

Perbandingan Optimizer dan RMSPROP Menggunakan Arsitektur ANN untuk Klasifikasi Pengenalan Tulisan Tangan

M. Nejatullah Sidqi^{1*}, Rio²

¹Program Studi Teknologi Informasi, Univesitas PGRI Silampari

²Program Studi Sistem Informasi, Univesitas PGRI Silampari

*neja.sidqi97@gmail.com

Abstrak

Penelitian ini membahas perbandingan kinerja dua algoritma optimisasi, Adam dan RMSProp, dalam melatih Artificial Neural Network (ANN) untuk tugas klasifikasi pengenalan tulisan tangan menggunakan dataset MNIST. Dataset MNIST berisi 70.000 citra digit tulisan tangan (0–9) berukuran 28×28 piksel. Tahapan penelitian meliputi input dataset, preprocessing (normalisasi piksel, pembagian data latih dan uji, serta konversi label menjadi one-hot encoding), inisialisasi bobot, dan pelatihan model ANN dengan variasi optimizer Adam dan RMSProp. Kinerja model diuji menggunakan akurasi, presisi, recall, dan F1-score. Hasil pengujian dilakukan menggunakan aplikasi google colab, skema pengujian dilakukan secara terstruktur dengan iterasi epoch sebanyak 50, 100, 150, 200 dan 250 pada optimizer adam dan RMSProp. Pengujian menunjukkan bahwa optimizer Adam umumnya memberikan akurasi lebih tinggi dan konvergensi lebih cepat dibandingkan RMSProp pada dataset MNIST, berkat kombinasi momentum dan adaptasi laju pembelajaran yang lebih stabil. Namun, RMSProp tetap unggul pada kestabilan pembaruan bobot di beberapa kasus dan cenderung lebih konsisten pada dataset dengan distribusi gradien yang fluktuatif.

Kata kunci : Ann, Adam, Dataset, MNIST, RMSProp

Abstract

This study discusses the performance comparison of two optimization algorithms, Adam and RMS Prop, in training an Artificial Neural Network (ANN) for handwriting recognition classification tasks using the MOIST dataset. The MOIST dataset contains 70,000 handwritten digit images (0–9) measuring 28×28 pixels. The research stages include dataset input, preprocessing (pixel normalization, training and testing data division, and label conversion to one-hot encoding), weight initialization, and ANN model training with variations of Adam and RMS Prop optimizers. Model performance was tested using accuracy, precision, recall, and F1-score. The test results were carried out using the Google Cola application; the testing scheme was carried out in a structured manner with epoch iterations of 50, 100, 150, 200 and 250 on the Adam and RMS Prop optimizers. The test shows that the Adam optimizer generally provides higher accuracy and faster convergence than RMS Prop on the MOIST dataset, thanks to a combination of momentum and a more stable learning rate adaptation. However, RMS Prop still excels in weight update stability in some cases and tends to be more consistent on datasets with fluctuating gradient distributions.

Keywords : Ann, Adam, Dataset, MNIST, RMSProp.

1. Pendahuluan

Kemajuan teknologi kecerdasan buatan, khususnya diranah Deep Learning, telah membawa kemajuan signifikan dalam pengolahan citra, termasuk pada aplikasi

pengenalan tulisan tangan (handwritten recognition)^[1]. Salah satu metode yang umum digunakan dalam tugas klasifikasi seperti ini adalah Artificial Neural Network (ANN), yang mampu mempelajari pola kompleks dari data

input tanpa aturan yang eksplisit. Salah satu pendekatan yang populer dalam bidang klasifikasi citra, termasuk pengenalan karakter tulisan tangan, adalah Artificial Neural Network (ANN)^[2] ANN mampu mengenali pola-pola kompleks dari data input tanpa perlu eksplisit memberikan aturan atau fitur secara manual. Dalam konteks pengenalan tulisan tangan, ANN sangat efektif dalam mengenali bentuk huruf dan angka yang bervariasi dari tulisan tangan manusia yang tidak konsisten, untuk mendapatkan performa optimal dari ANN, dibutuhkan algoritma optimasi yang efektif dalam mempercepat proses pelatihan dan meningkatkan akurasi model. Proses pelatihan yang dilakukan menggunakan sebuah data set MNIST^[3].

