



Rancang Bangun Alat Penganalisa Kerusakan Berdasarkan Getaran dan Kecepatan Putar Motor Menggunakan IoT

Adam Setiawan Tiyas^{1*}, Farrady Alif Fiolana², Diah Arie Widhining Kusumastutie³

¹⁻³ Program Studi Teknik Elektro, Universitas Islam Kediri, Indonesia

Email: adamjeck8@gmail.com¹, farradyalif@uniska-kediri.ac.id², diahariewk@uniska-kediri.ac.id³

*Penulis Korespondensi: adamjeck8@gmail.com

Abstract. *Electric motors are critical components across various industrial sectors; therefore, condition monitoring systems are essential to maintain performance and mitigate the risk of downtime. This study aims to design an Internet of Things (IoT)-based condition monitoring system for electric motors by integrating vibration and rotational speed sensors and employing the Random Forest machine learning algorithm for motor condition classification. The system utilizes an ESP32 microcontroller, an ADXL345 vibration sensor, and a TCRT5000 speed sensor. The research dataset comprises 1,200 data points across three motor condition categories: normal, loose bushing, and bent shaft. The Random Forest model was constructed using 15 decision trees with a maximum depth of 5. Testing on 240 data points demonstrated classification performance with accuracy, precision, recall, and F1-score values of 100% across all conditions. Furthermore, the system is capable of transmitting measurement data to a web-based IoT dashboard in real-time, with an average update interval of one second..*

Keywords: *Electric Motor; Internet of Things; Machine Learning; Random Forest; Vibration.*

Abstrak. Motor listrik merupakan komponen penting dalam berbagai sektor industri sehingga diperlukan sistem pemantauan kondisi untuk menjaga kinerja dan mengurangi risiko downtime. Penelitian ini bertujuan merancang sistem monitoring kondisi motor listrik berbasis Internet of Things (IoT) dengan mengintegrasikan sensor getaran dan sensor kecepatan putaran serta menerapkan algoritma machine learning Random Forest untuk klasifikasi kondisi motor. Sistem dirancang menggunakan mikrokontroler ESP32, sensor getaran ADXL345, dan sensor kecepatan TCRT5000. Dataset penelitian terdiri dari 1.200 data dengan tiga kategori kondisi motor, yaitu normal, bushing longgar, dan as bengkok. Model Random Forest dibangun menggunakan 15 decision tree dengan kedalaman maksimum 5. Hasil pengujian terhadap 240 data menunjukkan performa klasifikasi dengan nilai akurasi, precision, recall, dan F1-score sebesar 100% pada seluruh kondisi. Selain itu, sistem mampu mengirimkan data pengukuran ke dashboard IoT berbasis web secara real-time dengan waktu pembaruan rata-rata satu detik.

Kata kunci: *Getaran; Internet of Things; Motor Listrik; Pembelajaran Mesin; Random Forest.*

1. PENDAHULUAN

Motor listrik merupakan salah satu komponen penting yang banyak digunakan dalam berbagai sektor, seperti industri, pertanian, dan peralatan rumah tangga. Kinerja motor perlu dipantau secara berkala untuk memastikan kondisi tetap optimal dan mencegah terjadinya kerusakan. Parameter getaran dan kecepatan putaran dapat digunakan sebagai indikator kondisi motor. (Anuar *et. al*, 2024).

Pada industri skala kecil dan UMKM, pemantauan kondisi motor masih sering dilakukan secara manual melalui pemeriksaan langsung oleh operator. Cara tersebut memiliki keterbatasan karena membutuhkan waktu dan belum mampu memberikan informasi kondisi motor secara kontinu serta *real time*. Kondisi tersebut dapat meningkatkan risiko kerusakan komponen, memperpanjang waktu penghentian mesin, serta menimbulkan kerugian akibat terganggunya kegiatan produksi. Oleh karena itu, diperlukan sistem monitoring yang mampu

memberikan informasi kondisi motor secara cepat, akurat, dan berkelanjutan (Usman & Anhar, 2024).

Penerapan *Internet of Things* (IoT) dapat menjadi salah satu solusi untuk mengatasi keterbatasan tersebut. Teknologi IoT memungkinkan sensor memperoleh data kondisi motor, kemudian data diproses oleh mikrokontroler dan dikirimkan melalui jaringan sehingga kondisi motor dapat dipantau secara *real time*. Meskipun sistem monitoring motor berbasis IoT telah banyak dikembangkan, sebagian penelitian masih menggunakan satu parameter dalam proses identifikasi kondisi motor (Ibrahim *et al*, 2024). Oleh sebab itu, diperlukan integrasi kedua parameter tersebut dengan metode pengolahan data yang mampu menghasilkan klasifikasi kondisi motor secara lebih akurat.

