

**SISTEM DETEKSI TUMOR OTAK PADA CITRA MRI MENGGUNAKAN
ALGORITMA *NEURAL NETWORK* (NN)**

SKRIPSI

Oleh:

RIZQY NUR MAULIDDINAH
NIM. 220605110024



**PROGRAM STUDI TEKNIK INFORMATIKA
FAKULTAS SAINS DAN TEKNOLOGI
UNIVERSITAS ISLAM NEGERI MAULANA MALIK IBRAHIM
MALANG
2026**

**SISTEM DETEKSI TUMOR OTAK PADA CITRA MRI MENGGUNAKAN
ALGORITMA *NEURAL NETWORK* (NN)**

SKRIPSI

Diajukan kepada:
Universitas Islam Negeri Maulana Malik Ibrahim Malang
Untuk memenuhi Salah Satu Persyaratan dalam
Memperoleh Gelar Sarjana Komputer (S.Kom)

Oleh :
RIZQY NUR MAULIDDINAH
NIM. 220605110024

**PROGRAM STUDI TEKNIK INFORMATIKA
FAKULTAS SAINS DAN TEKNOLOGI
UNIVERSITAS ISLAM NEGERI MAULANA MALIK IBRAHIM
MALANG
2026**

HALAMAN PERSETUJUAN

**SISTEM DETEKSI TUMOR OTAK PADA CITRA MRI MENGGUNAKAN
ALGORITMA NEURAL NETWORK (NN)**

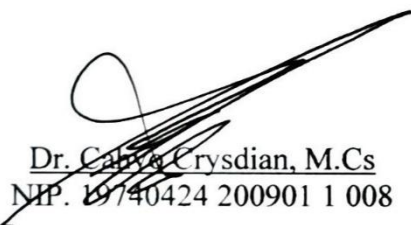
SKRIPSI

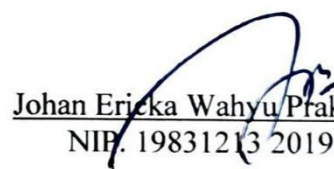
**Oleh :
RIZQY NUR MAULIDDINAH
NIM. 220605110024**

Telah Diperiksa dan Disetujui untuk Diuji:
Tanggal: 06 Maret 2026

Pembimbing I,

Pembimbing II,


Dr. Cahya Crysdian, M.Cs
NIP. 19740424 200901 1 008


Johan Ericka Wahyu Prakasa, M.Kom
NIP. 19831213 201903 1 004

Mengetahui,
Ketua Program Studi Teknik Informatika
Fakultas Sains dan Teknologi
Universitas Islam Negeri Maulana Malik Ibrahim Malang


Supriyono, M.Kom
NIP. 19841010 201903 1 012

HALAMAN PENGESAHAN

SISTEM DETEKSI TUMOR OTAK PADA CITRA MRI MENGGUNAKAN ALGORITMA *NEURAL NETWORK* (NN)

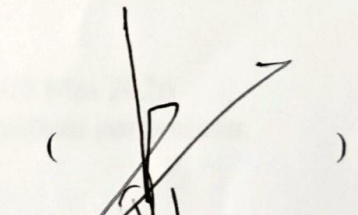
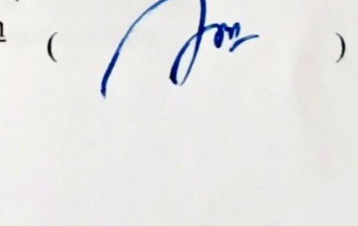
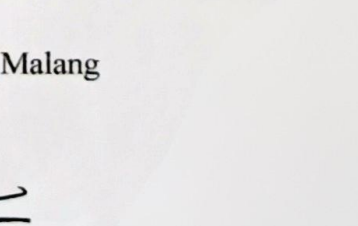
SKRIPSI

Oleh :
RIZQY NUR MAULIDDINAH
NIM. 22060511024

Telah Dipertahankan di Depan Dewan Penguji Skripsi
dan Dinyatakan Diterima Sebagai Salah Satu Persyaratan
Untuk Memperoleh Gelar Sarjana Komputer (S.Kom)
Tanggal: 12 Maret 2026

Susunan Dewan Penguji

Ketua Penguji	: <u>Dr. Irwan Budi Santoso, M.Kom</u> NIP. 19770103 201101 1 004
Anggota Penguji I	: <u>Nurizal Dwi Priandani, M.Kom</u> NIP. 19920830 202203 1 001
Anggota Penguji II	: <u>Dr. Cahyo Crysdian, M.Cs</u> NIP. 19740424 200901 1 008
Anggota Penguji III	: <u>Johan Ericka Wahyu Prakasa, M.Kom</u> NIP. 19831213 201903 1 004

()
()
()
()

Mengetahui dan Mengesahkan,
Ketua Program Studi Teknik Informatika
Fakultas Sains dan Teknologi
Universitas Islam Negeri Maulana Malik Ibrahim Malang




Supriyono, M.Kom
NIP. 19841010 201903 1 012

PERNYATAAN KEASLIAN TULISAN

Saya yang bertanda tangan di bawah ini:

Nama : Rizqy Nur Mauliddinah
NIM : 220605110024
Fakultas / Program Studi : Sains dan Teknologi / Teknik Informatika
Judul Skripsi : SISTEM DETEKSI TUMOR OTAK PADA CITRA
MRI MENGGUNAKAN ALGORITMA *NEURAL*
NETWORK (NN)

Menyatakan dengan sebenarnya bahwa Skripsi yang saya tulis ini benar-benar merupakan hasil karya saya sendiri, bukan merupakan pengambil alihan data, tulisan, atau pikiran orang lain yang saya akui sebagai hasil tulisan atau pikiran saya sendiri, kecuali dengan mencantumkan sumber cuplikan pada daftar pustaka.

Apabila dikemudian hari terbukti atau dapat dibuktikan skripsi ini merupakan hasil jiplakan, maka saya bersedia menerima sanksi atas perbuatan tersebut.

Malang, 05 Mei 2026
Yang membuat pernyataan,



Rizqy Nur Mauliddinah
NIM.220605110024

MOTTO

“Do Your Best, Trust Yourself, and Let Allah Do The Rest.”

“Bukan tentang seberapa cepat kamu sampai, tapi tentang seberapa kuat kamu bertahan hingga akhir.”

HALAMAN PERSEMBAHAN

Karya ini saya persembahkan kepada kedua orang tua saya tercinta, Bapak Choir dan Ibu Elly, yang telah memberikan dukungan penuh dengan segenap jiwa, raga, tenaga, dan segala pengorbanan. Meskipun proses ini memerlukan waktu lebih lama dari yang diharapkan, dan Bapak telah lebih dahulu berpulang ke rahmatullah sebelum karya ini selesai, namun tanpa doa, dukungan, dan pengorbanan almarhum Bapak, serta ridha Allah Subhanahu wa Ta'ala, saya tidak akan mampu menyelesaikan skripsi ini dan menjadi sarjana. Semoga Bapak tenang dan bahagia di sana, serta mendapatkan tempat terbaik di sisi-Nya. Aamiin.

Tak lupa, kepada Ibuku tercinta, yang tak pernah berhenti mendedikasikan seluruh hidupnya demi anak-anaknya. Terima kasih telah menjadi tempat pulang yang paling nyaman, seperti dermaga yang selalu setia menanti di tengah luasnya samudera kehidupan. Terima kasih juga untuk seluruh keluarga besar Bani Sohani yang telah membantu dan mendukung selama proses pengerjaan karya ini.

Karya ini juga saya persembahkan untuk kakak-kakak saya, Mas Rafli dan Mas Sultan, yang telah ikut berjuang dan senantiasa menjadi pelindung terdepan dalam setiap langkah perjalanan saya. Terima kasih atas segala kesabaran, dukungan moral maupun finansial, serta seluruh pengorbanan yang tidak ternilai. Semoga Mas Rafli dan Mas Sultan senantiasa diberikan kesehatan, rezeki yang berlimpah, serta kesuksesan di dunia dan akhirat.

Kepada sahabat saya, Tanti, yang selalu hadir di setiap suka maupun duka, yang dengan sabar mendengarkan setiap keluh kesah dan senantiasa mengingatkan ketika saya kehilangan arah. Darimu, saya belajar tentang makna ketulusan, rasa

syukur, dan keikhlasan yang sesungguhnya. Terima kasih atas segala dukungan dan kehadiranmu yang berarti.

Karya ini juga saya persembahkan untuk teman-teman Faza 19, teman-teman PKL, serta seluruh mahasiswa Teknik Informatika angkatan 2022. Perjalanan ini tidak akan terasa sama tanpa kebersamaan kita. Setiap tawa, perjuangan, dan kenangan yang kita ukir bersama menjadi bagian tak terlupakan dari perjalanan ini. Semoga kita semua senantiasa diberikan kemudahan dalam setiap langkah ke depan, dan kelak dapat meraih kesuksesan masing-masing dengan penuh kebanggaan.

Dan yang terakhir, karya ini saya persembahkan untuk diri saya sendiri. Untuk setiap malam yang terasa terlalu panjang, untuk setiap keraguan yang sempat menggoyahkan langkah, dan untuk setiap air mata yang jatuh dalam diam. Perjalanan ini tidak mudah, dan ada banyak momen di mana saya nyaris menyerah, tapi saya memilih untuk tetap bertahan. Terima kasih kepada diri sendiri yang tidak berhenti berjuang, meski lelah, meski sendirian, meski kehilangan. Syukur yang tak terhingga pada Allah Subhanahu wa Ta'ala atas segala kekuatan yang dititipkan, sehingga saya mampu berdiri sampai di titik ini. Karya ini adalah bukti bahwa saya mampu.

KATA PENGANTAR

Segala puji bagi Allah Subhānahu wa Ta‘ālā atas segala nikmat dan kasih sayang-Nya yang telah memudahkan penulis untuk menyelesaikan skripsi yang berjudul “SISTEM DETEKSI TUMOR OTAK PADA CITRA MRI MENGGUNAKAN ALGORITMA *NEURAL NETWORK* (NN)”. Shalawat serta salam senantiasa tercurah kepada junjungan kita Nabi Muhammad ﷺ ‘Alaihi Wasallam, keluarga, sahabat, serta para pengikutnya hingga akhir zaman. Semoga kita semua mendapatkan syafaat beliau di hari kiamat kelak, āmīn yā Rabbal ‘ālamīn.

Dengan rasa syukur yang mendalam, penulis menyadari bahwa penyelesaian skripsi ini tidak lepas dari bantuan, bimbingan, dan doa dari berbagai pihak. Oleh karena itu, penulis ingin menyampaikan ucapan terima kasih yang sebesar-besarnya kepada:

1. Prof. Dr. Hj. Ilfi Nurdiana, M.Si., CAHRM., CRMP, selaku Rektor Universitas Islam Negeri Maulana Malik Ibrahim Malang.
2. Dr. Agus Mulyono, M.Kes, selaku Dekan Fakultas Sains dan Teknologi Universitas Islam Negeri Maulana Malik Ibrahim Malang.
3. Supriyono, M.Kom., selaku Ketua Program Studi Teknik Informatika Universitas Islam Negeri Maulana Malik Ibrahim Malang.
4. Dr. Cahyo Crysdian, M.Cs., selaku Dosen Pembimbing I, yang dengan sabar, telaten, dan penuh dedikasi telah memberikan bimbingan, arahan, serta masukan yang sangat berharga sehingga penulis dapat menyelesaikan skripsi ini.

5. Johan Ericka Wahyu Prakasa, M.Kom, selaku Dosen Pembimbing II, yang telah meluangkan waktu dan pikiran untuk membimbing, mengarahkan, serta memberikan kritik dan saran yang konstruktif selama proses pengerjaan skripsi.
6. Dr. Irwan Budi Santoso, M.Kom dan Nurizal Dwi Priandani, M.Kom, selaku penguji yang telah memberikan masukan dan saran sehingga skripsi ini dapat disempurnakan.
7. Segenap dosen, staf administrasi, laboran, dan teman-teman mahasiswa Program Studi Teknik Informatika Fakultas Sains dan Teknologi yang telah memberikan dukungan moril maupun materiil selama penulis menempuh pendidikan dan mengerjakan skripsi ini.

Penulis menyadari bahwa skripsi ini masih jauh dari sempurna. Oleh karena itu, segala kritik, saran, dan masukan yang membangun sangat penulis harapkan demi kesempurnaan di masa mendatang. Semoga skripsi ini bermanfaat bagi penulis khususnya, dan bagi pembaca pada umumnya.

Malang, 12 Maret 2026

Penulis

DAFTAR ISI

HALAMAN PENGAJUAN	ii
HALAMAN PERSETUJUAN	iii
HALAMAN PENGESAHAN	iv
PERNYATAAN KEASLIAN TULISAN	v
MOTTO	vi
HALAMAN PERSEMBAHAN	vii
KATA PENGANTAR	ix
DAFTAR ISI	xi
DAFTAR GAMBAR	xiii
DAFTAR TABEL	xiv
ABSTRAK	xvi
ABSTRACT	xvii
البحث مستخلص	xviii
BAB I PENDAHULUAN	1
1.1 Latar Belakang	1
1.2 Pernyataan Masalah	7
1.3 Tujuan Penelitian	7
1.4 Manfaat Penelitian	7
1.5 Batasan Masalah	7
BAB II STUDI PUSTAKA	9
BAB III DESAIN DAN IMPLEMENTASI SISTEM	16
3.1 Desain Sistem.....	16
3.1.1 <i>Input Image</i>	16
3.1.2 <i>Data Preprocessing</i>	17
3.1.3 <i>Feature Extraction</i>	18
3.1.4 <i>Neural Network (NN)</i>	23
3.1.5 <i>Neural Network 1 Hidden layer</i>	27
3.1.6 <i>Neural Network 2 Hidden layer</i>	33
3.2 Implementasi Sistem.....	40
3.2.1 <i>Training Model</i>	41
3.3 Antarmuka Sistem.....	42
BAB IV UJI COBA DAN PEMBAHASAN	44
4.1 Desain Uji Coba	44
4.1.1 <i>Data Preparation</i>	45
4.1.2 Skenario Uji Coba	48
4.1.3 Ukuran Pengujian	53
4.2 Hasil Uji Coba.....	56
4.2.1 Hasil Skenario Uji Coba <i>Neural Network 1 Hidden layer</i>	56
4.2.1.1 Skenario 1-A-X.....	56
4.2.1.2 Skenario 1-A-Y	57
4.2.1.3 Skenario 1-A-Z	58
4.2.1.4 Skenario 1-B-X	60

4.2.1.5	Skenario 1-B-Y	61
4.2.1.6	Skenario 1-B-Z	62
4.2.1.7	Skenario 1-C-X	63
4.2.1.8	Skenario 1-C-Y	64
4.2.1.9	Skenario 1-C-Z	65
4.2.2	Hasil Skenario Pengujian <i>Neural Network 2 Hidden layer</i>	67
4.2.2.1	Skenario 2-BA-X	67
4.2.2.2	Skenario 2-BA-Y	68
4.2.2.3	Skenario 2-BA-Z	69
4.2.2.4	Skenario 2-CB-X	70
4.2.2.5	Skenario 2-CB-Y	72
4.2.2.6	Skenario 2-CB-Z	73
4.2.2.7	Skenario 2-CA-X	74
4.2.2.8	Skenario 2-CA-Y	75
4.2.2.9	Skenario 2-CA-Z	76
4.3	Pembahasan	77
BAB V Penutup		97
5.1	Kesimpulan	97
5.2	Saran	98
DAFTAR PUSTAKA		

DAFTAR GAMBAR

Gambar 3. 1 Desain Sistem.....	16
Gambar 3. 2 Visualisasi Formasi Matrix GLCM.....	19
Gambar 3. 3 Grafik Fungsi Aktivasi Sigmoid	25
Gambar 3. 4 Grafik Fungsi Aktivasi ReLU	26
Gambar 3. 5 Skema <i>Forward Propagation Neural Network 1 Hidden Layer</i>	28
Gambar 3. 6 Skema <i>Backward Propagation Neural Network 1 Hidden Layer</i>	31
Gambar 3. 7 Skema <i>Forward Propagation Neural Network 2 Hidden Layer</i>	34
Gambar 3. 8 Skema <i>Backward Propagation Neural Network 2 Hidden Layer</i>	37
Gambar 3. 9 <i>Flowchart</i> Proses <i>Training Model</i>	41
Gambar 3. 10 Antarmuka Sistem Deteksi Tumor Otak	43
Gambar 4. 1 <i>Flowchart</i> Proses <i>Testing Model</i>	45
Gambar 4. 2 <i>Sample Data MRI Images</i> Tumor Otak.....	46
Gambar 4. 3 <i>Sample Data MRI Images</i> Otak Sehat.....	46
Gambar 4. 4 Boxplot Distribusi Nilai <i>Accuracy</i> Semua Model.....	78
Gambar 4. 5 Boxplot Distribusi Nilai <i>Precision</i> Semua Model.....	81
Gambar 4. 6 Boxplot Distribusi Nilai <i>Recall</i> Semua Model.....	83
Gambar 4. 7 Boxplot Distribusi Nilai <i>F1-score</i> Semua Model	86

DAFTAR TABEL

Tabel 2. 1 Rangkuman Hasil Evaluasi Model.....	14
Tabel 3. 1 Definisi dan Fungsi Parameter Model.....	42
Tabel 4. 1 Persebaran Jumlah Data	48
Tabel 4. 2 Skenario Jumlah <i>Hidden Layer Neural Network</i>	49
Tabel 4. 3 Skenario Jumlah Neuron pada <i>Hidden Layer</i>	50
Tabel 4. 4 Skenario Nilai <i>Learning Rate Neural Network</i>	51
Tabel 4. 5 Kombinasi Skenario Pengujian	52
Tabel 4. 6 Confusion Matrix	55
Tabel 4. 7 Rangkuman Hasil <i>Training Model 1-A-X</i>	57
Tabel 4. 8 Rangkuman Performa Model 1-A-X.....	57
Tabel 4. 9 Rangkuman Hasil <i>Training Model 1-A-Y</i>	58
Tabel 4. 10 Rangkuman Performa Model 1-A-Y.....	58
Tabel 4. 11 Rangkuman Hasil <i>Training Model 1-A-Z</i>	59
Tabel 4. 12 Rangkuman Performa Model 1-A-Z	59
Tabel 4. 13 Rangkuman Hasil <i>Training Model 1-B-X</i>	60
Tabel 4. 14 Rangkuman Performa Model 1-B-X.....	60
Tabel 4. 15 Rangkuman Hasil <i>Training Model 1-B-Y</i>	61
Tabel 4. 16 Rangkuman Performa Model 1-B-Y	62
Tabel 4. 17 Rangkuman Hasil <i>Training Model 1-B-Z</i>	62
Tabel 4. 18 Rangkuman Performa Model 1-B-Z	63
Tabel 4. 19 Rangkuman Hasil <i>Training Model 1-C-X</i>	63
Tabel 4. 20 Rangkuman Performa Model 1-C-X.....	64
Tabel 4. 21 Rangkuman Hasil <i>Training Model 1-C-Y</i>	65
Tabel 4. 22 Rangkuman Performa Model 1-C-Y	65
Tabel 4. 23 Rangkuman Hasil <i>Training Model 1-C-Z</i>	66
Tabel 4. 24 Rangkuman Performa Model 1-C-Z.....	66
Tabel 4. 25 Rangkuman Hasil <i>Training Model 2-BA-X</i>	67
Tabel 4. 26 Rangkuman Performa Model 2-BA-X	68
Tabel 4. 27 Rangkuman Hasil <i>Training Model 2-BA-Y</i>	69
Tabel 4. 28 Rangkuman Performa Model 2-BA-Y	69
Tabel 4. 29 Rangkuman Hasil <i>Training Model 2-BA-Z</i>	70
Tabel 4. 30 Rangkuman Performa Model 2-BA-Z	70
Tabel 4. 31 Rangkuman Hasil <i>Training Model 2-CB-X</i>	71
Tabel 4. 32 Rangkuman Hasil Uji Coba Model 2-CB-X	71
Tabel 4. 33 Rangkuman Hasil <i>Training Model 2-CB-Y</i>	72
Tabel 4. 34 Rangkuman Performa Model 2-CB-Y	72
Tabel 4. 35 Rangkuman Hasil <i>Training Model 2-CB-Z</i>	73

Tabel 4. 36 Rangkuman Performa Model 2-CB-Z.....	73
Tabel 4. 37 Rangkuman Hasil <i>Training</i> Model 2-CA-X.....	74
Tabel 4. 38 Rangkuman Performa Model 2-CA-X.....	74
Tabel 4. 39 Rangkuman Hasil <i>Training</i> Model 2-CA-Y.....	75
Tabel 4. 40 Rangkuman Performa Model 2-CA-Y	75
Tabel 4. 41 Rangkuman Hasil <i>Training</i> Model 2-CA-Z	76
Tabel 4. 42 Rangkuman Performa Model 2-CA-Z	76
Tabel 4. 43 Hasil Uji <i>T-Test Accuracy</i> Model 2-CB-X terhadap Model Lainnya.	79
Tabel 4. 44 Hasil Uji <i>T-Test Precision</i> Model 2-BA-X terhadap Model Lainnya.	82
Tabel 4. 45 Hasil Uji <i>T-Test Recall</i> Model 2-CB-X terhadap Model Lainnya.....	84
Tabel 4. 46 Hasil Uji <i>T-Test F1-Score</i> Model 2-CB-X terhadap Model Lainnya .	88

ABSTRAK

Mauliddinah, Rizqy Nur. 2026. **Sistem Deteksi Tumor Otak pada Citra MRI menggunakan Algoritma Neural Network (NN)**. Skripsi. Jurusan Teknik Informatika Fakultas Sains dan Teknologi Universitas Islam Negeri Maulana Malik Ibrahim Malang. Pembimbing: (I) Dr. Cahyo Crysdiyan, M.Cs (II) Johan Ericka Wahyu Prakasa, M.Kom.

