

PENERAPAN ARSITEKTUR JST DALAM DEEP LEARNING UNTUK MENINGKATKAN AKURASI KLASIFIKASI GAMBAR DENGAN AUTOENCODER

¹ Citra Hudaya, ² Ardiansyah Gunawan, ³ Bintar Wijaya Tri
^{1,2,3} Prodi D4 Teknik Elkasista, Politeknik Angkatan Darat, Batu

¹ citrahudaya2117@gmail.com, ² ardiansyahgunawan1995@gmail.com, ³ bintarwijaya030796@gmail.com

Abstract - The development of artificial intelligence technology, particularly deep learning, has made significant contributions to digital image processing across various fields such as medicine, security, and manufacturing industries. This study aims to implement the autoencoder method within an Artificial Neural Network (ANN) architecture to optimally enhance image classification accuracy. The autoencoder is employed as an unsupervised learning technique to extract essential and relevant features from input images before passing them to the classification layer. The training process was carried out using a carefully curated image dataset, and the model was evaluated to measure classification performance based on accuracy, precision, and recall. The experimental results show that integrating an autoencoder into the ANN architecture can improve feature extraction efficiency, reduce noise, and deliver more accurate and consistent classification results compared to conventional approaches. This research demonstrates that the autoencoder can serve as a vital component in modern deep learning-based classification systems.

Keywords — *Deep Learning, Neural Network, Autoencoder, Image Classification, Feature Extraction.*

Abstrak—Perkembangan teknologi kecerdasan buatan pada era ini, khususnya deep learning, telah memberikan berbagai kontribusi yang sangat signifikan di dalam pengolahan citra digital di berbagai bidang seperti medis, keamanan, dan industry manufaktur. Penelitian ini dilakukan bertujuan untuk menerapkan metode autoencoder di dalam arsitektur Jaringan Syaraf Tiruan (JST) guna meningkatkan akurasi klasifikasi gambar secara lebih optimal. Autoencoder juga digunakan sebagai teknik pembelajaran yang tidak terawasi untuk mengekstraksi sebuah fitur penting dan relevan dari citra masukan sebelum kemudian diteruskan ke lapisan klasifikasi. Proses pelatihan ini dilakukan dengan menggunakan dataset gambar yang telah dikurasi secara ketat, dan model diuji untuk mengukur performa klasifikasi berdasarkan akurasi, presisi, dan recall. Hasil pengujian tersebut telah menunjukkan bahwa integrasi autoencoder dalam arsitektur JST mampu meningkatkan efisiensi ekstraksi fitur, mengurangi noise, serta memberikan hasil klasifikasi yang lebih akurat dan konsisten dibandingkan dengan pendekatan konvensional. Penelitian ini membuktikan bahwa autoencoder dapat menjadi komponen penting dalam sistem klasifikasi berbasis deep learning modern.

Kata Kunci—*Deep Learning, Jaringan Syaraf Tiruan, Autoencoder, Klasifikasi Gambar, Ekstraksi Fitur.*

PENDAHULUAN

Autoencoder adalah salah satu jenis arsitektur deep learning yang bertujuan untuk mempelajari representasi data dengan cara merekonstruksi input asli secara efisien. Autoencoder terdiri dari dua komponen utama, yaitu encoder dan decoder. Encoder bertugas mengubah input data menjadi representasi laten (code) yang memiliki dimensi lebih kecil dibandingkan data aslinya[1]. Sebaliknya, decoder berfungsi untuk merekonstruksi kembali data asli dari representasi laten tersebut. Proses ini memungkinkan autoencoder mengidentifikasi dan mempertahankan fitur-fitur penting dari data, sekaligus mengabaikan informasi yang kurang relevan atau redundan[2].