Berdasarkan permasalahan tersebut, penelitian ini mengembangkan sistem monitoring motor listrik berbasis IoT dengan menggunakan sensor ADXL345 untuk mengukur getaran dan TCRT5000 untuk mendeteksi kecepatan putaran. Data dari kedua sensor selanjutnya diolah menggunakan metode *machine learning* dengan algoritma *Random Forest* untuk mengklasifikasikan kondisi kipas angin menjadi tiga kategori, yaitu normal, *bushing*, dan as bengkok. Sistem dirancang untuk melakukan pemantauan dan pengolahan data secara *real time* dengan biaya yang relatif terjangkau serta antarmuka yang sederhana. Pengembangan sistem ini diharapkan dapat membantu mendeteksi gangguan motor secara lebih cepat dan akurat serta menjadi alternatif monitoring yang mudah diterapkan pada lingkungan industri skala kecil dan UMKM.

2. KAJIAN TEORI

Motor listrik. Motor listrik mengubah energi listrik menjadi energi mekanik. Gangguan mekanis dapat memengaruhi karakteristik getaran dan kecepatan putaran sehingga kedua parameter dapat digunakan sebagai indikator kondisi motor.

Internet of Things. IoT memungkinkan perangkat fisik terhubung melalui jaringan untuk melakukan pengumpulan dan pertukaran data. Pada penelitian ini ESP32 berfungsi sebagai unit pemrosesan dan komunikasi yang mengirim data sensor menuju web server.

Sensor ADXL345. ADXL345 digunakan untuk memperoleh percepatan getaran pada sumbu X, Y, dan Z. Data tersebut diolah menggunakan metode *Root Mean Square* (RMS) untuk memperoleh nilai efektif getaran. Sensor TCRT5000. Sensor inframerah reflektif digunakan untuk membaca putaran baling-baling kipas. Nilai pulsa dikonversi menjadi RPM menggunakan hubungan antara jumlah pulsa dan waktu pengukuran.

Random Forest merupakan algoritma ensemble yang menggabungkan sejumlah *decision tree* untuk menghasilkan keputusan klasifikasi. Prinsip dasarnya adalah menggabungkan model pembelajaran lemah untuk menghasilkan model yang lebih kuat (Hassan *et al*, 2024). Model pada penelitian ini menggunakan 15 *tree* dengan kedalaman 5.

3. METODE PENELITIAN

Penelitian dilakukan melalui studi pustaka, eksperimen, metode uji presentase *error*, dan *machine learning*. Tahapan penelitian meliputi perancangan *hardware*, perancangan *software*, pengambilan data sensor, pengolahan sinyal, pembentukan dataset, pelatihan Random Forest, pengujian sistem, serta pengiriman data ke *dashboard web*.

Perancangan *hardware*. Komponen utama terdiri dari ESP32, ADXL345, TCRT5000, OLED, *buzzer*, *power supply*, dan kipas angin sebagai objek pengujian. ESP32 mengolah data sensor sekaligus menyediakan koneksi Wi-Fi untuk komunikasi dengan web server. Pengolahan data. Data ADXL345 diperoleh dari tiga sumbu dan diolah menggunakan RMS.

Pengolahan sinyal juga memanfaatkan *Peak to Peak* dan *Fast Fourier Transform* (FFT) untuk melihat karakteristik frekuensi. Sensor TCRT5000 menghasilkan pulsa yang dikonversi menjadi nilai RPM. Dataset dan klasifikasi. Dataset terdiri atas 1.200 data dari tiga kondisi, yaitu normal, bushing longgar, dan as bengkok. Evaluasi model menggunakan 240 data uji.

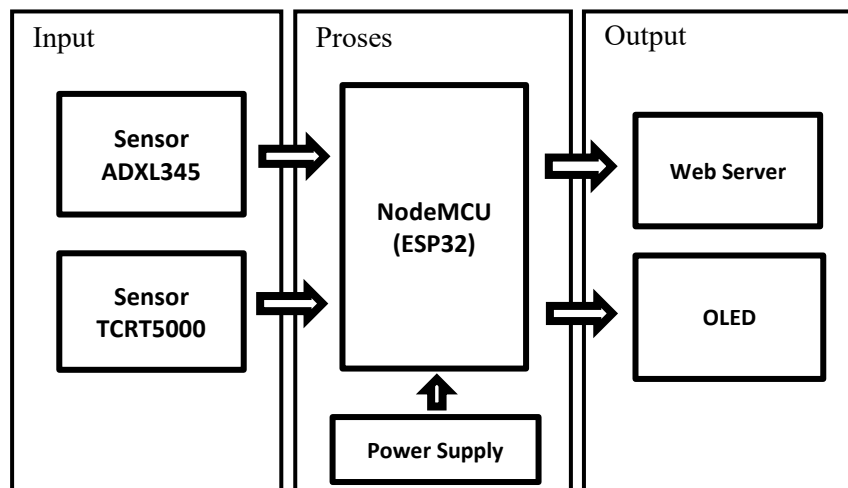
4. HASIL DAN PEMBAHASAN

Alat dan Bahan

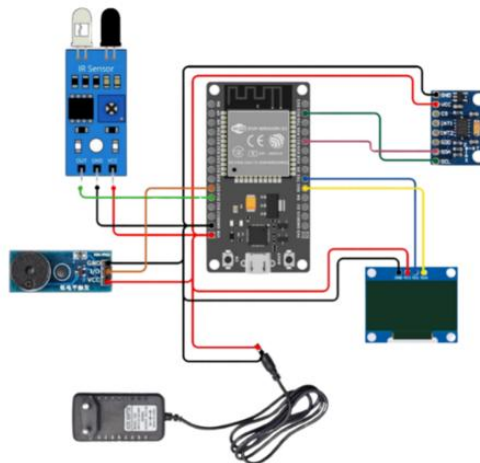
Tabel 1. Alat dan Bahan.

