akhir
Dingin	$\leq 32,5$	0,0
Normal	$30 < x < 40$	0,8
Panas	$\geq 37,5$	0,0

Untuk membership fuction suhu terdiri dari himpunan dingin, normal, panas. dingin

kurang dari sama dengan 32,5 °C, normal antara 30 °C dan 40 °C, dan panas lebih dari sama dengan 37,5 °C hingga batas atas 50 °C

- Output

Kesehatan ESC (%)	
Himpunan	
Buruk	≤ 55
Normal	$50 < x < 85$
Baik	≥ 80

Untuk membership Kesehatan ESC terdiri dari himpunan buruk, normal, baik. Buruk

bernilai kurang dari sama dengan 55 %, normal antara 50 % dan 85 %, dan baik lebih dari sama dengan 80 % hingga batas atas 100 %

- Rules/Aturan

Aturan										
Rule	-	Arus		Tegangan		Suhu	-	Output	Alpha	Z
Rule1	if	Rendah	AND	Rendah	AND	Dingin	THEN	Baik	0	0,0
Rule2	if	Rendah	AND	Rendah	AND	Normal	THEN	Baik	0	0,0
Rule3	if	Rendah	AND	Rendah	AND	Panas	THEN	Baik	0	0,0
Rule4	if	Rendah	AND	Sedang	AND	Dingin	THEN	Normal	0	0,0
Rule5	if	Rendah	AND	Sedang	AND	Normal	THEN	Normal	0	0,0
Rule6	if	Rendah	AND	Sedang	AND	Panas	THEN	Normal	0	0,0
Rule7	if	Rendah	AND	tinggi	AND	Dingin	THEN	Normal	0	0,0
Rule8	if	Rendah	AND	tinggi	AND	Normal	THEN	Normal	1	90,0
Rule9	if	Rendah	AND	tinggi	AND	Panas	THEN	Normal	0	0,0
Rule10	if	Sedang	AND	Rendah	AND	Dingin	THEN	Buruk	0	0,0
Rule11	if	Sedang	AND	Rendah	AND	Normal	THEN	Buruk	0	0,0
Rule12	if	Sedang	AND	Rendah	AND	Panas	THEN	Buruk	0	0,0
Rule13	if	Sedang	AND	Sedang	AND	Dingin	THEN	Normal	0	0,0
Rule14	if	Sedang	AND	Sedang	AND	Normal	THEN	Normal	0	0,0
Rule15	if	Sedang	AND	Sedang	AND	Panas	THEN	Normal	0	0,0
Rule16	if	Sedang	AND	tinggi	AND	Dingin	THEN	Normal	0	0,0
Rule17	if	Sedang	AND	tinggi	AND	Normal	THEN	Normal	0	0,0
Rule18	if	Sedang	AND	tinggi	AND	Panas	THEN	Normal	0	0,0
Rule19	if	tinggi	AND	Rendah	AND	Dingin	THEN	Normal	0	0,0
Rule20	if	tinggi	AND	Rendah	AND	Normal	THEN	Buruk	0	0,0
Rule21	if	tinggi	AND	Rendah	AND	Panas	THEN	Buruk	0	0,0
Rule22	if	tinggi	AND	Sedang	AND	Dingin	THEN	Buruk	0	0,0
Rule23	if	tinggi	AND	Sedang	AND	Normal	THEN	Buruk	0	0,0
Rule24	if	tinggi	AND	Sedang	AND	Panas	THEN	Buruk	0	0,0
Rule25	if	tinggi	AND	tinggi	AND	Dingin	THEN	Buruk	0	0,0
Rule26	if	tinggi	AND	tinggi	AND	Normal	THEN	Normal	0	0,0
Rule27	if	tinggi	AND	tinggi	AND	Panas	THEN	Buruk	0	0,0
									1	90,0
									68	

Rules yang dihasilkan dari masing masing membership function yaitu 27 rules.

- Defuzzyfikasi

Rumus persamaan defuzzyfikasi sebagai berikut :

$$= \text{IF}(I11 = \text{"Buruk"}; (50 - (J11 * (50 - 0))) * J11; \text{IF}(I11 = \text{"Normal"}; ((50 + J11 * (67,5 - 50)) * J11) + ((85 - J11 * (85 - 67,5)) * J11); \text{IF}(I11 = \text{"Baik"}; ((80 + J11 * (100 - 80)) * J11); 0)))$$

Dari rules diatas dihasilkannya rumus persamaan defuzzyfikasi yaitu jika output = buruk maka akan menghitung $(50 - (\text{Alpha} * (50 - 0))) * \text{Alpha}$, output = normal $(50 - (\text{Alpha} * (50 - 0))) * \text{Alpha}$, dan output = baik $((80 + \text{Alpha} * (100 - 80)) * \text{Alpha}; 0)))$

- Jelaskan hasil NN yang telah dirunning, mulai dari update bobot, bias, dan hasil dari NN.

Tampilkan table input, output.

