

Implementasi Algoritma Kompresi Data untuk Meningkatkan Kinerja Pendeteksian Gangguan Kualitas Daya Listrik

Mariana Syamsudin

Politeknik Negeri Pontianak, Indonesia
Jalan Ahmad Yani Pontianak Telepon: (0561) 736180
marianasyamsudin@gmail.com

INFO ARTIKEL

Alamat Web Artikel:

<https://journal.umy.ac.id/index.php/mt/article/view/18386>

DOI:

<https://doi.org/10.18196/mt.v5i1.18386>

Data Artikel:

Diterima:

09 Mei 2023

Direview:

29 Mei 2023

Direvisi :

04 Juni 2023

Disetujui :

12 Juni 2023

Korespondensi:

marianasyamsudin@gmail.com

ABSTRAK

Risiko terjadinya penurunan kualitas daya listrik dapat terjadi pada banyak tahapan yaitu, produksi, transformasi, distribusi, dan konsumsi energi. Salah satu cara untuk menangani masalah kualitas daya adalah dengan melakukan deteksi dan klasifikasi gangguan kualitas daya atau dalam istilah asing disebut *Power Quality Disturbances* (PQDs). Namun, penelitian sebelumnya hanya berfokus pada topik berikut: gangguan kebisingan (*noise*), kegagalan model dalam menggeneralisasi data (*overfitting*), dan waktu yang diperlukan untuk pelatihan dataset. Sebuah strategi baru disarankan untuk mengatasi masalah ini dengan menggabungkan kompresi dataset sinyal 1-Dimensi dengan algoritma klasifikasi *convolutional neural network* (CNN). Dua jenis algoritma kompresi yang diusulkan untuk dievaluasi adalah *wavelet transform* (WT) dan *autoencoder*. Data yang digunakan untuk evaluasi adalah kumpulan data sintetik menurut standar IEEE-1159, yaitu empat belas tipe PQDs yang berbeda. Selanjutnya, prosedur klasifikasi PQDs akan mengintegrasikan data terkompresi dengan algoritma klasifikasi CNN. Hasil akhir penelitian memperlihatkan, metode yang disarankan menunjukkan bahwa menggabungkan algoritma kompresi *autoencoder* dan metode klasifikasi CNN dapat mengenali PQDs secara efisien. Bahkan di lingkungan dengan tingkat *noise* 20db, pemrosesan sinyal PQDs mencapai akurasi hingga 97,14 persen dan berhasil memperkecil *overfitting*.

Kata Kunci: Algoritma Kompresi, CNN, *Framework* Klasifikasi.

ABSTRACT

At several points during the process of energy production, transformation, distribution, and consumption, there is some risk related to power quality. The classification of power quality disturbances (PQDs) provides the foundation for handling power quality issues. Previous studies, however, have concentrated on a particular issue—noise disturbance, the model failed to generalize the data (overfitting), and dataset training duration. The convolutional neural network (CNN) classification technique and effective 1-Dimensional dataset compression are combined in a novel approach to address this issue. First, wavelet transform and autoencoder are two forms of compression techniques that are suggested for evaluation. The synthetic dataset was created using fourteen different PQDs forms in accordance with the IEEE-1159 standard. Additionally, the CNN classification method was combined with compressed data in the PQDs classification process. Eventually, the proposed approach indicates that effectively identifying PQDs can be achieved by combining autoencoder compression and CNN classification algorithms. PQDs signal processing reached up to 97.14 percent accuracy and controlled the overfit even in noisy conditions.

Keywords: Classification Framework, CNN, Compression Algorithm.

1. PENDAHULUAN

Kualitas daya atau *power quality* (PQ) berkaitan erat dengan penggunaan energi pada peralatan listrik. Hal ini menjadi masalah penting karena usia pemakaian peralatan elektronik cenderung akan menurun sesuai durasi pemakaiannya [1]. Sedangkan perilaku arus atau tegangan yang khas atau tidak normal, yang dapat merusak fungsi sistem tenaga listrik, dianggap sebagai gangguan kualitas daya atau *Power Quality Disturbances* (PQDs) [2]. Berbagai gangguan elektromagnetik dapat terjadi di berbagai tempat di seluruh sistem, sehingga dapat mempengaruhi energi yang dikirimkan dari produsen listrik ke berbagai titik permintaan. Beberapa gangguan sistem yang paling sering terjadi adalah *Sags, Swells, Harmonics, Flicker, Transients, dan Interrupts* [3].

Fenomena ini membawa dampak teknis seperti mesin menjadi terlalu panas, pemicu kerusakan pada peralatan proteksi dan kerusakan isolasi, termasuk kerusakan perangkat industri, perangkat lunak komputer, dan perangkat keras komputer [4]. Berdasarkan hal yang telah dijelaskan tersebut, sangat penting untuk mengenali jenis gangguan ini untuk menentukan tindakan yang harus diambil untuk meminimalisasi gangguan. Sejauh ini, klasifikasi dan deteksi gangguan telah dikenal sebagai kunci untuk menjaga kualitas daya dalam konteks *smart grid*, dimana perangkat lunak dan teknologi informasi dan komunikasi diintegrasikan untuk memantau dan mengendalikan sistem tenaga listrik. Sensor kualitas daya berbasis *Internet of things* (IoT) dapat diinstal di sepanjang jaringan distribusi, dengan tujuan mengirimkan informasi tentang konsumsi dan gangguan ke perangkat deteksi melalui infrastruktur komunikasi dua arah.