Kata kunci: Tumor Otak, Citra MRI, GLCM, *Neural Network*.

Penelitian ini bertujuan untuk mengevaluasi performa model *Neural Network* dalam mendeteksi tumor otak pada citra MRI berdasarkan metrik evaluasi *accuracy*, *precision*, *recall*, dan *F1-score* menggunakan dataset 4514 pasien. Metode yang digunakan adalah arsitektur *Neural Network* yang dikombinasikan dengan ekstraksi fitur berbasis *Gray Level Co-Occurrence Matrix (GLCM)*. Terdapat 18 konfigurasi model yang diuji dengan memvariasikan kedalaman arsitektur, jumlah neuron, serta nilai *learning rate*, dengan evaluasi menggunakan *5-Fold Cross-Validation* dan *confusion matrix*. Hasil penelitian menunjukkan bahwa kedalaman arsitektur merupakan faktor determinan yang signifikan, di mana model dengan dua *hidden layer* secara konsisten mengungguli model dengan satu *hidden layer* dengan rentang akurasi 93,51%–96,61% dibandingkan 81,34% – 94,69%. Model optimal yang ditetapkan adalah 2-CB-X dengan arsitektur *NeuralNetworkV2* (256→128 neuron, aktivasi ReLU dan Sigmoid, *learning rate* 0,1), yang mencapai *mean accuracy* 96,61% $\pm$ 0,43%, *precision* 95,96% $\pm$ 0,42%, *recall* 97,81% $\pm$ 0,51%, dan *F1-score* 96,87% $\pm$ 0,40%.

ABSTRACT

Mauliddinah, Rizqy Nur. 2026. **Brain Tumor Detection System on MRI Images Using Neural Network (NN) Algorithm**. Thesis. Department of Informatics Engineering, Faculty of Science and Technology, Universitas Islam Negeri Maulana Malik Ibrahim Malang. Advisor: (I) Dr. Cahyo Crysdian, M.Cs (II) Johan Ericka Wahyu Prakasa, M.Kom.

This study aims to evaluate the performance of Neural Network models in detecting brain tumors on MRI Images based on accuracy, precision, recall, and F1-score evaluation metrics using a dataset of 4,514 images. The method employed is a Neural Network architecture combined with texture feature extraction based on the Gray Level Co-Occurrence Matrix (GLCM). A total of 18 model configurations were tested by varying the architecture depth, number of neurons, and *learning rate* values, with evaluation conducted using *5-Fold Cross-Validation* and confusion matrix. The results demonstrate that architecture depth is a significant determinant factor, where models with two hidden layers consistently outperformed models with one hidden layer, achieving an accuracy range of 93.51%–96.61% compared to 81.34%–94.69%. The optimal model established is 2-CB-X with a NeuralNetworkV2 architecture (256→128 neurons, ReLU and Sigmoid activation, *learning rate* 0.1), achieving a *mean* accuracy of 96.61% $\pm$ 0.43%, precision of 95.96% $\pm$ 0.42%, recall of 97.81% $\pm$ 0.51%, and F1-score of 96.87% $\pm$ 0.40%.

Key words: Brain Tumor, MRI Image, GLCM, Neural Network.

مستخلص البحث

مولدينه، رزقي نور. 2026. نظام كشف أورام الدماغ في صور الرنين المغناطيسي باستخدام خوارزمية الشبكة العصبية الاصطناعية. البحث الجامعي. قسم تقنية المعلومات، كلية العلوم والتكنولوجيا، جامعة مولانا مالك إبراهيم الإسلامية الحكومية مالانج. المشرف: (١) الدكتور جاهيو كريسديان، الماجستير (٢) جوهان إيريكا واهيو براكاسا، الماجستير.

الكلمات المفتاحية: ورم الدماغ، مصفوفة التضامن على المستوى الرمادي، صور الرنين المغناطيسي، الشبكة العصبية الاصطناعية.

يهدف هذا البحث إلى تقييم أداء نماذج الشبكة العصبية الاصطناعية في الكشف عن أورام الدماغ من صور الرنين المغناطيسي، استناداً إلى معايير التقييم المتمثلة في الدقة الكلية والدقة الإيجابية والاستدعاء ودرج F1 ، وذلك باستخدام مجموعة بيانات تضم 4514 صورة. تعتمد المنهجية على بنية الشبكة العصبية الاصطناعية المدججة مع استخراج السمات النسيجية القائمة على مصفوفة التضامن ذات المستوى الرمادي (GLCM). تم اختبار ثمانية عشر هيئة نموذجية من خلال تغيير عمق البنية المعمارية وعدد الخلايا العصبية وقيم معدل التعلم، مع إجراء التقييم باستخدام التحقق المتقاطع ذي الخمسة طيات ومصفوفة الالتباس. كشفت النتائج أن عمق البنية المعمارية يُعدّ عاملاً محدداً بالغ الأهمية، إذ تفوّقت النماذج ذات الطبقتين المخفيتين باستمرار على النماذج ذات الطبقة المخفية الواحدة، محققةً نطاق دقة يتراوح بين 93.51% و 96.61% مقارنةً بنطاق يتراوح بين 81.34% و 94.69%. تم تحديد النموذج الأمثل وهو النموذج 2-CB-X المبني على بنية NeuralNetworkV2 المكوّنة من 256 إلى 128 خلية عصبية مع دالة تنشيط ReLU إلى Sigmoid ومعدل تعلم 0.1، إذ حقق متوسط دقة كلية 96.61% بانحراف معياري 0.43%، ودقة إيجابية 95.96% بانحراف معياري 0.42%، واستدعاء 97.81% بانحراف معياري 0.51%، ودرجة F1 تبلغ 96.87% بانحراف معياري 0.40%.

BAB I

PENDAHULUAN

1.1 Latar Belakang

Otak adalah organ pusat sistem saraf dalam diri manusia yang berperan vital dalam mengatur berbagai fungsi tubuh, mulai dari pengendalian gerak, persepsi sensorik, pemikiran, emosi, hingga fungsi otonom seperti pernapasan dan juga detak jantung. Fungsi otak yang kompleks ini menjadikan otak sebagai organ yang berperan sangat penting bagi kelangsungan hidup dan kualitas hidup seorang manusia (Chen et al., 2024). Sedangkan tumor merupakan pertumbuhan sel abnormal yang bersifat jinak atau non-malignan dan ganas atau malignan. Apabila tumor ini berada pada otak, maka dampaknya akan sangat berbahaya karena dapat menekan jaringan otak normal, mengganggu fungsi saraf, serta menyebabkan berbagai gejala, seperti kejang, gangguan bicara, kelemahan anggota gerak, hingga perubahan perilaku (Pichaivel et al., 2022). Selain itu, apabila tumor sudah menyerang bagian otak, maka dampaknya akan sangat fatal karena otak tidak memiliki ruang yang cukup untuk menampung pertumbuhan massa baru yang dapat menyebabkan tekanan intrakranial meningkat sehingga akan terjadi kerusakan permanen pada jaringan otak, kecacatan neurologis, hingga kematian (Ostrom et al., 2022).

Deteksi dini tumor otak sangat penting dilakukan karena memungkinkan intervensi lebih awal sehingga dapat meningkatkan peluang keberhasilan terapi, serta menurunkan angka mortalitas pada pasien penderita (Ijaz et al., 2025a). Namun, kondisi aktual pada masyarakat mengungkap bahwa deteksi dini pada tumor otak

masih menghadapi berbagai tantangan. Pertama, terdapat kesenjangan signifikan antara jumlah radiologi dengan beban kerja analisis citra MRI. Data Kementerian Kesehatan Indonesia menunjukkan rasio radiolog terhadap populasi hanya 1:250.000, jauh dibawah standar WHO dengan rasio 1:50.000 (Ijaz et al., 2025b). Sementara itu, jumlah pemeriksaan MRI terus meningkat setiap tahunnya. Kondisi ini mengakibatkan waktu tunggu diagnosis yang cukup lama dan berisiko menunda penanganan pada kasus-kasus kritis.

Selain itu, interpretasi citra MRI tumor otak secara manual oleh radiolog memiliki variabilitas diagnostik yang tinggi (*inter-observer variability*) berkisar 15-25%, terutama pada tumor stadium awal atau tumor dengan karakteristik visual yang ambigu. Hal ini disebabkan oleh kompleksnya struktur otak manusia, bervariasinya bentuk serta ukuran tumor, dan kemiripan dengan jaringan sehat, sehingga membuat identifikasi manual sangat menantang dan rawan perbedaan interpretasi antar radiolog. Kondisi tersebut dapat menyebabkan kesalahan diagnosis yang berdampak negatif pada penanganan pasien. Di samping itu, studi oleh Taylor-Phillips & Stinton (2019) menunjukkan bahwasanya kelelahan radiolog setelah membaca banyak citra dapat menurunkan akurasi diagnosis hingga 4–8% serta meningkatkan risiko *error* dan variabilitas.

Lebih lanjut, kurangnya sistem *second opinion* yang terjangkau dan cepat di fasilitas kesehatan tingkat primer dan sekunder. Banyak rumah sakit di berbagai daerah Indonesia yang tidak memiliki radiolog spesialis *neuro-imaging*, sehingga harus merujuk kasus ke rumah sakit pusat dengan biaya dan waktu yang signifikan (Handayani et al., 2021). Keterbatasan alat pencitraan, kurangnya pelatihan tenaga

medis, dan infrastruktur yang tidak memadai juga memperparah keterlambatan diagnosis (Rahmartani et al., 2024). Kondisi ini menciptakan *barrier* akses terhadap diagnosis berkualitas, terutama bagi pasien di wilayah terpencil atau dari kalangan ekonomi menengah ke bawah.

Berbagai tantangan tersebut mengakibatkan banyaknya kasus tumor otak yang baru terdiagnosis pada stadium lanjut, dimana gejala yang dialami sudah berat dan pilihan terapi menjadi terbatas sehingga prognosis pasien memburuk (Shakir et al., 2024). Secara ideal, deteksi dini diharapkan dapat dilakukan melalui kombinasi teknologi pencitraan mutakhir dan kecerdasan buatan yang dapat memberikan *second opinion* otomatis, mengurangi beban kerja radiolog, dan meminimalkan variabilitas diagnostik. Namun, kenyataannya, akses terhadap teknologi ini masih belum merata dan implementasinya di layanan kesehatan primer sangat terbatas (Kumar, 2025).

Teknologi *machine learning* telah mengalami perkembangan pesat dalam deteksi tumor otak, khususnya melalui analisis citra *Magnetic Resonance Imaging* (MRI). Berbagai metode *machine learning*, seperti *Support Vector Machine* (SVM), *Random Forest*, dan model hibrida, telah digunakan untuk klasifikasi dan segmentasi tumor otak, dimana metode-metode tersebut menunjukkan potensi dalam meningkatkan akurasi diagnosis (Pimpalkar et al., 2023). Selain itu, pendekatan berbasis *deep learning*, seperti *Convolutional Neural Network* (CNN), telah menjadi standar utama dalam penelitian sebelumnya karena kemampuannya mengekstraksi fitur kompleks dari citra MRI secara otomatis, menghasilkan akurasi

deteksi yang sangat tinggi, bahkan mencapai lebih dari 97% pada beberapa studi (Khan et al., 2022a) (Sharma et al., 2022).

Model *deep learning* seperti CNN memang menawarkan akurasi tinggi untuk deteksi tumor otak, namun model tersebut juga memiliki beberapa keterbatasan signifikan untuk implementasi praktis di Indonesia. Model CNN ini membutuhkan hardware *GPU high-end* dan waktu *training model* yang cukup lama, sehingga biaya operasional dan *maintenance* dapat meningkat. Namun, hanya sebagian kecil rumah sakit di Indonesia yang memiliki komputasi AI berbasis GPU, sehingga membatasi adopsi teknologi ini secara luas (Salehi et al., 2023). Selain itu, CNN sering dianggap model “*black box*” karena sulit menjelaskan dasar keputusan yang dihasilkan, sehingga menurunkan kepercayaan dan akuntabilitas klinis (Xie et al., 2022). Model ini juga rentan terhadap *overfitting* pada dataset kecil atau menengah tanpa teknik regularisasi dan augmentasi data yang tepat (Mohammed et al., 2024).

Penelitian ini memilih pendekatan *Neural Network* (NN) sederhana dengan arsitektur *Multi-Layer Perception* (MLP) yang dikombinasikan dengan ekstraksi fitur manual berbasis *Gray Level Co-Occurance Matrix* (GLCM) untuk analisis citra MRI otak. Pendekatan ini dipilih karena fleksibilitas NN dalam memodelkan hubungan non-linier dalam data citra MRI dan kemampuannya untuk memproses data berdimensi tinggi dengan cepat dan akurat, mengatasi keterbatasan metode tradisional yang bergantung pada analisis manual dan rentan terhadap kesalahan manusia. GLCM mengekstraksi fitur tekstur seperti kontras, korelasi, dan homogenitas dari citra MRI, yang sangat relevan untuk mengidentifikasi

karakteristik tumor otak tanpa memerlukan pemrosesan yang kompleks. Penggunaan NN dengan GLCM membutuhkan sumber daya komputasi yang rendah, sehingga cocok untuk fasilitas kesehatan primer dan sekunder. Selain itu, fitur GLCM memiliki makna fisik yang jelas, sehingga mudah dipahami oleh tenaga medis dan meningkatkan kepercayaan klinis. Studi oleh Kang et al., (2021) menunjukkan bahwa kombinasi GLCM dan NN mampu mencapai akurasi tinggi untuk klasifikasi tumor otak pada dataset MRI kecil hingga menengah. Dengan demikian, pendekatan ini menawarkan keseimbangan optimal antara akurasi, efisiensi komputasi, dan interpretabilitas, yang menjadikannya solusi yang sesuai untuk mendukung *workflow* klinis yang transparan di fasilitas kesehatan Indonesia dengan sumber daya terbatas.

Dalam perspektif Islam, upaya deteksi dini tumor otak melalui pemanfaatan teknologi kecerdasan buatan, seperti *Neural Network* (NN) pada citra MRI, sejalan dengan perintah Allah untuk memelihara kehidupan dan kesehatan manusia. Allah SWT berfirman dalam Al-Qur'an Surah Al-Maidah ayat 32 :

وَمَنْ أَحْيَاهَا فَكَأَنَّمَا أَحْيَا النَّاسَ جَمِيعًا ﴿٣٢﴾

“Barang siapa memelihara kehidupan seorang manusia, maka seakan-akan dia telah memelihara kehidupan semua manusia” (QS. Al-Maidah: 32).

Pada cuplikan ayat tersebut menegaskan bahwa menyelamatkan satu jiwa memiliki nilai yang setara dengan menyelamatkan seluruh umat manusia, yang menunjukkan betapa mulianya upaya dalam menjaga dan memelihara kehidupan. Prinsip ini menggarisbawahi kewajiban umat Islam untuk saling tolong-menolong, khususnya dalam bidang kesehatan sebagai bagian dari amanah kemanusiaan. Hal

tersebut menunjukkan kewajiban umat untuk saling menjaga dan memelihara kehidupan manusia. Salah satu upaya dalam penjagaan tersebut yaitu melalui tolong menolong dalam kepentingan kesehatan. Pengembangan teknologi berbasis *Neural Network* untuk deteksi tumor otak merupakan manifestasi nyata dari nilai-nilai Islam dalam memanfaatkan ilmu pengetahuan untuk kemaslahatan umat, sebagaimana sabda Rasulullah SAW pada Hadits: “*Sebaik-baik manusia adalah yang paling bermanfaat bagi manusia lainnya.*” (HR. Ahmad). Melalui penelitian ini, terwujudlah tanggung jawab kemanusiaan untuk meningkatkan akses terhadap diagnosis medis yang akurat dan terjangkau, terutama bagi masyarakat di wilayah dengan fasilitas kesehatan terbatas, sehingga lebih banyak jiwa yang dapat diselamatkan melalui deteksi dini dan penanganan yang tepat waktu.

Dengan demikian, penelitian ini akan membangun model berbasis *Neural Network* (NN) untuk analisis citra MRI dalam mendeteksi tumor otak secara otomatis. Model ini dirancang untuk mengatasi tantangan deteksi dini dengan memanfaatkan kemampuan model dalam mempelajari hubungan non-linier dalam data citra medis, sehingga meminimalkan ketergantungan pada tenaga ahli radiologi yang terbatas dan mempercepat proses diagnosis. Penelitian ini diharapkan dapat menghasilkan model NN yang efisien untuk meningkatkan akurasi dan efisiensi diagnosis, mendukung transformasi digital di bidang kesehatan, dan secara signifikan meningkatkan angka kelangsungan hidup pasien tumor otak di berbagai komunitas.

1.2 Pernyataan Masalah

Berapa tingkat *accuracy*, *precision*, *recall*, dan *F1-score* yang dapat dicapai oleh model *Neural Network* dalam mendeteksi tumor otak?

1.3 Tujuan Penelitian

Mengevaluasi performa model *Neural Network* berdasarkan metrik evaluasi *accuracy*, *precision*, *recall*, dan *F1-score*.

1.4 Manfaat Penelitian

Penelitian ini diharapkan dapat memberikan manfaat sebagai berikut:

1. Bagi Institusi Kesehatan, penelitian ini diharapkan bisa menyediakan tools bantu yang dapat meningkatkan efisiensi proses diagnosis tumor otak dan mengurangi beban kerja radiologis.
2. Bagi Tenaga Medis, penelitian ini diharapkan dapat membantu dokter dalam proses *screening* awal untuk diagnosis tumor otak, terutama di fasilitas kesehatan dengan keterbatasan tenaga ahli.
3. Bagi Pasien, penelitian ini diharapkan memungkinkan deteksi dini yang lebih cepat dan akurat, sehingga dapat meningkatkan peluang keberhasilan pengobatan.

1.5 Batasan Masalah

Untuk memfokuskan penelitian, penelitian ini dibatasi pada hal berikut :

1. Penelitian ini menggunakan dataset publik citra MRI otak yang tersedia di Kaggle.

2. Jenis *MRI Images* yang diklasifikasi adalah *MRI Images* otak pasien dengan tumor dan *MRI Images* pasien dengan otak sehat.

BAB II

STUDI PUSTAKA

Neural Network (NN) merupakan model komputasi yang meniru struktur dan fungsi jaringan saraf otak manusia, terdiri dari lapisan-lapisan neuron buatan yang saling terhubung dan mampu mempelajari pola kompleks dari data melalui proses pelatihan. Dalam konteks deteksi tumor pada otak, *neural network* menunjukkan keunggulan signifikan dibandingkan metode lainnya karena kemampuannya menangkap hubungan non-linier yang kompleks, fleksibilitas dalam memproses berbagai jenis data citra, serta adaptasi terhadap data beranotasi melalui pelatihan berbasis algoritma optimisasi seperti *backpropagation* (Abdalla & Esmail, 2018). Selain itu, NN dapat mengidentifikasi fitur penting seperti tekstur, bentuk, dan intensitas piksel pada citra MRI, yang sangat krusial untuk membedakan jenis dan tingkat keparahan tumor otak. Penelitian menunjukkan bahwa model NN, khususnya CNN dan DNN mampu mencapai akurasi deteksi hingga 97–100%, jauh melampaui metode tradisional yang bergantung pada analisis manual dan rentan terhadap kesalahan manusia (Rahman & Islam, 2023). Kemajuan teknologi NN ini telah mendorong berbagai penelitian untuk mengeksplorasi pendekatan yang lebih efisien, akurat, dan dapat diandalkan dalam deteksi dan klasifikasi tumor otak.

Seiring dengan perkembangan tersebut, penelitian terkait deteksi tumor otak menggunakan teknologi *machine learning* maupun *deep learning* telah menunjukkan kemajuan yang signifikan pada beberapa tahun terakhir. Berbagai pendekatan telah banyak diterapkan, mulai dari metode tradisional hingga model

berbasis *deep learning*, untuk meningkatkan akurasi dan efisiensi dalam kalsifikasi dan segmentasi tumor otak. (Naseer et al., 2021) mengembangkan sistem *Computer-Aided Diagnosis (CAD)* berbasis *Convolutional Neural Network (CNN)* guna diagnosis dini tumor otak melalui peningkatan akurasi menggunakan augmentasi citra MRI, dengan fokus pada peningkatan ketahanan model terhadap data *unseen* untuk meningkatkan rasio kelangsungan hidup pasien. Penelitian ini menggunakan dataset benchmark BR35H yang berisi citra MRI tumor otak, dengan evaluasi keberlanjutan pada enam dataset lain yaitu BMI-I, BTI, BMI-II, BTS, BMI-III, dan BD-BT, serta menerapkan augmentasi geometris (seperti rotasi, flipping) dan standardisasi statistik untuk mengatasi keterbatasan data. Model CNN yang diusulkan dilatih untuk klasifikasi biner (tumor vs. non-tumor), dengan hiperparameter seperti *learning rate* dan *epoch* yang dioptimalkan. Evaluasi menunjukkan performa superior dengan akurasi rata-rata 98.8%, spesifisitas 0.99, dan akurasi 100% pada dataset BTS serta BD-BT, melebihi sistem eksisting seperti SVM, KNN, dan CNN lain. Kelebihan model ini meliputi ketahanan tinggi terhadap variasi data *unseen*, peningkatan akurasi diagnosis dini yang mendukung intervensi medis tepat waktu, serta perbandingan komprehensif dengan *state-of-the-art*, namun keterbatasannya terletak pada ketergantungan pada dataset publik yang mungkin kurang variatif secara klinis, risiko *overfitting* pada augmentasi berlebih, interpretabilitas terbatas pada model *deep learning*, serta kurangnya evaluasi pada subtype tumor spesifik atau data multimodal *real-time*.