Autoencoder juga sebagai salah satu kelas yang khusus dari jaringan saraf tiruan, yang dimana berperan sangat penting dalam kemajuan *deep representation learning*. Jaringan ini dirancang guna mengompresi data input kemudian representasi yang lebih ringkas dan selanjutnya merekonstruksi lagi data masukan tersebut dengan meminimalisir kemungkinan kehilangannya informasi. Pada proses ganda tersebut sangat membantu dalam mengurangi dimensi data dan juga sekaligus mempelajari berbagai fitur penting tanpa membutuhkan asistensi manusia yang ekstensif[10], [11]

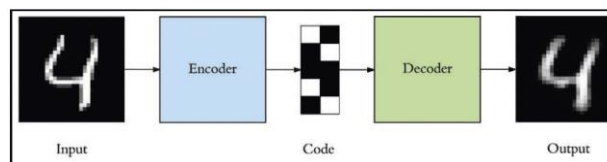
Autoencoder diperlukan karena kemampuannya dalam menyederhanakan representasi data yang kompleks. Dengan mereduksi dimensi data, autoencoder tidak hanya membantu dalam pengolahan data yang lebih efisien, tetapi juga meningkatkan kinerja model pada tugas-tugas tertentu seperti klasifikasi, deteksi anomali, serta visualisasi data dalam ruang dimensi rendah. Selain itu, autoencoder dapat dimanfaatkan untuk memperbaiki kualitas data dengan cara mengurangi noise atau gangguan yang terdapat dalam dataset[3].

Dalam mendesain autoencoder, terdapat beberapa hal penting yang harus diperhatikan, seperti pemilihan arsitektur jaringan (jumlah layer dan unit neuron), fungsi aktivasi, metode optimasi, serta ukuran representasi laten. Desain yang tepat akan memastikan autoencoder mampu menangkap fitur esensial dari data input secara optimal. Penentuan representasi laten yang terlalu kecil dapat menyebabkan hilangnya informasi penting, sementara representasi laten yang terlalu besar akan mengurangi efektivitas autoencoder dalam menangkap fitur esensial[2], [3].

Pada bab ini, mahasiswa akan mempelajari dan menerapkan teknik rekonstruksi gambar dengan metode autoenkoder. Selain itu, salah satu pengembangan metode autoenkoder yang sangat populer, yaitu denoising autoencoder juga akan dipelajari. Denoising autoenkoder adalah metode untuk merekonstruksi data asli yang telah mengalami gangguan atau noise. Fungsi utama denoising autoencoder adalah meningkatkan ketahanan model terhadap gangguan dalam data, serta memperbaiki kualitas data dengan menghilangkan noise[4], [5]. Metode ini sangat bermanfaat dalam aplikasi seperti pra-pemrosesan data, restorasi citra, dan pengenalan pola pada data yang terkontaminasi.

Secara historis, autoencoder ini memiliki peran besar dalam mendukung perkembangan unsupervised learning dan semi-supervised learning, terutama saat data yang berlabel terbatas ataupun mahal untuk didapatkan. Autoencoder juga sering digunakan pada pendeteksian anomaly, pemrosesan citra, serta generative modeling, yang kemudian menegaskan luasnya aplikasi autoencoder dalam machine learning modern[10], [12]

Autoencoder adalah sebuah arsitektur jaringan saraf tiruan yang dirancang terutama untuk reduksi dimensi, ekstraksi fitur, dan pembelajaran representasi data[6]. Autoencoder termasuk dalam paradigma pembelajaran tak terbimbing (unsupervised learning), artinya jaringan ini belajar dari data yang tidak memiliki label. Tujuan utama autoencoder adalah untuk mengompresi data input ke dalam representasi berdimensi lebih rendah (encoding), kemudian merekonstruksi kembali data tersebut dari representasi yang telah dikompresi (decoding)[1]. Gambar 1 menunjukkan contoh proses pengolahan sebuah gambar dengan arsitektur autoencoder.



Gambar 1. Proses pengolahan gambar dengan autoencoder. **Encoder** berfungsi untuk merubah input menjadi representasi laten (code). **Decoder** berfungsi untuk merubah code menjadi bentuk yang sesuai dengan input.