Fitur seperti informasi statistik, parameter spasial-temporal, dan perilaku sinyal PQ stasioner dan non-stasioner digunakan untuk mengklasifikasikan gangguan kualitas daya [5]. Para akademisi sebelumnya telah melakukan penelitian tentang kompresi data PQDs dengan *Wavelet Transform* (WT) [6]. Salah satu studi mengusulkan algoritma klasifikasi masalah kualitas daya berdasarkan metode *Multilayer Perceptron Neural Network*. WT dan analisis sensitivitas digunakan untuk mengekstrak fitur dan meminimalkan dimensi. Dengan akurasi klasifikasi 99,81 persen, secara akurat dapat mengenali enam kelainan mendasar PQ [7].

Meskipun metode WT menunjukkan hasil yang baik dalam mendeteksi gangguan daya dan berhasil mengurangi ukuran *file* dataset gangguan daya, metode ini menunjukkan beberapa kelemahan, seperti ketergantungan akurasi terhadap pemilihan *mother wavelet*, performa klasifikasinya juga sangat bergantung pada pemilihan fitur dan metode klasifikasi yang digunakan [8]. Karena itu, metode baru untuk pemrosesan sinyal dengan berbagai aplikasi dan kemampuan anti-*noise* dipandang perlu untuk dirancang.

Deep learning (DL) dan teknik data *mining* lainnya telah digunakan untuk melakukan kompresi sinyal. *Deep Stacked Auto-Encoder* digunakan untuk melakukan kompresi data dari *smart meter*. Kompresor nonlinear menghasilkan tingkat kompresi yang tinggi dengan jumlah kehilangan informasi yang rendah [9]. Walaupun pada penelitian sebelumnya telah mengimplementasikan algoritma kompresi yang dijelaskan di atas, masih terdapat masalah lain yang berhubungan dengan kualifikasi PQDs. Kualitas hasil klasifikasi memberikan informasi tentang bobot karakteristik input tetapi tidak mempengaruhi hasil ekstraksinya. Karena kurangnya karakteristik yang diekstraksi untuk mengenali PQDs menyebabkan kesalahan dalam proses klasifikasi PQDs, terutama ketika dataset yang digunakan sebagai sampel mengandung *noise*. Karena itu, kerangka analisis baru harus digunakan untuk mempersingkat proses analisis dan meningkatkan akurasi.

Mempertimbangkan masalah yang telah disebutkan sebelumnya, berikut adalah kontribusi utama dari artikel ini:

- a. Klasifikasi menggunakan algoritma CNN telah terbukti jauh lebih unggul dalam mengekstrak karakteristik data, umumnya digunakan untuk klasifikasi dataset dalam bentuk gambar (*image*) atau 2-Dimensi. Fakta ini adalah salah satu celah penelitian yang harus diperiksa kelayakannya apabila diimplementasikan pada data PQD 1-Dimensi.
- b. Untuk memvalidasi pendekatan klasifikasi CNN yang diusulkan, penelitian ini melakukan perbandingan detail terhadap waktu pelatihan, akurasi, dan parameter model dengan menggunakan dua algoritma kompresi, yaitu *wavelet transform*, dan *autoencoder*.

2. METODE PENELITIAN

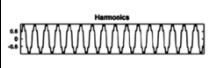
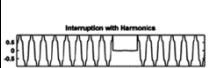
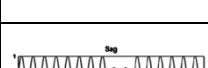
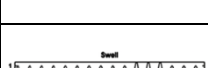
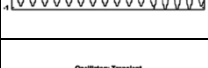
Pada bagian ini dijelaskan metode yang digunakan untuk menghasilkan dataset PQDs sintetis, algoritma data kompresi yang diusulkan dan *framework* algoritma klasifikasi berbasis CNN.

2.1. Menghasilkan Dataset PQD Sintetis

Kumpulan data sintetis dibuat menggunakan empat belas kategori PQDs yang berbeda, yaitu *flicker*, *flicker-harmonic*, *flicker-sag*, *flicker-swell*, *harmonic*, *interrupt*, *interrupt-harmonic*, *normal*, *notch*, *sag*, *sag-harmonic*, *swell*, *swell-harmonic*, dan *transient-oscillation*. Model matematis jenis gangguan dan parameter yang digunakan sesuai dengan standar IEEE-1159 [10].