Artificial Neural Network, yang terdiri dari proses pemodelan dari input dataset. Algoritma ANN terdapat proses dimana untuk menentukan hasil klasifikasi terdapat fungsi aktivasi^[4]. Fungsi aktivasi yang digunakan pada algoritma ANN saat ini dengan fungsi aktivasi Adam dan RMSProp yang berfungsi akan mencari terbaik dari jenis fungsi aktivasi. Namun, keberhasilan pelatihan model ANN sangat bergantung pada algoritma optimasi yang digunakan. Optimizer berperan dalam memperbarui bobot jaringan selama proses pelatihan dengan tujuan meminimalkan fungsi kerugian (loss function)^[5].

Fungsi aktivasi yang tersedia, Adam dan RMSProp merupakan dua algoritma yang paling

banyak digunakan karena kemampuan adaptifnya dalam menyesuaikan learning rate selama proses pelatihan^[6].

2. Tinjauan Pustaka

2.1. Penelitian Terkait

Adapun penelitian sebelumnya yang relevan dengan penelitian ini antara lain sebagai berikut :

1. Penelitian dilakukan oleh Mochammad Toyib, Tegar Decky Kurniawan dan Ibnu Aqil dengan judul “ Penerapan Algoritma CNN untuk mendeteksi tulisan tangan dengan augmentasi data” (2024) menghasilkan teknik bahwa Teknik penambahan data terbukti sangat membantu dalam meningkatkan kinerja model Convolutional Neural Network (CNN) untuk mendeteksi angka Romawi yang ditulis tangan^[7].

2. Penelitian terkait yang dilakukan oleh Laurenza Setiana Riva, Felicia Febriana, Mutihara, dkk dengan judul “ Perbandingan algoritma CNN dan ANN dengan Projection Histogram untuk klasifikasi citra tulisan tangan berupa angka” (2022) menghasilkan Hasil pengujian algoritma CNN pada data validasi menunjukkan akurasi sebesar 0. 9752, dan pada data pengujian mencapai 0. 9689. Sementara itu, algoritma projection histogram+ANN meraih akurasi 0. 8904 untuk data validasi dan 0. 8962 untuk data pengujian. Berdasarkan temuan ini, model yang menggunakan Algoritma CNN terbukti memberikan kinerja yang lebih baik daripada

projection histogram+ANN^[8].

3. Terdapat penelitian lain yang dilakukan oleh Antonius Angga Kurniawan, Rama Dian Syah, dan Rizki Ariyani dengan judul “Klasifikasi Citra Digital Tulisan Tangan Angka Menggunakan Metode Convolutional Neural Network” (2022) menghasilkan Kumpulan data latih, data uji, serta pemeriksaan dan evaluasi hasilnya. Penelitian ini sukses mengidentifikasi tulisan tangan dengan tingkat ketepatan 0,9800 dan nilai kekeliruan 0,0665^[9].

4. Selanjutnya terdapat penelitian yang dilakukan oleh Dewi Aulian tanjung, Tiara Amanda lestari, dkk dengan judul “Implementasi Algoritma Convolutional Neural Network (CNN) untuk Pengenalan Angka Tulisan Tangan yang menghasilkan bahwa penerapan algoritma Convolutional Neural Network (CNN) yang telah terbukti ampuh dalam mengambil fitur visual dengan sendirinya melalui proses konvolusi dan pooling. Dataset MNIST yang berisi 60. 000 data pelatihan serta 10. 000 data pengujian dimanfaatkan dalam studi ini sebagai landasan untuk melatih model. Verifikasi yang dilakukan mengungkap bahwa rancangan model CNN berhasil meraih tingkat keakuratan yang luar biasa, yaitu mencapai 99% dalam mengidentifikasi digit tulisan tangan di dataset MNIST^[10]. Pada penelitian ini akan dilakukan optimasi pada fungsi optimizer algoritma ANN menggunakan optimzer adam dan RMSProp

untuk mencari hasil akurasi terbaik pada algoritma yang digunakan.

2.2. Landasan Teori

1. Klasifikasi

Klasifikasi merupakan proses membagi entitas yang berbeda kedalam beberapa kelas disebut klasifikasi, untuk melatih suatu label yang digunakan sebagai mencocokkan atribut dengan salah satu label kelas [11]. Klasifikasi adalah teknik pengolahan data yang membagi objek menjadi beberapa kelas sesuai dengan jumlah kelas yang diinginkan [12] . Dari beberapa pengertian Klasifikasi proses mengkategorikan data ke dalam kelas atau label tertentu berdasarkan fitur atau karakteristik yang dimilikinya. Dalam konteks machine learning, klasifikasi merupakan tugas supervised learning, di mana sistem belajar dari data berlabel untuk memprediksi label dari data baru yang tidak diketahui.

