

KOMPRESI DATA SENSOR MENGGUNAKAN OPTIMALISASI PARAMETER DCT UNTUK SISTEM MONITORING IKLIM

Dimas Aditya Putra Wardhana¹, Mamat Septyan², Karisma Trinanda Putra³, Syahri Muharom⁴

e-mail: dimasapw@untag-sby.ac.id, mamat.septyan@untag-sby.ac.id

^{1,2}Teknologi Listrik, Fakultas Vokasi

Universitas 17 Agustus 1945 Surabaya, Indonesia

e-mail : karisma@ft.umy.ac.id

³Teknik Elektro, Fakultas Teknik Elektro

Universitas Muhammadiyah Yogyakarta, Indonesia

e-mail : Syahrimuharom@itats.ac.id

⁴Teknik Elektro, Fakultas Teknik Elektro dan Informasi

Institut Teknologi Adhi Tama Surabaya

ABSTRAK

Teknik kompresi telah banyak digunakan dalam distribusi media digital untuk penghematan ruang penyimpanan. Terdapat dua jenis teknik kompresi, yaitu *lossy* dan *lossless*. Keuntungan dari kompresi *lossy* terletak pada penggunaan ruang penyimpanan yang lebih kecil dengan mempertahankan data yang penting. Kompresi *lossy* memiliki potensi untuk diimplementasikan kedalam suatu sistem dengan perangkat sensor yang banyak melalui eksplorasi teknik DCT. Kami juga melakukan percobaan untuk menentukan parameter DCT terbaik agar data yang direkonstruksi tidak jauh berbeda dari data asli dengan menjaga rasio penghematan data. Tujuan utamanya yaitu untuk mengurangi penambahan kapasitas penyimpanan di pusat data terkait dengan sistem pemantauan iklim. Kami menggunakan kumpulan data terbuka, yang terdiri dari enam sensor berbeda yang mewakili urutan data nyata berdasarkan pengukuran di lapangan. Hasil percobaan menunjukkan bahwa konsentrasi energi mempengaruhi kualitas data yang direkonstruksi. Rasio penghematan data terbaik diperoleh dengan menggunakan variasi ambang konsentrasi energi antara 0.9 hingga 0.99, tergantung pada jenis data sensor yang menghasilkan tingkat kesalahan di bawah 5%.

Kata kunci: DCT, monitoring iklim, Rasio penghematan data, Teknik kompresi

PENDAHULUAN

Internet of Things (IoT) merupakan sebuah konsep atau program dimana sebuah objek memiliki kemampuan untuk mengirim atau menerima data melalui jaringan tanpa menggunakan bantuan komputer dan manusia. Saat ini, IoT telah mengalami banyak perkembangan yang dapat dilihat dari tingkat konvergensi teknologi nirkabel, mikroelektromekanis (MEMS), internet, dan Kode QR (*Quick Response*). IoT juga sering diidentifikasi dengan *Radio Frequency Identification* (RFID) sebagai metode komunikasinya.

Selain itu, IoT juga mencakup teknologi berbasis sensor, seperti teknologi nirkabel dan Kode QR. Kemampuan IoT sendiri tidak diragukan lagi. Banyak teknologi telah mengimplementasikan sistem IoT, contohnya sensor cahaya dan sensor suara dari teknologi Google AI dan Amazon Alexa. Kemudian yang terbaru yaitu implementasi *Smart City* di beberapa negara maju. Dengan implementasi tersebut, segala bentuk aktivitas penduduk suatu kota dapat dipantau dengan baik oleh sebuah sistem dengan jaringan database berskala besar.