Komponen dalam autoencoder ada tiga komponen utama. Yang pertama, Encoder, terdiri dari beberapa lapisan jaringan saraf yang secara bertahap mengompresi data berdimensi tinggi ke representasi berdimensi lebih rendah (ruang laten)[7]. Setiap lapisan menerapkan transformasi linear yang diikuti oleh fungsi aktivasi non-linear seperti ReLU, sigmoid, atau tanh, memungkinkan ekstraksi fitur yang semakin abstrak dan ringkas dari data input. Encoder mengurangi redundansi dengan membuang fitur yang kurang penting atau tidak informatif sehingga hanya fitur-fitur penting yang dipertahankan[7].

Kedua, Representasi Laten(code), Merupakan bagian inti dan paling kritis dari autoencoder, terletak di antara encoder dan decoder. Umumnya memiliki jumlah neuron yang jauh lebih sedikit dibandingkan dengan lapisan input, sehingga memaksa jaringan untuk mengidentifikasi dan menyimpan karakteristik data yang paling penting[1]. Representasi pada bottleneck ini menghasilkan fitur-fitur abstrak yang ringkas dan efisien dari data asli.

Ketiga, Decoder, Memiliki struktur yang merupakan cerminan dari encoder, bertugas merekonstruksi data asli dari representasi laten. Menggunakan arsitektur jaringan yang mirip dengan encoder, biasanya simetris, untuk memastikan

rekonstruksi data asli yang akurat. Decoder bertujuan untuk meminimalkan kesalahan rekonstruksi (reconstruction error) sehingga mampu menghasilkan data yang semirip mungkin dengan input asli[1].

METODE PENELITIAN

Pada praktikum ini, kita akan mempelajari bagaimana kita dapat mengolah data berupa gambar dengan menggunakan dua arsitektur Jaringan Syaraf Tiruan yang berbeda, yaitu MLP dan CNN[8]. Dua dataset yang berbeda akan digunakan dalam praktikum ini. Pertama adalah MNIST dan yang kedua adalah CIFAR-10.

MNIST merupakan dataset yang berisikan ribuan gambar hitam putih dari angka 0-9 yang ditulis tangan. Setiap gambar memiliki ukuran 28x28 piksel. Sedangkan CIFAR-10 berisikan ribuan gambar berwarna yang diklasifikasikan ke dalam 10 kategori (pesawat, mobil, burung, kucing, rusa, anjing, katak, kuda, kapal, dan truk). Setiap gambar memiliki ukuran 32x32 piksel[9].

Struktur Notebook. Pertama berisi model yang digunakan untuk klasifikasi gambar angka pada dataset MNIST dengan metode MLP dan CNN. MNIST adalah dataset yang berisikan ribuan gambar angka 0-9 dan berwarna hitam putih. Setiap gambar memiliki ukuran 28x28 piksel dan satu channel. Kedua berisi model yang digunakan untuk melakukan proses denoising atau pembersihan gambar angka pada dataset MNIST metode CNN. Gambar kotor dibuat dengan cara menambahkan noise atau gangguan yang berasal dari Gaussian noise pada gambar asli.

Memahami parameter yang dapat diubah, dalam kedua notebook ini, terdapat beberapa parameter yang dapat diubah untuk meningkatkan performa model. Beberapa parameter utama yang akan kita eksplorasi meliputi, ukuran dari representasi laten (code) yang mengatur seberapa besar kompresi data yang kita inginkan, dan factor noise untuk menentukan seberapa besar gaussian noise akan ditambahkan pada gambar asli.

Eksperimen dan evaluasi, setelah mengubah parameter tertentu, kita akan melatih ulang model dan menganalisis hasilnya menggunakan metrik evaluasi yang sesuai. Tujuan utama dari eksperimen ini adalah mengetahui efek dari ukuran representasi laten (code) serta noise factor dalam proses rekonstruksi dan pembersihan gambar.