Variabel frekuensi sampel ditetapkan pada 3200 Hz pada konfigurasi awal. Frekuensi dasar diatur ke 60 hertz, dan setiap sampel memiliki sepuluh siklus (533 poin). Setiap gangguan berisi 10.000 sampel data pelatihan untuk mencapai kualitas dataset pembelajaran. Hasilnya, total sampel yang digunakan untuk pelatihan data adalah 140.000, dengan 80 persen digunakan untuk proses pembelajaran dan 20 persen digunakan untuk validasi. Selain itu, setiap kategori menerima 1.000 data pengujian, dengan jumlah total 14.000 data pengujian dari 14 kategori. Dalam penelitian ini, data dikumpulkan sebanyak dua kali. Yang pertama menggunakan dataset sintetis tanpa *noise*, sedangkan yang kedua menggunakan dataset yang telah dikontaminasi oleh *Gaussian White Noise* (GWN) dengan nilai variabel *Signal-Noise Ratio* (SNR) 40 dB dan 20 dB [11]. Seperti ditunjukkan pada Tabel 1, kumpulan data dihasilkan dengan mengimplementasikan model matematika PQDs dalam program MATLAB.

Tabel 1. Model Matematis PQDs

Kategori	Formula Matematika	Parameter
	$y(t) = A[1 + \lambda \sin(\omega_f t)] \sin(\omega t)$	$8 \leq f_f \leq 25 \text{ Hz}$, $\omega_f = 2\pi f_f$, $0.05 \leq \lambda \leq 0.1$
	$y(t) = A[1 + \lambda \sin(\omega_f t)] [\alpha_1 \sin(\omega t) + \alpha_3 \sin(3\omega t) + \alpha_5 \sin(5\omega t)]$	$0.05 \leq \lambda \leq 0.1$, $8 \leq f_f \leq 25 \text{ Hz}$, $0.05 \leq \alpha_3, \alpha_5, \alpha_7 \leq 0.15$, $\sum \alpha_i^2 = 1$
	$y(t) = A[1 + \lambda \sin(\omega_f t) (1 - \alpha(u(t-t_1) - u(t-t_2)))] \sin(\omega t)$	$0.1 \leq \alpha \leq 0.9$, $T \leq t_2 - t_1 \leq 9T$, $0.05 \leq \lambda \leq 0.1$, $8 \leq f_f \leq 25 \text{ Hz}$
	$y(t) = A[1 + \lambda \sin(\omega_f t) (1 + \alpha(u(t-t_1) - u(t-t_2)))] \sin(\omega t)$	$0.1 \leq \alpha \leq 0.8$, $T \leq t_2 - t_1 \leq 9T$, $0.05 \leq \lambda \leq 0.1$, $8 \leq f_f \leq 25 \text{ Hz}$
	$y(t) = A[\alpha_1 \sin(\omega t) + \alpha_3 \sin(3\omega t) + \alpha_5 \sin(5\omega t) + \alpha_7 \sin(7\omega t)]$	$0.05 \leq \alpha_3, \alpha_5, \alpha_7 \leq 0.15$, $\sum \alpha_i^2 = 1$
	$y(t) = A[1 - \alpha(u(t-t_1) - u(t-t_2))] \sin(\omega t)$	$0.9 \leq \alpha \leq 1$, $T \leq t_2 - t_1 \leq 9T$
	$y(t) = A[1 - \alpha(u(t-t_1) - u(t-t_2))] [\alpha_1 \sin(\omega t) + \alpha_3 \sin(3\omega t) + \alpha_5 \sin(5\omega t)]$	$0.9 \leq \alpha \leq 1$, $T \leq t_2 - t_1 \leq 9T$, $0.05 \leq \alpha_3, \alpha_5, \alpha_7 \leq 0.15$, $\sum \alpha_i^2 = 1$
	$y(t) = A[1 \pm \alpha(u(t-t_1) - u(t-t_2))] \sin(\omega t)$	$\alpha \leq 0.1$, $T \leq t_2 - t_1 \leq 9T$, $\omega = 2\pi f$
	$y(t) = \sin(\omega t) - \text{sign}(\sin(\omega t)) \times \{\sum_{n=0}^9 k [u(t - (t_1 - sn) - u(t - (t_2 - sn)))]\}$	$0.01T \leq t_2 - t_1 \leq 0.05T$, $t_2 \leq s$, $t_1 \geq 0$, $0.1 \leq k \leq 0.4$, $c = \{1, 2, 4, 6\}$, $s = \frac{T}{c}$
	$y(t) = A[1 - \alpha(u(t-t_1) - u(t-t_2))] \sin(\omega t)$	$0.1 \leq \alpha \leq 0.9$, $T \leq t_2 - t_1 \leq 9T$
	$y(t) = A[1 - \alpha(u(t-t_1) - u(t-t_2))] [\alpha_1 \sin(\omega t) + \alpha_3 \sin(3\omega t) + \alpha_5 \sin(5\omega t)]$	$0.1 \leq \alpha \leq 0.9$, $T \leq t_2 - t_1 \leq 9T$, $0.05 \leq \alpha_3, \alpha_5, \alpha_7 \leq 0.15$, $\sum \alpha_i^2 = 1$
	$y(t) = A[1 + \alpha(u(t-t_1) - u(t-t_2))] \sin(\omega t)$	$0.1 \leq \alpha \leq 0.8$, $T \leq t_2 - t_1 \leq 9T$
	$y(t) = A[1 + \alpha(u(t-t_1) - u(t-t_2))] [\alpha_1 \sin(\omega t) + \alpha_3 \sin(3\omega t) + \alpha_5 \sin(5\omega t)]$	$0.1 \leq \alpha \leq 0.8$, $T \leq t_2 - t_1 \leq 9T$, $0.05 \leq \alpha_3, \alpha_5, \alpha_7 \leq 0.15$, $\sum \alpha_i^2 = 1$
	$y(t) = A[\sin(\omega t) + \beta e^{-(t-t_1)/\tau} \sin(\omega_n(t - t_1))](u(t-t_2) - u(t-t_1))$	$300 \leq f_n \leq 900$, $\omega_n = 2\pi f_n$, $0.5T \leq t_2 - t_1 \leq \frac{Nc}{3.33} T$, $8 \text{ ms} \leq \tau \leq 40 \text{ ms}$, $0.1 \leq \beta \leq 0.8$