No	Alat	Bahan	
		Nama	Keterangan
1.	ESP32	Kabel Port	1 Unit
2.	Sensor ADXL345	Kabel <i>Female to Female</i>	Secukupnya
3.	Sensor TCRT5000	Kabel <i>Male to Female</i>	Secukupnya
4.	Buzzer	<i>Double Tipe</i>	Secukupnya
5.	Oled	Box Kontrol X4	1 Unit
6.	Laptop	<i>Spacer</i>	4 Unit
7.	Adaptor AC to DC 9 Volt		
8.	ESP32 Baseboard		
9.	Kipas Angin 6 inch		
10.	Solder		

Perancangan *Hardware*



Gambar 1. Blok Diagram Rancangan Alat.

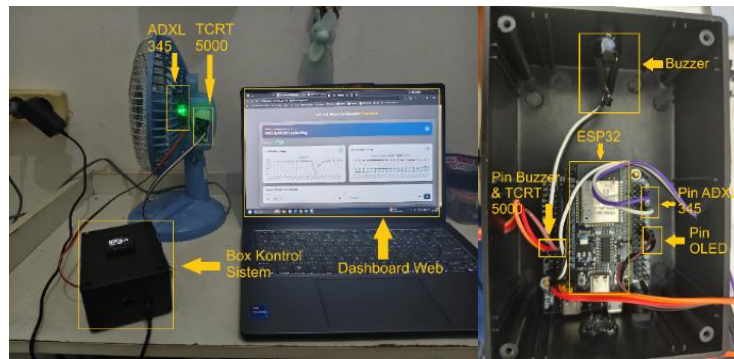


Gambar 2. Wiring Diagram Alat.

Keterangan:

- 1) Sumber 9 V
- 2) ESP32
- 3) Sensor getaran ADXL345
- 4) Sensor kecepatan TCRT5000
- 5) LCD Oled
- 6) Buzzer

Hasil Perancangan Penelitian



Gambar 3. Hasil Perancangan Penelitian.

Berdasarkan gambar tersebut dapat dilihat bahwa perancangan alat pada penelitian ini sudah baik dan berintegritas. Hal tersebut dibuktikan ketika kipas hidup OLED dapat menampilkan data nilai RPM, RMS dan hasil dari *random forest* ke tampilan OLED. Kemudian *dashboard web* dapat menampilkan 2 grafik pembacaan data RPM, RMS dan hasil *random forest* secara *real-time*.

Hasil Pengujian Sensor dan Sinyal

Tabel 1. Hasil Pengujian Dan Perhitungan Nilai *Output* Sensor ADXL 345.

Hasil Analisa	Posisi Sensor	Mean	Standar Deviasi	Koefisien Variasi (CV)
RMS X	Atas	2,2461	0,0516	0,0230
	Samping	1,0327	0,0814	0,0788
	Belakang	2,9314	0,0849	0,0290
RMS Y	Atas	0,2772	0,0128	0,0463
	Samping	0,5723	0,0318	0,0556
	Belakang	0,5357	0,0267	0,0499
RMS Z	Atas	0,6971	0,0157	0,0227
	Samping	1,1645	0,0266	0,0228
	Belakang	0,2239	0,0166	0,0741

Berdasarkan pengujian tersebut posisi atas dipilih sebagai posisi sensor yang paling optimal untuk proses akuisisi data getaran pada penelitian ini, karena mampu menghasilkan data yang lebih representatif untuk proses ekstraksi fitur dan klasifikasi menggunakan algoritma *Random Forest*.

Tabel 2. Hasil Analisa RPM (Sensor TCRT 5000).

No.	Posisi	Mean RPM	Standar Deviasi	Koefisien Variasi (CV)
1.	Depan	2387,32	9,64	0.0040
2.	Samping	2204,81	121,29	0.0550
3.	Belakang	2287,45	50,05	0.0219

Berdasarkan pengujian tersebut diperoleh hasil berupa posisi depan memberikan kestabilan pengukuran terbaik berdasarkan nilai CV. Namun, dalam mempertimbangkan keseimbangan antara kestabilan dan representasi nilai RPM yang mendekati kondisi

sebenarnya, posisi samping dapat dipertimbangkan sebagai lokasi alternatif yang lebih representatif.

Tabel 3. Perbandingan Sensor TCRT 5000 Dengan DT-2234C⁺.