Tabel 1. Data ukuran laten

No	Ukuran Laten	MSE
1	4	0,0177
2	8	0,0105
3	16	0,0082
4	32	0,0064

Membandingkan hasil, setelah melakukan beberapa kali eksperimen dengan parameter representasi laten (code) dan noise factor yang berbeda-beda, kita akan membandingkan kualitas hasil luaran decoder.

Tabel 2. Data ukuran laten

No	Ukuran Laten	MSE
1	2	0,0388
2	4	0,0370
3	8	0,0318
4	32	0,0254

Analisa ini akan membantu memahami bagaimana parameter tertentu mempengaruhi kinerja model serta bagaimana cara meningkatkan kualitas dari proses rekonstruksi gambar.

HASIL DAN PEMBAHASAN

A. Hasil

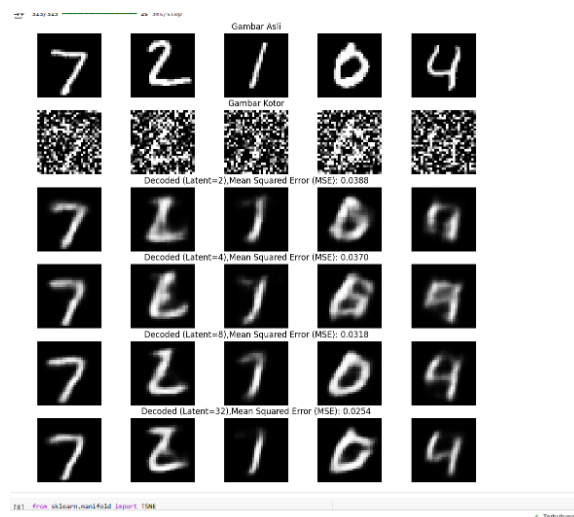
Membandingkan hasil, setelah melakukan beberapa kali eksperimen dengan parameter representasi laten (code) dan noise factor yang berbeda-beda, kita akan membandingkan kualitas hasil luaran decoder. Analisa ini akan membantu memahami bagaimana parameter tertentu mempengaruhi kinerja model serta bagaimana cara meningkatkan kualitas dari proses rekonstruksi gambar.

Selanjutnya menjalankan program dengan menggunakan 2 data, yaitu `cnn_autoencoder_latent.ipynb` dan `cnn_denoising_autoencoder.ipynb`. Melakukan percobaan dengan menjalankan notebook pertama (`cnn_autoencoder_latent.ipynb`). Setelah melakukan percobaan, catat hasil Mean Squared Errors (MSE) dari hasil rekonstruksi gambar dan jelaskan mengapa hasil seperti itu bisa didapatkan. Hasil analisis pertama yaitu semakin besar ukuran laten, semakin banyak informasi yang dapat dipertahankan selama rekonstruksi gambar, kemudian MSE menurun karena dengan lebih banyak parameter, autoencoder dapat menyimpan lebih banyak detail dari gambar asli, sehingga hasil rekonstruksi lebih mendekati aslinya, namun ukuran laten yang terlalu besar dapat menyebabkan overfitting, sementara ukuran laten yang terlalu kecil dapat menyebabkan hilangnya detail penting.