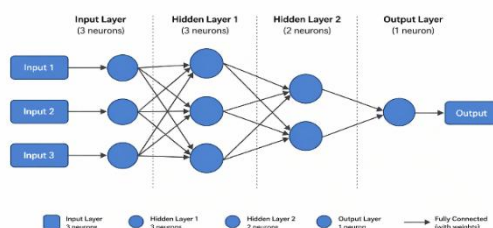
2. Dataset

Dataset adalah sekumpulan data yang terorganisir secara sistematis dalam bentuk tabel, file, atau struktur digital lainnya, yang digunakan untuk berbagai keperluan analisis, pelatihan model, pengujian sistem, atau visualisasi informasi.

3. Artificial Neural Network (ANN)

Algoritma Artificial Neural Network (ANN) adalah suatu metode komputasi yang terinspirasi dari

cara kerja jaringan saraf biologis pada otak manusia dalam memproses informasi. ANN terdiri dari sekumpulan unit pemroses sederhana yang disebut neuron buatan, yang saling terhubung melalui bobot (weights) dan bekerja secara paralel untuk memecahkan suatu permasalahan, seperti klasifikasi, regresi, prediksi, maupun pengenalan pola. Secara umum, ANN berfungsi Kesadaran terhadap sistem pembelajaran kompleks pada otak biologis menjadi inspirasi terciptanya ANN. Kemudian ANN juga memiliki kemampuan untuk mendeteksi interaksi yang mungkin antara variabel prediksi. ANN mampu untuk mendeteksi secara komplit tanpa keraguan meskipun dalam hubungan non linear antara variabel bebas dan terikat [13]. Berikut proses kerja dari algoritma ANN terdapat pada gambar 1.



Gambar 1 Arsitektur ANN

4. Optimizer Algoritma

Optimizer adalah metode atau algoritma yang digunakan dalam proses pelatihan model pembelajaran mesin (machine learning) atau jaringan saraf tiruan (ANN) untuk menyesuaikan bobot (weights) dan bias secara bertahap agar error atau loss menjadi sekecil mungkin. Pada

penelitian ini menggunakan fungsi optimizer Adam dan RMSProp (Root Mean Square Propagation).

a. Optimizer Adam

Optimizer Adam (Adaptive Moment Estimation) adalah algoritma optimisasi yang digunakan untuk memperbarui bobot dan bias pada model pembelajaran mesin, khususnya Artificial Neural Network (ANN), selama proses pelatihan

b. Optimizer RMSProp

Optimizer RMSProp (Root Mean Square Propagation) adalah algoritma optimisasi yang digunakan dalam proses pelatihan jaringan saraf tiruan (Artificial Neural Network / ANN) untuk memperbarui bobot dan bias. RMSProp bekerja dengan cara menyesuaikan learning rate secara adaptif untuk setiap parameter berdasarkan rata-rata eksponensial kuadrat gradien terbaru.

5. Confusion Matrix

Konfusi Matriks adalah tabel yang digunakan untuk mengevaluasi hasil dari kinerja pada model klasifikasi termasuk pada ANN, dengan membandingkan prediksi model terhadap label sebenarnya. Struktur pada confusion matrik terdapat pada gambar 2.

		Nilai Sebenarnya	
		Positive	Negative
Nilai Prediksi	Positive	TP (True Positive)	FP (False Positive)
	Negative	FN (False Negative)	TN (True Negative)

Gambar 2 Confusion Matrix

3. Metode Penelitian

3.1. Lokasi Penelitian

Penelitian ini dilakukan pada Google Colabority (G-Colab) dengan melakukan penarikan dataset dari UCI yang dengan mengambil data mengenai MNIST atau karakter angka tulisan tangan, kemudian akan dilakukan klasifikasi menggunakan algoritma ANN untuk menentukan hasil akurasi terbaik serta mendapatkan model algoritma yang terbaik. Berikut tampilan Google Colabority pada gambar 3.



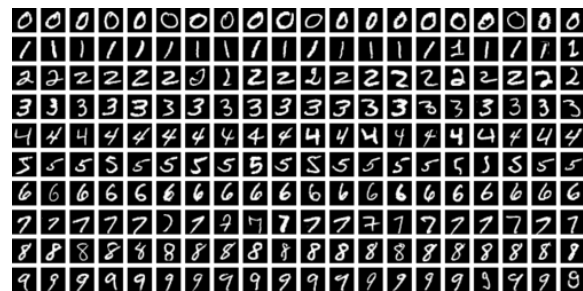
Gambar 3 Google Colabority

3.2. Metode Pengumpulan Data

Adapun data yang diambil pada penelitian ini menggunakan Data Skunder , Data ini biasanya dikumpulkan oleh peneliti lain, lembaga, atau instansi untuk tujuan tertentu, dan kemudian digunakan kembali untuk penelitian baru. Sedangkan pada penelitian ini datanya bersumber dari

<https://archive.ics.uci.edu/dataset/683/mnist+database+of+handwritten+digits>

Berikut contoh dataset MNIST pada gambar 4.