Penelitian oleh Shah et al., (2022) bertujuan untuk mengembangkan pendekatan *robust* dalam deteksi tumor otak pada citra *Magnetic Resonance*

Imaging (MRI) guna meningkatkan akurasi klasifikasi dan deteksi tumor secara otomatis. Penelitian ini menggunakan dataset dari Kaggle yang berisi 3762 citra MRI, termasuk 1500 citra tumor dan 1500 citra non-tumor, dengan pembagian data berupa 80% untuk pelatihan dan 20% untuk validasi. Metode yang diusulkan melibatkan *fine-tuning Convolutional Neural Network (CNN) EfficientNet-B0* dengan *layer* tambahan berupa *flattening*, *dropout*, dan *sigmoid classifier*, didukung oleh *transfer learning* dari dataset *ImageNet* dan optimasi dengan Adam serta fungsi *loss binary cross-entropy*. Hasil percobaan penelitian menunjukkan akurasi validasi sebesar 98.87%, presisi 99.4%, sensitivitas 99.5%, spesifisitas 99.2%, dan AUC 0.988. Kelebihan model ini yaitu ukuran model yang ringan (16.8MB) dengan performa tinggi, meskipun terdapat potensi *overfitting* pada data kecil dan kompleksitas komputasi.

Penelitian oleh (Anantharajan et al., 2024) bertujuan untuk mengembangkan metode deteksi tumor otak pada citra MRI menggunakan pendekatan *deep learning* dan *machine learning* guna meningkatkan identifikasi dini dan akurasi diagnosis. Penelitian ini menggunakan dataset citra MRI dari repositori publik yaitu Kaggle, dengan total sekitar 4600 gambar, berupa 2513 citra tumor dan 2087 citra *healthy*. Model yang diusulkan yaitu *Ensemble Deep Neural Support Vector Machine (EDN-SVM)*, dimana model ini dipilih untuk mengintegrasikan kekuatan *deep neural network* dengan *support vector machine* dalam mengklasifikasikan jaringan abnormal. Teknik *preprocessing* yang digunakan pada penelitian ini ialah *Adaptive Contrast Enhancement Algorithm (ACEA)* dan median filter untuk meningkatkan kontras dan mengurangi *noise*,

segmentasi *fuzzy c-means* untuk memisahkan area tumor, serta ekstraksi fitur *Gray-level co-occurrence matrix (GLCM)* yang mencakup *energy*, *mean*, *entropy*, dan *contrast*. Evaluasi menunjukkan performa superior dengan akurasi 97.93%, sensitivitas 92%, dan spesifisitas 98% dalam membedakan jaringan tumor dari normal. Kelebihan model ini meliputi efisiensi dalam menangani data medis dengan akurasi tinggi, kemampuan deteksi dini yang mendukung pengambilan keputusan klinis, serta integrasi *hybrid DL-ML* yang *robust* terhadap variasi citra. Namun keterbatasannya terletak pada ketergantungan pada fitur GLCM yang mungkin kurang optimal untuk dataset sangat kompleks, interpretabilitas terbatas pada model *ensemble*, kurangnya evaluasi pada subtype tumor spesifik, serta potensi *overfitting* pada dataset publik yang relatif kecil tanpa augmentasi lanjutan.

Penelitian oleh Jabbar et al., (2023) bertujuan untuk mengembangkan model otomatis berbasis *deep learning* untuk mendeteksi dan mengklasifikasikan tumor otak dengan segmentasi *multi-grade*. Penelitian ini menggunakan data dari dataset Brats-2020 dan Brats-2019 yang berisi citra MRI berkualitas tinggi dari pasien glioma, dengan pembagian data 80% untuk pelatihan dan 20% untuk validasi. Model yang diusulkan yaitu Caps-VGGNet, dimana model ini dipilih untuk mengatasi kebutuhan dataset yang besar pada model CNN konvensional. Model ini mengintegrasikan CapsNet untuk invariansi rotasi dan VGGNet untuk ekstraksi fitur otomatis. Evaluasi menunjukkan akurasi sebesar 0.99, spesifisitas 0.99, dan sensitivitas 0.98 pada dataset Brats-2020. Kelebihan model ini meliputi performa tinggi, efisiensi komputasi, dan kemampuan segmentasi presisi tinggi, namun

keterbatasannya terletak pada interpretabilitas terbatas, kurangnya evaluasi pada dataset lebih besar, dan potensi tantangan generalisasi ke kondisi medis lain.

Penelitian yang dilakukan oleh Asiri et al., (2024) bertujuan untuk mengembangkan pendekatan berbasis dua modul untuk mendeteksi tumor otak secara optimal melalui peningkatan citra MRI dan klasifikasi tumor. Penelitian ini memanfaatkan dataset CE-MRI yang terdiri dari 3064 citra dari 233 pasien, mencakup tumor meningioma, glioma, dan pituitary, dengan resolusi 512×512 piksel, yang dibagi 70% untuk pelatihan dan 30% untuk pengujian. Metode yang diusulkan meliputi modul pertama untuk peningkatan citra menggunakan teknik seperti filter *Wiener* adaptif, jaringan saraf RBF, dan analisis komponen independen (ICA1). Modul kedua yang diusulkan pada penelitian ini digunakan untuk segmentasi dan klasifikasi tumor menggunakan SVM. Evaluasi model tersebut menunjukkan akurasi sebesar 98.9%, sensitivitas 0.99, spesifisitas 0.99, dan *Dice Score* 0.981.

Penelitian yang dilakukan oleh Santoso et al., (2024) bertujuan untuk mengembangkan metode berbasis *deep learning* guna mendeteksi tumor otak pada citra MRI dengan akurasi tinggi, mengatasi tantangan kompleksitas struktur otak dan variasi bentuk serta ukuran tumor. Penelitian ini memanfaatkan empat dataset MRI dari Kaggle dan situs lain, yang terdiri dari citra tumor (termasuk *glioma*, *meningioma*, dan *pituitary*) dan non-tumor. Dataset tersebut dibagi menjadi data pelatihan dengan 6123 citra, data pengujian 1 dengan 394 citra, dan data pengujian 2 dengan 1311 citra. Metode yang diusulkan meliputi model *Convolutional Neural Network* (CNN) bernama tpCNN + voting dengan tiga jalur jaringan paralel

menggunakan kernel ukuran berbeda (3×3 , 5×5 , 7×7) dan metode voting baru dengan kondisi jika satu model mendeteksi tumor, hasil akhir menyatakan adanya tumor. Evaluasi model ini menunjukkan akurasi terbaik sebesar 99.24% pada pengujian 1 dan 99.92% pada pengujian 2.

Rangkuman hasil evaluasi model dari penelitian sebelumnya sebagaimana disajikan pada Tabel 2.1 berikut :

Tabel 2. 1 Rangkuman Hasil Evaluasi Model

Peneliti	Metode/Model	Hasil
(Naseer et al., 2021)	<i>Convolutional Neural Network (CNN)</i>	Akurasi : 98.8%
(Shah et al., 2022)	Fine-tuning EfficientNet-B0	Akurasi : 98.87% Presisi : 99.4% AUC : 98.8% Sensitivitas : 99.5% Spesifitas : 99.2%
(Anantharajan et al., 2024)	<i>Ensemble Deep Neural Support Vector Machine (EDN-SVM)</i>	Akurasi : 97.93% Sensitivitas : 92% Spesifitas : 98%
(Jabbar et al., 2023)	Caps-VGGNet (CapsNet + VGGNet)	Akurasi : 99% Recall : 98% Sensitivitas : 98% Spesifitas : 99%
(Asiri et al., 2024)	2 modul (Wiener, RBF, ICA1, SVM)	Akurasi : 98.9% Sensitivitas : 99% Spesifitas : 99% Dice Score : 98.1%
(Santoso et al., 2024)	tpCNN + Voting	Akurasi Uji 1 : 99.24% Akurasi Uji 2 : 99.92%

Secara keseluruhan, semua model pada penelitian tersebut telah menunjukkan akurasi yang tinggi dalam melakukan deteksi pada tumor otak. Pada penelitian ini menggunakan metode *Neural Network* untuk deteksi tumor otak pada citra MRI. Model ini dipilih karena arsitekturnya yang sederhana, dengan kompleksitas rendah, sehingga mudah dilatih dan efisien secara komputasi. Melalui

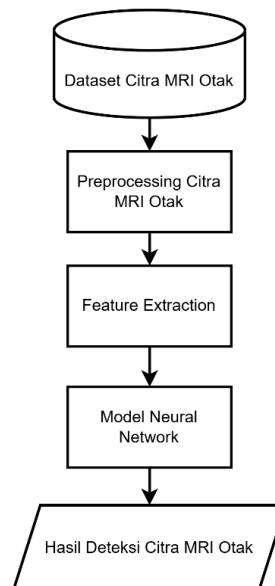
pendekatan ini, diharapkan proses deteksi tumor otak pada citra MRI dapat lebih akurat dan efisien.

BAB III

DESAIN DAN IMPLEMENTASI SISTEM

3.1 Desain Sistem

Desain sistem dirancang secara sistematis untuk memastikan setiap tahapan pada penelitian dapat berjalan dengan efisien dan memberikan hasil yang akurat. Gambar 3.1 berikut merupakan desain sistem yang dirancang untuk membangun sistem deteksi tumor otak menggunakan algoritma *Neural Network* .



Gambar 3. 1 Desain Sistem

3.1.1 *Input Image*

Citra MRI otak dari dataset yang sebelumnya telah didapatkan akan digunakan sebagai masukan utama pada sistem yang akan dibuat. Data citra MRI memiliki berbagai macam format diantaranya yaitu format *Joint Photographic Group* (*.jpg), *Joint Photographic Experts Group* (*.jpeg), *Portable Network Graphics* (*.png), dan *Tagged Image Format* (*.tif). Semua data masukan yang

digunakan telah memiliki label *Ground truth* yang diperoleh dari interpretasi medis yang divalidasi, sehingga menjadikan data ini representative untuk pelatihan dan evaluasi model deteksi tumor otak.

3.1.2 Data Preprocessing

Sebelum menjadi masukan utama dalam model *neural network* yang dibangun, semua citra MRI harus melewati beberapa tahap *preprocessing* untuk memastikan konsistensi dan kompatibilitas dengan arsitektur yang digunakan. Tahap pertama yaitu berupa *resizing* untuk menyesuaikan ukuran setiap citra dari ukuran aslinya dengan mengubah semua ukuran citra menjadi dimensi yang seragam yaitu 128 x 128 piksel.

Tahap kedua dari *preprocessing* ini yaitu tahap normalisasi. Tahap ini dilakukan dengan mengubah nilai intensitas piksel setiap citra, yang semula berada dalam rentang [0, 255] dinormalisasi ke dalam rentang [0, 1]. Proses ini dilakukan dengan membagi nilai piksel dengan 255. Tahap normalisasi ini dapat membantu mempercepat proses konvergensi selama pelatihan dan menjaga stabilitas gradien (Santoso et al., 2024).

$$X_n = \frac{X - X_{min}}{X_{max} - X_{min}} \quad (3.1)$$

Keterangan:

X_n = nilai data setelah proses normalisasi (dalam rentang 0-1)

X = nilai data asli.

X_{max} = nilai maksimum dalam dataset.

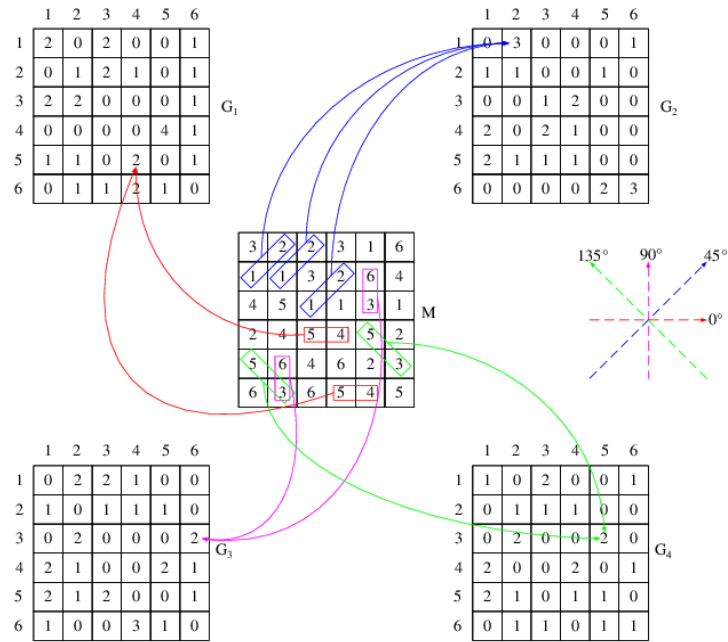
X_{min} = nilai minimum dalam dataset.

3.1.3 *Feature Extraction*

Feature Extraction yang digunakan pada penelitian ini adalah *Gray-Level Co-occurrence Matrix (GLCM)*, sebuah metode statistik yang efektif untuk mengekstraksi fitur tekstur dari citra MRI otak. GLCM bekerja dengan menghitung frekuensi kemunculan pasangan nilai intensitas piksel pada jarak dan sudut tertentu, sehingga merepresentasikan hubungan spasial antar piksel secara akurat (Dheepak et al., 2024). Metode ini dipilih karena kemampuannya dalam menangkap informasi tekstur yang relevan untuk deteksi tumor otak tanpa memerlukan pemrosesan kompleks seperti pada *deep learning*, serta menghasilkan fitur yang mudah diinterpretasi dalam medis.

Untuk menangkap heterogenitas tekstur secara lebih halus dan mengurangi pengaruh *noise* pada citra MRI, dilakukan *preprocessing* dengan menerapkan *uniform filter (mean filter)* berukuran 5×5 piksel sebelum ekstraksi fitur. *Uniform filter* bekerja dengan mengganti nilai setiap piksel pusat dengan rata-rata intensitas dari 25 piksel di sekitarnya (kernel 5×5), sehingga menghasilkan citra yang lebih halus, mengurangi variasi acak akibat *noise* pemindaian, dan mempertahankan struktur utama seperti batas tumor tanpa menghilangkan informasi tekstur esensial. Setelah proses filtering selesai, GLCM dihitung secara global pada citra yang telah difilter dengan empat arah (0° , 45° , 90° , dan 135°) serta jarak antar piksel 1. Dari setiap arah dan dua kondisi simetri (*symmetric* dan *asymmetric*), diekstrak 6 fitur Haralick yaitu *Contrast*, *Dissimilarity*, *Homogeneity*, *Energy*, *Correlation*, dan *Entropy*, menghasilkan vektor fitur akhir berukuran 48 dimensi yang mewakili karakteristik tekstur keseluruhan citra. Pendekatan ini memastikan representasi

tekstur yang lebih stabil, *robust* terhadap *noise*, dan tetap efektif dalam membedakan pola jaringan sehat dengan area tumor, sesuai dengan kebutuhan diagnostik klinis. Proses ekstraksi GLCM sebagaimana ditunjukkan pada Gambar 3.5, yang diadaptasi dari (Vyas et al., 2017).



Gambar 3. 2 Visualisasi Formasi Matrix GLCM

Contrast digunakan untuk mengukur perbedaan intensitas lokal pada citra MRI otak. Nilai tinggi menunjukkan tekstur kasar dengan banyak variasi intensitas, yang umumnya mengindikasikan adanya ketidakaturan jaringan seperti pada area tumor. Nilai rendah menunjukkan tekstur halus dan seragam yang biasanya ditemukan pada jaringan otak normal. Dalam konteks deteksi tumor otak, fitur ini sangat penting untuk membedakan antara jaringan tumor yang memiliki densitas tidak homogen dengan jaringan otak sehat yang cenderung lebih *uniform* (Fikriah et al., 2024).

$$Contrast = \sum_i \sum_j (i - j)^2 p_{(i,j)} \quad (3.2)$$

Keterangan:

i = intensitas piksel referensi (baris)

j = intensitas piksel tetangga (kolom)

$p_{(i,j)}$ = probabilitas kemunculan pasangan intensitas (i,j) yang sudah dinormalisasi

Dissimilarity merupakan ukuran variasi intensitas yang lebih sensitif terhadap perbedaan kecil dibandingkan *contrast*. Nilai tinggi mengindikasikan variasi intensitas yang besar pada citra MRI. Fitur ini efektif dalam mendeteksi perubahan halus pada tekstur jaringan otak yang mungkin merupakan tanda awal pertumbuhan tumor. *Dissimilarity* membantu sistem dalam mengidentifikasi transisi bertahap antara jaringan sehat dan jaringan tumor, terutama pada kasus tumor dengan batas yang tidak jelas.

$$Dissimilarity = \sum_i \sum_j p(i,j) |i - j| \quad (3.3)$$

Keterangan:

i = intensitas piksel referensi (baris)

j = intensitas piksel tetangga (kolom)

$p_{(i,j)}$ = probabilitas kemunculan pasangan intensitas (i,j) yang sudah dinormalisasi

Homogeneity mengukur kedekatan distribusi elemen GLCM terhadap diagonal matriksnya. Nilai tinggi (mendekati 1) menunjukkan tekstur yang homogen dan seragam, karakteristik dari jaringan otak normal. Nilai rendah mengindikasikan tekstur heterogen yang sering ditemukan pada area tumor otak karena adanya sel-sel abnormal dengan densitas bervariasi. Fitur ini sangat berguna untuk membedakan *Region of Interest (ROI)* yang mengandung tumor dari jaringan otak sehat, karena tumor cenderung memiliki struktur internal yang tidak teratur.

$$Homogeneity = \sum_i \sum_j \frac{p(i,j)}{1 + |i - j|} \quad (3.4)$$

Keterangan:

i = intensitas piksel referensi (baris)

j = intensitas piksel tetangga (kolom)

$p_{(i,j)}$ = probabilitas kemunculan pasangan intensitas (i,j) yang sudah dinormalisasi

Correlation mengukur hubungan linear antara piksel referensi dan piksel tetangganya pada citra MRI otak. Nilai korelasi yang tinggi menunjukkan adanya pola tekstur yang terkorelasi dan terorganisir dengan baik, umumnya ditemukan pada jaringan otak sehat. Nilai korelasi yang rendah atau mendekati nol mengindikasikan tekstur acak dan tidak beraturan, yang sering dijumpai pada area tumor karena pertumbuhan sel yang tidak teratur. Fitur ini membantu sistem dalam mengenali pola spasial yang konsisten atau tidak konsisten pada jaringan otak, sehingga dapat digunakan untuk membedakan antara struktur normal dan patologis.

$$Correlation = \sum_i \sum_j \frac{[(i - \mu_i)(j - \mu_j) \times p(i,j)]}{(\sigma_i \times \sigma_j)} \quad (3.5)$$

Keterangan:

i = intensitas piksel referensi (baris)

j = intensitas piksel tetangga (kolom)

μ_i = mean / rata-rata untuk i

μ_j = mean / rata-rata untuk j

σ_i = standar deviasi untuk i

σ_j = standar deviasi untuk j

$p_{(i,j)}$ = probabilitas kemunculan pasangan intensitas (i,j) yang sudah dinormalisasi

Entropy adalah salah satu fitur utama dalam *Gray-Level Co-occurrence Matrix (GLCM)* yang digunakan untuk mengukur ketidakteraturan dan kompleksitas distribusi tingkat keabuan (intensitas piksel) pada citra MRI otak. Nilai tinggi dari *entropy* menunjukkan tekstur yang sangat tidak seragam, heterogen, dan memiliki variasi intensitas yang besar, yang umumnya menjadi ciri khas jaringan tumor akibat proliferasi sel abnormal yang tidak terkontrol.