Kemunculan IoT menyediakan paradigma baru dalam penelitian yang berkaitan dengan ilmu komputer, seperti bidang pertanian, mekanik, teknik sipil, atau medis. Sistem dengan kemampuan pemantauan cerdas dan analisis data dapat menyediakan informasi yang lebih aktual, memungkinkan untuk memberikan prediksi secara akurat, dan mengurangi biaya tambahan lainnya seperti pengelolaan data. Beberapa tahun lalu, kita telah melihat beberapa konsep mutakhir mengenai sistem pemantauan pintar berbasis IoT yang berguna di industri. Konsep ini melibatkan banyak sensor yang terhubung dengan perangkat IoT. Setiap perangkat IoT terhubung dengan sebuah koordinator. Kemudian beberapa koordinator akan terhubung ke pusat data pada *cloud*, sehingga menciptakan sistem kompleks yang membutuhkan kapasitas penyimpanan data yang besar.

Salah satu hambatan dalam implementasi IoT di industri adalah jumlah besar data yang dihasilkan dari berbagai jenis sensor. Selain itu, jumlah sensor juga meningkat seiring dengan bertambahnya jumlah variabel dan area yang perlu dipantau. Pengumpulan data secara terus-menerus dan *real-time* akan meningkatkan penggunaan sumber daya pemrosesan,

bahkan penggunaan daya listrik. Pada kenyataannya, perangkat IoT umumnya memiliki kemampuan pemrosesan data yang rendah dengan konsumsi energi minimal. Dengan demikian, diperlukan algoritma yang diimplementasikan pada perangkat IoT agar menghasilkan volume data yang efisien dan integritas data yang baik.

Salah satu tantangan utama untuk memperluas kemampuan perangkat IoT adalah bagaimana mengelola data besar dengan efisien dan dapat diproses oleh mikroprosesor sederhana. Salah satu solusi untuk menangani volume data yang besar adalah dengan menerapkan teknik kompresi data sehingga beban penyimpanan dan komunikasi dapat dikurangi (Moon et al, 2018; Marrium, et al, 2021). Berbagai algoritma kompresi telah diterapkan untuk memenuhi berbagai persyaratan aplikasi dan banyak di antaranya merupakan algoritma *lossy* seperti Walsh-Hadamart, *Wavelet*, dan *Cosines-based* (Setyaningsih & Harjoko, 2017). Kompresi *lossy* (Okasa et al, 2020) dapat membantu mengurangi ukuran data secara signifikan, tetapi tingkat kesalahan dan kehilangan kualitas data tidak mudah dikendalikan. Beberapa pendekatan untuk peningkatan penggunaan metode kompresi *lossy* telah diusulkan (Zhou et al, 2018; Sun et al, 2021). Dalam beberapa aplikasi sebelumnya, kompresi sering menunjukkan perubahan kecil dalam data, bahkan deformasi yang cukup tajam muncul di area data kecil. Namun, dalam aplikasi pengamatan seperti analisis data aliran *turbulence* (Marin et al, 2018) dan data iklim (Hubbe et al, 2013), kompresi *lossy* masih memungkinkan analisis yang bermakna untuk dieksplorasi dari data yang direkonstruksi. Selain itu, data yang direkonstruksi dapat menunjukkan hasil rata-rata yang sama dengan tingkat kompresi hingga 20% dari data asli dan dapat direkonstruksi sehingga secara statistik tidak dapat dibedakan dari data asli (Liang et al, 2022).

Sebuah kumpulan data terbuka yang terdiri dari sinar matahari, suhu, kelembaban, kecepatan angin, arah angin, dan tekanan udara digunakan untuk memverifikasi teknik kompresi. Kumpulan data ini direkam setiap 5 menit di stasiun cuaca HI-SEAS dalam rentang waktu September hingga Desember 2016. Setiap data sensor merupakan rangkaian dari 2016 data floating point yang mewakili nilai pengukuran nyata. Untuk melihat seberapa efektif algoritma ini, kami menghitung kesalahan rata-rata antara data asli dan data yang direkonstruksi.