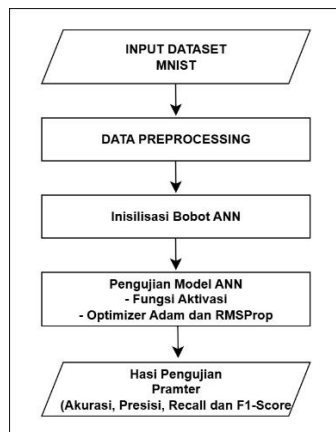


Gambar 4 Dataset MNIST

Dataset ini berisi gambar angka tulisan tangan dari 0 hingga 9, yang telah diproses dalam format seragam sehingga mudah digunakan untuk pelatihan dan pengujian model

3.3 Tahapan Penelitian

Kerangka kerja penelitian yang dilakukan yaitu dengan mengumpulkan data dari Dataset UCI Repository, kemudian dilakukan inisialisasi bobot secara otomatis dari algoritma ANN kemudian dilakukan klasifikasi serta di optimasikan dengan fungsi Optimizer seperti Adam dan RMSProp. Pada saat melakukan klasifikasi, pengujian model di lakukan beberapa iterasi. Iteraasi yang dilakukan setiap optimizer dilakukan sebanyak 5 kali, sehingga mendapatkan hasil terbaik. Setelah itu hasil klasifikasi dari seluruh validasi model, akan dilakukan analisis dari hasil matrix parameter seperti akurasi, presisi, recall serta f1-score. Proses dari tahapan penelitian dapat dilihat pada gambar 5.



Gambar 5 Tahapan Penelitian

Pada tahapan penelitian ini akan dilakukan mulai dari:

1. Input Dataset

MNIST yang di mana dataset terdiri dari gambar grayscale berukuran 28 x 28 piksel kemudian Setiap piksel memiliki nilai intensitas antara 0 sampai 255, di mana 0 berarti hitam (background) dan 255 berarti putih (tulisan).

2. Data Preprocessing

Data preprocessing adalah serangkaian proses atau langkah awal yang dilakukan untuk menyiapkan data mentah agar menjadi lebih bersih.

3. Inisialisasi Bobot

Tahapan ini sangat penting karena bobot awal memengaruhi bagaimana jaringan belajar dan seberapa cepat model dapat mencapai konvergensi (solusi optimal).

4. Fungsi Aktivasi ANN

Fungsi aktivasi bertugas mengubah sinyal input neuron menjadi output yang non-linear, sehingga jaringan bisa belajar pola kompleks.

Beberapa fungsi aktivasi populer Sigmoid dan ReLu

5. Tahapan Pengujian ANN

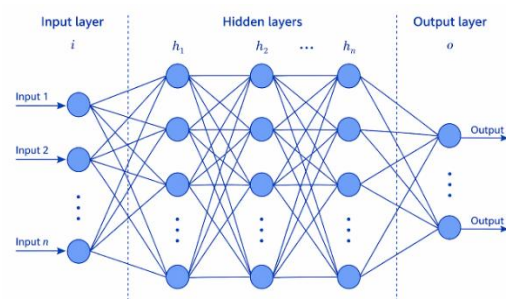
Fungsi aktivasi bertugas mengubah sinyal input neuron menjadi output yang non-linear, sehingga jaringan bisa belajar pola kompleks. Beberapa fungsi aktivasi populer Sigmoid dan ReLu.

6. Hasil Pengujian

Setelah melakukan pengujian model Artificial Neural Network (ANN) pada data uji, beberapa metrik evaluasi yang umum digunakan untuk menilai performa model klasifikasi, kemudian mengukur Accuracy, Precision, Recall dan F1-Score.

3.4 Klasifikasi Menggunakan ANN

Proses klasifikasi dataset MNIST menggunakan algoritma ANN. Berikut adalah arsitektur ANN yang diusulkan pada Gambar 6.



Gambar 6. Arsitektur ANN yang di Usulkan

Jaringan neuron buatan memiliki dari suatu kelompok neuron yang ditata pada 3 lapisan yaitu: lapisan Input berfungsi yang menghubungkan jaringan ke dunia luar.