Detik ke-n	RPM TCRT 5000	RPM DT-2234C ⁺	Error (%)
1	2360	2420	2.54
2	2360	2420	2.54
3	2360	2420	2.54
4	2138	2420	13.19
5	2307	2420	4.90
6	2307	2420	4.90
7	2307	2420	4.90
8	2307	2440	5.77
9	2307	2440	5.77
10	2307	2440	5.77
11	2360	2420	2.54
12	2360	2420	2.54
13	2360	2420	2.54
14	2360	2420	2.54
15	2360	2420	2.54
Error Rata-Rata			4,37

Berdasarkan tabel tersebut diperoleh hasil bahwa simpangan atau selisih data nilai RPM antara sensor TCRT 5000 dengan DT-2234C⁺ memiliki keluaran nilai *error* rata-rata sebesar 4,37 % dari total ke 15 data tersebut dan menurut peneliti presentase *error* tersebut sudah sangat baik karena mendekati 0% dan tidak perlu dilakukan adanya kalibrasi *output* sensor pada program karena nilai *output* RPM sudah sesuai dengan spesifikasi kipas angin.

Tabel 4. Hasil Input ADXL 345.

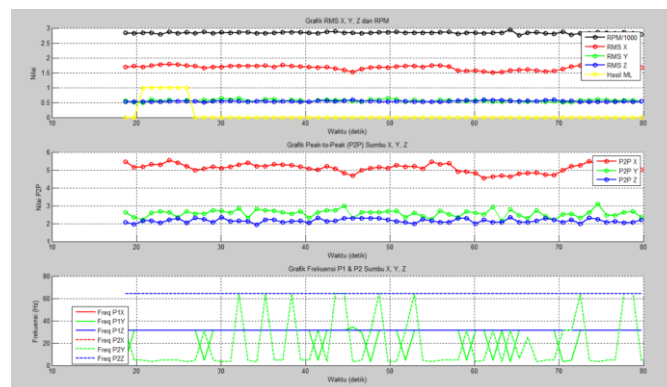
Detik Ke-n	PPS	RMS X (m/s ²)	RMS Y (m/s ²)	RMS Z (m/s ²)
1	900	0.71	0.44	0.56
2	1224	0.7	0.44	0.56
3	1210	0.74	0.43	0.56
4	1107	0.71	0.41	0.57
5	1080	0.72	0.42	0.58
6	1186	0.72	0.42	0.57
7	1094	0.72	0.43	0.55
8	1311	0.71	0.41	0.55
9	1177	0.73	0.39	0.57
10	1068	0.75	0.44	0.58
11	1167	0.93	0.49	0.67
12	1126	0.97	0.47	0.66
13	1194	1.0	0.47	0.66
14	1231	1.0	0.46	0.65
15	1123	0.97	0.45	0.65

Tabel diatas menunjukkan hasil dari pengujian komputasi sinyal *input* sensor ADXL 345. Pada data PPS (*Pulse Per Second*) selama 15 detik diketahui hasil rata-rata PPS nya sebesar 1146,53. Ini menandakan bahwa rata-rata pulsa atau sinyal *input* sensor ADXL 345 sebanyak 1146,53 kali dalam 1 detik.

Tabel 5. Hasil *Input* TCRT 5000.

Detik ke-n	PPS	RPM
1	120	2400
2	119	2380
3	119	2380
4	119	2380
5	119	2380
6	120	2400
7	119	2380
8	119	2380
9	119	2380
10	119	2380
11	119	2380
12	119	2380
13	119	2380
14	119	2380
15	119	2380

Tabel diatas menunjukkan hasil dari komputasi sensor TCRT 5000 dimana dapat dilihat nilai rata-rata PPS atau *pulse per second* yang dihasilkan ialah sebesar 119,13. Hal ini berarti dalam setiap detik mikrokontroller menerima *input* sinyal dari TCRT 5000 rata-rata sebanyak 119,13 kali.



Gambar 4. Grafik Data Pengujian Akurasi.

Terdapat 60 data yang diperoleh dari pengujian yang dilakukan selama 60 detik ini. Dari 60 data yang telah diujikan terhadap model algoritma klasifikasi *random forest* yang telah dibuat tersebut, hasil *output random forest* yang menyimpang hanya 6 data. Artinya, data *output* ML memperoleh keakuratan sebesar 90% sehingga dapat disimpulkan algoritma klasifikasi *random forest* yang dipakai dapat mengklasifikasikan pengunjung dengan baik dan akurat. Itupun data *error* 10% yang tidak akurat terjadi pada saat kipas angin baru menyala.

Jadi sangat wajar jika terjadi lonjakan getaran yang abnormal. Oleh karena itu, algoritma yang dipakai dalam penelitian ini dapat disimpulkan berjalan secara efisien.

Pengujian Komputasi *Machine Learning*

Tabel 6. Pengujian Komputasi *Machine Learning*.