Sebaliknya, nilai *entropy* yang rendah mengindikasikan tekstur yang teratur, seragam, dan konsisten, yang merupakan karakteristik dari jaringan otak normal dengan distribusi intensitas yang stabil. Fitur ini berfungsi sebagai indikator kuat untuk membedakan antara pola tekstur jaringan sehat dan jaringan yang mengalami gangguan patologis, sehingga sangat efektif dalam mengidentifikasi keberadaan anomali struktural pada citra MRI otak.

$$Entropy = \sum_i \sum_j p_{(i,j)} \times \log_2\{p_{(i,j)}\} \quad (3.6)$$

Keterangan:

i = intensitas piksel referensi (baris)

j = intensitas piksel tetangga (kolom)

$p_{(i,j)}$ = probabilitas kemunculan pasangan intensitas (i,j) yang sudah dinormalisasi

Energy mengukur keseragaman dan uniformitas tekstur pada citra MRI otak. Nilai tinggi (mendekati 1) menunjukkan tekstur yang sangat teratur dengan pola repetitif, umumnya ditemukan pada jaringan otak normal yang memiliki struktur konsisten. Nilai rendah mengindikasikan tekstur tidak teratur dan acak, yang merupakan ciri khas dari jaringan tumor. Fitur ini membantu sistem dalam mengidentifikasi area dengan pola tekstur yang menyimpang dari normalitas jaringan otak, sehingga dapat digunakan sebagai indikator potensial keberadaan tumor.

$$Energy = \sum_i \sum_j \{p_{(i,j)}\}^2 \quad (3.7)$$

Keterangan:

i = intensitas piksel referensi (baris)

j = intensitas piksel tetangga (kolom)

$p_{(i,j)}$ = probabilitas kemunculan pasangan intensitas (i,j) yang sudah dinormalisasi

Keenam fitur tekstur GLCM tersebut diekstraksi dari setiap arah, yaitu 0° (horizontal), 45° (diagonal kanan atas), 90° (vertikal), dan 135° (diagonal kiri atas) dengan jarak $d=1$ piksel, untuk membentuk vektor fitur berukuran 48 dimensi yang merepresentasikan karakteristik tekstur jaringan secara komprehensif. Kombinasi 48 fitur ini memberikan informasi komprehensif tentang pola, keseragaman, dan variasi intensitas pada citra dari berbagai orientasi, yang menjadi representasi numerik penting untuk membedakan antara jaringan tumor dan jaringan normal.

Sebelum digunakan sebagai *input* untuk model *neural network*, vektor fitur ini terlebih dahulu dinormalisasi menggunakan untuk memastikan semua fitur berada dalam rentang yang sama yaitu rentang $[0,1]$, sehingga tidak ada fitur yang mendominasi proses pembelajaran. Vektor fitur yang telah dinormalisasi kemudian digunakan sebagai *input* untuk model *neural network*, yang akan mempelajari pola kompleks dan hubungan non-linear antar fitur untuk melakukan deteksi tumor otak secara otomatis dengan tingkat akurasi yang tinggi.

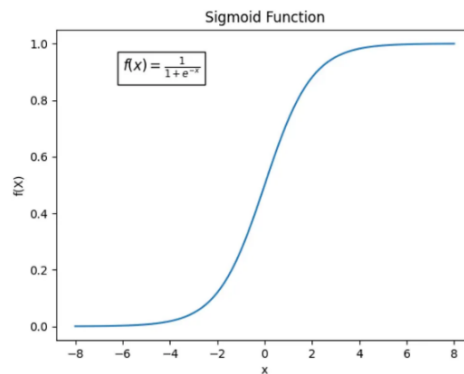
3.1.4 *Neural Network* (NN)

Neural Network (NN) merupakan model komputasi yang meniru struktur dan fungsi jaringan saraf otak manusia. Model ini terdiri dari lapisan-lapisan neuron buatan yang saling terhubung dan mampu mempelajari pola kompleks dari data melalui proses pelatihan. Secara arsitektural, model *neural network* memiliki tiga komponen utama yang bekerja secara berurutan, yaitu *input layer* untuk menerima data masukan dari lingkungan eksternal, *hidden layer* untuk pemrosesan dan ekstraksi fitur serta hubungan non-linear dalam data, serta *output layer* untuk

menghasilkan prediksi akhir serta mengubah representasi internal menjadi bentuk yang dapat dimengerti sistem.

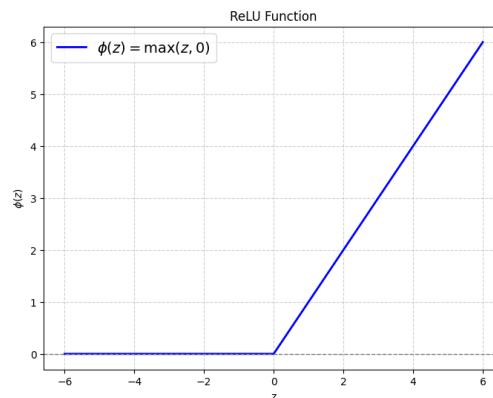
Dalam *neural network* terdapat proses utama yang disebut dengan *forward propagation*, Dimana data *input* diteruskan secara berurutan dari *input layer*, melalui *hidden layer*, hingga ke *output layer* untuk menghasilkan prediksi akhir (Park et al., 2024). Pada setiap *layer*, data diolah dengan bobot dan fungsi aktivasi, sehingga jaringan dapat memetakan *input* menjadi *output* yang diinginkan. Tanpa fungsi aktivasi, model *neural network* hanya akan menjadi kombinasi linear sederhana, sehingga tidak mampu mempelajari pola yang kompleks dalam data. Fungsi aktivasi memberikan sifat non-linear pada jaringan, yang memungkinkan model untuk mempelajari hubungan yang rumit antara *input* dan *output*.

Fungsi sigmoid merupakan fungsi aktivasi non-linier yang bekerja dengan cara mengubah *input* menjadi *output* dalam rentang 0 hingga 1 (Zhang et al., 2025). Karakteristik penting fungsi sigmoid adalah bahwa nilai *input* yang sangat negatif akan menghasilkan *output* mendekati 0, nilai sekitar 0 akan menghasilkan *output* sekitar 0.5, dan nilai yang sangat positif akan menghasilkan *output* mendekati 1. Gradien yang halus dari fungsi ini memungkinkan pembelajaran yang stabil selama proses *training*. Pada penelitian ini, fungsi sigmoid diterapkan pada model *neural network* dengan satu *hidden layer* serta pada *hidden layer* kedua dan lapisan *output* pada model *neural network* dengan dua *hidden layer*. Grafik fungsi aktivasi sigmoid sebagaimana yang tertera pada Gambar 3.3.



Gambar 3. 3 Grafik Fungsi Aktivasi Sigmoid

Fungsi aktivasi *ReLU* (*Rectified Linear Unit*) menjadi pilihan utama pada arsitektur *deep learning* modern karena sederhana, efisien secara komputasi, dan efektif mengatasi masalah *vanishing gradient* yang kerap muncul pada sigmoid. Secara matematis, ReLU didefinisikan sebagai $\text{ReLU}(x) = \max(0, x)$, sehingga nilai *output* sama dengan *input* jika $x > 0$ dan nol jika $x \leq 0$. Fungsi ini mempertahankan gradien sebesar 1 pada sisi positif, menghasilkan aktivasi sparse yang meningkatkan efisiensi pemrosesan, serta sering kali memberikan performa lebih baik pada berbagai tugas seperti klasifikasi citra. Meskipun demikian, ReLU memiliki keterbatasan berupa fenomena “*dying ReLU*”, yaitu neuron yang menjadi permanen tidak aktif jika *input* selalu negatif. Dalam penelitian ini, fungsi ReLU diterapkan pada *hidden layer* pertama pada *neural network* dengan dua *hidden layer* untuk memanfaatkan keunggulan non-linearitas dan stabilitas gradiennya. Grafik fungsi aktivasi ReLU sebagaimana yang tertera pada Gambar 3.4.



Gambar 3. 4 Grafik Fungsi Aktivasi ReLU

Setelah proses *forward propagation* menghasilkan prediksi, langkah selanjutnya yaitu melakukan proses *backward propagation* untuk memperbaiki kesalahan pada model. *Backward propagation* adalah proses pembelajaran pada sebuah model, dimana *error* berupa selisih antara *output* prediksi dan target dihitung, lalu *error* ini disebarakan dengan cara mundur dari *output layer* ke *input layer* untuk memperbarui bobot jaringan (Wright et al., 2022). *Backward propagation* memungkinkan model *neural network* belajar dari kesalahan dan meningkatkan akurasi prediksi dengan mengoptimalkan bobot secara bertahap sehingga model dapat menyesuaikan parameter secara bertahap menuju konvergensi.

Pada penelitian ini, diterapkan dua jenis arsitektur *neural network*, yaitu *neural network* dengan satu *hidden layer* yang menggunakan fungsi aktivasi sigmoid pada *hidden layer* dan *output layer*, serta *neural network* dengan dua *hidden layer* yang menggunakan fungsi sigmoid pada *hidden layer* kedua dan lapisan *output*, serta fungsi ReLU pada *hidden layer* pertama. Kedua varian tersebut dirancang untuk membandingkan kemampuan representasi fitur dan performa

klasifikasi dalam mendeteksi keberadaan tumor otak pada citra MRI, dengan memanfaatkan kekuatan *forward propagation* dan *backward propagation* serta pemilihan fungsi aktivasi yang sesuai untuk menangkap pola kompleks pada data.

3.1.5 *Neural Network 1 Hidden layer*

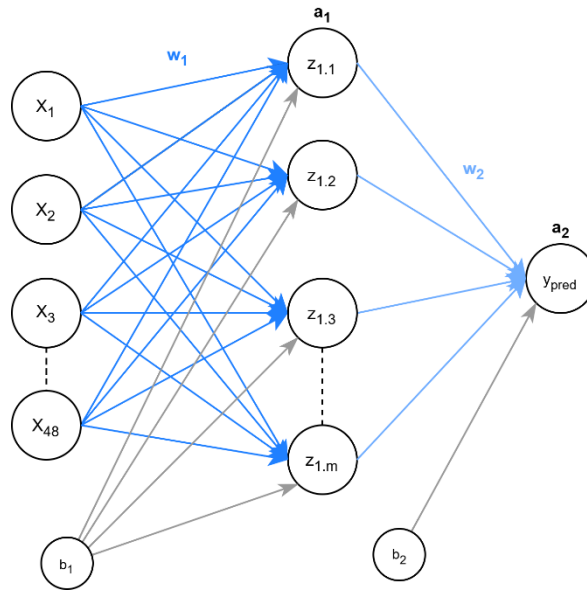
Implementasi model *Neural Network* (NN) dengan satu *hidden layer* diwujudkan melalui kelas *NeuralNetworkV1*. Model ini dirancang untuk klasifikasi biner deteksi tumor otak dengan *input* 24 fitur tekstur. Berikut ini proses yang terjadi dalam penerapan *neural network* dengan 1 *hidden layer*:

1. *Data Input* dan Inisialisasi

Data input yang digunakan merupakan hasil ekstraksi fitur GLCM yang diinisialisasikan dengan variabel (X). Selanjutnya inisialisasi nilai bobot (W_1, W_2) dan bias (b_1, b_2) dilakukan dengan menggunakan fungsi *Xavier/Glorot Initialization* ($np.sqrt(\frac{1.0}{N})$) untuk memastikan stabilitas gradien pada lapisan sigmoid.

2. *Forward Propagation*

Proses *forward propagation*, sebagai tahap utama pemetaan *input* ke *output*, diilustrasikan pada Gambar 3.5 yang menunjukkan langkah-langkah perhitungan nilai mentah dan aktivasi secara berurutan pada masing-masing lapisan.



Gambar 3. 5 Skema *Forward Propagation Neural Network 1 Hidden Layer*

Berikut ini akan diuraikan proses *forward propagation* :

- Perhitungan *Hidden layer* dilakukan dengan menghitung nilai mentah (*self.z1*) di *hidden layer* dengan mengalikan *batch input* (X) dengan bobot (W_1) dan ditambah bias (b_1).

$$z_1 = XW_1 + b_1 \quad (3.8)$$

Keterangan:

z_1 = hasil perhitungan node *hidden layer* 1

X = node *input*

W_1 = bobot yang menghubungkan *input layer* dengan *hidden layer*

b_1 = bias yang menghubungkan *input layer* dengan *hidden layer*

- Hasil perhitungan nilai pada (z_1) kemudian diaktivasi menggunakan Fungsi Sigmoid (*self.sigmoid*). Hasil aktivasi disimpan sebagai (a_1) atau (*self.a1*).

$$a_1 = \frac{1}{1 + e^{-z_1}} \quad (3.9)$$

Keterangan:

a_1 = output fungsi sigmoid pada *hidden layer* 1

e = basis logaritma natural (sekitar 2.71828), digunakan untuk menghitung eksponensial negatif dari z

z_1 = hasil perhitungan node *hidden layer* 1

- c. Hasil aktivasi (a_1) dikalikan dengan bobot (W_2) yang menyambungkan *hidden layer* ke *output layer*, kemudian ditambah bias (b_2) untuk mendapatkan nilai mentah (z_2) atau (*self.z1*).

$$z_2 = a_1 W_2 + b_2 \quad (3.10)$$

Keterangan:

z_2 = hasil perhitungan *output layer*

a_1 = hasil aktivasi pada node *hidden layer*

W_2 = bobot yang menghubungkan *hidden layer* dengan *output layer*

b_2 = bias yang menghubungkan *hidden layer* dengan *output layer* indeks

- d. Nilai (z_2) diaktivasi kembali menggunakan Fungsi Sigmoid untuk mendapatkan probabilitas prediksi akhir y_{pred} atau (*self.a1*).

$$y_{pred} = a_2 = \frac{1}{1 + e^{-z_2}} \quad (3.11)$$

Keterangan:

y_{pred} = output model

a_2 = hasil aktivasi pada node *output layer*

e = basis logaritma natural (sekitar 2.71828), digunakan untuk menghitung eksponensial negatif dari z

z_2 = hasil perhitungan *output layer*

3. Loss Function

Setelah didapatkan probabilitas prediksi (y_{pred}), nilai *loss* (L) dihitung menggunakan fungsi *Weighted Binary Cross-Entropy* untuk mengukur perbedaan antara hasil nilai prediksi (y_{pred}) dan *ground truth* (y). Variabel (s_1) merupakan *sample weight* yang diterapkan pada setiap sampel dengan nilai 2.0 untuk tumor, serta nilai 1.0 untuk sehat.

$$L = -\frac{1}{m} \sum_{i=1}^m s_1[y \log y_pred + (1 - y) \log(1 - y_pred)] \quad (3.12)$$

Keterangan :

L = nilai fungsi *loss*, yang mengukur kesalahan prediksi model terhadap label sebenarnya dan menjadi target untuk diminimalkan selama pelatihan.

m = jumlah total sampel dalam dataset, yang menentukan ukuran rata-rata *loss*.

y = label sebenarnya untuk sampel ke- i , dengan nilai 0 (sehat) atau 1 (tumor).

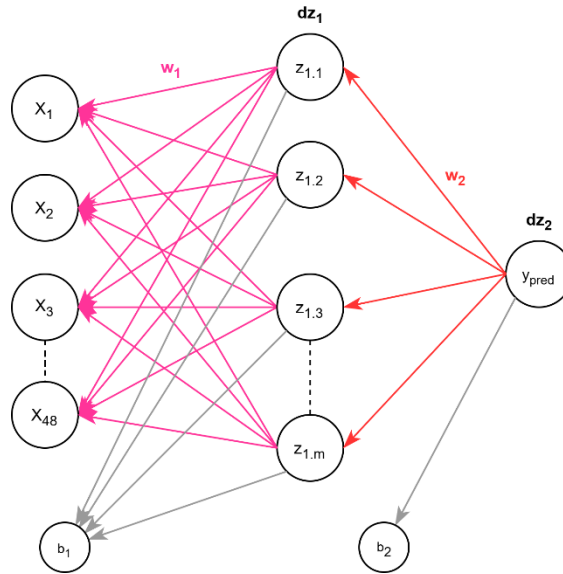
y_pred = probabilitas prediksi model untuk sampel ke- i , yang dihasilkan oleh fungsi sigmoid di *output layer*.

$\sum_{i=1}^m$ = operasi penjumlahan untuk semua sampel dari 1 hingga m .

Proses *training* model *neural network* ini akan berhenti jika jumlah *epoch* maksimum tercapai atau kondisi *early stopping* dengan nilai *patience* 120 terpenuhi.

4. *Backward Propagation*

Backward propagation merupakan proses penyebaran *error* secara mundur untuk memperbarui bobot dan bias pada setiap lapisan jaringan. Proses ini dimulai setelah *forward propagation* menghasilkan prediksi dan *loss function* dihitung. Skema *backward propagation* sebagaimana yang tertera pada Gambar 3.6.



Gambar 3. 6 Skema *Backward Propagation Neural Network 1 Hidden Layer*

Berikut ini proses yang terjadi pada *backward propagation*

- Menghitung error pada *output layer* (dz_2) dengan cara mengurangi nilai probabilitas prediksi (y_{pred}) dengan nilai *grund truth* (y).

$$dz_2 = y_{pred} - y \quad (3.13)$$

Keterangan:

dz_2 = nilai *error* pada *output layer*

y_{pred} = nilai probabilitas prediksi

y = nilai label asli (*ground truth*)

- Error output layer* (dz_2) dipropagasi mundur untuk menghitung nilai *error* pada *hidden layer* (dz_1), disesuaikan dengan kontribusi (W_2) dan sensitivitas aktivasi sigmoid. Hal ini dilakukan dengan cara mengalikan error (dz_2) dengan transpose bobot (W_2) dan turunan aktivasi sigmoid yaitu ($a1(1 - a1)$).

$$da_1 = dz_2 \cdot W_2^T \quad (3.14)$$

$$dz_1 = da_1 \cdot a1(1 - a1) \quad (3.15)$$

Keterangan:

da_1 = gradien *loss* terhadap aktivasi *hidden layer*

dz_2 = nilai *error* pada *output layer*

w_2^T = nilai transpose matriks bobot antara *hidden layer* dan *output layer*

dz_1 = nilai *error* pada *hidden layer*

a_1 = *output* fungsi sigmoid pada *hidden layer 1*

- c. Perhitungan gradien bobot (dw_2) dan bias (db_2) dilakukan menggunakan *output* fungsi sigmoid pada *hidden layer 1* dan nilai *error* pada *output layer*.

$$dw_2 = \frac{1}{m} \cdot a_1^T \cdot dz_2 \quad (3.16)$$

$$db_2 = \frac{1}{m} \cdot \sum_{batch} dz_2 \quad (3.17)$$

Keterangan:

dw_2 = gradien *loss* terhadap bobot w_2

a_1^T = nilai transpose matriks pada hasil aktivasi sigmoid pada *hidden layer*

dz_2 = nilai *error* pada *output layer*

m = jumlah total sampel dalam dataset

db_2 = gradien *loss* terhadap bias b_2

- d. Perhitungan gradien bobot (dw_1), dan bias (db_1) dilakukan menggunakan *input* (X) dan nilai *error* yang telah dipropagasi (dz_1).

$$dw_1 = \frac{1}{m} \cdot X^T \cdot dz_1 \quad (3.18)$$

$$db_1 = \frac{1}{m} \cdot \sum_{batch} dz_1 \quad (3.19)$$

Keterangan:

dw_1 = gradien *loss* terhadap bobot w_1

dz_1 = nilai *error* pada *hidden layer*

X^T = nilai transpose matriks pada *input layer*

m = jumlah total sampel dalam dataset

db_1 = gradien *loss* terhadap bias b_1

- e. Terakhir, pembaruan nilai bobot (W_1, W_2) dan bias (b_1, b_2) dengan cara mengalikan (*learning_rate*) dengan gradien masing-masing.

$$W_1 = learning_rate \times dw_1 \quad (3.20)$$

$$b_1 = learning_rate \times db_1 \quad (3.21)$$

$$W_2 = learning_rate \times dw_2 \quad (3.22)$$

$$b_2 = learning_rate \times db_2 \quad (3.23)$$

Keterangan:

W_1 = nilai bobot w_1 baru setelah pembaruan

b_1 = nilai bias b_1 baru setelah pembaruan

W_2 = nilai bobot w_2 baru setelah pembaruan

b_2 = nilai bias b_2 baru setelah pembaruan

5. Prediksi

Output probabilitas (y_{pred}) dikonversi menjadi label kelas biner (0 atau 1)

menggunakan threshold 0.5.

$$Kelas = \begin{cases} 1 & \text{jika } y_{pred} \geq 0.5 \text{ (Tumor)} \\ 0 & \text{jika } y_{pred} < 0.5 \text{ (No Tumor)} \end{cases} \quad (3.24)$$

3.1.6 Neural Network 2 Hidden layer

Implementasi model *Neural Network* (NN) dengan dua *hidden layer* diwujudkan melalui kelas *NeuralNetworkV2*. Model ini dirancang untuk klasifikasi biner deteksi tumor otak dengan *input* 48 fitur tekstur.