Neuron ini tidak melakukan apapun pada data, hanya meneruskan data ini ke lapisan berikutnya. Lapisan Tersembunyi (Hidden Layer) Hasil klasifikasi ANN menggunakan ukuran parameter yang akan diuji validitas dengan Confusion Matrix berupa Accuracy, Precision dan Recall.

4. Hasil dan Pembahasan

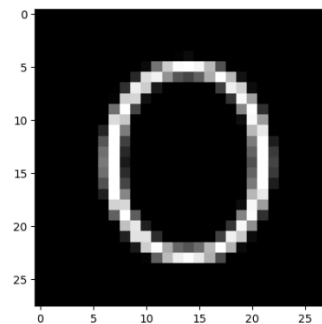
Pada bagian ini membahas hasil penelitian pada klasifikasi tulisan tangan dengan menggunakan dataset MNIST yang dimana hendak dilaksanakan klasifikasi menggunakan algoritma ANN. Pada proses klasifikasi akan melakukan percobaan skenario dengan menggunakan 2 fungsi optimizer yang berbeda. Adapun fungsi optimizer yang dilakukan menggunakan Optimizer Adam dan Optimizer RMSProp. Dari kedua validasi tersebut akan mendapatkan hasil klasifikasi, kemudian dilakukan analisis untuk menentukan model yang terbaik serta performa akurasi tertinggi setiap validasi klasifikasi tersebut

4.1. Hasil Penelitian

1. Dataset

Pada klasifikasi dengan model ANN dilakukan pada dataset MNIST dengan ukuran Dataset terdiri dari 29.900 data (baris) dan 786 variabel (kolom). Dalam penelitian ini, terdapat 10 nilai yang berbeda dalam kolom "labels". Label Unik : Kolom "labels" berisi 10 nilai yang berbeda, yang

menandakan bahwa ada 10 kategori atau kelas yang ingin diklasifikasikan oleh model. Berikut contoh gambar dataset yang masih dalam bentuk gambar atau image terdapat pada Gambar 7



Gambar 7 Contoh Dataset MNIST

Gambar 7 suatu baris data telah diubah bentuknya menjadi gambar 28x28 piksel menggunakan metode reshape dan kemudian menampilkan angka 0 yang ditulis tangan dalam bentuk gambar. Pada pembagian dataset menggunakan split data dengan ukuran 80 : 20 yang dimana, 80% untuk data training dan 20% untuk data testing.

2. Parameter Pengujian

Pengujian yang dilakukan pada model DNN menggunakan beberapa parameter pengukuran, setiap layer akan dilakukan pengujian menggunakan 50 epoch, 100 epoch, 150 epoch, 200 epoch dan 250 epoch. Selain dari pada itu fungsi aktivasi yang akan digunakan pada setiap layer menggunakan fungsi aktivasi ReLU. Sedangkan pada parameter lapisan input pada tahap pertama yang menerima data mentah yang masuk, pada lapisan ini terdapat 784 neuron untuk setiap piksel dalam gambar. Selanjutnya

terdapat lapisan output yang merupakan tahap terakhir yang menghasilkan prediksi dari model, dalam arsitektur ini terdapat 10 neuron. Kemudian terdapat lapisan tersembunyi yang merupakan lapisan yang tidak termasuk lapisan input maupun lapisan output, untuk arsitektur di lapisan ini menggunakan 128 neuron. Sedangkan bobot neuron Nilai bobot yang diberikan pada setiap nilai masukan dari neuron tertentu. Bobot ini disesuaikan selama pelatihan untuk meningkatkan performa model. Pada neuron Unit dalam lapisan jaringan saraf yang bertanggung jawab untuk menerima masukan dan menghasilkan keluaran, dan terdapat fungsi aktivasi yang menentukan keluaran neuron berdasar jumlah tertimbang dari inputnya. untuk arsitektur fungsi aktivasi untuk lapisan tersembunyi menggunakan 'relu', sedangkan untuk lapisan output menggunakan 'softmax'. Arsitektur Model ANN yang di terapkan pada gambar 8

```
# Mendefinisikan Model
model = keras.Sequential([
    keras.layers.Dense(128, input_shape=(784,), activation='relu'),
    keras.layers.Dense(128, input_shape=(784,), activation='relu'),
    keras.layers.Dense(10, activation='softmax')
])

# Mengompilasi Model
model.compile(optimizer='Adam', loss='sparse_categorical_crossentropy', metrics=['accuracy'])
```

Gambar 8 Arsitektur Model

Arsitektur model pada ann yang dibuat menjelaskan bahwa epoch merujuk pada satu iterasi penuh melalui seluruh dataset pelatihan, kemudian loss pelatihan adalah ukuran seberapa baik performa model pada data pelatihan,

terdapat di tabel 1 untuk mendefinisikan arsitektur yang akan di gunakan.