Tabel 2 menampilkan ringkasan data sintetik PQDs,

Tabel 2. Dataset yang Digunakan dalam Pengujian

	<i>Total Amount of Samples</i>	<i>The Level of Noise</i>
<i>Validation set</i>	140,000 x 0.2 = 28,000	no noise, SNR 40 dB, SNR 20 dB
<i>Learning set</i>	140,000 x 0.8 = 112,000	
<i>Testing set</i>	1,000 x 14 = 14,000	
<i>Total</i>	154,000	

2.2. Algoritma Kompresi Data

Kompresi data adalah cara merepresentasikan berbagai jenis data, yang dapat berupa numerik, teks, gambar, video, audio, atau jenis lainnya dengan menggunakan ukuran data minimum. Ada dua jenis kompresi data yaitu *lossy* dan *lossless*. Kompresi data jenis *lossy* menunjukkan bahwa data yang didekompresi berbeda dari aslinya, metode ini lebih disarankan untuk mengompresi data video, audio, dan gambar karena rasio akurasi dan kompresinya yang tinggi sedangkan *file* aslinya terlalu besar untuk dikirim. Berbeda dengan kompresi data jenis *lossless* yang menyiratkan bahwa data asli dan data yang didekompresi adalah identik, sehingga sangat baik digunakan untuk kompresi data teks, angka atau simbol. Untuk penelitian ini digunakan kompresi data jenis *lossless* karena data PQDs dalam format numerik dan teks.

Studi ini akan memberikan wawasan tentang dua algoritma kompresi data, yaitu *Wavelet Transform*, dan *Autoencoder*.

a. Wavelet Transform (WT)

Model WT yang digunakan untuk mengkompresi sinyal gangguan dalam penelitian ini adalah seperti yang digunakan pada pendekatan yang telah digunakan pada penelitian sebelumnya [12]. Pendekatan dekomposisi *wavelet* dipilih untuk mengidentifikasi dan menemukan gangguan sinyal. Untuk melebarkannya dan menggeser bentuk gelombang, pada permulaan digunakan *mother wavelet* $\psi(t)$ (1).

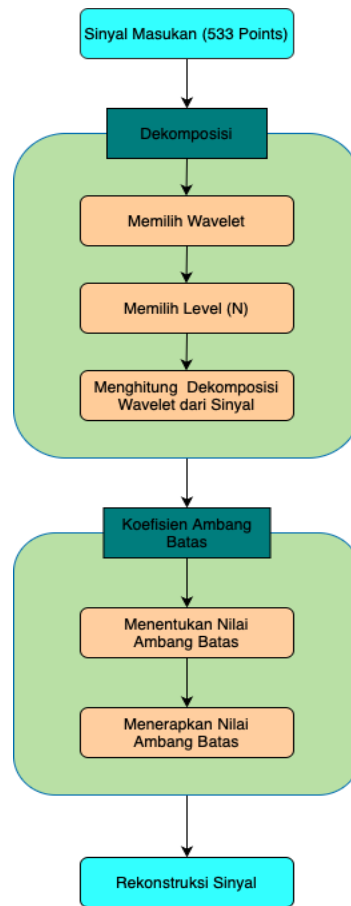
$$\psi_{j,k}(t) = \frac{1}{\sqrt{a}} \psi\left(\frac{t-b}{a}\right) \quad (1)$$

dimana (a) dan (b) mewakili parameter skala dan parameter pergeseran posisi pada sumbu t .

$$DWT_{\psi}x(m, n) = 2^{-m/2} \int_{-\infty}^{\infty} x(t) \psi * \left(\frac{t-n2^m}{2^m}\right) dt \quad (2)$$

sedangkan m mewakili parameter skala dan n mewakili parameter pergeseran waktu.

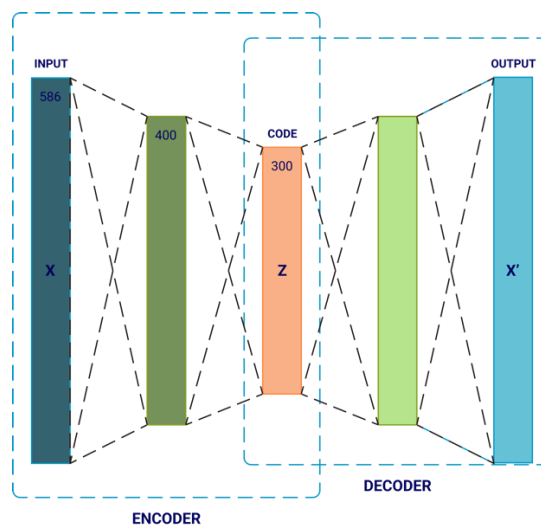
Mother Daubechies Wavelet (Db4) digunakan dalam percobaan ini [13]. Tingkat dekomposisi ditetapkan pada tingkat 4, dan nilai ambang batas ditetapkan pada 62,91. Proses kompresi data terdiri dari langkah-langkah berikut untuk mencapai hasil kompresi yang efisien, seperti yang diilustrasikan pada Gambar 3.