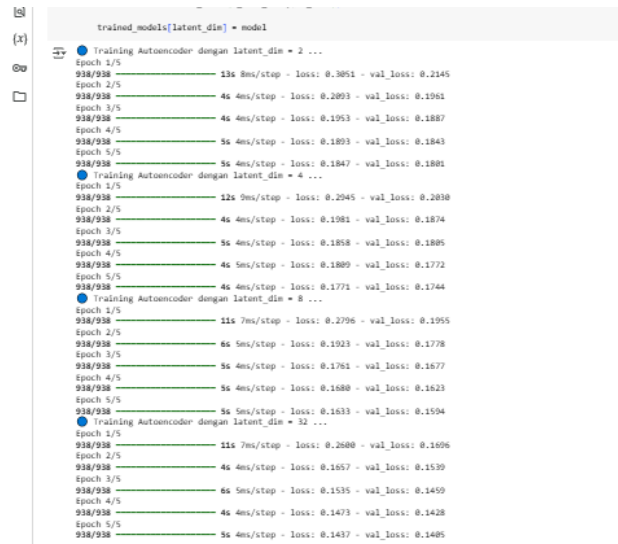
Kesimpulan dari percobaan pertama adalah ukuran laten berpengaruh signifikan terhadap kualitas rekonstruksi. Semakin besar ukuran laten, semakin kecil error rekonstruksi, tetapi perlu keseimbangan agar model tetap efisien dan tidak overfitting. Pada percobaan yang kami laksanakan dapat kami simpulkan semakin besar nilai laten yang diberikan maka akan mempengaruhi nilai Errors (MSE) semakin kecil.

Kemudian melakukan percobaan dengan menjalankan notebook kedua (`cnn_denoising_autoencoder.ipynb`) kemudian catat nilai MSE dengan data uji. Analisa dari percobaan tersebut yaitu MSE menurun seiring bertambahnya ukuran laten, karena model dapat menyimpan lebih banyak informasi untuk merekonstruksi gambar yang lebih bersih dari noise, ukuran laten yang kecil menyebabkan rekonstruksi kurang optimal, karena terlalu sedikit informasi yang disimpan, dan ukuran laten yang lebih besar memungkinkan model menangkap lebih banyak fitur dari gambar asli, sehingga mampu menghilangkan noise lebih efektif.

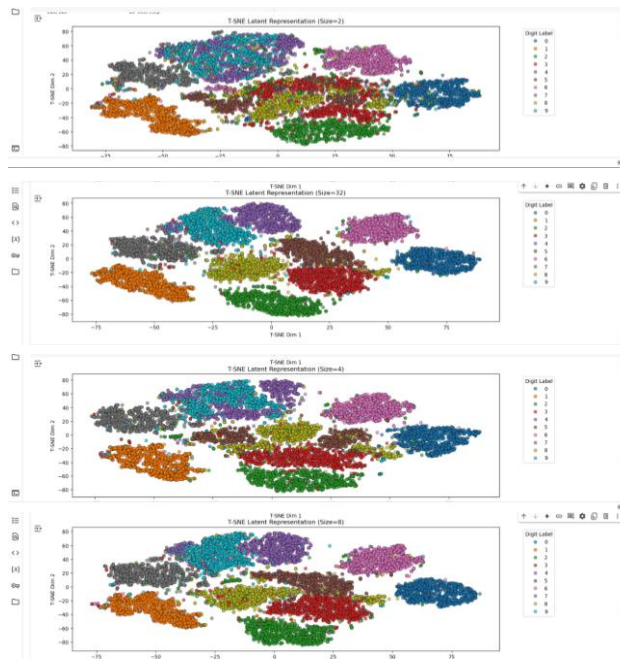
Kesimpulan dari percobaan kedua adalah Semakin besar ukuran laten, semakin baik autoencoder dalam menghilangkan noise dari gambar, dengan error yang lebih kecil. Namun, perlu keseimbangan agar model tetap efisien tanpa overfitting. Tampilan dari plot T-SNE mulai dari model yang terbaik hingga terburuk dari hasil eksperimen pada gambar 2, 3, dan 4.



Gambar 2. Hasil eksperimen dari model terbaik hingga terburuk



Gambar 3. Tampilan Plot T-SNE



Gambar 4. Tampilan Plot T-SNE

B. Pembahasan

Penelitian ini bertujuan untuk memahami bagaimana autoencoder, yang merupakan salah satu bentuk jaringan saraf tiruan (JST), dapat digunakan untuk merekonstruksi citra serta bagaimana pengaruh variasi parameter, seperti ukuran representasi laten dan faktor noise, terhadap performa model.