Tabel 1 Paramter ANN

NO	Optimizer	Epoch
1	Adam	50
		100
		150
		200
		250
2	RMSProp	50
		100
		150
		200
		250

Pada tabel 1 parameter ANN menggunakan 2 fungsi optimizer yaitu Adam dan RMSProp dengan tingkatan epoch sebanyak 50 dari masing-masing setiap pengujian.

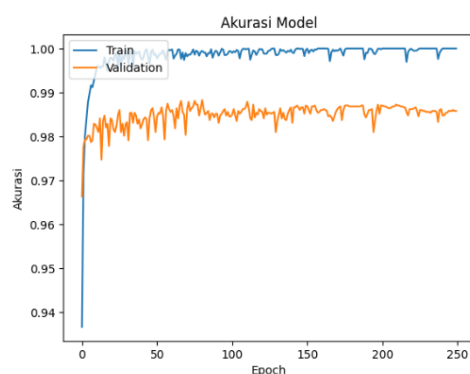
3. Hasil Pengujian Validasi ANN

Hasil validasi klasifikasi ANN dengan inisialisasi optimizer dengan Adam dan RMSProp pada data set MNIST menggunakan beberapa layer dan epoch menghasilkan hasil klasifikasi yang di tampilkan pada Tabel 2.

Tabel 2 Hasil Pengujian Validasi ANN

Optimizer	Epoch	Akurasi
Adam	50	98.31%
	100	98.58%
	150	98.52%
	200	98.39%
	250	98.67%
RMSProp	50	98.42
	100	98.41
	150	98.37
	200	98.46
	250	98.61

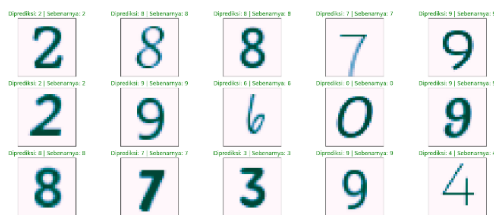
Pengujian model ANN dengan menggunakan fungsi Optimizer adam, menghasilkan nilai terendah dengan akurasi sebesar 98.31% menggunakan epoch sebesar 50. Sedangkan pengujian dengan hasil akurasi tertinggi mendapatkan hasil maksimal dengan akurasi sebesar 98.67% dengan menggunakan parameter 250 epoch. Hal menunjukkan bahwa dengan epoch 250 model ANN akan bekerja dengan hasil yang sangat baik dalam melakukan klasifikasi dataset MNIST. Kemudian untuk hasil dari fungsi optimizer RMSProp mendapatkan nilai akurasi terendah sebesar 98.37% dengan Epoch 150 sedangkan untuk hasil akurasi tertinggi diperoleh dengan hasil sebesar 98.61% pada epoch 250. Dari kedua hasil fungsi optimizer tersebut dapat diartikan bahwa akurasi terbaik di peroleh pada Fungsi Optimizer Adam. Selanjutnya dari akurasi tertinggi tersebut dapat digambarkan dengan visualisasi gambar akurasi model supaya untuk mempermudah dalam melakukan analisis dari hasil yang di dapatkan serta hasil yang didapatkan lebih optimal. Gambar grafik dapat di tampilkan pada Gambar 9



Gambar 9 Akurasi ANN + Adam

Grafik ini menampilkan perkembangan akurasi pelatihan dan akurasi validasi yang bekerja sebanyak 250 epoch. Kedua kurva tersebut berfungsi untuk menilai sejauh mana model dapat belajar dengan baik dan apakah terjadi kelebihan fitting. Secara detail, grafik akurasi menunjukkan bahwa kurva berwarna biru mewakili akurasi pelatihan, yang di awal epoch meningkat dengan cepat dari sekitar 0,93 ke lebih dari 0,99. Setelah mencapai titik yang stabil (sekitar epoch ke-20 dan seterusnya), akurasi pelatihan mendekati 1,00 (100%) dan garisnya sangat stabil dengan sedikit fluktuasi. Model ANN terbaik ini dapat memahami data pelatihan dengan sangat baik. Akurasi yang hampir sempurna menunjukkan bahwa model mampu mempelajari pola data secara efektif. Akurasi pelatihan yang mendekati 1,0 menunjukkan kemampuan model yang baik dalam mengenali pola data, sedangkan akurasi validasi yang stabil di atas 0,98 menunjukkan kemampuan generalisasi yang strong. Tidak ada indikasi besar dari overfitting meskipun akurasi pelatihan sangat tinggi.