Gambar 1. Tahapan Kompresi Data Menggunakan Algoritma *Wavelet Transform*

b. Autoencoder

Autoencoder (AE) pada dasarnya adalah jaringan syaraf tiruan umpan maju atau *feed-forward Artificial Neural Network* (ANN) yang mencoba membuat ulang lapisan input pada lapisan output. Gambar. 2 mengilustrasikan struktur kedua jaringan *autoencoder*.



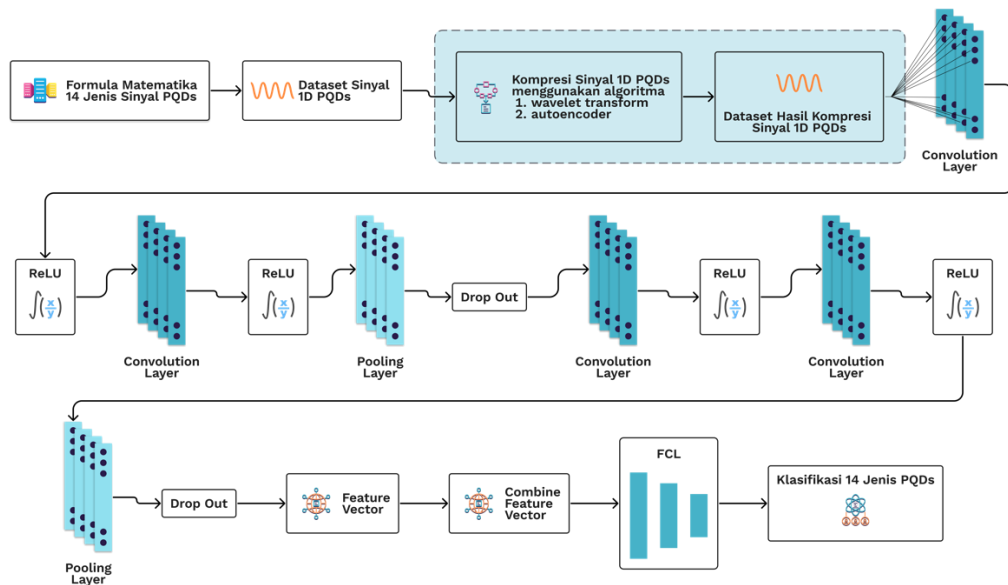
Gambar 2. Representasi Dasar *Autoencoder*

Encoder mengontrol transformasi input dari fitur spasial dimensi tinggi ke dimensi rendah. Sementara *Decoder* merekonstruksi sinyal asli dari kode masing-masing. Dengan menggunakan desain ini, kedua jaringan dilatih secara bersamaan dengan terlebih dahulu menyesuaikan bobot *decoder* dan kemudian bobot *encoder*. Tujuan utama dari desain ini adalah untuk meminimalkan kesenjangan antara keluaran (rekonstruksi) dan masukan (sinyal asli). Sebagai alternatif, jumlah *node* pada lapisan tersembunyi secara substansial lebih kecil dari jumlah *node* pada lapisan input, sehingga memungkinkan *encoder* secara efektif mewakili data input.

Pada skenario penelitian ini, implementasi *autoencoder* di *Keras* akan menggunakan vektor 586 dengan angka antara [0,1], 400 *node* di lapisan tersembunyi, dan ukuran kode 300.

2.3. Framework algoritma klasifikasi berbasis CNN

Secara detail, *framework* yang diusulkan dari arsitektur klasifikasi CNN pada sinyal 1-D diilustrasikan pada Gambar 3. Model ini diadaptasi dari model yang telah dipergunakan pada penelitian sebelumnya [14]. Namun, data yang digunakan sebelum ditransfer ke model klasifikasi CNN melalui lapisan jaringan adalah dataset yang telah dikompresi terlebih dahulu menggunakan WT atau *autoencoder*.