Skema yang kami usulkan dapat menghasilkan parameter terbaik untuk setiap sensor yang meningkatkan rasio kompresi seraya memperkecil tingkat kesalahan menggunakan optimasi pada kompresi *Discrete Cosine Transform* (DCT). Dengan menggunakan data input yang telah dikuantisasi, didapatkan hasil yang lebih baik dibanding penelitian sebelumnya (Nakano et al, 2015). Kami juga mencatat bahwa *threshold* konsentrasi energi (q) sangat mempengaruhi rasio kompresi, sehingga untuk

mencapai rasio optimal, diperlukan penyesuaian nilai ini. Kami juga mencatat bahwa penggunaan nilai q antara 0,99% hingga 0,99 menghasilkan akurasi optimal sambil mempertahankan rasio penyimpanan data yang rendah. Hasil-hasil ini jelas menunjukkan bahwa kompresi DCT lebih unggul dalam manajemen data yang efisien dari perangkat IoT dengan mengurangi kebutuhan penyimpanan sambil mempertahankan keakuratan data.

Tabel 1. Kriteria statistik dataset

	SD	NSD	Skewness	Kurtosis
Radiasi matahari	399.44	0.38	0.74	-1.18
Suhu	6.39	0.34	0.21	-1.65
Kelembaban	17.06	0.25	1.04	0.34
Tekanan udara	2.45	0.17	0.62	0.98
Arah angin	80.55	0.22	0.69	0.98
Kecepatan angin	0.03	0.31	-0.33	-1.14

DCT dan *Discrete Wavelet Transform* (DWT) merupakan fungsi matematis yang paling berhasil dalam pemrosesan sinyal di dua puluh tahun terakhir. DWT menggunakan filter *wavelet* tertentu untuk membagi data menjadi komponen frekuensi atau skala yang berbeda, kemudian menganalisis setiap komponen dengan resolusi yang sesuai dengan skala tersebut. DWT ideal untuk mengekstraksi informasi dari sinyal yang bersifat stasioner atau transien. Sehingga sebuah bank filter DWT dapat digunakan untuk memisahkan berbagai pita frekuensi, tanpa pengetahuan eksplisit tentang parameter radar seperti *Pulse Repetition Frequency* (PRF). Karakteristik transformasi pada DWT dipengaruhi oleh *wavelet* induk yang digunakan. DCT menghasilkan *wavelet* yang mengonversi sinyal menjadi komponen frekuensi dasarnya. DCT memiliki hubungan erat dengan DFT, sehingga data direpresentasikan dalam komponen frekuensi. Hasil pemrosesan sinyal menggunakan DCT dianalisis dan dibandingkan dengan hasil pemrosesan menggunakan DWT.

METODE PENELITIAN

Teknik kompresi *lossy* memanfaatkan sifat data dimana meskipun terdapat beberapa ketidakberaturan, secara umum data tersebut memiliki pola periodik yang rata. Apabila ketidakberaturan ini terlalu luas dan tidak dapat diprediksi, teknik kompresi *lossy* akan menghasilkan rasio kompresi yang kurang baik. Oleh karena itu, teknik kompresi *lossy* sangat cocok digunakan pada data yang dihasilkan oleh sensor IoT, terutama data yang terkait dengan pemantauan cuaca atau iklim, dimana data tersebut cenderung stabil secara periodik dengan tingkat ketidakberaturan yang rendah.

A. Karakteristik Dataset

Kami melakukan evaluasi efektivitas algoritma kompresi DCT menggunakan dataset dunia nyata dari stasiun iklim HI-SEAS. Enam variabel digunakan sebagai data pengujian, terdiri dari radiasi matahari, suhu, kelembaban, tekanan udara, arah angin, dan kecepatan angin, dari data iklim yang dikumpulkan selama September hingga Desember 2016. Data dikumpulkan setiap 5 menit atau 288 data dalam sehari. Bila tipe data dalam bentuk floating point, dibutuhkan 4 byte untuk menyimpan sebuah nilai. Karena digunakan 6 jenis instrumen pengukuran, sehingga dalam satu hari dihasilkan 6.912 byte (288 x 6 x 4 byte) data. Jadi dalam seminggu dan sebulan, volume data mencapai sekitar 48 KB dan 207 KB secara berturut-turut.