Klasikasi Model yang telah dilakukan menghasilkan sebuah gambar yang menunjukan hasil dari pengujian model tersebut. Adapun pengujian model yang dilakukan dapat dilihat pada Gambar 10.



Gambar 10 Hasil Klasifikasi Model ANN

Gambar menunjukkan prediksi digit MNIST untuk sekitar 15 contoh. Semua prediksi benar, menunjukkan model bekerja sangat baik pada sampel ini. Model mampu mengenali variasi penulisan angka yang berbeda. Visualisasi ini cocok sebagai bukti bahwa model melakukan klasifikasi dengan benar, meskipun grafik sebelumnya menunjukkan tanda-tanda overfitting dalam jangka panjang.

Kemudian dari Confusion Matrix tersebut enunjukkan tingkat kesalahan sangat rendah. Meskipun validation loss tinggi, itu tidak berarti akurasi rendah. Sedangkan hasil Precision, recal dan F1 Score terdapat pada tabel 3.

Tabel 3 Hasil Confusion Matrix

Kelas	Precision	Recall	F1-score
0	0.9849	0.9932	0.9932
1	0.9860	0.9860	0.9860
2	0.9948	0.9880	0.9880
3	0.9733	0.9898	0.9898
4	0.9900	0.9884	0.9884
5	0.9884	0.9740	0.9740
6	0.9805	0.9918	0.9918
7	0.9965	0.9880	0.9880
8	0.9873	0.9842	0.9842
9	0.9851	0.9834	0.9834

4.2. Pembahasan

Model klasifikasi digit tulisan tangan (dataset MNIST) telah dilatih menggunakan jaringan saraf dalam (ANN). Evaluasi dilakukan melalui beberapa metrik, yaitu akurasi, loss, visualisasi prediksi, confusion matrix, serta perhitungan recall, precision, dan F1-score pada setiap label output luaran. Pada Analisis Grafik Akurasi Grafik akurasi menunjukkan perkembangan akurasi data pelatihan dan akurasi data validasi proses training. Akurasi training meningkat dengan cepat dan mencapai hampir 100% pada epoch awal. Akurasi validasi stabil pada rentang 98% – 98.8%, dengan fluktuasi kecil yang masih normal

5. Kesimpulan

Penelitian yang telah dilakukan menghasilkan Akurasi training dan validasi tinggi menunjukkan performa klasifikasi yang sangat baik pada Model ANN dengan fungsi Optimizer adam menghasilkan akurasi tertinggi pada epoch 250 dengan hasil akurasi 98.67%, kemudian validation loss yang meningkat mengindikasikan adanya overfitting, meskipun tidak berdampak signifikan pada akurasi, sedangkan visualisasi prediksi memperlihatkan bahwa model mampu mengklasifikasikan digit tulisan tangan secara tepat pada berbagai variasi bentuk. Hasil Confusion matrix menunjukkan performa stabil pada seluruh kelas digit, dengan kesalahan yang minimal dan wajar dan precision, recall, dan F1-

score tinggi mengonfirmasi bahwa model tidak hanya akurat tetapi juga konsisten dan handal. Secara keseluruhan, model memiliki performa sangat baik untuk klasifikasi digit MNIST, meskipun tetap memerlukan perbaikan untuk mengurangi overfitting pada nilai loss.