Berikut ini proses yang terjadi dalam penerapan *Neural Network* dengan 2 *hidden layer*:

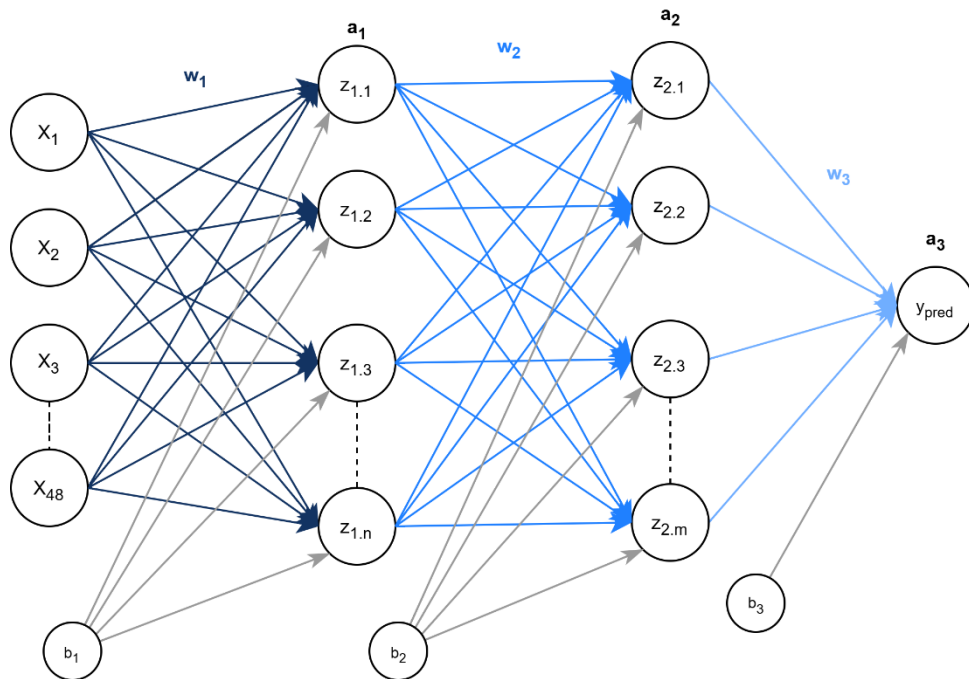
1. Data Input dan Inisialisasi

Data input yang digunakan merupakan hasil ekstraksi fitur GLCM yang diinisialisasikan dengan variabel (X). Selanjutnya inisialisasi nilai bobot (W_1, W_2) dilakukan dengan menggunakan fungsi *He initialization* ($np.sqrt(\frac{2.0}{N})$) untuk

fungsi aktivasi ReLU serta inisialisasi nilai bobot (W_3) menggunakan *Xavier/Glorot Initialization* ($np.sqrt(\frac{1.0}{N})$) untuk fungsi aktivasi sigmoid.

2. Forward propagation

Proses *forward propagation*, sebagai tahap utama pemetaan *input* ke *output*, diilustrasikan pada Gambar 3.7 yang menunjukkan langkah-langkah perhitungan nilai mentah dan aktivasi secara berurutan pada masing-masing lapisan.



Gambar 3. 7 Skema *Forward Propagation Neural Network 2 Hidden Layer*

Berikut ini akan diuraikan proses *forward propagation* :

- a. Perhitungan dan Aktivasi *hidden layer* 1 menggunakan ReLU dilakukan dengan memproses nilai *input* (X) secara linier menjadi (z_1), kemudian diaktivasi menggunakan fungsi ReLU.

$$\text{ReLU}(z) = \max(0, z) \quad (3.25)$$

$$z_1 = XW_1 + b_1 \quad (3.26)$$

$$a_1 = \text{ReLU}(z_1) \quad (3.27)$$

Keterangan:

$\text{ReLU}(z)$ = hasil aplikasi fungsi aktivasi ReLu terhadap nilai mentah z

z_1 = hasil perhitungan node *hidden layer* 1

X = node *input*

W_1 = bobot yang menghubungkan *input layer* dengan *hidden layer* 1

b_1 = bias yang menghubungkan *input layer* dengan *hidden layer* 1

a_1 = hasil penerapan fungsi aktivasi ReLu pada *hidden layer* 1

Pada lapisan ini fungsi aktivasi ReLU dipilih karena kemampuannya mengatasi masalah *vanishing gradient* dan meningkatkan efisiensi komputasi, yang sesuai untuk lapisan yang lebih dalam.

- b. Perhitungan dan Aktivasi *Hidden layer* 2 menggunakan Sigmoid dilakukan dengan cara memproses *output* pada *hidden layer* 1 (a_1) secara linier menjadi (z_2) , kemudian diaktivasi menggunakan sigmoid sehingga menghasilkan a_2 .

$$z_2 = a_1 W_2 + b_2 \quad (3.28)$$

$$a_2 = \text{Sigmoid}(z_2) \quad (3.29)$$

Keterangan:

z_2 = hasil perhitungan node *hidden layer* 2

a_1 = hasil perhitungan *hidden layer* 1

W_2 = bobot yang menghubungkan *hidden layer* 1 dengan *hidden layer* 2

b_2 = bias yang menghubungkan *hidden layer* 1 dengan *hidden layer* 2

a_2 = hasil penerapan fungsi aktivasi Sigmoid pada *hidden layer* 2

- c. Perhitungan pada *output layer* dilakukan dengan memproses *output* pada *hidden layer* 2 (a_2) secara linier menjadi (z_3) , kemudia nilai (z_3) diaktivasi kembali menggunakan Fungsi Sigmoid untuk mendapatkan probabilitas prediksi akhir y_{pred} .

$$z_3 = a_2 W_3 + b_3 \quad (3.30)$$

$$y_{pred} = \text{Sigmoid}(z_3) \quad (3.31)$$

Keterangan:

z_3 = hasil perhitungan pada *output layer*

a_2 = hasil aktivasi pada node *output layer*

W_3 = bobot yang menghubungkan *hidden layer 2* dengan *output layer*

b_3 = bias yang menghubungkan *hidden layer 2* dengan *output layer*

y_{pred} = *output* model setelah diterapkan fungsi aktivasi sigmoid

3. Loss Function

Setelah didapatkan probabilitas prediksi (y_{pred}), nilai *loss* (L) dihitung menggunakan fungsi *Weighted Binary Cross-Entropy*, sebagaimana pada *neural network 1 hidden layer*, untuk mengukur perbedaan antara hasil nilai prediksi (y_{pred}) dan *ground truth* (y). Variabel (s_1) merupakan *sample weight* yang diterapkan pada setiap sampel dengan nilai 2.0 untuk tumor, serta nilai 1.0 untuk sehat.

$$L = -\frac{1}{m} \sum_{i=1}^m s_1 [y \log y_{pred} + (1 - y) \log(1 - y_{pred})] \quad (3.32)$$

Keterangan :

L = nilai fungsi *loss*, yang mengukur kesalahan prediksi model terhadap label sebenarnya dan menjadi target untuk diminimalkan selama pelatihan.

m = jumlah total sampel dalam dataset, yang menentukan ukuran rata-rata *loss*.

y = label sebenarnya untuk sampel ke- i , dengan nilai 0 (sehat) atau 1 (tumor).

y_{pred} = probabilitas prediksi model untuk sampel ke- i , yang dihasilkan oleh fungsi sigmoid di *output layer*.

$\sum_{i=1}^m$ = operasi penjumlahan untuk semua sampel dari 1 hingga m .

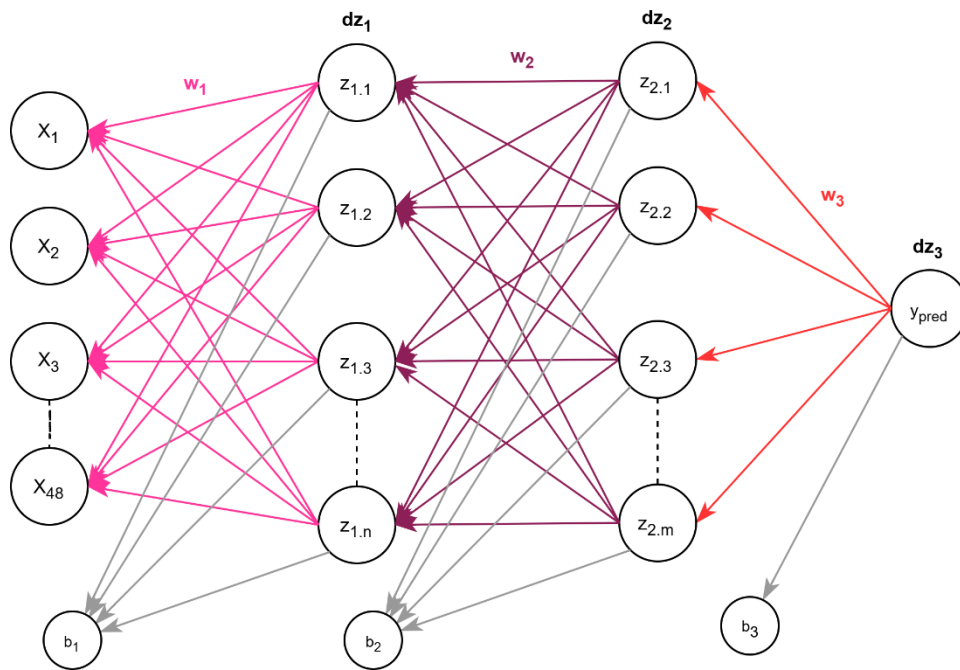
Proses *training* model *neural network* ini akan berhenti jika jumlah *epoch* maksimum tercapai atau kondisi *early stopping* dengan nilai *patience* 100 terpenuhi.

4. Backward Propagation

Backward propagation merupakan proses penyebaran error secara mundur untuk memperbarui bobot dan bias pada setiap lapisan jaringan. Proses ini dimulai

setelah *forward propagation* menghasilkan prediksi dan *loss function* dihitung.

Skema *backward propagation* sebagaimana tertera pada Gambar 3.8.



Gambar 3. 8 Skema *Backward Propagation Neural Network 2 Hidden Layer*

Berikut ini akan proses *backward propagation* :

- Menghitung *error* pada *output layer* (dz_3) dengan cara mengurangi nilai probabilitas prediksi (y_{pred}) dengan nilai *ground truth* (y) untuk menghitung kesalahan awal prediksi.

$$dz_3 = y_{pred} - y \quad (3.33)$$

Keterangan:

dz_3 = nilai *error* pada *output layer*

y_{pred} = nilai probabilitas prediksi

y = nilai label asli (*ground truth*)

- b. Error *output layer* (dz_3) dipropagasi mundur untuk menghitung nilai *error* pada *hidden layer 2* (dz_2) dengan melewati bobot (W_3) dan dikalikan dengan turunan aktivasi sigmoid pada a_2 .

$$da_2 = dz_3 \cdot W_3^T \quad (3.34)$$

$$dz_2 = da_2 \cdot a_2(1 - a_2) \quad (3.35)$$

Keterangan:

da_2 = gradien *loss* terhadap aktivasi *hidden layer 2*

dz_3 = nilai *error* pada *output layer*

w_3^T = nilai transpose matriks bobot antara *hidden layer 2* dan *output layer*

dz_2 = nilai *error* pada *hidden layer 2*

a_2 = output fungsi sigmoid pada *hidden layer 2*

- c. Error pada *hidden layer 2* (dz_2) dipropagasi mundur untuk menghitung nilai *error* pada *hidden layer 1* (dz_1) dengan melewati bobot (W_2) dan dikalikan dengan turunan aktivasi ReLU pada z_1 .

$$da_1 = dz_2 \cdot W_2^T \quad (3.36)$$

$$\frac{d}{dz} \text{ReLU}(z) = \text{relu}_{\text{derivative}} = \begin{cases} 1 & \text{jika } z > 0 \\ 0 & \text{jika } z \leq 0 \end{cases} \quad (3.37)$$

$$dz_1 = da_1 \cdot \text{relu}_{\text{derivative}}(z_1) \quad (3.38)$$

Keterangan:

da_1 = gradien *loss* terhadap aktivasi *hidden layer 1*

dz_2 = nilai *error* pada *hidden layer 2*

w_2^T = nilai transpose matriks bobot antara *hidden layer 1* dan *hidden layer 2*

$\text{relu}_{\text{derivateive}}$ = turunan fungsi aktivasi ReLU

dz_1 = nilai *error* pada *hidden layer 1*

- d. Perhitungan gradien pada *output layer* untuk bobot (dw_3) dan bias (db_3).

$$dw_3 = \frac{1}{m} \cdot a_2^T \cdot dz_3 \quad (3.39)$$

$$db_3 = \frac{1}{m} \cdot \sum_{\text{batch}} dz_3 \quad (3.40)$$

Keterangan:

dw_3 = gradien *loss* terhadap bobot w_3

a_2^T = nilai transpose matriks pada hasil aktivasi sigmoid pada *hidden layer 2*

dz_3 = nilai *error* pada *output layer*

m = jumlah total sampel dalam dataset

db_3 = gradien *loss* terhadap bias b_3

- e. Perhitungan gradien pada *hidden layer 2* untuk bobot (dw_2) dan bias (db_2).

$$dw_2 = \frac{1}{m} \cdot a_1^T \cdot dz_2 \quad (3.41)$$

$$db_2 = \frac{1}{m} \cdot \sum_{batch} dz_2 \quad (3.42)$$

Keterangan:

dw_2 = gradien *loss* terhadap bobot w_2

a_1^T = nilai transpose matriks pada hasil aktivasi sigmoid pada *hidden layer 1*

dz_2 = nilai *error* pada *hidden layer 2*

m = jumlah total sampel dalam dataset

db_2 = gradien *loss* terhadap bias b_2

- f. Perhitungan gradien pada *hidden layer 1* untuk bobot (dw_1) dan bias (db_1).

$$dw_1 = \frac{1}{m} \cdot X^T \cdot dz_1 \quad (3.43)$$

$$db_1 = \frac{1}{m} \cdot \sum_{batch} dz_1 \quad (3.44)$$

Keterangan:

dw_1 = gradien *loss* terhadap bobot w_1

X^T = nilai transpose matriks pada *input layer*

dz_1 = nilai *error* pada *hidden layer 1*

m = jumlah total sampel dalam dataset

db_1 = gradien *loss* terhadap bias b_1

- g. Terakhir, pembaruan nilai bobot (W_1, W_2) dan bias (b_1, b_2) dengan cara mengalikan (*learning_rate*) dengan gradien masing-masing.

$$W_1 = learning_rate \times dw_1 \quad (3.45)$$

$$b_1 = learning_rate \times db_1 \quad (3.46)$$

$$W_2 = learning_rate \times dw_2 \quad (3.47)$$

$$b_2 = learning_rate \times db_2 \quad (3.48)$$

$$W_3 = learning_rate \times dw_3 \quad (3.49)$$

$$b_3 = learning_rate \times db_3 \quad (3.50)$$

Keterangan:

W_1 = nilai bobot w_1 baru setelah pembaruan

b_1 = nilai bias b_1 baru setelah pembaruan

W_2 = nilai bobot w_2 baru setelah pembaruan

b_2 = nilai bias b_2 baru setelah pembaruan

W_3 = nilai bobot w_3 baru setelah pembaruan

b_3 = nilai bias b_3 baru setelah pembaruan

5. Prediksi

Output probabilitas (y_{pred}) dikonversi menjadi label kelas biner (0 atau 1)

menggunakan *threshold* 0.5.

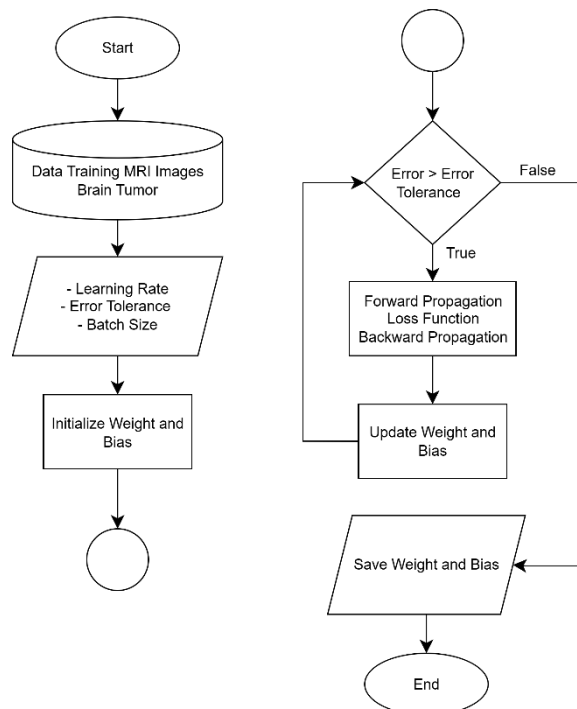
$$Kelas = \begin{cases} 1 & \text{jika } y_{pred} \geq 0.5 \text{ (Tumor)} \\ 0 & \text{jika } y_{pred} < 0.5 \text{ (No Tumor)} \end{cases} \quad (3.51)$$

3.2 Implementasi Sistem

Sebelum model diimplementasikan untuk melakukan deteksi tumor pada citra MRI otak, model harus dilatih terlebih dahulu agar dapat mengenali pola yang sesuai dengan *ground truth*. Tahap pertama implementasi sistem ini yaitu melakukan *training* terhadap model. Proses *training* tersebut menghasilkan nilai error yang digunakan untuk mengoptimasi nilai bobot yang ada didalam jaringan model. Setelah model dilatih, maka nilai bobot yang ada didalam jaringan model disimpan untuk kemudian digunakan pada tahap *testing*.

3.2.1 Training Model

Tahap *training* model merupakan proses pembelajaran dimana *neural network* mempelajari pola-pola yang terdapat dalam dataset untuk dapat membedakan antara citra MRI otak yang mengandung tumor dan yang sehat. Pada tahap ini, model *neural network* akan melalui beberapa proses penting. Pertama, *forward propagation* dilakukan untuk menghitung prediksi berdasarkan *input* citra MRI yang diberikan. Kemudian, *loss function* dihitung untuk mengukur perbedaan antara prediksi model dengan ground truth yang sebenarnya. Setelah itu, *backward propagation* dilakukan untuk menghitung gradien dan memperbarui bobot-bobot dalam jaringan *neural network* menggunakan algoritma optimisasi. Proses ini diulang untuk setiap *batch* data selama beberapa *epoch* hingga model mencapai konvergensi atau performa yang optimal. Alur dari proses *training* tersebut sebagaimana tertera pada Gambar 3.9.



Gambar 3. 9 Flowchart Proses Training Model

Dalam proses *training* model *neural network*, terdapat beberapa parameter penting yang mempengaruhi proses dan hasil *training*, yaitu *epoch*, *batch size*, *learning rate*, dan *error tolerance*. Definisi masing-masing parameter tersebut sebagaimana pada Tabel 3.1.

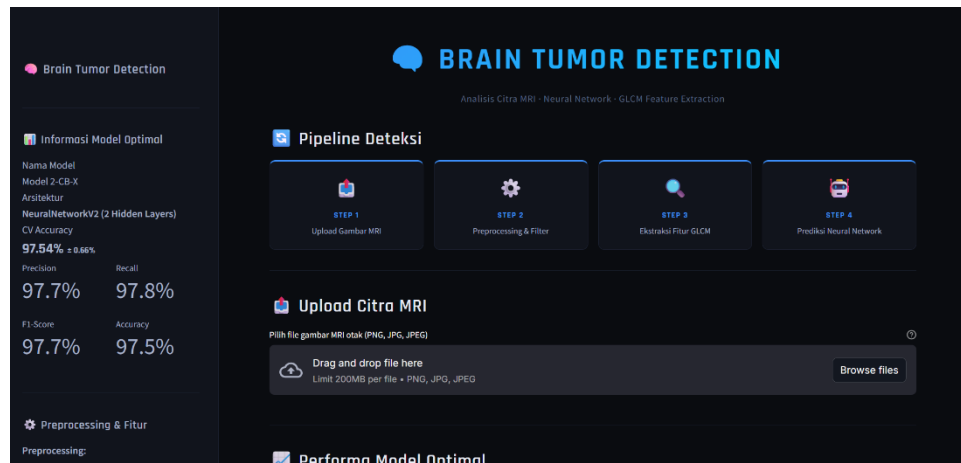
Tabel 3. 1 Definisi dan Fungsi Parameter Model

Istilah	Definisi & Fungsi
<i>Epoch</i>	Satu <i>epoch</i> adalah satu kali seluruh dataset digunakan untuk melatih model. Proses <i>Training</i> model biasanya dilakukan dalam banyak <i>epoch</i> agar model belajar secara bertahap dan tidak <i>overfitting/underfitting</i> (Afaq & Rao, 2020)
<i>Batch Size</i>	Jumlah sampel data yang diproses sebelum parameter model diperbarui. <i>Batch size</i> kecil sering menghasilkan generalisasi lebih baik, sedangkan <i>batch size</i> besar mempercepat komputasi tapi bisa menurunkan akurasi (Kandel & Castelli, 2020)
<i>Learning Rate</i>	Nilai yang menentukan seberapa besar langkah pembaruan bobot pada setiap iterasi. <i>Learning rate</i> terlalu besar bisa membuat <i>training</i> tidak stabil, terlalu kecil membuat <i>training</i> lambat (Granziol et al., 2020)
<i>Error Tolerance</i>	Batas toleransi error yang digunakan sebagai kriteria pemberhentian proses <i>training</i> atau sebagai ukuran ketahanan model terhadap kesalahan, baik pada data maupun perangkat keras (Torres-Huitzil & Girau, 2017)

3.3 Antarmuka Sistem

Setelah model optimal berhasil dibuat melalui proses pelatihan dan evaluasi, model tersebut disimpan untuk diterapkan pada data baru. Proses penyimpanan dilakukan dengan menyimpan bobot dan konfigurasi model sehingga memungkinkan penggunaan ulang tanpa perlu melatih ulang model dari awal. Selain itu, untuk meningkatkan aksesibilitas sistem, dikembangkan sebuah antarmuka sederhana berbasis web menggunakan Streamlit yang memungkinkan pengguna untuk mengunggah citra MRI secara langsung. Dalam antarmuka ini, pengguna dapat memilih file citra MRI dalam format yang didukung, lalu sistem akan secara otomatis memproses citra melalui tahap *preprocessing* hingga

penerapan model *neural network* yang telah disimpan, dan menghasilkan prediksi berupa "Tumor" atau "Healthy" dengan cepat, sehingga mendukung interpretasi klinis secara praktis dan efisien. Implementasi antarmuka sistem tersebut sebagaimana tertera pada Gambar 3.10.