Selain itu, jumlah node sensor tentu mempengaruhi ketepatan pengukuran berdasarkan karakteristik spasial dan temporal suatu wilayah. Semakin banyak node sensor yang digunakan, semakin besar pula data yang dihasilkan. Pertukaran data yang besar meningkatkan beban jaringan dan daya pemrosesan. Oleh karena itu, perangkat IoT akan lebih sering aktif memproses dan mengirim data sehingga konsumsi daya juga meningkat. Akibatnya, masa pakai perangkat berkurang karena pada umumnya perangkat IoT dirancang dengan menggunakan baterai atau tanpa pasokan daya eksternal.

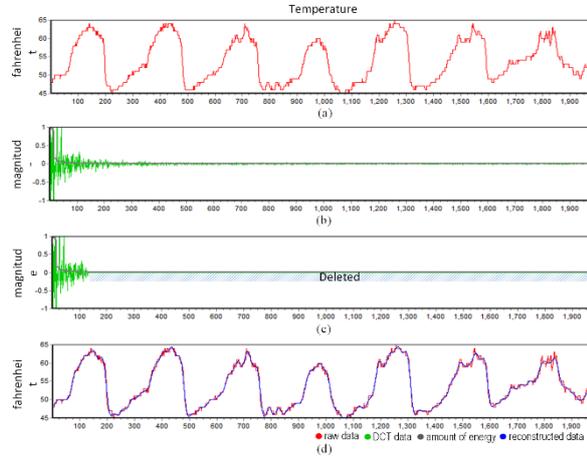
Tabel 1 menyajikan data statistik yang terdiri dari simpangan baku (SD), simpangan baku terstandarisasi (NSD), *skewness*, dan *kurtosis* dari data asli. Simpangan baku adalah nilai statistik yang digunakan untuk menentukan bagaimana data didistribusikan dalam sampel dan seberapa dekat titik data individual dengan nilai rata-rata sampel. Sementara itu, NSD adalah simpangan baku yang diterapkan pada data yang dinormalisasi. *Skewness* adalah nilai yang menunjukkan derajat asimetri dari distribusi. Sedangkan *kurtosis* adalah derajat distorsi dari distribusi.

B. Discrete Cosine Transform

Sinyal pada DCT direpresentasikan dengan magnitude dan frekuensi yang bervariasi. DCT digunakan dalam kompresi sinyal *lossy* karena memiliki pemadatan energi yang baik, yaitu dengan mengkonsentrasikan sebagian besar energi sinyal dalam koefisien yang lebih rendah kemudian informasi yang penting disimpan dengan menggunakan jumlah bit yang sedikit. Untuk menerapkan transformasi DCT pada suatu sinyal dapat diawali dengan melakukan pra-pemrosesan pada sinyal. Kemudian akan terlihat, setelah DCT diterapkan, lebih dari 90% data akan berada dalam komponen frekuensi yang lebih rendah. Sebagai contoh, kita ambil sebuah data *array* berukuran 2016 dengan semua nilainya berupa fungsi sinus sinyal. Dengan implementasi DCT akan didapatkan sekitar 201 data yang mencapai *threshold* pemadatan energi dan data sisanya yang berada dibawah *threshold* dapat diabaikan.

C. Peningkatan Kemampuan Kompresi DCT

Kami menambahkan pra-pemrosesan yang terdiri dari normalisasi data dan penghapusan DC (*Direct Current*). Proses-proses ini mengubah rentang sinyal dan menyesuaikan data dengan nilai rata-rata dari sinyal tersebut. Prosedur lebih rinci dijelaskan dalam Gambar 1.