Gambar 3. *Framework* Klasifikasi CNN pada Sinyal 1-D

Struktur jaringan dibagi menjadi tiga bagian: lapisan input, lapisan ekstraksi fitur, dan lapisan klasifikasi [15].

a. Lapisan Input (*Input Layer*)

Pada lapisan *input*, data sinyal satu dimensi diterima. Data tersebut terdiri dari 14 jenis data PQDs tanpa *noise* dan data PQDs yang telah dikontaminasi dengan *noise*.

b. Lapisan Ekstraksi Fitur (*Feature Extraction*)

Bagian ini menyajikan properti penting dari kumpulan data pelatihan yang dibagi menjadi tiga lapisan: *input*, persiapan data, dan lapisan konvolusional. *Input* model ditangani di lapisan pertama oleh lapisan *input*. Jika dataset pelatihan dalam format yang sesuai, maka akan dikirimkan langsung ke lapisan konvolusional dari fase pembelajaran fitur; jika tidak, maka data akan dipersiapkan terlebih dahulu, dikonversi dengan format data WT atau autoencoder pada lapisan data pra-pemrosesan. Lapisan konvolusional akan mengontrol pembelajaran fitur dan menerapkan konvolusi ke data yang masuk. Terdapat empat jumlah lapisan konvolusional, setiap lapisan terdiri dari 10 filter, 3 ukuran *Kernel*, satu *stride*. Model juga dilengkapi dengan satu *pooling layer* yang bertugas

memetakan data untuk meminimalkan dimensi data, *dropout layer* yang berfungsi membuang lapisan secara acak selama proses pelatihan untuk mencegah terjadinya *overfitting* dan ReLU sebagai aktivator. Tantangan utama di bagian ini adalah menentukan jumlah lapisan karena peningkatan jumlah lapisan secara linear akan meningkatkan waktu komputasi.

c. Lapisan Klasifikasi (*Classification Layers*)

Bagian ini menunjukkan label kelas yang dibagi menjadi tiga bagian: lapisan *reshape*, lapisan prediksi kelas, dan lapisan keluaran. Sebelum dikirim ke tahap prediksi, *input* harus berupa vektor yang dibutuhkan oleh layer berikutnya. Tugas ini dijalankan di lapisan *reshape*. Fungsi utama dari lapisan ini adalah untuk memprediksi kelas. Pada bagian *fully connected layer*, yang mencakup *flatten layer*, *dense layer* dengan aktivasi ReLU, dan *dense layer* dengan aktivasi SoftMax, digunakan di bagian ini. Selama proses pelatihan, parameter pada tiap lapisan dimodifikasi untuk menghasilkan model terbaik.

3. HASIL DAN PEMBAHASAN

Pada bagian ini dijelaskan tentang hasil penelitian diantaranya perbandingan pembelajaran algoritma kompresi dan hasil pengujian berbasis akurasi.

3.1. Perbandingan Pembelajaran Algoritma Kompresi Data

Seperti ditunjukkan pada Tabel 3, bagian ini membandingkan algoritma kompresi dan kerjanya dalam model klasifikasi CNN untuk PQD menggunakan kumpulan data tanpa *noise*. Eksperimen pertama menunjukkan bahwa proses klasifikasi dengan menggunakan pendekatan WT, model mencapai akurasi 99,52 persen setelah dua puluh enam episode dalam lima belas menit enam detik. Sementara itu, model hanya dapat mencapai akurasi 99,03 persen pada menit ke-24 pada *epoch* ke-59 ketika menerapkan pendekatan kompresi *autoencoder*. Hal ini disebabkan oleh ketergantungan metode WT pada pemilihan *mother wavelet* yang sesuai untuk menghasilkan akurasi, selain itu kinerja klasifikasinya juga sangat bergantung pada pemilihan fitur dan algoritma klasifikasi yang terhubung. Pada penelitian ini, dengan menerapkan *mother Daubechies wavelet* (db4) dapat mendeteksi ketidaknormalan kualitas daya secara akurat.

Tabel 3. Performansi Pembelajaran Model Klasifikasi CNN

<i>Jenis Algoritma Kompresi</i>	<i>Validation Loss (%)</i>	<i>Validation Accuracy (%)</i>	<i>Accuracy (%)</i>	<i>Epoch of Time</i>	<i>Time consumes for each epoch (detik)</i>	<i>Training Time (menit,detik)</i>
<i>Wavelet Transform</i>	1.58	99.46	99.52	26	36 detik	15m6d
<i>Autoencoder</i>	2.95	99.09	99.03	59	25 detik	24m58d