Gambar 3. 10 Antarmuka Sistem Deteksi Tumor Otak

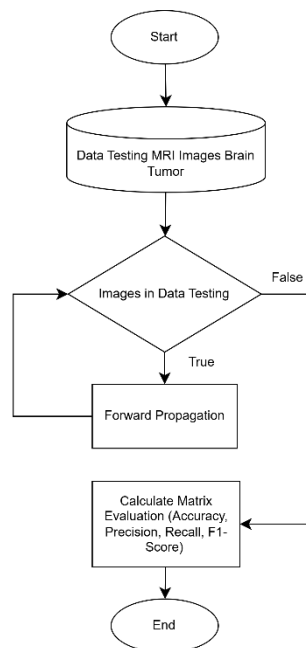
BAB IV

UJI COBA DAN PEMBAHASAN

4.1 Desain Uji Coba

Untuk melakukan deteksi tumor otak pada citra MRI, model *neural network* harus dilatih terlebih dahulu agar bisa mendeteksi ada atau tidaknya tumor pada gambar MRI sesuai dengan *ground truth* yang ada. Setelah proses *training* model selesai, maka tahap selanjutnya yaitu tahap *testing* model. Tahap *testing* model merupakan evaluasi *final* untuk mengukur kemampuan model yang telah dilatih dalam mendeteksi tumor otak pada citra MRI yang belum pernah dilihat sebelumnya. Proses ini menggunakan *test set* yang telah dipisahkan sejak awal.

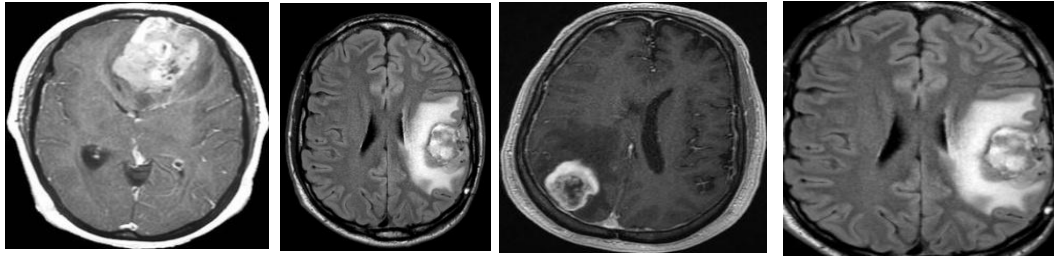
Pada tahap *testing*, model yang telah dilatih akan melakukan prediksi terhadap seluruh data dalam *test set* tanpa melakukan pembaruan bobot. Hasil prediksi kemudian dibandingkan dengan label yang sebenarnya untuk menghitung berbagai metrik evaluasi. Proses *testing* ini bersifat objektif karena menggunakan data yang benar-benar independen dan tidak pernah digunakan selama proses *training*. Alur proses *testing model* tersebut sebagaimana tertera pada Gambar 4.1.



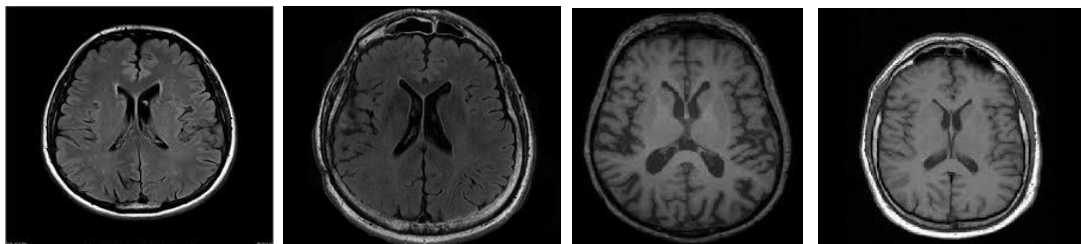
Gambar 4. 1 *Flowchart* Proses *Testing Model*

4.1.1 *Data Preparation*

Pada penelitian ini dataset yang digunakan ialah data *MRI Images* otak yang didapatkan dari repositori publik yaitu Kaggle (<https://www.kaggle.com/datasets/preetviradiya/brian-tumor-dataset/data>). Data ini terdiri dari dua bagian, pertama yaitu data dengan label '*Brain Tumor*' yang berisi *MRI Images* pasien yang telah didiagnosa tumor otak, kedua, merupakan data dengan label '*Healthy*' yang berisi *MRI Images* pasien yang tidak terindikasi tumor pada otaknya. Jumlah dari semua data yaitu 4600 data, dengan 2513 *MRI Images brain tumor* dan 2087 *MRI Images* untuk *brain healthy*.



Gambar 4. 2 *Sample Data MRI Images Tumor Otak*



Gambar 4. 3 *Sample Data MRI Images Otak Sehat*

Gambar tersebut menampilkan dua kelompok sampel data MRI otak, yaitu Gambar 4.2 untuk otak dengan tumor (*Brain Tumor*) dan Gambar 4.3 untuk otak sehat (*Healthy Brain*). Pada gambar 4.2, citra MRI otak dengan tumor ditandai dengan adanya sel-sel yang berbeda, seperti area putih atau abu-abu yang tidak biasa, yang menunjukkan pertumbuhan tumor yang mengganggu jaringan otak sehat di sekitarnya. Sebaliknya, pada Gambar 4.3, citra MRI menunjukkan struktur otak yang normal dengan distribusi jaringan yang seragam dan tidak terdapat anomali yang signifikan. Perbedaan ini menjadi dasar penting untuk analisis lebih lanjut dalam mendeteksi dan mendiagnosis kondisi tumor otak menggunakan teknologi pengolahan citra.

Ground truth atau label sebenarnya pada dataset ini telah tersedia dan terintegrasi dalam nama file setiap citra. Sebagai contoh file dengan nama "*Cancer*"

(1).jpg” menunjukkan bahwa citra tersebut memiliki label ”tumor” , sedangkan file dengan nama *”Not Cancer (1).jpg*” menunjukkan bahwa citra tersebut memiliki label ”normal”. Sistem pelabelan ini memudahkan proses ekstraksi label untuk keperluan pelatihan dan evaluasi model, karena label dapat diperoleh secara otomatis melalui *parsing* nama file tanpa memerlukan file metadata terpisah. Kualitas *ground truth* data yang sudah terverifikasi ini menjadi dasar yang kuat untuk mengembangkan dan mengevaluasi kinerja model deteksi tumor otak dengan lebih akurat.

Langkah awal yang dilakukan sebelum memasuki proses pengujian yaitu mempersiapkan dataset yang akan digunakan dalam penelitian. Dataset yang digunakan pada penelitian ini diambil dari website Kaggle. Dataset tersebut berjumlah 4600 hasil citra MRI otak pasien sehat dan pasien yang mengidap tumor. Namun, setelah dilakukan proses loading dan verifikasi integritas file, ditemukan sebanyak 86 citra yang tidak dapat dibaca karena *file corrupt*, memiliki ukuran 0 *byte*, atau format yang tidak didukung oleh *library* pengolahan citra. Oleh karena itu, jumlah citra yang berhasil dimuat dan dapat digunakan dalam penelitian ini adalah sebanyak 4.514 citra atau setara dengan 98,13 % dari total citra awal.

Data citra tersebut selanjutnya menjalani tahap *preprocessing* yang meliputi konversi citra ke mode *grayscale* menggunakan metode *convert('L')*, *resize* menjadi 128×128 piksel, serta normalisasi intensitas piksel ke rentang $[0, 1]$. Setelah *preprocessing* selesai, dilakukan tahapan ekstraksi fitur tekstur menggunakan *Gray Level Co-occurrence Matrix (GLCM)* pada empat sudut (0° , 45° , 90° , dan 135°) dengan jarak antar piksel 1. Dari setiap GLCM diekstrak enam fitur Haralick, yaitu

dissimilarity, *correlation*, *homogeneity*, *contrast*, *entropy*, dan *energy*, sehingga menghasilkan 48 nilai fitur per *patch*. Seluruh *patch* pada satu citra kemudian dirata-ratakan untuk memperoleh satu vektor fitur global berukuran 48 dimensi yang mewakili karakteristik tekstur keseluruhan citra. Proses ini menghasilkan matriks fitur akhir berukuran (4.514, 48).

Setelah matriks fitur diperoleh, dilakukan pembagian data menjadi dua kelompok utama yaitu data latih (*training set*) untuk melatih model dan data uji (*test set*) untuk menguji akurasi model secara independen. Proporsi pembagian yang digunakan adalah 70% untuk data latih dan 30% untuk data uji. Pembagian tersebut diterapkan menggunakan teknik *stratified split* melalui fungsi *train_test_split* dari *library scikit-learn*. Pembagian stratifikasi ini memastikan proporsi kelas Brain Tumor dan *Healthy* tetap seimbang pada kedua himpunan data. Persebaran jumlah data pada masing-masing set tersebut tertera pada Tabel 4.1.

Tabel 4. 1 Persebaran Jumlah Data

Kategori	<i>Training Set</i> (70 %)	<i>Testing Set</i> (30%)	Total Data
<i>Brain Tumor</i>	1698	729	2427
<i>Healthy</i>	1460	627	2087
Total	3158	904	4514

4.1.2 Skenario Uji Coba

Pada penelitian ini, skenario uji coba dirancang untuk mengetahui kombinasi parameter mana yang menunjukkan hasil kinerja terbaik untuk melakukan deteksi tumor otak pada citra MRI. Untuk menentukan konfigurasi optimal yang menghasilkan performa terbaik, dilakukan variasi terhadap beberapa *hyperparameter*, yaitu jumlah neuron per *hidden layer* dan *learning rate* dengan

tujuan mengidentifikasi kombinasi yang memberikan nilai akurasi dan generalisasi tertinggi sekaligus mencegah *overfitting* (Anantharajan et al., 2024).

Skenario pertama ialah terkait jumlah *hidden layer* yang digunakan pada pembangunan model *neural network*. Pemilihan variasi jumlah *hidden layer* ini didasarkan pada prinsip keseimbangan antara kompleksitas model dan performa prediksi. Studi yang dilakukan oleh (Seidi et al., 2025) menemukan bahwa penambahan *hidden layer* dari satu ke dua secara signifikan meningkatkan akurasi prediksi, kecepatan konvergensi, dan mengurangi variasi performa, karena *layer* kedua memungkinkan ekstraksi fitur yang lebih abstrak dan diskriminatif. Namun, penambahan ke tiga *layer* tidak memberikan perbaikan bermakna dan justru meningkatkan waktu komputasi tanpa keuntungan proporsional, sehingga penelitian ini membatasi variasi pada satu dan dua *hidden layer* untuk menjaga efisiensi sambil memaksimalkan kemampuan model dalam mendeteksi pola tekstur pada citra MRI otak. Skenario jumlah *hidden layer* tersebut sebagaimana ditampilkan pada Tabel 4.2.

Tabel 4. 2 Skenario Jumlah *Hidden layer Neural Network*

Skenario	Jumlah <i>Hidden layer</i>
1	1
2	2

Skenario Kedua berfokus pada jumlah *neuron* pada *hidden layer* yang digunakan, yaitu 64, 128, dan 256 neuron. Arsitektur dengan satu *hidden layer* berukuran 64–128 neuron telah terbukti efektif pada berbagai studi terkini, baik sebagai bagian akhir klasifikasi pada CNN, dengan kemampuan mencapai akurasi tinggi pada dataset citra tumor otak (Saeedi et al., 2023). Studi yang dilakukan oleh

Khaliki & Başarslan (2024) menunjukkan bahwa model CNN dengan jumlah *node dense layer* 128 neuron berhasil mencapai akurasi 98 % dalam klasifikasi glioma, meningioma, dan pituitari. Sedangkan studi oleh Ferreira et al., (2025) menunjukkan konfigurasi model CNN dengan *dense layer* 64 neuron mencapai performa sekitar 95 % pada dataset serupa. Dalam konteks ini, 64 neuron cenderung menghasilkan model yang lebih ringan, lebih cepat, dan memiliki risiko *overfitting* lebih rendah, tapi kurang ekspresif pada data yang kompleks. Sebaliknya, 128 neuron menawarkan kompromi optimal antara kapasitas representasi dan kompleksitas, sehingga mampu menangkap variasi tekstur, bentuk, dan lokasi tumor dengan lebih baik tanpa menambah beban komputasi yang berlebihan. Sementara itu, 256 neuron dipilih karena dapat meningkatkan kapasitas model secara signifikan, sehingga dapat bermanfaat pada dataset sangat besar dan beragam. Namun, pada tugas medis dengan volume data moderat, konfigurasi ini justru berpotensi memperbesar risiko *overfitting* serta waktu pelatihan tanpa memberikan peningkatan performa yang sepadan (Khan et al., 2022b). Skenario jumlah *hidden layer* tersebut sebagaimana tertera pada Tabel 4.3.

Tabel 4. 3 Skenario Jumlah Neuron pada *Hidden layer*

Skenario	Jumlah Node <i>Hidden layer</i>
A	64
B	128
C	256

Skenario ketiga menguji pengaruh nilai *learning rate* yang digunakan pada pembangunan model *neural network*. Paramater *learning rate* yang berfungsi untuk mengatur seberapa besar bobot jaringan diperbarui pada setiap iterasi pelatihan.

Pada percobaan ini, nilai *learning rate* tersebut divariasikan pada nilai 0.1, 0.05, dan 0.01. Nilai 0.01 dipilih karena cenderung menghasilkan pelatihan yang stabil dan akurasi tinggi, sebagaimana studi oleh (Mahmud et al., 2023) yang menggunakan CNN untuk deteksi tumor otak dengan nilai *learning rate* 0.01 mencapai akurasi validasi 93,3% dan AUC 98,43% pada 3264 citra MRI. Selanjutnya, nilai *learning rate* 0.05 dipilih sebagai alternatif yang terbukti dapat memberikan keseimbangan optimal antara kecepatan konvergensi dan stabilitas pelatihan (Faruq et al., 2024). Selain itu, penelitian oleh (Dong, 2025) yang menganalisis pengaruh *learning rate* pada *EfficientNetB0* untuk klasifikasi tumor otak menunjukkan bahwa *learning rate* yang lebih tinggi dapat membantu mencapai performa optimal jika dikombinasikan dengan *optimizer* dan *batch size* yang sesuai. Oleh karena itu, pada penelitian ini juga menggunakan nilai *learning rate* 0.1 karena cenderung mempercepat proses konvergensi dan menghasilkan error yang kecil pada banyak kasus. Skenario variasi nilai *learning rate* tersebut sebagaimana ditampilkan pada Tabel 4.4.

Tabel 4. 4 Skenario Nilai *Learning Rate Neural Network*

Skenario	<i>Learning Rate</i>
X	0.1
Y	0.05
Z	0.01

Dengan demikian, Kombinasi dari ketiga skenario di atas menghasilkan beberapa kombinasi skenario untuk mengetahui kombinasi skenario terbaik. Kombinasi skenario pengujia model sebagaimana disajikan pada Tabel 4.5.

Tabel 4. 5 Kombinasi Skenario Pengujian

No. Skenario	Skenario Pengujian
1	1-A-X
2	1-A-Y
3	1-A-Z
4	1-B-X
5	1-B-Y
6	1-B-Z
7	1-C-X
8	1-C-Y
9	1-C-Z
10	2-BA-X
11	2-BA-Y
12	2-BA-Z
13	2-CB-X
14	2-CB-Y
15	2-CB-Z
16	2-CA-X
17	2-CA-Y
18	2-CA-Z

Sebelum masuk kedalam pembahasan setiap skenario, nilai dari parameter harus ditentukan terlebih dahulu. Seluruh skenario tersebut menggunakan parameter tetap dengan nilai *batch size* sebesar 32, *early stopping* atau *patience* 120, dengan nilai maksimum *epoch* yaitu 1000. Nilai maksimum *epoch* sebesar 1000 dipilih karena pada *epoch* di bawah 1000 model masih menunjukkan peningkatan performa yang signifikan, sedangkan pada *epoch* di atas 1000 peningkatan tersebut menjadi sangat kecil dan tidak signifikan, sehingga 1000 *epoch* dianggap sebagai titik optimal untuk mencapai konvergensi tanpa risiko

overfitting dan pemborosan waktu komputasi. Inisialisasi bobot menggunakan *He Initialization* untuk *layer ReLU* dan *Xavier* untuk *layer sigmoid*.

4.1.3 Ukuran Pengujian

Tahapan evaluasi model pada penelitian ini melibatkan dua komponen evaluasi utama yang saling melengkapi. Pertama adalah *K-Fold Cross Validation* yang memberikan estimasi performa model secara menyeluruh dengan membagi data menjadi beberapa *fold* dan melakukan evaluasi secara berulang. Kedua adalah analisis *Confusion Matrix* yang memberikan gambaran detail tentang jenis kesalahan yang dibuat model, termasuk perhitungan metrik evaluasi seperti akurasi, presisi, *recall*, dan *F1-score*. Kombinasi kedua metode evaluasi ini memberikan pemahaman yang komprehensif tentang kinerja model dan menjadi dasar untuk menilai kelayakan model untuk diimplementasikan dalam aplikasi nyata.

K-Fold Cross Validation adalah teknik evaluasi model yang bekerja dengan membagi dataset menjadi k bagian (*fold*) yang berukuran sama, kemudian melakukan proses pelatihan dan pengujian secara berulang sebanyak k iterasi (Li, 2023). Pada setiap iterasi, satu *fold* digunakan sebagai data validasi, sementara $k-1$ *fold* sisanya digunakan sebagai data pelatihan. Proses ini diulang hingga setiap *fold* pernah menjadi data validasi. Hasil evaluasi dari semua iterasi kemudian dirata-ratakan untuk mendapatkan estimasi performa model yang *final*.

Penggunaan *K-Fold Cross Validation* memiliki beberapa tujuan utama, yaitu memberikan estimasi akurasi model yang lebih stabil dibandingkan dengan pembagian data tunggal, mengurangi risiko *overfitting* yang dapat terjadi akibat bias dalam pembagian data, dan memaksimalkan pemanfaatan seluruh dataset

untuk proses pelatihan dan pengujian (Wong & Yeh, 2020). Dalam penelitian ini, nilai k yang dipilih yaitu $k=5$ karena memberikan hasil yang stabil dengan kebutuhan komputasi yang masih dapat diterima (Marcot & Hanea, 2021).

Penerapan *K-Fold Cross Validation* dalam evaluasi dan uji coba model klasifikasi tumor ini akan menghasilkan nilai rata-rata untuk setiap metrik evaluasi pada *confusion matrix*. *Confusion Matrix* adalah tabel evaluasi yang digunakan untuk menilai kinerja model klasifikasi. Perhitungan ini dilakukan untuk mengukur akurasi dan kesalahan model klasifikasi dengan membandingkan hasil prediksi dengan label sebenarnya. Hasil dari *confusion matrix* terdiri dari 4 komponen utama, diantaranya *True Positive (TP)*, *False Positive (FP)*, *False Negative (FN)*, dan *True Negative (TN)*. Pada penelitian ini, kelas dengan nilai sebenarnya positif adalah kelas tumor, sedangkan kelas dengan nilai sebenarnya negatif adalah kelas normal atau *no tumor*. *Confusion matrix* untuk kelas tumor dan normal tersebut sebagaimana ditampilkan pada Tabel 4.2. Berikut ini penjelasan masing-masing komponen dari *confusion matrix* :

1. *True Positive (TP)* merupakan kelas positif yang diprediksi benar oleh sistem sebagai kelas positif.
2. *False Positive (FP)* merupakan kelas positif yang diprediksi salah oleh sistem sebagai kelas negatif.
3. *False Negative (FN)* merupakan kelas negatif yang diprediksi salah oleh sistem sebagai kelas positif.
4. *True Negative (TN)* merupakan kelas negatif yang diprediksi benar oleh sistem sebagai kelas negatif.