Gambar 1. Hasil kompresi DCT (a) data input, (b) data DCT sebelum terkompresi, (c) data DCT terkompresi

1. Pra-pemrosesan diterapkan pada data mentah menggunakan normalisasi mean data. Normalisasi digunakan untuk menyesuaikan kumpulan data heterogen sehingga dapat dibandingkan secara relevan. Normalisasi menyederhanakan pengaturan *threshold* dalam berbagai algoritma *threshold* dengan menyesuaikan dataset dalam rentang [0:1], kemudian mengatur titik tengah sinyal pada nilai rata-ratanya.

$$x'_i = \frac{x_i - x_{mean}}{x_{max} - x_{min}} \tag{1}$$

dimana x , x' , x_{mean} , x_{max} , x_{min} menyatakan data asli, data hasil normalisasi, nilai rata-rata data, nilai data terbesar, dan data terkecil.

2. Mentransformasi data dari domain spasial ke domain frekuensi menggunakan DCT. Hasilnya disajikan sebagai jumlah magnitude dan frekuensi yang bervariasi dari input. DCT didefinisikan sebagai berikut:

$$x'_j = \frac{1}{\sqrt{N}} \sum_{i=1}^N x_i \cos\left(\frac{\pi(i + 0.5)j}{N}\right) \tag{2}$$

dimana x , x' , N menyatakan data asli, data hasil transformasi, dan panjang dari x .

3. Menghitung pemadatan energi dengan mengonversi data asli ke dalam bentuk vektor dasar dari DCT. Kemudian perlu didefinisikan koefisien i pada koefisien vektor $\{x'_n, x'_{(n-1)}, x'_{(n-2)}, \dots, x'_1\}$, yang menentukan

berapa banyak koefisien yang diperlukan untuk merepresentasikan jumlah energi dalam sinyal (δ), di mana $0 < \delta < 1$. Nilai $\text{norm}()$ dihitung dengan normalisasi Euclidean (p -norm, $p = 2$) dari vektor $|x'|$.

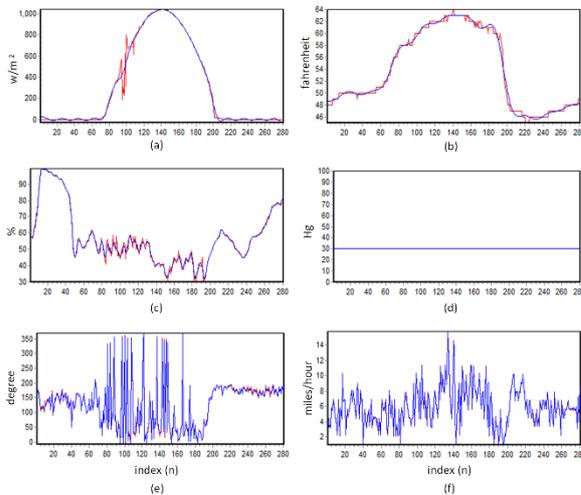
4. Koefisien yang lebih kecil dari nilai *threshold* δ dihapus. Dengan kata lain, nilai-nilai yang tidak menonjol dihilangkan.

$$\text{Remove}(x'_i \in |x'|) \text{ where } (e_i < \delta) \quad (3)$$

5. Untuk merekonstruksi data dari koefisien DCT, diperlukan penerapan inverse DCT yang didefinisikan sebagai berikut:

$$x''_j = \frac{2}{\sqrt{N}} \sum_{i=1}^N x'_i \cos\left(\frac{\pi i(j+0.5)}{N}\right) \quad (4)$$

dimana x' , x'' , N menyatakan koefisien DCT yang tersisa, data rekonstruksi, dan panjang dari x' .



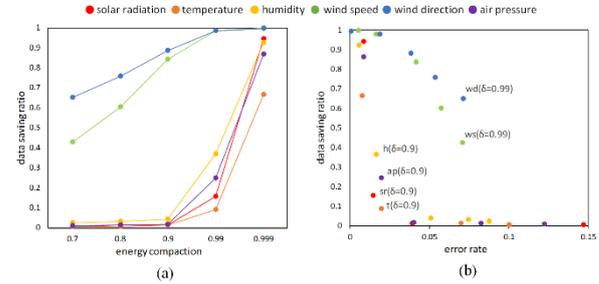
Gambar 2. Perbandingan data yang dipulihkan untuk enam variabel cuaca: (a) radiasi matahari, (b) suhu, (c) kelembaban, (d) tekanan udara, (e) arah angin, dan (f) kecepatan angin menggunakan 99% energi DCT