3.2. Perbandingan Hasil Pengujian Berbasis Akurasi

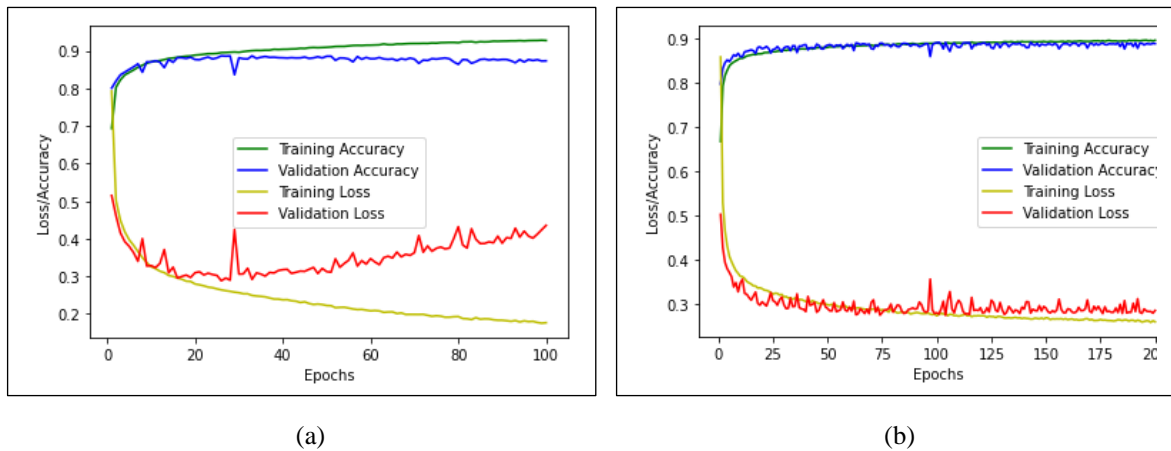
Beberapa metrik yang sering digunakan untuk mengevaluasi model pada data uji secara komprehensif adalah kriteria akurasi (*accuracy*), sensitifitas (*recall*), presisi (*precision*), dan skor F1 (*F1-score*). Akurasi adalah rasio prediksi benar terhadap keseluruhan data, umumnya dianggap sebagai cara utama untuk mengevaluasi PQDs. Dalam percobaan ini, dua algoritma kompresi diterapkan pada empat dataset menggunakan metode klasifikasi CNN. Keakuratan hasil klasifikasi untuk setiap strategi kompresi data ditunjukkan pada Tabel 4. Untuk dataset tanpa *noise*, WT mencapai akurasi sebesar 99,93 persen, diikuti sedikit lebih rendah oleh *Autoencoder* sebesar 99,92 persen. Begitu juga untuk data terkontaminasi-*noise* dengan SNR 40 dB, WT juga menunjukkan kinerja sedikit lebih baik, dengan mencapai nilai akurasi sebesar 98,46 persen sementara metode *autoencoder* menunjukkan nilai akurasi sebesar 98,43 persen. Berbeda dengan data yang terkontaminasi-*noise* dengan SNR 20 dB, algoritma klasifikasi CNN dengan kombinasi algoritma kompresi data *autoencoder* justru menunjukkan performansi jauh lebih baik dari WT, dengan akurasi sebesar 97,14 persen, dibandingkan dengan WT sebesar 95,72 persen. Kondisi ini terjadi karena metode WT sangat peka terhadap interferensi sinyal *noise* sementara dengan menerapkan jumlah

node dan ukuran kode yang tepat, metode *autoencoder* menghasilkan tingkat kompresi data yang tinggi dengan kehilangan informasi yang rendah pada saat proses kompresi berlangsung.

Tabel 4. Detail Kinerja Algoritma Kompresi

Method	ACC (%)	Recall (%)	Precision (%)	F1 (%)
AU_Tanpa <i>noise</i>	99.92	99.41	99.42	99.41
WT_Tanpa <i>noise</i>	99.93	99.52	99.52	99.52
AU_40dB	98.43	88.74	89.82	88.73
WT_40dB	98.46	88.91	89.11	88.96
AU_20dB	97.14	97.14	97.14	97.00
WT_20dB	95.72	95.85	95.64	95.71

Dalam berbagai kondisi, PQDs di lapangan dapat terkontaminasi *noise*. Efek *noise* sangat penting untuk kompresi PQDs, seperti mengakibatkan model *overfitting* karena algoritma mengonseptualisasikan *noise* atau gangguan acak dalam kumpulan data pelatihan [16]. Kurva pembelajaran pada Gambar 4(a) menggambarkan model *overfit* dari dataset PQDs. Tahap di mana nilai *training loss* terus berkurang sementara nilai *validation loss* telah mencapai ke tingkat minimal dan akan terus meningkat. Kondisi *overfit* tidak dapat dihilangkan dalam *supervised machine learning*, tetapi dapat diminimalisasi. Gambar 4(b) menunjukkan bahwa *overfitting* telah diperbaiki setelah menerapkan pendekatan pelatihan metodologis seperti pengoptimalan dengan memilih metode optimalisasi yang tepat, menambahkan *dropout layer* dan pengaturan nilai *hyperparameter* dalam model sistem.



Gambar 4. Dataset PQDs Terkontaminasi *Noise*: (a) Dataset dengan *Overfitting* (B) Dataset dengan Pengurangan *Overfitting*

4. KESIMPULAN

Hasil akhir dari pengujian ini mengusulkan *framework* yang menggabungkan kompresi dataset 1-Dimensi dengan klasifikasi CNN. Setelah mengevaluasi dua jenis algoritma kompresi: *wavelet transform* dan *autoencoder*, model klasifikasi CNN dengan algoritma kompresi WT lebih unggul dari algoritma kompresi *autoencoder* untuk dataset tanpa *noise* dan dataset dengan SNR 40dB yang diuji pada gangguan kualitas daya 1-Dimensi, dalam hal akurasi, sensitifitas, presisi, dan F1 dengan nilai masing-masing sebagai berikut: 99,93 persen, 99,52 persen, 99,52 persen, dan 99,52 persen untuk data tanpa *noise* dan 98,46 persen, 88,91 persen, 89,11 persen, dan 88,96 persen untuk data terkontaminasi-*noise* dengan SNR 40dB. Model ini juga memiliki waktu pembelajaran lebih singkat. Sedangkan untuk data terkontaminasi *noise* dengan SNR 20dB sangat dianjurkan menggunakan algoritma kompresi data *autoencoder* bersama dengan model klasifikasi CNN. Karena *framework* yang diusulkan menunjukkan nilai akurasi lebih baik, yaitu sebesar 97,14 persen, dibandingkan