Tabel 4. 6 Confusion Matrix

Kelas Aktual	Kelas Prediksi	
	Tumor	Normal
Tumor	<i>True Positive (TP)</i>	<i>False Negative (FN)</i>
Normal	<i>False Positive (FP)</i>	<i>True Negative (TN)</i>

Komponen-komponen dalam *confusion matrix* tersebut selanjutnya akan digunakan dalam perhitungan metrik evaluasi yaitu *accuracy*, *F1-score*, *precision*, dan *recall*. Akurasi mengukur proporsi prediksi yang benar dari total prediksi, sementara presisi menunjukkan seberapa banyak prediksi positif yang benar-benar benar (Chen et al., 2023). *Recall* mengukur kemampuan model dalam mengidentifikasi semua instance positif yang benar-benar terdeteksi oleh model dari seluruh kasus positif yang ada, dan *F1-score* merupakan rata-rata harmonik antara presisi dan *recall*, memberikan gambaran seimbang tentang performa model (Takahashi et al., 2022). Hasil dari metrik-metrik ini digunakan untuk menilai keefektifan model dan memberikan dasar untuk perbaikan lebih lanjut.

$$Accuracy = \frac{TP + TN}{TP + TN + FP + FN} \quad (4.1)$$

$$Precision = \frac{TP}{TP + FP} \quad (4.2)$$

$$Recall = \frac{TP}{TP + FN} \quad (4.3)$$

$$F1 - Score = \frac{Presisi \times Recall}{Presisi + Recall} \quad (4.4)$$

4.2 Hasil Uji Coba

4.2.1 Hasil Skenario Uji Coba *Neural Network 1 Hidden layer*

Terdapat enam skenario pengujian yang menggunakan 1 *hidden layer* pada NN, yakni skenario 1-A-X, 1-A-Y, 1-B-X, 1-B-Y, 1-C-X, dan 1-C-Y. Pada setiap model, pelatihan dilakukan dengan menggunakan pendekatan *5-fold stratified cross-validation* untuk memperoleh estimasi performa yang stabil dan tidak bias terhadap pembagian data. Pada setiap *fold*, model dievaluasi menggunakan data validasi yang tidak terlibat dalam proses pelatihan. Setelah proses pelatihan selesai, dilakukan proses *testing* pada semua model menggunakan *5-fold cross validation*. Performa klasifikasi diukur menggunakan confusion matrix pada masing-masing *fold*, yang kemudian digunakan untuk menghitung metrik evaluasi utama yaitu *accuracy*, *precision*, *recall*, dan *F1-score* sesuai dengan Persamaan 4.1, 4.2, 4.3, dan 4.4. Penjelasan masing-masing skenario akan dipaparkan selanjutnya pada sub-bab.

4.2.1.1 Skenario 1-A-X

Uji coba skenario 1-A-X menggunakan model *neural network* dengan 1 *hidden layer* (64 neuron) dan *learning rate* 0.1. *Training* dilakukan menggunakan *5-Fold Cross-Validation* pada 70% data, menghasilkan rata-rata *validation accuracy* $93.63\% \pm 0.56\%$ dengan *loss* 0.1054 ± 0.0179 sebagaimana yang tertera pada Tabel 4.7.

Tabel 4. 7 Rangkuman Hasil *Training* Model 1-A-X

<i>Fold</i>	<i>Final Loss</i>	<i>Training Accuracy</i>	<i>Validation Accuracy</i>
1	0.079889	0.983373	0.930380
2	0.102858	0.984561	0.930380
3	0.103753	0.987332	0.938291
4	0.114204	0.982588	0.936609
5	0.126215	0.977048	0.944532

Model terbaik kemudian dievaluasi pada *test* set menggunakan *5-Fold Cross-Validation*. Hasil *testing* pada Tabel 4.8 menunjukkan *mean accuracy* $92.55\% \pm 1.22\%$, *precision* $92.44\% \pm 0.90\%$, *recall* $93.83\% \pm 2.26\%$, dan *F1-score* $93.11\% \pm 1.20\%$.

Tabel 4. 8 Rangkuman Performa Model 1-A-X

<i>Fold</i>	<i>TN</i>	<i>FP</i>	<i>FN</i>	<i>TP</i>	<i>Accuracy</i>	<i>Precision</i>	<i>Recall</i>	<i>F1-score</i>
1	113	13	5	141	0.933824	0.915584	0.965753	0.940000
2	114	12	11	134	0.915129	0.917808	0.924138	0.920962
3	115	10	14	132	0.911439	0.929577	0.904110	0.916667
4	113	12	9	137	0.922509	0.919463	0.938356	0.928814
5	116	9	6	140	0.944649	0.939597	0.958904	0.949153
<i>Mean</i>					0.925510	0.924406	0.938252	0.931119
<i>Standart Deviation</i>					0.012249	0.008981	0.022573	0.012012

Secara kumulatif, model berhasil mengklasifikasikan tumor 684 (TP) dan *healthy* 571 (TN), serta gagal memprediksi kelas tumor 45 (FN) dan *healthy* 56 (FP). Penurunan performa sekitar 1% dari *validation* ke *testing* mengindikasikan generalisasi yang baik dengan *overfitting* minimal.

4.2.1.2 Skenario 1-A-Y

Uji coba skenario 1-A-Y menggunakan model *neural network* dengan 1 *hidden layer* (64 neuron) dan *learning rate* 0.05. *Training* dilakukan menggunakan *5-Fold Cross-Validation* pada 70% data, menghasilkan rata-rata *validation accuracy*

90.94% $\pm$ 1.00% dengan *loss* 0.2225 $\pm$ 0.0172 sebagaimana yang tertera pada Tabel 4.9.

Tabel 4. 9 Rangkuman Hasil *Training* Model 1-A-Y

<i>Fold</i>	<i>Final Loss</i>	<i>Final Accuracy</i>	<i>Validation Accuracy</i>
1	0.196012	0.958828	0.901899
2	0.215126	0.949723	0.900316
3	0.228942	0.949327	0.920886
4	0.240723	0.948556	0.901743
5	0.232098	0.948160	0.922345

Model terbaik kemudian dievaluasi pada *test* set menggunakan *5-Fold Cross-Validation*. Hasil *testing* pada Tabel 4.10 menunjukkan *mean accuracy* 89.75% $\pm$ 2.04%, *precision* 90.07% $\pm$ 1.34%, *recall* 90.95% $\pm$ 2.72%, dan *F1-score* 90.50% $\pm$ 1.94%.

Tabel 4. 10 Rangkuman Performa Model 1-A-Y

<i>Fold</i>	<i>TN</i>	<i>FP</i>	<i>FN</i>	<i>TP</i>	<i>Accuracy</i>	<i>Precision</i>	<i>Recall</i>	<i>F1-score</i>
1	110	16	14	132	0.889706	0.891892	0.904110	0.897959
2	111	15	13	132	0.896679	0.897959	0.910345	0.904110
3	108	17	18	128	0.870849	0.882759	0.876712	0.879725
4	112	13	15	131	0.896679	0.909722	0.897260	0.903448
5	113	12	6	140	0.933579	0.921053	0.958904	0.939597
<i>Mean</i>					0.897498	0.900677	0.909466	0.904968
<i>Standart Deviation</i>					0.020367	0.013436	0.027186	0.019435

Secara kumulatif, model berhasil mengklasifikasikan tumor 663 (TP) dan *healthy* 554 (TN), serta gagal memprediksi kelas tumor 66 (FN) dan *healthy* 73 (FP). Penurunan performa sekitar 1.2% dari *validation* ke *testing* menunjukkan konsistensi yang baik, meskipun *learning rate* yang lebih rendah menghasilkan akurasi yang lebih rendah dibanding 1-A-X.

4.2.1.3 Skenario 1-A-Z

Uji coba skenario 1-A-Z menggunakan model *neural network* dengan 1 *hidden layer* (64 neuron) dan *learning rate* 0.01. *Training* dilakukan menggunakan *5-Fold*

Cross-Validation pada 70% data, menghasilkan rata-rata *validation accuracy* $81.92\% \pm 1.83\%$ dengan *loss* 0.5728 ± 0.0201 sebagaimana yang tertera pada Tabel 4.11.

Tabel 4. 11 Rangkuman Hasil *Training* Model 1-A-Z

<i>Fold</i>	<i>Final Loss</i>	<i>Final Accuracy</i>	<i>Validation Accuracy</i>
1	0.563380	0.827395	0.816456
2	0.545524	0.846793	0.799051
3	0.600609	0.827395	0.849684
4	0.577195	0.840918	0.828843
5	0.577111	0.827463	0.801902

Model terbaik kemudian dievaluasi pada *test set* menggunakan *5-Fold Cross-Validation*. Hasil *testing* pada Tabel 4.12 menunjukkan *mean accuracy* $81.78\% \pm 1.60\%$, *precision* $82.40\% \pm 1.41\%$, *recall* $84.09\% \pm 2.34\%$, dan *F1-score* $83.22\% \pm 1.59\%$.

Tabel 4. 12 Rangkuman Performa Model 1-A-Z

<i>Fold</i>	<i>TN</i>	<i>FP</i>	<i>FN</i>	<i>TP</i>	<i>Accuracy</i>	<i>Precision</i>	<i>Recall</i>	<i>F1-score</i>
1	101	25	21	125	0.830882	0.833333	0.856164	0.844595
2	96	30	21	124	0.811808	0.805195	0.855172	0.829431
3	98	27	30	116	0.789668	0.811189	0.794521	0.802768
4	102	23	22	124	0.833948	0.843537	0.849315	0.846416
5	99	26	22	124	0.822878	0.826667	0.849315	0.837838
<i>Mean</i>					0.817837	0.823984	0.840897	0.832210
<i>Standart Deviation</i>					0.016031	0.014097	0.023364	0.015883

Secara kumulatif, model berhasil mengklasifikasikan tumor 613 (TP) dan *healthy* 496 (TN), serta gagal memprediksi kelas tumor 116 (FN) dan *healthy* 131 (FP). Performa yang relatif stabil antara *validation* dan *testing* (penurunan $\sim 0.14\%$) mengindikasikan tidak ada *overfitting*, namun *learning rate* yang terlalu rendah menyebabkan *underfitting* dengan akurasi keseluruhan terendah di antara varian 1-A.

4.2.1.4 Skenario 1-B-X

Uji coba skenario 1-B-X menggunakan model *neural network* dengan 1 *hidden layer* (128 neuron) dan *learning rate* 0.1. *Training* dilakukan menggunakan *5-Fold Cross-Validation* pada 70% data, menghasilkan rata-rata *validation accuracy* $94.43\% \pm 0.95\%$ dengan *loss* 0.0800 ± 0.0071 sebagaimana yang tertera pada Tabel 4.13.

Tabel 4. 13 Rangkuman Hasil *Training* Model 1-B-X

<i>Fold</i>	<i>Final Loss</i>	<i>Final Accuracy</i>	<i>Validation Accuracy</i>
1	0.086527	0.990895	0.933544
2	0.073623	0.992082	0.938291
3	0.079207	0.988519	0.941456
4	0.072419	0.988128	0.949287
5	0.088007	0.989315	0.958796

Model terbaik kemudian dievaluasi pada *test set* menggunakan *5-Fold Cross-Validation*. Hasil *testing* pada Tabel 4.14 menunjukkan *mean accuracy* $93.95\% \pm 0.80\%$, *precision* $94.15\% \pm 0.95\%$, *recall* $94.65\% \pm 1.33\%$, dan *F1-score* $94.39\% \pm 0.75\%$.

Tabel 4. 14 Rangkuman Performa Model 1-B-X

<i>Fold</i>	<i>TN</i>	<i>FP</i>	<i>FN</i>	<i>TP</i>	<i>Accuracy</i>	<i>Precision</i>	<i>Recall</i>	<i>F1-score</i>
1	116	10	5	141	0.944853	0.933775	0.965753	0.949495
2	120	6	9	136	0.944649	0.957746	0.937931	0.947735
3	115	10	9	137	0.929889	0.931973	0.938356	0.935154
4	116	9	10	136	0.929889	0.937931	0.931507	0.934708
5	117	8	6	140	0.948339	0.945946	0.958904	0.952381
<i>Mean</i>					0.939524	0.941474	0.946490	0.943895
<i>Standart Deviation</i>					0.007975	0.009454	0.013335	0.007469

Secara kumulatif, model berhasil mengklasifikasikan tumor 690 (TP) dan *healthy* 584 (TN), serta gagal memprediksi kelas tumor 39 (FN) dan *healthy* 43 (FP). Penurunan performa sekitar 0.5% dari *validation* ke *testing* mengindikasikan

generalisasi yang sangat baik. Peningkatan jumlah neuron dari 64 ke 128 meningkatkan akurasi sekitar 1.4% dibanding 1-A-X.

4.2.1.5 Skenario 1-B-Y

Uji coba skenario 1-B-Y menggunakan model *neural network* dengan 1 *hidden layer* (128 neuron) dan *learning rate* 0.05. *Training* dilakukan menggunakan *5-Fold Cross-Validation* pada 70% data, menghasilkan rata-rata *validation accuracy* $92.24\% \pm 1.73\%$ dengan *loss* 0.1991 ± 0.0117 sebagaimana yang tertera pada Tabel 4.15.

Tabel 4. 15 Rangkuman Hasil *Training* Model 1-B-Y

<i>Fold</i>	<i>Final Loss</i>	<i>Final Accuracy</i>	<i>Validation Accuracy</i>
1	0.201233	0.963579	0.931962
2	0.195714	0.961203	0.909810
3	0.202027	0.955265	0.925633
4	0.180947	0.963197	0.914422
5	0.215632	0.956074	0.930269

Model terbaik kemudian dievaluasi pada *test set* menggunakan *5-Fold Cross-Validation*. Hasil *testing* pada Tabel 4.16 menunjukkan *mean accuracy* $90.71\% \pm 1.23\%$, *precision* $92.17\% \pm 0.47\%$, *recall* $90.40\% \pm 2.29\%$, dan *F1-score* $91.26\% \pm 1.25\%$.

Tabel 4. 16 Rangkuman Performa Model 1-B-Y

<i>Fold</i>	<i>TN</i>	<i>FP</i>	<i>FN</i>	<i>TP</i>	<i>Accuracy</i>	<i>Precision</i>	<i>Recall</i>	<i>F1-score</i>
1	115	11	10	136	0.922794	0.925170	0.931507	0.928328
2	116	10	14	131	0.911439	0.929078	0.903448	0.916084
3	114	11	20	126	0.885609	0.919708	0.863014	0.890459
4	113	12	14	132	0.904059	0.916667	0.904110	0.910345
5	113	12	12	134	0.911439	0.917808	0.917808	0.917808
<i>Mean</i>					0.907068	0.921686	0.903977	0.912605
<i>Standart Deviation</i>					0.012289	0.004710	0.022924	0.012505

Secara kumulatif, model berhasil mengklasifikasikan tumor 659 (TP) dan *healthy* 571 (TN), serta gagal memprediksi kelas tumor 70 (FN) dan *healthy* 56 (FP). Penurunan performa sekitar 1.5% dari *validation* ke *testing* menunjukkan generalisasi yang cukup baik, meskipun *learning rate* 0.05 menghasilkan performa yang lebih rendah dibanding 1-B-X dengan *learning rate* 0.1.

4.2.1.6 Skenario 1-B-Z

Uji coba skenario 1-B-Z menggunakan model *neural network* dengan 1 *hidden layer* (128 neuron) dan *learning rate* 0.01. *Training* dilakukan menggunakan 5-Fold *Cross-Validation* pada 70% data, menghasilkan rata-rata *validation accuracy* $81.10\% \pm 1.84\%$ dengan *loss* 0.5729 ± 0.0094 sebagaimana yang tertera pada Tabel 4.17.

Tabel 4. 17 Rangkuman Hasil *Training* Model 1-B-Z

<i>Fold</i>	<i>Final Loss</i>	<i>Final Accuracy</i>	<i>Validation Accuracy</i>
1	0.571731	0.822249	0.811709
2	0.557699	0.834521	0.783228
3	0.584897	0.822249	0.832278
4	0.575013	0.823506	0.811410
5	0.575212	0.828651	0.816165

Model terbaik kemudian dievaluasi pada *test set* menggunakan 5-Fold *Cross-Validation*. Hasil *testing* pada Tabel 4.18 menunjukkan *mean accuracy*

81.34% $\pm$ 1.99%, *precision* 81.48% $\pm$ 2.60%, *recall* 84.64% $\pm$ 2.06%, dan *F1-score* 82.99% $\pm$ 1.67%.

Tabel 4. 18 Rangkuman Performa Model 1-B-Z

<i>Fold</i>	<i>TN</i>	<i>FP</i>	<i>FN</i>	<i>TP</i>	<i>Accuracy</i>	<i>Precision</i>	<i>Recall</i>	<i>F1-score</i>
1	99	27	19	127	0.830882	0.824675	0.869863	0.846667
2	88	38	22	123	0.778598	0.763975	0.848276	0.803922
3	100	25	28	118	0.804428	0.825175	0.808219	0.816609
4	101	24	22	124	0.830258	0.837838	0.849315	0.843537
5	98	27	21	125	0.822878	0.822368	0.856164	0.838926
<i>Mean</i>					0.813409	0.814806	0.846368	0.829932
<i>Standart Deviation</i>					0.019856	0.025986	0.020570	0.016734

Secara kumulatif, model berhasil mengklasifikasikan tumor 617 (TP) dan *healthy* 486 (TN), serta gagal memprediksi kelas tumor 112 (FN) dan *healthy* 141 (FP). Peningkatan kecil sekitar 0.24% dari *validation* ke *testing* menunjukkan tidak ada *overfitting*, namun *learning rate* yang terlalu rendah menyebabkan konvergensi yang buruk dengan akurasi terendah di kelompok 1-B.

4.2.1.7 Skenario 1-C-X

Uji coba skenario 1-C-X menggunakan model *neural network* dengan 1 *hidden layer* (256 neuron) dan *learning rate* 0.1. *Training* dilakukan menggunakan 5-Fold *Cross-Validation* pada 70% data, menghasilkan rata-rata *validation accuracy* 94.54% $\pm$ 0.95% dengan *loss* 0.0741 $\pm$ 0.0052 sebagaimana yang tertera pada Tabel 4.19.

Tabel 4. 19 Rangkuman Hasil *Training* Model 1-C-X

<i>Fold</i>	<i>Final Loss</i>	<i>Final Accuracy</i>	<i>Validation Accuracy</i>
1	0.071941	0.992874	0.935127
2	0.076496	0.990895	0.935127
3	0.077983	0.990103	0.950949
4	0.078379	0.989711	0.947702
5	0.065628	0.993273	0.958796

Model terbaik kemudian dievaluasi pada *test set* menggunakan *5-Fold Cross-Validation*. Hasil *testing* pada Tabel 4.20 menunjukkan *mean accuracy* $94.69\% \pm 0.37\%$, *precision* $94.71\% \pm 0.92\%$, *recall* $95.47\% \pm 0.70\%$, dan *F1-score* $95.08\% \pm 0.32\%$.

Tabel 4. 20 Rangkuman Performa Model 1-C-X

<i>Fold</i>	<i>TN</i>	<i>FP</i>	<i>FN</i>	<i>TP</i>	<i>Accuracy</i>	<i>Precision</i>	<i>Recall</i>	<i>F1-score</i>
1	117	9	6	140	0.944853	0.939597	0.958904	0.949153
2	120	6	7	138	0.952030	0.958333	0.951724	0.955017
3	119	6	8	138	0.948339	0.958333	0.945205	0.951724
4	116	9	7	139	0.940959	0.939189	0.952055	0.945578
5	116	9	5	141	0.948339	0.940000	0.965753	0.952703
<i>Mean</i>					0.946904	0.947091	0.954728	0.950835
<i>Standart Deviation</i>					0.003740	0.009183	0.007012	0.003232

Secara kumulatif, model berhasil mengklasifikasikan tumor 696 (TP) dan *healthy* 588 (TN), serta gagal memprediksi kelas tumor 33 (FN) dan *healthy* 39 (FP). Peningkatan performa sekitar 0.15% dari *validation* ke *testing* menunjukkan generalisasi yang sangat baik. Model 1-C-X mencapai performa tertinggi di antara semua varian *NeuralNetworkV1* dengan 256 neuron memberikan kapasitas representasi yang optimal.

4.2.1.8 Skenario 1-C-Y

Uji coba skenario 1-C-Y menggunakan model *neural network* dengan 1 *hidden layer* (256 neuron) dan *learning rate* 0.05. *Training* dilakukan menggunakan *5-Fold Cross-Validation* pada 70% data, menghasilkan rata-rata *validation accuracy* $92.43\% \pm 0.91\%$ dengan *loss* 0.2023 ± 0.0053 sebagaimana yang tertera pada Tabel 4.21.

Tabel 4. 21 Rangkuman Hasil *Training* Model 1-C-Y

<i>Fold</i>	<i>Final Loss</i>	<i>Final Accuracy</i>	<i>Validation Accuracy</i>
1	0.194394	0.961203	0.922468
2	0.202057	0.958432	0.920886
3	0.211289	0.957245	0.939873
4	0.203753	0.954887	0.912837
5	0.200025	0.955679	0.925515

Model terbaik kemudian dievaluasi pada *test set* menggunakan *5-Fold Cross-Validation*. Hasil *testing* pada Tabel 4.22 menunjukkan *mean accuracy* 91.15% $\pm$ 1.17%, *precision* 92.04% $\pm$ 1.37%, *recall* 91.49% $\pm$ 2.48%, dan *F1-score* 91.74% $\pm$ 1.17%.

Tabel 4. 22 Rangkuman Performa Model 1-C-Y

<i>Fold</i>	<i>TN</i>	<i>FP</i>	<i>FN</i>	<i>TP</i>	<i>Accuracy</i>	<i>Precision</i>	<i>Recall</i>	<i>F1-score</i>
1	113	13	11	135	0.911765	0.912162	0.924658	0.918367
2	117	9	14	131	0.915129	0.935714	0.903448	0.919298
3	113	12	18	128	0.889299	0.914286	0.876712	0.895105
4	116	9	12	134	0.922509	0.937063	0.917808	0.927336
5	110	15	7	139	0.918819	0.902597	0.952055	0.926667
<i>Mean</i>					0.911504	0.920365	0.914936	0.917355
<i>Standart Deviation</i>					0.011670	0.013670	0.024793	0.011715

Secara kumulatif, model berhasil mengklasifikasikan tumor 667 (TP) dan *healthy* 569 (TN), serta gagal memprediksi kelas tumor 62 (FN) dan *healthy* 58 (FP). Penurunan performa sekitar 1.3% dari *validation* ke *testing* menunjukkan generalisasi yang baik, meskipun *learning rate* 0.05 menghasilkan akurasi yang lebih rendah dibanding 1-C-X.