HASIL DAN PEMBAHASAN

Pada bagian ini, kami akan menjelaskan tingkat fidelitas data dan menunjukkan hasil perbaikan dibandingkan dengan metode dasar. Data *array* sebanyak 2016 yang diperoleh dari pengukuran selama 1 minggu dikompresi untuk menghasilkan rangkaian koefisien DCT. Kemudian kami menghitung pemadatan energi dari koefisien DCT. Dari rangkaian pemadatan energi tersebut, dipilih 5 nilai spesifik sebagai *threshold*, yaitu 0.7, 0.8, 0.9, 0.99, dan 0.999.

Data dengan energi di bawah *threshold* mewakili nilai yang tidak memiliki dampak signifikan dan dapat diabaikan. Nilai ini dihapus sehingga rangkaian koefisien DCT akan secara drastis berkurang. Dari nilai koefisien

DCT yang tersisa, dilakukan rekonstruksi sinyal. Untuk mengevaluasi fidelitas data, kami membandingkan data rekonstruksi dengan data asli. Semakin kecil kesalahan, semakin efisien algoritma kompresi tersebut. Seperti yang ditunjukkan dalam Gambar 3, pada skenario kompresi data tanpa pra-pemrosesan, tingkat kesalahan sedikit lebih tinggi. Di sisi lain, dengan menggunakan pra-pemrosesan yang melibatkan penskalaan data dan penghapusan dc, tingkat kesalahan sedikit lebih rendah.



Gambar 3. Perbandingan rasio penghematan data dengan (a) rasio pemadatan energi, (b) tingkat kesalahan pada data yang telah dikompresi.

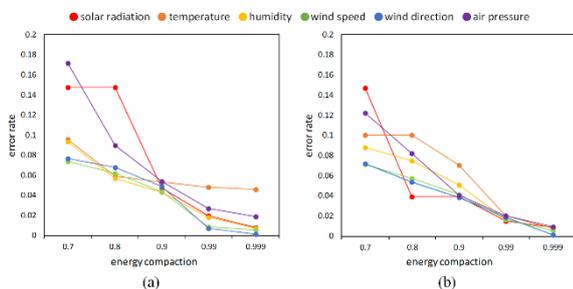
Dari enam variabel data yang diuji, tingkat kesalahan terbesar terdapat pada data suhu dan tekanan udara. Berdasarkan pengamatan visual, sinyal dari kedua variabel tersebut hampir datar. Hasil rekonstruksi sinyal pada gelombang datar sulit diakomodasi oleh DCT. Bentuk awal dan akhir dari data rekonstruksi terlihat osilasi. Namun, dengan pra-pemrosesan, sinyal yang awalnya dalam bentuk mentah diproses terlebih dahulu pada skala $0 < s[n] < 1$.

Kemudian dilakukan proses penyetalan untuk menyesuaikan sinyal dengan nilai rata-ratanya. Dengan penyesuaian ini, osilasi yang terjadi pada awal dan akhir data dapat diminimalkan. Hal ini akan memudahkan algoritma kompresi untuk merekonstruksi sinyal yang telah diskalakan dengan amplitudo yang sangat tinggi. Secara umum, semakin besar nilai *threshold* energi, semakin rendah tingkat kesalahan pada kedua skenario uji, dengan atau tanpa pra-pemrosesan. Kesalahan mulai berkurang ketika *threshold* energi diatur lebih besar atau sama dengan 0,9 untuk semua variabel data. Secara visual, sinyal rekonstruksi dapat menyerupai sinyal asli seperti yang ditunjukkan dalam Gambar 2.