Millis	Selisih (ms)	RP M	RMS X	RMS Y	RMS Z	Hasil ML	STATUS
5239	0						Start loop awal
6236	997	0	-0.39	-0.16	8.94		Ambil raw data sensor
6247	11	0	0.4	0.56	0.4		Pengolahan data sensor
6259	12	0	0.4	0.56	0.4	MESIN MATI	Pengambilan keputusan ML
6298	39	0	0.4	0.56	0.4	MESIN MATI	Olah data ke Oled dan server
6310	12						Start loop awal
7308	998	624	0.39	-0.16	8.63		Ambil raw data sensor
7319	11	6031	0.42	0.7	0.39		Pengolahan data sensor
7331	12	6031	0.42	0.7	0.39	BUSHING LGR	Pengambilan keputusan ML
7370	39	6031	0.42	0.7	0.39	BUSHING LGR	Olah data ke Oled dan server
7382	12						Start loop awal

Berdasarkan tabel diatas diketahui waktu komputasi algoritma klasifikasi *random forest* yang dipakai ialah 1 siklus atau *loop* dalam waktu 1 detik. Selanjutnya waktu yang dibutuhkan tiap proses mulai dari ambil data mentah dari sensor dibutuhkan waktu 997 ms. Selanjutnya untuk pengolahan data mentah sensor menjadi nilai getaran dan kecepatan selama 11 ms. Proses pengambilan keputusan yang dilakukan oleh model *random forest* juga sangat cepat yaitu selama 12 ms. Dan terakhir proses olah data sensor pada oled dan kirim ke server yaitu selama 39 ms. Oleh karena itu, algoritma yang dipakai dalam penelitian ini dapat disimpulkan berjalan secara efisien.

Evaluasi *Random Forest*

Tabel 7. Evaluasi Model *Random Forest*.

Parameter	Nilai (%)
Akurasi	100
<i>Precision</i>	100
<i>Recall</i>	100
<i>F1-Score</i>	100

Berdasarkan tabel diatas model *random forest* mendapatkan akurasi sebesar 100%. Hasil tersebut menunjukkan bahwa model yang telah dibuat mampu mengklasifikasikan semua data dengan benar. Lalu diketahui pula nilai *precision* sebesar 100% menunjukkan bahwa prediksi yang diberikan model memiliki tingkat ketepatan yang tinggi. Nilai *recall* sebesar 100% menunjukkan kemampuan model dalam mengenali data pada setiap kategori dengan baik. Selain itu, nilai *F1-Score* sebesar 100% menunjukkan keseimbangan yang baik antara *precision* dan *recall*.

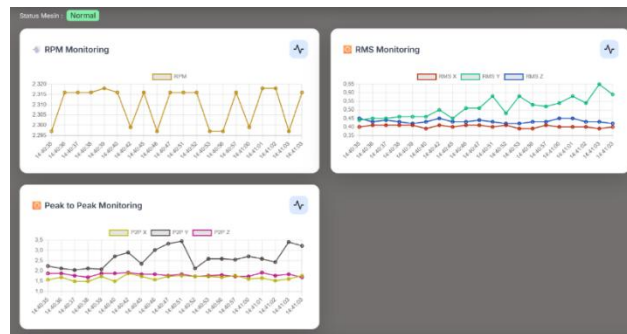
Pengujian Sistem Secara Keseluruhan

Pengujian ini dilakukan dengan 3 kondisi kipas angin yang berbeda-beda. Pertama ialah kondisi kipas angin normal, kedua kondisi kipas angin yang batang as nya telah bengkok dan ketiga ialah kondisi kipas yang *bushing* atau rumah batang as nya telah rusak. Data hasil pengukuran dikirim menggunakan metode POST HTTP. Rata-rata waktu pembaruan data ke *web server* sekitar satu detik. *Dashboard* menampilkan grafik RPM, RMS, dan *Peak to Peak* sehingga kondisi motor dapat dipantau secara *real-time*.

Tabel 8. Hasil Pengujian Kipas Angin Normal.

NO	RPM	RMS X (m/s ²)	RMS Y (m/s ²)	RMS Z (m/s ²)	P2P X (m/s ²)	P2P Y (m/s ²)	P2P Z (m/s ²)
1	2299	0.4	0.43	0.45	1.13	1.22	1.27
2	2316	0.41	0.45	0.43	1.16	1.27	1.22
3	2316	0.41	0.45	0.44	1.16	1.27	1.24
4	2316	0.41	0.49	0.44	1.16	1.39	1.24
5	2299	0.41	0.47	0.44	1.16	1.33	1.24
6	2316	0.41	0.46	0.43	1.16	1.30	1.22
7	2297	0.4	0.47	0.43	1.13	1.33	1.22
8	2318	0.41	0.46	0.42	1.16	1.30	1.19
9	2316	0.39	0.46	0.43	1.10	1.30	1.22
10	2299	0.41	0.5	0.45	1.16	1.41	1.27