Autoencoder terdiri dari dua komponen utama, yaitu encoder dan decoder. Encoder berfungsi mengubah data input citra menjadi representasi berdimensi lebih rendah (dikenal sebagai kode laten), sedangkan decoder bertugas untuk merekonstruksi citra dari kode laten tersebut. Performa rekonstruksi diukur menggunakan Mean Squared Error (MSE), yang mencerminkan selisih antara citra asli dengan hasil rekonstruksi.

Pada eksperimen pertama, dilakukan pengujian dengan variasi ukuran representasi laten tanpa penambahan noise. Hasilnya menunjukkan bahwa semakin besar ukuran laten yang digunakan, semakin rendah nilai MSE yang dihasilkan. Hal ini mengindikasikan bahwa dimensi laten yang lebih besar mampu menyimpan informasi lebih banyak dari citra asli, sehingga menghasilkan rekonstruksi yang lebih akurat. Sebagai contoh, pada ukuran laten 4 diperoleh MSE sebesar 0,0177, sedangkan pada ukuran laten 32 nilai MSE menurun menjadi 0,0064. Ini menegaskan bahwa peningkatan kapasitas representasi memungkinkan model untuk menangkap fitur yang lebih kompleks dari citra input.

Selanjutnya, pada eksperimen kedua, dilakukan penambahan noise terhadap data input untuk menguji ketahanan (robustness) model. Penambahan noise menyebabkan peningkatan nilai MSE pada semua ukuran laten, namun tren penurunan MSE terhadap peningkatan ukuran laten tetap konsisten. Sebagai contoh, pada ukuran laten 2 dengan noise, MSE mencapai 0,0388, dan turun menjadi 0,0254 pada ukuran laten 32. Hal ini menunjukkan bahwa meskipun terdapat gangguan pada input, autoencoder tetap mampu merekonstruksi citra dengan cukup baik, khususnya jika ukuran laten mencukupi.

Dari kedua eksperimen tersebut, dapat disimpulkan bahwa:

1. Ukuran representasi laten memiliki pengaruh signifikan terhadap kualitas hasil rekonstruksi. Semakin besar ukuran laten, semakin baik kualitas rekonstruksi, ditunjukkan oleh penurunan nilai MSE.
2. Penambahan noise menurunkan kualitas rekonstruksi, tetapi model tetap menunjukkan performa yang stabil saat ukuran laten meningkat. Ini menunjukkan potensi autoencoder sebagai metode praproses yang robust terhadap gangguan.
3. Trade-off antara kompleksitas model dan akurasi perlu diperhatikan. Meskipun ukuran laten besar memberikan hasil rekonstruksi lebih baik, hal ini juga meningkatkan kompleksitas komputasi.

Secara keseluruhan, eksperimen ini menunjukkan bahwa pemahaman terhadap parameter-parameter dalam arsitektur autoencoder sangat penting untuk menghasilkan model yang efektif dan efisien dalam tugas rekonstruksi citra.

Pembahasan hasil eksperimen cnn autoencoder. Pada percobaan ini, dilakukan implementasi dan pengujian model Convolutional Autoencoder dengan berbagai variasi ukuran representasi laten (latent size) guna mengevaluasi pengaruhnya terhadap kualitas rekonstruksi gambar. Berdasarkan hasil pengamatan terhadap nilai Mean Squared Error (MSE), diperoleh bahwa semakin besar ukuran laten yang digunakan, maka nilai MSE cenderung semakin kecil. Hal ini menunjukkan bahwa rekonstruksi gambar menjadi semakin mendekati gambar aslinya.

Secara teknis, ukuran laten yang lebih besar memberikan kapasitas representasi yang lebih tinggi bagi autoencoder untuk menyimpan informasi penting dari citra input. Akibatnya, model dapat mempertahankan lebih banyak detail spasial dan tekstur gambar selama proses encoding dan decoding, sehingga menghasilkan gambar hasil rekonstruksi yang lebih akurat. Sebaliknya, ketika ukuran laten terlalu kecil, representasi yang dihasilkan menjadi terlalu sederhana dan tidak mampu merepresentasikan kompleksitas fitur gambar secara menyeluruh. Kondisi ini menyebabkan hilangnya detail-detail penting, yang berdampak pada peningkatan nilai MSE sebagai indikator bahwa hasil rekonstruksi menyimpang dari gambar asli.