Pada skenario berikutnya, kami menguji rasio penghematan data dengan mengubah *threshold* pemadatan energi. Rasio penghematan data mencerminkan perbandingan antara jumlah data yang dikompresi dan jumlah data asli. Semakin kecil rasio ini, semakin baik algoritma kompresi yang digunakan, karena data asli dapat dikompresi menjadi data yang lebih kecil. Ini berarti kapasitas penyimpanan yang dapat dihemat menjadi lebih besar. Jika kita melihat pengaruh perubahan *threshold* pemadatan energi terhadap rasio penghematan, hasilnya akan berbentuk linear.

Semakin rendah *threshold* pemadatan energi, semakin kecil rasio penghematan data. Dari eksperimen ini, terdapat 2 variabel yang memiliki karakteristik yang berbeda, yaitu data kecepatan angin dan arah angin. Rasio penyimpanan tampaknya lebih tinggi dibandingkan dengan hasil empat variabel lainnya. Hal ini disebabkan karena data asli kecepatan angin dan arah angin terlihat lebih acak dengan amplitudo yang rentan berubah dalam waktu singkat. Sedangkan, pada empat variabel lainnya, sinyal terlihat lebih halus dan stabil. Hal ini dibuktikan oleh karakteristik statistik kecepatan angin dan arah angin yang memiliki nilai *kurtosis* yang tinggi.

Nilai *kurtosis* yang tinggi melambangkan data dengan rentang nilai yang cukup luas, atau dapat dikatakan bahwa data memiliki tingkat acak yang tinggi. Namun, rasio kompresi data minimal dapat mencapai 0.4 dan 0.65 untuk data kecepatan angin dan arah angin, masing-masing, dengan menggunakan *threshold* 0.7 dengan kompensasi tingkat kesalahan yang lebih tinggi. Ini berarti bahwa DCT tetap dapat diterapkan pada variabel-variabel ini, meskipun hanya dapat mencapai penghematan penyimpanan yang sedikit.



Gambar 4. Perbandingan tingkat kesalahan kompresi DCT (a) tanpa pra-pemrosesan (b) dengan pra-pemrosesan

Selanjutnya, perlu diamati efek dari rasio penyimpanan data terhadap tingkat kesalahan. Kami ingin menemukan parameter terbaik untuk enam variabel data yang memberikan hasil yang paling optimal. Hasil tersebut dianggap optimal jika rasio penyimpanan data rendah dan diikuti oleh tingkat kesalahan yang rendah. Pada gambar (b), variasi hasil data untuk beberapa *threshold* pemadatan energi yang berbeda ditampilkan. Nilai optimal diperoleh untuk setiap variabel termasuk radiasi matahari, suhu, kelembaban, dan tekanan udara adalah $d = 0,9$, yang mencapai rasio penghematan data yang rendah. Sementara itu, untuk data kecepatan angin dan arah angin, nilai optimal yang dipilih adalah $d = 0,99$. Nilai tersebut dipilih karena tingkat kesalahan sinyal berada dibawah 5%. Namun, nilai rasio penghematan data masih cukup tinggi dalam kisaran 0,6-0,8. Ini berarti, untuk kedua jenis data ini, jumlah penyimpanan yang dapat dihemat tidak terlalu besar,

PENUTUP

Peningkatan pesat terhadap penggunaan teknologi IoT di berbagai bidang menimbulkan tantangan baru terkait kebutuhan kapasitas ruang penyimpanan. Oleh karena itu, diperlukan solusi yang efektif dan efisien dalam pengumpulan data untuk mengurangi jumlah data dan sumber daya pemrosesan yang dibutuhkan. Penelitian ini bertujuan untuk mengevaluasi efektivitas kompresi data, terutama pada data cuaca yang sangat relevan dengan aplikasi IoT. Kami membandingkan seluruh parameter yang terkait dengan proses kompresi data dan berusaha mencari nilai batas kompresi energi terbaik yang menghasilkan kompresi optimal, dengan tingkat kesalahan tetap rendah dibawah 5%. Hasil eksperimen menunjukkan bahwa kompresi DCT dapat mencapai penghematan data yang signifikan dengan kerugian kualitas data yang rendah. Terutama untuk data cuaca yang terdiri dari radiasi matahari, suhu, kelembaban, dan tekanan udara mencapai rasio penghematan data optimal dengan *threshold* kompresi energi sebesar 0,99. Meskipun pada data kecepatan angin dan arah angin dihasilkan penghematan data yang kurang signifikan dengan menggunakan *threshold* energi sebesar 0,9, namun tingkat kesalahan dapat dipertahankan pada nilai yang rendah. Secara keseluruhan, hasil kompresi dari beberapa jenis data cukup baik dengan penyesuaian yang tepat untuk memastikan keandalan data. Disarankan untuk menggunakan *threshold* kompresi energi yang lebih tinggi untuk data yang memiliki tingkat keacakannya yang tinggi. Pengaturan ini dapat mengurangi data, meskipun tidak secara signifikan, sambil tetap menjaga tingkat kesalahan rendah.