dengan nilai akurasi WT sebesar 95.72 persen. Dibandingkan dengan model klasifikasi sebelum menambahkan *dropout layer*, menggunakan *dropout layer* pada metode *deep convolution neural network* terbukti dapat mengurangi *overfitting* dan mempercepat pelatihan secara efektif, terutama saat memproses data yang terkontaminasi *noise*. Temuan ini mendukung hipotesis awal penelitian ini bahwa algoritma kompresi WT dan *autoencoder* cocok untuk diimplementasikan pada dataset PQDs dan sangat direkomendasikan untuk digunakan bersama dengan model klasifikasi 1D-CNN untuk mencapai kinerja yang maksimal pada persyaratan sistem. Untuk penelitian lebih lanjut dapat dilakukan evaluasi dengan menggunakan algoritma kompresi lainnya, seperti algoritma kompresi CNN.

DAFTAR PUSTAKA

- [1] Y.-C. Chen, M. Syamsudin, and W. Xu, "An Internet of Things Thermostat Sensor Developed with an Arduino Device Using a Recursively Digital Optimization Algorithm," *J. Inf. Hiding Multim. Signal Process.*, vol. 10, no. 3, pp. 434-446, 2019.
- [2] R. Singh, S. R. Mohanty, N. Kishor, and A. Thakur, "Real-time implementation of signal processing techniques for disturbances detection," *IEEE Transactions on Industrial Electronics*, vol. 66, no. 5, pp. 3550-3560, 2018.
- [3] Y.-C. Chen, M. Syamsudin, and S. S. Berutu, "Pretrained Configuration of Power-Quality Grayscale-Image Dataset for Sensor Improvement in Smart-Grid Transmission," *Electronics*, vol. 11, no. 19, p. 3060, 2022.
- [4] M. Schael and C. Sourkounis, "Influences of power supply quality on electric equipment in production processes," in *IECON 2013-39th Annual Conference of the IEEE Industrial Electronics Society*, 2013: IEEE, pp. 2081-2086.
- [5] M. Pérez-Ortiz, S. Jiménez-Fernández, P. A. Gutiérrez, E. Alexandre, C. Hervás-Martínez, and S. Salcedo-Sanz, "A review of classification problems and algorithms in renewable energy applications," *Energies*, vol. 9, no. 8, p. 607, 2016.
- [6] B. Eristi, O. Yildirim, H. Eristi, and Y. Demir, "A new embedded power quality event classification system based on the wavelet transform," *International Transactions on Electrical Energy Systems*, vol. 28, no. 9, p. e2597, 2018.
- [7] S. Chen and H. Y. Zhu, "Wavelet transform for processing power quality disturbances," *EURASIP Journal on Advances in Signal Processing*, vol. 2007, pp. 1-20, 2007.
- [8] Y. Shen, M. Abubakar, H. Liu, and F. Hussain, "Power quality disturbance monitoring and classification based on improved PCA and convolution neural network for wind-grid distribution systems," *Energies*, vol. 12, no. 7, p. 1280, 2019.
- [9] X. Huang, T. Hu, C. Ye, G. Xu, X. Wang, and L. Chen, "Electric load data compression and classification based on deep stacked auto-encoders," *Energies*, vol. 12, no. 4, p. 653, 2019.
- [10] Y.-C. Chen, S. S. Berutu, L.-C. Hung, and M. Syamsudin, "A New Approach for Power Signal Disturbances Classification Using Deep Convolutional Neural Networks," 2023.
- [11] Y. Chen, "Improved energy detector for random signals in Gaussian noise," *IEEE Transactions on Wireless Communications*, vol. 9, no. 2, pp. 558-563, 2010.
- [12] J. Ning, J. Wang, W. Gao, and C. Liu, "A wavelet-based data compression technique for smart grid," *IEEE Transactions on Smart Grid*, vol. 2, no. 1, pp. 212-218, 2010.
- [13] C.-I. Chen, S. S. Berutu, Y.-C. Chen, H.-C. Yang, and C.-H. Chen, "Regulated Two-Dimensional Deep Convolutional Neural Network-Based Power Quality Classifier for Microgrid," *Energies*, vol. 15, no. 7, p. 2532, 2022.
- [14] M. Syamsudin, "Power Quality Disturbances Recognition Using a Pretrained Algorithm and an Efficient Compression Combination Framework," Asia University, 2022.
- [15] Y.-C. Chen, M. Syamsudin, and S. Berutu, "Regulated 2D Grayscale Image for Finding Power Quality Abnormalities in Actual Data," in *Journal of Physics: Conference Series*, 2022, vol. 2347, no. 1: IOP Publishing, p. 012018.
- [16] X. Ying, "An overview of overfitting and its solutions," in *Journal of Physics: Conference Series*, 2019, vol. 1168, no. 2: IOP Publishing, p. 022022.