NO	FREQ P1X (Hz)	FREQ P1Y (Hz)	FREQ P1Z (Hz)	FREQ P2X (Hz)	FREQ P2Y (Hz)	FREQ P2Z (Hz)	CURRENT STATUS	CREATED AT
1	39	20	39	56	39	55	NORMAL	2026-07-28 14:40:36
2	39	39	39	56	28	56	NORMAL	2026-07-28 14:40:36
3	39	20	39	56	39	56	NORMAL	2026-07-28 14:40:37
4	39	39	39	56	39	56	NORMAL	2026-07-28 14:40:37
5	39	20	39	56	37	56	NORMAL	2026-07-28 14:40:38
6	39	20	39	56	39	56	NORMAL	2026-07-28 14:40:38
7	39	20	39	56	39	56	NORMAL	2026-07-28 14:40:39
8	39	20	39	56	37	56	NORMAL	2026-07-28 14:40:39
9	39	20	39	56	37	56	NORMAL	2026-07-28 14:40:40
10	39	39	39	56	37	56	NORMAL	2026-07-28 14:40:42



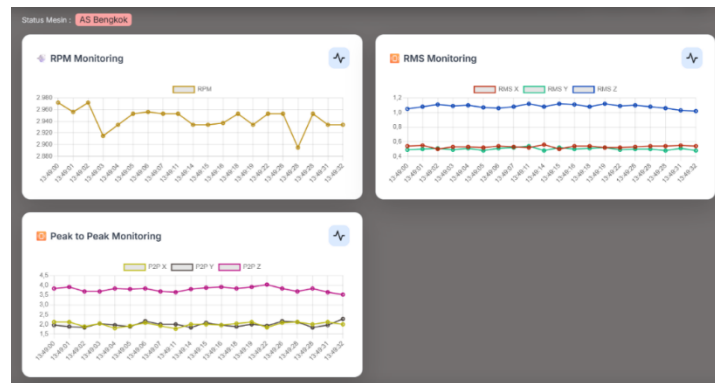
Gambar 5. Tampilan *Dashboard Web* Pengujian Normal.

Diketahui bahwa dari pengujian kipas angin normal hasil pengolahan algoritma klasifikasi *random forest* didapatkan tidak ada data pengolahan *random forest* yang tidak sesuai dengan kondisi kipas yang sebenarnya. Dari 10 data yang dihasilkan semuanya menghasilkan data hasil pengolahan *random forest* “Normal”. Dari pengujian tersebut menunjukkan bahwa dari 10 data yang dihasilkan hasil *output* dapat terbilang stabil dan baik.

Tabel 9. Hasil Pengujian Kipas Angin As Bengkok.

NO	RPM	RMS X (m/s ²)	RMS Y (m/s ²)	RMS Z (m/s ²)	P2P X (m/s ²)	P2P Y (m/s ²)	P2P Z (m/s ²)
1	2934	0.54	0.48	1.02	1.53	1.36	2.88
2	2934	0.56	0.48	1.03	1.58	1.36	2.91
3	2917	0.56	0.47	1	1.58	1.33	2.83
4	2915	0.56	0.47	1.02	1.58	1.33	2.88
5	2915	0.54	0.48	1.04	1.53	1.36	2.94
6	2953	0.56	0.47	0.99	1.58	1.33	2.80
7	2972	0.54	0.49	0.99	1.53	1.39	2.80
8	2975	0.53	0.5	1.04	1.50	1.41	2.94
9	2934	0.54	0.49	1.05	1.53	1.39	2.97
10	2972	0.54	0.51	1.04	1.53	1.44	2.94

NO	FREQ P1X (Hz)	FREQ P1Y (Hz)	FREQ P1Z (Hz)	FREQ P2X (Hz)	FREQ P2Y (Hz)	FREQ P2Z (Hz)	CURRENT STATUS	CREATED AT
1	28	28	28	67	67	40	AS BENGKOK	2026-07-28 13:49:32
2	28	28	28	67	67	40	AS BENGKOK	2026-07-28 13:49:36
3	28	28	28	67	67	40	AS BENGKOK	2026-07-28 13:49:37
4	28	28	28	67	67	40	AS BENGKOK	2026-07-28 13:49:38
5	28	28	28	67	68	40	AS BENGKOK	2026-07-28 13:49:41
6	28	28	28	67	65	40	AS BENGKOK	2026-07-28 13:49:42
7	28	28	28	67	67	40	AS BENGKOK	2026-07-28 13:49:43
8	28	28	28	67	40	40	AS BENGKOK	2026-07-28 13:49:44
9	28	28	28	67	67	40	AS BENGKOK	2026-07-28 13:49:45
10	26	26	26	68	68	40	AS BENGKOK	2026-07-28 13:49:48



Gambar 6. Tampilan *Dashboard Web* Pengujian As Bengkok.