UCAPAN TERIMA KASIH

Kami ingin mengungkapkan rasa terima kasih kepada semua pihak yang telah membantu dalam penulisan artikel ini, terutama kepada lembaga-lembaga yang terlibat dalam penelitian ini, yaitu Universitas 17 Agustus Surabaya, Universitas Muhammadiyah Yogyakarta, Institut Teknologi Adhi Tama Surabaya, dan editor jurnal yang telah membantu dalam penerbitan artikel ini.

DAFTAR PUSTAKA

- Moon, A., Kim, J., Zhang, J., & Son, S. W. (2018). Evaluating fidelity of lossy compression on spatiotemporal data from an IoT enabled smart farm, *Computers and Electronics in Agriculture*, 154, 304–313.
- Marriam, S., Masood, K., & Rashid, H. U (2021). Compression Techniques for Different Applications of Real-Time Communication, *LGU R.R.J. Computer Science IT*, 2(1), 7-19.
- Setyaningsih, E., Harjoko, A. (2017). Survey of Hybrid Image Compression Techniques.

- International Journal of Electrical and Computer Engineering*, 7(4), 2206-2214.
- Okasa, A. J., Ngantcha, J. P., Allogho, G., Joseph, A., Okassa, O., & Ele, P. (2020). Compression of Vibration Data by the Walsh-Hadamard Transform. *Journal of Engineering and Applied Sciences*, 15(10), 2256-2260.
- Zhou, J., & Kwan, C. (2018). A Hybrid Approach for Wind Tunnel Data Compression. *Data Compression Conference*.
- Sun, J., Yan, T., Sun, H., Lin, H., & Sun, G. (2021). Lossy Compression of Communication Traces Using Recurrent Neural Networks. *IEEE Transactions on Parallel and Distributed Systems*, 33(11), 3106-3116.
- Marin, O., Schanen, M., & Fischer, P.F. (2018). Lossy Data Compression Effects on Wall-bounded Turbulence: Bounds on Data Reduction. *Flow Turbulence and Combustion*, 101, 365-387.
- Hubbe, N., Wegener, A., Kunkel, J. M., Ling, Y., & Ludwig, T. (2013). Evaluating Lossy Compression on Climate Data. *International Supercomputing Conference*, 7905, 343-356
- Liang, X., Zhao, K., Di, S., Li, S., Underwood, R., Gok, A. M., Tian, J., Deng, J., Calhoun, J. C., Tao, D., Chen, Z., & Cappello, F. (2022). SZ3: A Modular Framework for Composing Prediction-Based Error-Bounded Lossy Compressors. *IEEE Transactions on Big Data*, 9(2), 485-498.
- Nakano, T., UoZomi, S., Johansson, R., Ohnishi, K. (2015). A Quantization Method for Haptic Data Lossy Compression. *IEEE International Conference on Mechatronics*, 126-131.
- Naveenkumar, R., Jagadale, B.N., & Sandeepa, K.S. (2016). Performance Evolution of Image Compression in Walsh Wavelet transform using Wavelet Threshold. *Interantional Conference on Electronics and communication systems*, 3, 500-503.