Namun demikian, perlu diperhatikan bahwa peningkatan ukuran laten secara berlebihan tidak selalu menguntungkan. Meskipun error menurun, penggunaan ukuran laten yang sangat besar dapat menyebabkan overfitting, di mana model terlalu menyesuaikan diri pada data pelatihan dan gagal melakukan generalisasi pada data uji. Selain itu, ukuran laten yang besar juga meningkatkan kebutuhan komputasi dan waktu pelatihan.

KESIMPULAN

Kesimpulan pembahasan dari eksperimen ini dapat disimpulkan bahwa pemilihan ukuran laten yang tepat merupakan faktor penting dalam desain arsitektur autoencoder. Ukuran laten yang optimal akan menghasilkan keseimbangan antara akurasi rekonstruksi (nilai MSE yang rendah) dan efisiensi model, sehingga kinerja autoencoder dapat dimaksimalkan tanpa menimbulkan overfitting atau beban komputasi yang berlebihan.

Pembahasan hasil eksperimen Denoising Autoencoder. Pada eksperimen tersebut dilakukan implementasi dan pengujian terhadap CNN-based Denoising Autoencoder, dengan tujuan untuk melihat bagaimana variasi ukuran laten (latent space) mempengaruhi kemampuan model dalam merekonstruksi gambar yang telah terkontaminasi noise. Nilai Mean Squared Error (MSE) digunakan sebagai metrik evaluasi terhadap kualitas hasil rekonstruksi gambar dari data uji yang mengandung noise.

Hasil pengamatan menunjukkan bahwa semakin besar ukuran laten yang digunakan, semakin rendah nilai MSE yang diperoleh. Hal ini menandakan bahwa model dengan ukuran laten yang lebih besar mampu menyimpan informasi yang lebih kompleks dan akurat dari gambar asli, meskipun telah terdistorsi oleh noise. Autoencoder tersebut lebih efektif dalam mengenali pola-pola utama dari citra dan mengabaikan informasi yang bersifat acak atau tidak penting (noise), sehingga hasil rekonstruksinya lebih bersih dan mendekati gambar asli.

Sebaliknya, pada ukuran laten yang kecil, model memiliki keterbatasan dalam menangkap fitur penting dari citra, yang menyebabkan hasil rekonstruksi menjadi kabur atau kehilangan detail penting. Hal ini dikarenakan representasi yang dihasilkan terlalu sederhana dan tidak mampu membedakan antara fitur penting dan noise secara efektif.

Namun, seperti halnya autoencoder biasa, penggunaan ukuran laten yang terlalu besar juga dapat mengakibatkan overfitting, di mana model terlalu menghafal data pelatihan dan tidak mampu bekerja dengan baik pada data baru. Oleh karena itu, diperlukan kompromi antara ukuran laten, performa, dan efisiensi model.

Keimpulan pembahasan dari eksperimen denoising autoencoder dapat disimpulkan bahwa ukuran laten sangat berpengaruh terhadap kemampuan autoencoder dalam melakukan denoising. Ukuran laten yang lebih besar secara umum memberikan hasil rekonstruksi yang lebih bersih dan akurat, dengan nilai MSE yang lebih rendah. Namun, perlu dilakukan pemilihan ukuran laten yang optimal agar model tidak terlalu kompleks dan tetap mampu melakukan generalisasi dengan baik terhadap data uji. Untuk memperkuat analisis, dilakukan visualisasi hasil representasi laten menggunakan metode t-distributed Stochastic Neighbor Embedding (t-SNE) pada dua model dengan performa terbaik dan terburuk berdasarkan nilai MSE. Model Terbaik (ukuran laten besar): Plot t-SNE menunjukkan klusterisasi yang jelas antar kelas gambar, yang mengindikasikan bahwa representasi laten yang dihasilkan memiliki pemisahan fitur yang baik dan padat, sehingga mempermudah proses rekonstruksi dan penghilangan noise. Sedangkan model terburuk (ukuran laten kecil): Plot t-SNE tampak lebih menyebar dan tidak terstruktur, menandakan bahwa representasi laten tidak cukup informatif atau terlalu tumpang tindih antar kelas, sehingga kualitas rekonstruksi menurun dan noise sulit dihilangkan secara efektif.