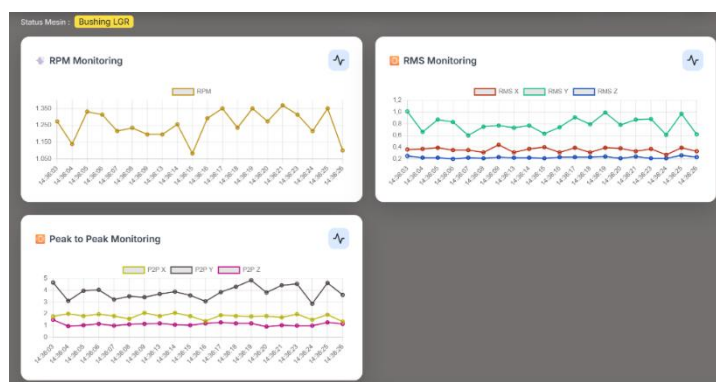
Diketahui bahwa dari pengujian kipas angin as bengkok hasil pengolahan algoritma klasifikasi *random forest* didapatkan tidak ada data pengolahan *random forest* yang tidak sesuai dengan kondisi kipas yang sebenarnya. Dari semua 10 data yang sudah diuji, dihasilkan semuanya menghasilkan data hasil pengolahan *random forest* yaitu “AS Bengkok”. Dari pengujian tersebut menunjukkan bahwa dari 10 data yang dihasilkan hasil *output* dapat terbilang stabil dan baik.

Tabel 10. Hasil Pengujian Kipas Angin *Bushing Rusak*.

NO	RPM	RMS X (m/s ²)	RMS Y (m/s ²)	RMS Z (m/s ²)	P2P X (m/s ²)	P2P Y (m/s ²)	P2P Z (m/s ²)
1	1332	0.39	0.87	0.22	1.10	2.46	0.62
2	1314	0.35	0.83	0.2	0.99	2.35	0.57
3	1216	0.35	0.6	0.22	0.99	1.70	0.62
4	1235	0.31	0.75	0.21	0.88	2.12	0.59
5	1196	0.44	0.77	0.23	1.24	2.18	0.65
6	1196	0.31	0.73	0.22	0.88	2.06	0.62
7	1389	0.41	0.91	0.26	1.16	2.57	0.74
8	1256	0.37	0.77	0.22	1.05	2.18	0.62
9	1351	0.36	0.89	0.25	1.02	2.52	0.71
10	1082	0.4	0.63	0.21	1.13	1.78	0.59

NO	FREQ P1X (Hz)	FREQ P1Y (Hz)	FREQ P1Z (Hz)	FREQ P2X (Hz)	FREQ P2Y (Hz)	FREQ P2Z (Hz)	CURRENT STATUS	CREATED AT
1	15	14	17	28	10	28	BUSHING LGR	2026-07-28 14:36:05
2	15	12	26	25	14	39	BUSHING LGR	2026-07-28 14:36:06
3	15	10	25	32	14	32	BUSHING LGR	2026-07-28 14:36:07
4	14	12	37	15	15	25	BUSHING LGR	2026-07-28 14:36:08
5	12	12	35	25	10	23	BUSHING LGR	2026-07-28 14:36:09
6	14	12	35	29	14	28	BUSHING LGR	2026-07-28 14:36:13
7	17	14	37	28	15	29	BUSHING LGR	2026-07-28 14:36:14

8	15	12	37	15	14	29	BUSHING LGR	2026-07-28 14:36:14
9	15	14	37	15	10	28	BUSHING LGR	2026-07-28 14:36:15
10	15	10	35	43	21	42	BUSHING LGR	2026-07-28 14:36:15



Gambar 7. Tampilan *Dashboard Web* Pengujian *Bushing Rusak*.

Diketahui bahwa dari pengujian kipas angin *bushing* rusak hasil pengolahan algoritma klasifikasi *random forest* didapatkan tidak ada data pengolahan *random forest* yang tidak sesuai dengan kondisi kipas yang sebenarnya. Dari semua 10 data yang sudah diuji, dihasilkan semuanya menghasilkan data hasil pengolahan *random forest* yaitu “Bushing LGR”. Dari pengujian tersebut menunjukkan bahwa dari 10 data yang dihasilkan hasil *output* dapat terbilang stabil dan baik.

Berdasarkan seluruh hasil pengujian dapat diketahui bahwa sistem monitoring yang dirancang mampu melakukan pengukuran percepatan getaran menggunakan sensor ADXL345, pengukuran kecepatan putaran menggunakan sensor TCRT5000, pengolahan data menggunakan metode RMS, serta klasifikasi kondisi motor menggunakan algoritma *Random Forest*. Data hasil pengukuran juga berhasil dikirimkan ke *dashboard Web Server* secara *real-time* sehingga kondisi motor dapat dipantau dengan lebih mudah.

5. KESIMPULAN

Dari beberapa pengujian yang telah dilaksanakan dalam penelitian kali ini, dapat diambil beberapa kesimpulan yaitu antara lain: (1) Alat yang telah dibuat berhasil menerapkan sistem yang direncanakan yaitu mengintegrasikan sensor putaran TCRT 5000 dan sensor getaran ADXL 345 dengan mikrokontroler ESP32 sebagai kontrol unitnya dengan platform IoT berbasis Web melalui metode pengiriman Post HTTP. Hal ini dapat dibuktikan pada bab pengujian web server dengan rata-rata prosesnya 1 detik. (2) Penerapan *Machine Learning* dengan algoritma *Random Forest* pada penelitian ini memiliki siklus waktu program yang

sangat cepat, dapat dibuktikan pada bab pengujian komputasi machine learning dengan loop counter sebesar 12 ms per loop. (3) Dashboard web yang dibuat memiliki tingkat penggunaan yang baik bagi pengguna Teknik dan non Teknik karena memiliki implementasi visualisasi data dalam bentuk grafik real-time dan juga mempunyai fitur export data ke excel. (4) Algoritma *Random Forest* yang diterapkan pada penelitian ini berhasil mengklasifikasikan kondisi dari kipas angin normal, bushing dan as bengkok. Penggunaan *decision tree* sebanyak 15 dan kedalam tiap *tree* sebesar 5 membuat keputusan yang diproses oleh model ini terbukti sangat baik untuk proses klasifikasi. Berdasarkan hasil pengujian dengan python, nilai akurasi model diketahui sebesar 100%.