6. Daftar Pustaka

- [1] N. Lestari, "Penerapan Deep Learning untuk Pengenalan Pola Belajar Siswa," *J. Pendidik. Indones.*, vol. 9, no. 4, pp. 311–320, 2020, [Online]. Available: <https://ejournal.undiksha.ac.id/index.php/JPI/article/view/32071>
- [2] L. Anggraini and Y. Yamasari, "Klasifikasi Citra Wajah Untuk Rentang Usia Menggunakan Metode Artificial Neural Network," *J. Informatics Comput. Sci.*, vol. 5, no. 02, pp. 185–192, 2023, doi: 10.26740/jinacs.v5n02.p185-192.
- [3] R. K. Mohapatra, B. Majhi, and S. K. Jena, "Classification performance analysis of MNIST Dataset utilizing a Multi-resolution Technique," in *2015 International Conference on Computing, Communication and Security (ICCCS)*, 2015, pp. 1–5. doi: 10.1109/CCCS.2015.7374136.
- [4] I. Wahidah, H. M. Putra, and T. Akbar, "Infotek : Jurnal Informatika dan Teknologi Analisa Komparatif Klasifikasi Citra Sayuran dengan Algoritma Support Vector Machine dan Convolutional Neural Network," vol. 9, no. 1, 2026.
- [5] M. R. Raharjo, Indra Riyana Rahadjeng, M. N. H. Siregar, and P. Alkhairi, "Optimasi Fungsi Aktivasi pada Artificial Neural Network untuk Prediksi Gagal Jantung Secara Akurat," *Explorer (Hayward)*, vol. 5, no. 1, pp. 42–51, 2025, doi: 10.47065/explorer.v5i1.1840.
- [6] B. Ihsan and D. Y. Sukma, "Pengaruh Masukan dan Fungsi Aktivasi terhadap Kecepatan Pelatihan Jaringan Syaraf Tiruan (JST) Modular sebagai Klasifikasi dan Estimasi Lokasi Gangguan pada Saluran Distribusi Bawah Tanah PT. Pertamina RU II Dumai," *J. Online Mhs. Fak. Tek. Univ. Riau*, vol. 4, no. 1, pp. 1–8, 2017.
- [7] A. J. Matematika, I. Alam, K. Angkasa, and N. Mei, "Penerapan Algoritma CNN Untuk Mendeteksi Tulisan Tangan Angka Romawi dengan Augmentasi Data Mochammad Toyib Tegar Decky Kurniawan Pratama Ibnu Aqil dapat Salah satu solusi yang menjanjikan adalah penggunaan Convolutional Neural Network (CNN), sebuah alg," vol. 2, no. 3, pp. 108–120, 2024.
- [8] L. S. Riva, F. Febriana, M. Panghurina, R. A. Maulida, B. Najah, and A. D. Yasiin, "Perbandingan Algoritma Cnn Dan Ann Dengan Projection Histogram Untuk Klasifikasi Citra Tulisan Tangan Berupa Angka," in *Prosiding Seminar Nasional Mahasiswa Bidang Ilmu Komputer dan Aplikasinya*, 2022, pp. 899–913.
- [9] Antonius Angga Kurniawan, Rama Dian Syah, and Rizki Ariyani, "Klasifikasi Citra Digital Tulisan Tangan Angka Menggunakan Metode Convolutional Neural Network," *J. Ilm. Tek.*, vol. 1, no. 1, pp. 36–41, 2022, doi: 10.56127/juit.v1i1.1718.
- [10] T. Amanda Lestari, D. Aulia Tanjung, L. Sofinah Harahap, and M. Rahma Kesuma, "Implementasi Algoritma Convolutional Neural Network (CNN) Untuk Pengenalan Angka Tulisan Tangan," *Data Sci. Indones.*, vol. 5, no. 2, pp. 93–100, 2025, doi: 10.47709/dsi.v5i2.7357.
- [11] H. Hananti and K. Sari, "Perbandingan Metode Support Vector Machine (SVM) dan Artificial Neural Network (ANN) pada Klasifikasi Gizi Balita," *Semin. Nas. Off. Stat.*, vol. 2021, no. 1, pp. 1036–1043, 2021, doi: 10.34123/semnasoffstat.v2021i1.1014.

- [12] M. Rosyada, I. Z. Putri, S. A. Virda, D. Aditya, and A. Saputra, "Pengenalan Angka Tulisan Tangan Menggunakan Metode Machine Learning Random Forest," vol. 9675, pp. 74–80, 2025.
- [13] V. No, M. Iqbal, Y. N. Dewi, and V. No, "Optimalisasi Prediksi Dalam Kelulusan Berbasis Deep Learning : Perbandingan Kinerja Multi- Layer Perceptron dan Deep Neural Networ," vol. 8, no. 2, pp. 630–641, 2025.
- [14] A. Sudianto, B. A. C. Permana, Muhammad Wasil, and Harianto, "Penerapan Sistem Payment Gateway Pada E-Commerce Sebagai Upaya Peningkatan Penjualan", *INFOTEK*, vol. 8, no. 1, pp. 271–279, Jan. 2025. doi: 10.29408/jit.v8i1.28323
- [15] Ida Wahidah, Hadian Mandala Putra, Suhartini, and Taufik Akbar, "Analisa Komparatif Klasifikasi Citra Sayuran dengan Algoritma Support Vector Machine dan Convolutional Neural Network", *INFOTEK*, vol. 9, no. 1, pp. 12–23, Jan. 2026.