DAFTAR PUSTAKA

- [1] K. Aitken, "Understanding How Encoder-Decoder Architectures Attend."
- [2] E. Mushtaq, A. Zameer, M. Umer, and A. A. Abbasi, "A two-stage intrusion detection system with auto-encoder and LSTMs," *Appl Soft Comput*, vol. 121, p. 108768, May 2022, doi: 10.1016/J.ASOC.2022.108768.
- [3] P. Li, Y. Pei, and J. Li, "A comprehensive survey on design and application of autoencoder in deep learning," *Appl Soft Comput*, vol. 138, p. 110176, May 2023, doi: 10.1016/J.ASOC.2023.110176.
- [4] A. E. Ilesanmi and T. O. Ilesanmi, "Methods for image denoising using convolutional neural network: a review," *Complex and Intelligent Systems*, vol. 7, no. 5, pp. 2179–2198, Oct. 2021, doi: 10.1007/s40747-021-00428-4.
- [5] M. Elad, B. Kawar, and G. Vaksman, "Image Denoising: The Deep Learning Revolution and Beyond—A Survey Paper," *SIAM J Imaging Sci*, vol. 16, no. 3, pp. 1594–1654, 2023, doi: 10.1137/23M1545859.
- [6] R. Hammouche, A. Attia, S. Akhrouf, and Z. Akhtar, "Gabor filter bank with deep autoencoder based face recognition system," *Expert Syst Appl*, vol. 197, p. 116743, Jul. 2022, doi: 10.1016/J.ESWA.2022.116743.
- [7] V. Badrinarayanan, A. Kendall, and R. Cipolla, "SegNet: A Deep Convolutional Encoder-Decoder Architecture for Image Segmentation," *IEEE Trans Pattern Anal Mach Intell*, vol. 39, no. 12, pp. 2481–2495, Dec. 2017, doi: 10.1109/TPAMI.2016.2644615.

- [8] M. D. Wuryandari and I. Afrianto, “PERBANDINGAN METODE JARINGAN SYARAF TIRUAN BACKPROPAGATION DAN LEARNING VECTOR QUANTIZATION PADA PENGENALAN WAJAH,” 2012.
- [9] S. Ghosh, N. Das, I. Das, and U. Maulik, “Understanding Deep Learning Techniques for Image Segmentation,” *ACM Comput. Surv.*, vol. 52, no. 4, Aug. 2019, doi: 10.1145/3329784.
- [10] I. D. Mienye and T. G. Swart, “Deep Autoencoder Neural Networks: A Comprehensive Review and New Perspectives,” 2025, *Springer Science and Business Media B.V.* doi: 10.1007/s11831-025-10260-5.
- [11] Y. FAROOQ and S. SAVAŞ, “Noise Removal from the Image Using Convolutional Neural Networks-Based Denoising Auto Encoder,” *Journal of Emerging Computer Technologies*, vol. 3, no. 1, pp. 21–28, Mar. 2024, doi: 10.57020/ject.1390428.
- [12] A. Mohamed, A. S. Tag Eldien, M. M. Fouda, and R. S. Saad, “LSTM-Autoencoder Deep Learning Technique for PAPR Reduction in Visible Light Communication,” *IEEE Access*, vol. 10, pp. 113028–113034, 2022, doi: 10.1109/ACCESS.2022.3216574.