DAFTAR REFERENSI

- Anuar, M., Rostan, M., & Hawa, A. (2024). IoT based motor vibration and temperature detector for real-time monitoring, *5*(2), 10–22.
- Arciniegas, S., Rivero, D., Piñan, J., Diaz, E., & Rivas, F. (2025). IoT device for detecting abnormal vibrations in motors using TinyML. *Discover Internet of Things*. <https://doi.org/10.1007/s43926-025-00142-4>
- Chan, K., & Dixit, S. (2026). Data-driven predictive maintenance of induction motors using self-supervised and federated learning on noisy current and vibration signals. *Scientific Reports*.
- Chen, J., Qi, Z., Yang, N., & Luo, W. (2022). Development and application of vibration comfort detection device for escalator based on MEMS acceleration sensor. *Journal of Physics: Conference Series*. <https://doi.org/10.1088/1742-6596/2366/1/012004>
- DQLab. (2022). *Studi kasus random forest machine learning untuk pemula data*. DQLab. <https://dqlab.id>
- Hassan, I. U., Panduru, K., & Walsh, J. (2024). An in-depth study of vibration sensors for condition monitoring.
- Heryadi, Y., & Wahyuno, T. (2020). *Machine learning: Konsep dan implementasi* (1st ed.). Gava Media.
- Ibrahim, S. M., Eun-myeong, G., & Iranda, J. (2024). Scalable and secure IoT-driven vibration monitoring: Advancing predictive maintenance in industrial systems. *[Journal/Publisher information unavailable]*, *3*(3), 370–381.
- Kiswanto, A., & Fajri, M. (2024). Transformasi proteksi tegangan: Sistem monitoring IoT untuk pemantauan real-time, *11*(2), 119–128.
- Kusuma, D. T. (2020). Fast Fourier Transform (FFT) dalam transformasi sinyal frekuensi suara sebagai upaya perolehan average energy (AE) musik. *Petir*, *14*(1), 28–35. <https://doi.org/10.33322/petir.v14i1.1022>
- Liao, T., Yan, J., & Chen, C. (2025). Real-time motor fault detection system using microcontroller, *37*(3), 1259–1270.
- Liu, Y., Wang, Y., & Zhang, J. (2012). New machine learning algorithm: Random forest. In *Lecture Notes in Computer Science (LNCS)* (Vol. 7473, pp. 246–252). Springer.

https://doi.org/10.1007/978-3-642-34062-8_32

- Mamatha, G., & Thejaswi, A. H. (2021). Induction motor condition monitoring and controlling based on IoT, *4*(9), 220–225.
- Nasution, E. S., & Hasibuan, A. (2018). Pengaturan kecepatan motor induksi 3 phasa dengan merubah frekuensi menggunakan inverter ALTIVAR 12P. *Journal Sistem Informasi*, *2*(1).
- Rahmatulloh, A., & MSN, F. (2017). Implementasi load balancing web server menggunakan Haproxy dan sinkronisasi file pada sistem informasi akademik Universitas Siliwangi. *Jurnal Nasional Teknologi dan Sistem Informasi*, *3*(2), 241–248. <https://doi.org/10.25077/teknosi.v3i2.2017.241-248>
- Ullah, I., Khan, N., Memon, S. A., Kim, W., Saleem, J., & Manzoor, S. (2025). Vibration-based anomaly detection for induction motors using machine learning, 1–21.
- Usman, S., & Anhar, M. (2024). *Condition monitoring for induction motor using wireless vibration monitoring system* (Vol. 2023). Atlantis Press International BV. <https://doi.org/10.2991/978-94-6463-364-1>
- Yousuf, M., Alsuwian, T., Amin, A. A., Fareed, S., & Hamza, M. (2024). IoT-based health monitoring and fault detection of industrial AC induction motor for efficient predictive maintenance. *Measurement and Control*, *57*(8), 1146–1160. <https://doi.org/10.1177/00202940241231473>
- Yuniko, F. T., & Putra, F. K. (2017). Penerapan teknologi informasi web programming untuk meningkatkan pelayanan publik dalam bidang kebijakan administrasi kependudukan. *Journal of Information System and Informatics Engineering*, *1*(1), 1–68.
- Zaki, H., & Marzolina. (2018). Pengaruh beban kerja dan kompensasi terhadap kepuasan kerja dan turnover intention karyawan. *Jurnal Akuntansi dan Ekonomika*, *8*(1), 109–117.