KNN adalah sebuah algoritma non-parametrik dan tidak melibatkan bobot atau bias. KNN membuat prediksi berdasarkan kelas mayoritas dari K tetangga terdekatnya. Tidak ada fase pelatihan eksplisit yang melibatkan penyesuaian bobot atau bias. Dikarenakan KNN itu sendiri tidak melibatkan bobot dan juga bias, menganntinya menggunakan data yang telah diambil sendiri yaitu menggunakan data pelatihan dan juga data uji. Dari kedua data tesebut kami mengambil data sendiri saat pesawat diterbangkan di udara lalu data kami ambil melalui program dan data tersebut yang dinamakan data logger dimasukkan ke dalam exel berbentuk csv. Dari

kedua data tersebut data pelatihan dan data uji mendapatkan hasil prediksi KNN sebesar 0,83% sebuah angka yang cukup bagus hasil dari KNN.

- c. Bandingkan performan output yang dihasilkan antara hasil pada sistem Fuzzy dan sistem NN.

Hasil dari prediksi KNN

	precision	recall	f1-score	support
Baik	0.93	1.00	0.97	56
Buruk	0.64	0.66	0.65	65
Normal	0.91	0.86	0.88	113
Rusak	0.69	0.75	0.72	36
Sehat	1.00	0.88	0.93	24
accuracy			0.83	294
macro avg	0.83	0.83	0.83	294
weighted avg	0.83	0.83	0.83	294

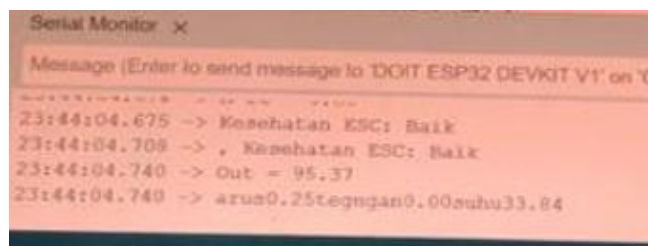
```
X_new= pd.DataFrame([[1,10,34]], columns=['Arus', 'Tegangan', 'Suhu'])
y_pred=knn_clf.predict(X_new)
print("Predicted Class:",y_pred)
```

Predicted Class: ['Sehat']

```
X_new = pd.DataFrame([[9.74, 15.3, 68.99]], columns=['Arus', 'Tegangan', 'Suhu'])
y_pred=knn_clf.predict(X_new)
print("Predicted Class:", y_pred)
```

Predicted Class: ['Normal']

Hasil dari Fuzzy logic



1. Ketidakpastian:

- Fuzzy Logic menyediakan bagian kerja dari mengatasi ketidakpastian dalam sistem dengan mengenalkan konsep seperti keanggotaan sebagian (fuzzy membership).

2. Variabel Fuzzy:

- Dalam Fuzzy Logic, variabel ini memiliki nilai keanggotaan sebagian di berbagai macam himpunan fuzzy. Ini kemungkinan representasi variabel yang tidak hanya true or no, tetapi juga mendapatkan tingkat kebenarannya di dalam suatu interval.

3. Aturan Fuzzy:

- Fuzzy Logic menggunakan aturan yang sifatnya linguistik yang biasa disebut dengan istilah seperti “jika x adalah A, maka y adalah B,” di mana A dan B adalah himpunan fuzzy.

4. Aplikasi:

- Fuzzy Logic sangat banyak digunakan dalam sistem kontrol, kecerdasan buatan, dan pemrosesan sinyal, terjadi ketika ada ketidakpastian dan variasi.

K-Nearest Neighbors (KNN):

1. Representasi Berbasis Instansi:

- KNN adalah algoritma berbasis instansi yang menyimpan seluruh data pelatihan dan melakukan prediksi berdasarkan ketergantungan dari tetangga terdekat.

2. Pengukuran Jarak:

- KNN menggunakan metrik jarak (seperti Euclidean) untuk melihat dan mengukur kemiripan data uji dengan data pelatihan dan memilih K tetangga terdekat.

3. Klasifikasi dan Regresi:

- KNN biasa digunakan baik untuk klasifikasi (dengan memilih kelas mayoritas dari tetangga terdekat) maupun untuk regresi (dengan memilih nilai rata-rata dari tetangga terdekat).

4. Aplikasi:

- KNN biasa digunakan dalam klasifikasi pola, pengenalan pola, dan prediksi menurut struktur data yang ada.

Perbedaan Utama:

- Konsep:

- Fuzzy Logic mengenalkan konsep keanggotaan sebagian dan mengatasi ketidakpastian, sementara itu KNN berfokus pada ketergantungan pada tetangga terdekat dan pengukuran jarak.
- Aplikasi:
- Fuzzy Logic biasanya digunakan dalam domain di mana ketidakpastian dan keabstrakan diperlukan, sementara KNN sering digunakan dalam tugas klasifikasi learning dan regresi berbasis data.

Memilih antara Fuzzy Logic dan KNN tergantung pada proyek dan kebutuhan spesifik dari masalah yang akan dihadapi serta karakteristik data yang digunakan

E. Kesimpulan

Dari Proyek yang kami buat kami menemukan bahwa Fuzzy Logic dan KNN sangat berguna di Proyek kami. Hasil dari Fuzzy digunakan untuk melihat keadaan ESC saat di gunakan sedangkan KNN digunakan untuk melihat berapakah akurasi dari Fuzzy yang kita buat.

F. Daftar Pustaka