4.2.1.9 Skenario 1-C-Z

Uji coba skenario 1-C-Z menggunakan model *neural network* dengan 1 *hidden layer* (256 neuron) dan *learning rate* 0.01. *Training* dilakukan menggunakan *5-Fold Cross-Validation* pada 70% data, menghasilkan rata-rata *validation accuracy*

80.32% $\pm$ 2.14% dengan *loss* 0.6027 $\pm$ 0.0138 sebagaimana yang tertera pada Tabel 4.23.

Tabel 4. 23 Rangkuman Hasil *Training* Model 1-C-Z

<i>Fold</i>	<i>Final Loss</i>	<i>Final Accuracy</i>	<i>Validation Accuracy</i>
1	0.589690	0.813539	0.806962
2	0.590739	0.821853	0.773734
3	0.621495	0.817102	0.833861
4	0.608355	0.808073	0.811410
5	0.603654	0.812822	0.789223

Model terbaik kemudian dievaluasi pada *test set* menggunakan *5-Fold Cross-Validation*. Hasil *testing* pada Tabel 4.24 menunjukkan *mean accuracy* 81.56% $\pm$ 1.69%, *precision* 83.64% $\pm$ 2.01%, *recall* 81.76% $\pm$ 2.61%, dan *F1-score* 82.65% $\pm$ 1.68%.

Tabel 4. 24 Rangkuman Peforma Model 1-C-Z

<i>Fold</i>	<i>TN</i>	<i>FP</i>	<i>FN</i>	<i>TP</i>	<i>Accuracy</i>	<i>Precision</i>	<i>Recall</i>	<i>F1-score</i>
1	102	24	23	123	0.827206	0.836735	0.842466	0.839590
2	96	30	25	120	0.797048	0.800000	0.827586	0.813559
3	103	22	34	112	0.793358	0.835821	0.767123	0.800000
4	104	21	26	120	0.826568	0.851064	0.821918	0.836237
5	105	20	25	121	0.833948	0.858156	0.828767	0.843206
<i>Mean</i>					0.815626	0.836355	0.817572	0.826518
<i>Standart Deviation</i>					0.016915	0.020069	0.026112	0.016822

Secara kumulatif, model berhasil mengklasifikasikan tumor 596 (TP) dan *healthy* 510 (TN), serta gagal memprediksi kelas tumor 133 (FN) dan *healthy* 117(FP). Peningkatan performa sekitar 1.24% dari *validation* ke *testing* mengindikasikan model underfitting pada fase *training* akibat *learning rate* yang terlalu rendah, menghasilkan akurasi terendah di kelompok 1-C.

4.2.2 Hasil Skenario Pengujian *Neural Network 2 Hidden layer*

Terdapat Sembilan skenario pengujian yang menggunakan 2 *hidden layer* pada NN, yakni skenario 2-BA-X, 2-BA-Y, 2-BA-Z, 2-CB-X, 2-CB-Y, 2-CB-Z, 2-CA-X, 2-CA-Y, 2-CA-Z. Pelatihan model dilakukan menggunakan pendekatan *5-fold stratified cross-validation* untuk memperoleh estimasi performa yang stabil dan tidak bias terhadap pembagian data. Pada *setiap fold*, model dievaluasi menggunakan data validasi yang tidak terlibat dalam proses pelatihan. Setelah proses pelatihan selesai, dilakukan proses *testing* pada semua model menggunakan *5-fold cross validation*. Performa klasifikasi diukur menggunakan confusion matrix pada masing-masing *fold*, yang kemudian digunakan untuk menghitung metrik evaluasi utama yaitu *accuracy*, *precision*, *recall*, dan *F1-score* sesuai dengan Persamaan 4.1, 4.2, 4.3, dan 4.4. Penjelasan masing-masing skenario akan dipaparkan selanjutnya pada sub-bab.

4.2.2.1 Skenario 2-BA-X

Uji coba skenario 2-BA-X menggunakan model *neural network* dengan 2 *hidden layers* (128→64 neuron) dan *learning rate* 0.1. *Training* dilakukan menggunakan *5-Fold Cross-Validation* pada 70% data, menghasilkan rata-rata *validation accuracy* $95.88\% \pm 0.87\%$ dengan *loss* 0.0024 ± 0.0006 sebagaimana yang tertera pada Tabel 4.25.

Tabel 4. 25 Rangkuman Hasil *Training* Model 2-BA-X

<i>Fold</i>	<i>Final Loss</i>	<i>Final Accuracy</i>	<i>Validation Accuracy</i>
1	0.003262	0.998416	0.952532
2	0.002483	1.000000	0.949367
3	0.001896	1.000000	0.971519
4	0.002385	1.000000	0.957211
5	0.001766	1.000000	0.963550

Model terbaik kemudian dievaluasi pada *test set* menggunakan *5-Fold Cross-Validation*. Hasil *testing* pada Tabel 4.26 menunjukkan *mean accuracy* $96.17\% \pm 0.37\%$, *precision* $96.33\% \pm 1.19\%$, *recall* $96.57\% \pm 0.75\%$, dan *F1-score* $96.44\% \pm 0.32\%$.

Tabel 4. 26 Rangkuman Performa Model 2-BA-X

<i>Fold</i>	<i>TN</i>	<i>FP</i>	<i>FN</i>	<i>TP</i>	<i>Accuracy</i>	<i>Precision</i>	<i>Recall</i>	<i>F1-score</i>
1	119	7	4	142	0.959559	0.95302	0.972603	0.962712
2	122	4	5	140	0.966790	0.972222	0.965517	0.968858
3	120	5	5	141	0.963100	0.965753	0.965753	0.965753
4	122	3	7	139	0.963100	0.978873	0.952055	0.965278
5	117	8	4	142	0.955720	0.946667	0.972603	0.959459
<i>Mean</i>					0.961653	0.963307	0.965706	0.964412
<i>Standart Deviation</i>					0.003746	0.011921	0.007504	0.003154

Secara kumulatif, model berhasil mengklasifikasikan tumor 704 (TP) dan *healthy* 600 (TN), serta gagal memprediksi kelas tumor 25 (FN) dan *healthy* 27 (FP). Peningkatan performa sekitar 0.29% dari *validation* ke *testing* menunjukkan generalisasi yang sangat baik. Arsitektur 2 *hidden layers* memberikan peningkatan signifikan dibanding *NeuralNetworkV1*, mencapai ranking ke-3 secara keseluruhan.

4.2.2.2 Skenario 2-BA-Y

Uji coba skenario 2-BA-Y menggunakan model *neural network* dengan 2 *hidden layers* (128→64 neuron) dan *learning rate* 0.05. *Training* dilakukan menggunakan *5-Fold Cross-Validation* pada 70% data, menghasilkan rata-rata *validation accuracy* $96.02\% \pm 0.79\%$ dengan *loss* 0.0058 ± 0.0011 sebagaimana yang tertera pada Tabel 4.27.

Tabel 4. 27 Rangkuman Hasil *Training* Model 2-BA-Y

<i>Fold</i>	<i>Final Loss</i>	<i>Final Accuracy</i>	<i>Validation Accuracy</i>
1	0.007730	0.998416	0.957278
2	0.005881	1.000000	0.950949
3	0.006036	1.000000	0.969937
4	0.005238	1.000000	0.954041
5	0.004509	1.000000	0.968304

Model terbaik kemudian dievaluasi pada *test set* menggunakan *5-Fold Cross-Validation*. Hasil *testing* pada Tabel 4.28 menunjukkan *mean accuracy* $96.02\% \pm 0.15\%$, *precision* $96.19\% \pm 0.96\%$, *recall* $96.43\% \pm 1.09\%$, dan *F1-score* $96.30\% \pm 0.14\%$.

Tabel 4. 28 Rangkuman Performa Model 2-BA-Y

<i>Fold</i>	<i>TN</i>	<i>FP</i>	<i>FN</i>	<i>TP</i>	<i>Accuracy</i>	<i>Precision</i>	<i>Recall</i>	<i>F1-score</i>
1	121	5	6	140	0.959559	0.965517	0.958904	0.962199
2	120	6	4	141	0.963100	0.959184	0.972414	0.965753
3	118	7	4	142	0.959410	0.95302	0.972603	0.962712
4	122	3	8	138	0.959410	0.978723	0.945205	0.961672
5	118	7	4	142	0.959410	0.95302	0.972603	0.962712
<i>Mean</i>					0.960177	0.961893	0.964346	0.963010
<i>Standart Deviation</i>					0.001462	0.009604	0.010931	0.001425

Secara kumulatif, model berhasil mengklasifikasikan tumor 703 (TP) dan *healthy* 599 (TN), serta gagal memprediksi kelas tumor 26 (FN) dan *healthy* 28 (FP). Performa yang sangat stabil antara *validation* dan *testing* (perbedaan $\sim 0\%$) menunjukkan generalisasi yang sempurna. Model ini menempati ranking ke-5 dengan konsistensi tertinggi (std paling rendah 0.15%).

4.2.2.3 Skenario 2-BA-Z

Uji coba skenario 2-BA-Z menggunakan model *neural network* dengan 2 *hidden layers* (128→64 neuron) dan *learning rate* 0.01. *Training* dilakukan menggunakan *5-Fold Cross-Validation* pada 70% data, menghasilkan rata-rata

validation accuracy $93.73\% \pm 0.96\%$ dengan *loss* 0.1399 ± 0.0369 sebagaimana yang tertera pada Tabel 4.29.

Tabel 4. 29 Rangkuman Hasil *Training* Model 2-BA-Z

<i>Fold</i>	<i>Final Loss</i>	<i>Final Accuracy</i>	<i>Validation Accuracy</i>
1	0.142805	0.973872	0.939873
2	0.109561	0.983373	0.935127
3	0.205740	0.948535	0.931962
4	0.137920	0.973091	0.923930
5	0.103974	0.979422	0.955626

Model terbaik kemudian dievaluasi pada *test set* menggunakan *5-Fold Cross-Validation*. Hasil *testing* pada Tabel 4.30 menunjukkan *mean accuracy* $93.51\% \pm 0.64\%$, *precision* $92.12\% \pm 0.80\%$, *recall* $96.16\% \pm 0.55\%$, dan *F1-score* $94.09\% \pm 0.58\%$.

Tabel 4. 30 Rangkuman Performa Model 2-BA-Z

<i>Fold</i>	<i>TN</i>	<i>FP</i>	<i>FN</i>	<i>TP</i>	<i>Accuracy</i>	<i>Precision</i>	<i>Recall</i>	<i>F1-score</i>
1	113	13	6	140	0.930147	0.915033	0.958904	0.936455
2	113	13	6	139	0.929889	0.914474	0.958621	0.936027
3	112	13	6	140	0.929889	0.915033	0.958904	0.936455
4	115	10	6	140	0.940959	0.933333	0.958904	0.945946
5	114	11	4	142	0.944649	0.928105	0.972603	0.949833
<i>Mean</i>					0.935107	0.921195	0.961587	0.940943
<i>Standart Deviation</i>					0.006393	0.007952	0.005509	0.005805

Secara kumulatif, model berhasil mengklasifikasikan tumor 701 (TP) dan *healthy* 567 (TN), serta gagal memprediksi kelas tumor 28 (FN) dan *healthy* 60 (FP). Penurunan performa sekitar 0.22% dari *validation* ke *testing* menunjukkan generalisasi yang baik, meskipun *learning rate* yang rendah menghasilkan akurasi lebih rendah dibanding 2-BA-X dan 2-BA-Y.

4.2.2.4 Skenario 2-CB-X

Uji coba skenario 2-CB-X menggunakan model *neural network* dengan 2 *hidden layers* (256→128 neuron) dan *learning rate* 0.1. *Training* dilakukan

menggunakan *5-Fold Cross-Validation* pada 70% data, menghasilkan rata-rata *validation accuracy* $96.23\% \pm 0.82\%$ dengan *loss* 0.0014 ± 0.0001 sebagaimana yang tertera pada Tabel 4.31.

Tabel 4. 31 Rangkuman Hasil *Training* Model 2-CB-X

<i>Fold</i>	<i>Final Loss</i>	<i>Final Accuracy</i>	<i>Validation Accuracy</i>
1	0.001534	1.000000	0.957278
2	0.001432	1.000000	0.950949
3	0.001492	1.000000	0.968354
4	0.001399	1.000000	0.961965
5	0.001353	1.000000	0.973059

Model terbaik kemudian dievaluasi pada *test set* menggunakan *5-Fold Cross-Validation*. Hasil *testing* pada Tabel 4.32 menunjukkan *mean accuracy* $96.61\% \pm 0.43\%$, *precision* $95.96\% \pm 0.42\%$, *recall* $97.81\% \pm 0.51\%$, dan *F1-score* $96.87\% \pm 0.40\%$.

Tabel 4. 32 Rangkuman Hasil Uji Coba Model 2-CB-X

<i>Fold</i>	<i>TN</i>	<i>FP</i>	<i>FN</i>	<i>TP</i>	<i>Accuracy</i>	<i>Precision</i>	<i>Recall</i>	<i>F1-score</i>
1	119	7	3	143	0.963235	0.953333	0.979452	0.966216
2	120	6	3	142	0.966790	0.959459	0.97931	0.969283
3	120	5	2	144	0.974170	0.966443	0.986301	0.976271
4	119	6	4	142	0.963100	0.959459	0.972603	0.965986
5	119	6	4	142	0.96310	0.959459	0.972603	0.965986
<i>Mean</i>					0.966079	0.959631	0.978054	0.968749
<i>Standart Deviation</i>					0.004285	0.004151	0.005118	0.003964

Secara kumulatif, model berhasil mengklasifikasikan tumor 713 (TP) dan *healthy* 597 (TN), serta gagal memprediksi kelas tumor 16 (FN) dan *healthy* 30 (FP). Peningkatan performa sekitar 0.38% dari *validation* ke *testing* menunjukkan generalisasi yang sangat baik.

4.2.2.5 Skenario 2-CB-Y

Uji coba skenario 2-CB-Y menggunakan model *neural network* dengan 2 *hidden layers* (256→128 neuron) dan *learning rate* 0.05. *Training* dilakukan menggunakan *5-Fold Cross-Validation* pada 70% data, menghasilkan rata-rata *validation accuracy* $96.57\% \pm 0.69\%$ dengan *loss* 0.0037 ± 0.0003 sebagaimana yang tertera pada Tabel 4.33.

Tabel 4. 33 Rangkuman Hasil *Training* Model 2-CB-Y

<i>Fold</i>	<i>Final Loss</i>	<i>Final Accuracy</i>	<i>Validation Accuracy</i>
1	0.003807	1.000000	0.968354
2	0.003300	1.000000	0.954114
3	0.003999	1.000000	0.968354
4	0.004024	1.000000	0.965135
5	0.003520	1.000000	0.971474

Model terbaik kemudian dievaluasi pada *test set* menggunakan *5-Fold Cross-Validation*. Hasil *testing* pada Tabel 4.34 menunjukkan *mean accuracy* $95.94\% \pm 0.77\%$, *precision* $95.80\% \pm 0.87\%$, *recall* $96.71\% \pm 1.18\%$, dan *F1-score* $96.24\% \pm 0.73\%$.

Tabel 4. 34 Rangkuman Performa Model 2-CB-Y

<i>Fold</i>	<i>TN</i>	<i>FP</i>	<i>FN</i>	<i>TP</i>	<i>Accuracy</i>	<i>Precision</i>	<i>Recall</i>	<i>F1-score</i>
1	118	8	4	142	0.955882	0.946667	0.972603	0.959459
2	119	7	6	139	0.952030	0.952055	0.958621	0.955326
3	119	6	2	144	0.970480	0.960000	0.986301	0.972973
4	119	6	7	139	0.952030	0.958621	0.952055	0.955326
5	121	4	5	141	0.966790	0.972414	0.965753	0.969072
<i>Mean</i>					0.959442	0.957951	0.967067	0.962432
<i>Standart Deviation</i>					0.007725	0.008671	0.011824	0.007280

Secara kumulatif, model berhasil mengklasifikasikan tumor 705 (TP) dan *healthy* 596 (TN), serta gagal memprediksi kelas tumor 24 (FN) dan *healthy* 31 (FP). Penurunan performa sekitar 0.63% dari *validation* ke *testing* menunjukkan generalisasi yang baik.

4.2.2.6 Skenario 2-CB-Z

Uji coba skenario 2-CB-Z menggunakan model *neural network* dengan 2 *hidden layers* (256→128 neuron) dan *learning rate* 0.01. *Training* dilakukan menggunakan *5-Fold Cross-Validation* pada 70% data, menghasilkan rata-rata *validation accuracy* $94.55\% \pm 0.65\%$ dengan *loss* 0.1016 ± 0.0234 sebagaimana yang tertera pada Tabel 4.35.

Tabel 4. 35 Rangkuman Hasil *Training* Model 2-CB-Z

<i>Fold</i>	<i>Final Loss</i>	<i>Final Accuracy</i>	<i>Validation Accuracy</i>
1	0.089068	0.988124	0.941456
2	0.092881	0.987332	0.935127
3	0.142435	0.969517	0.949367
4	0.097671	0.984962	0.947702
5	0.085810	0.990107	0.954041

Model terbaik kemudian dievaluasi pada *test set* menggunakan *5-Fold Cross-Validation*. Hasil *testing* pada Tabel 4.36 menunjukkan *mean accuracy* $94.54\% \pm 0.79\%$, *precision* $95.68\% \pm 0.79\%$, *recall* $94.10\% \pm 0.93\%$, dan *F1-score* $94.88\% \pm 0.74\%$.

Tabel 4. 36 Rangkuman Performa Model 2-CB-Z

<i>Fold</i>	<i>TN</i>	<i>FP</i>	<i>FN</i>	<i>TP</i>	<i>Accuracy</i>	<i>Precision</i>	<i>Recall</i>	<i>F1-score</i>
1	119	7	7	139	0.948529	0.952055	0.952055	0.952055
2	121	5	8	137	0.952030	0.964789	0.944828	0.954704
3	117	8	11	135	0.929889	0.944056	0.924658	0.934256
4	120	5	9	137	0.948339	0.964789	0.938356	0.951389
5	119	6	8	138	0.948339	0.958333	0.945205	0.951724
<i>Mean</i>					0.945425	0.956804	0.941020	0.948826
<i>Standart Deviation</i>					0.007894	0.007936	0.009259	0.007379

Secara kumulatif, model berhasil mengklasifikasikan tumor 686 (TP) dan *healthy* 596 (TN), serta gagal memprediksi kelas tumor 43 (FN) dan *healthy* 31 (FP). Performa yang sangat stabil antara *validation* dan *testing* (perbedaan $\sim 0.01\%$)

menunjukkan generalisasi yang sangat baik, meskipun *learning rate* rendah menghasilkan akurasi lebih rendah dibanding 2-CB-X dan 2-CB-Y.

4.2.2.7 Skenario 2-CA-X

Uji coba skenario 2-CA-X menggunakan model *neural network* dengan 2 *hidden layers* (256→64 neuron) dan *learning rate* 0.1. *Training* dilakukan menggunakan *5-Fold Cross-Validation* pada 70% data, menghasilkan rata-rata *validation accuracy* $96.55\% \pm 0.51\%$ dengan *loss* 0.0017 ± 0.0002 sebagaimana yang tertera pada Tabel 4.37.

Tabel 4. 37 Rangkuman Hasil *Training* Model 2-CA-X

<i>Fold</i>	<i>Final Loss</i>	<i>Final Accuracy</i>	<i>Validation Accuracy</i>
1	0.001763	1.000000	0.962025
2	0.001616	1.000000	0.958861
3	0.002018	1.000000	0.965190
4	0.001660	1.000000	0.969889
5	0.001698	1.000000	0.971474

Model terbaik kemudian dievaluasi pada *test set* menggunakan *5-Fold Cross-Validation*. Hasil *testing* pada Tabel 4.38 menunjukkan *mean accuracy* $96.24\% \pm 0.64\%$, *precision* $95.93\% \pm 0.44\%$, *recall* $97.12\% \pm 0.80\%$, dan *F1-score* $96.52\% \pm 0.59\%$.

Tabel 4. 38 Rangkuman Performa Model 2-CA-X

<i>Fold</i>	<i>TN</i>	<i>FP</i>	<i>FN</i>	<i>TP</i>	<i>Accuracy</i>	<i>Precision</i>	<i>Recall</i>	<i>F1-score</i>
1	120	6	3	143	0.966912	0.959732	0.979452	0.969492
2	120	6	4	141	0.963100	0.959184	0.972414	0.965753
3	120	5	3	143	0.970480	0.966216	0.979452	0.972789
4	119	6	5	141	0.959410	0.959184	0.965753	0.962457
5	118	7	6	140	0.952030	0.952381	0.958904	0.955631
<i>Mean</i>					0.962386	0.959339	0.971195	0.965225
<i>Standart Deviation</i>					0.006366	0.00438	0.007981	0.005923

Secara kumulatif, model berhasil mengklasifikasikan tumor 708 (TP) dan *healthy* 597 (TN), serta gagal memprediksi kelas tumor 21 (FN) dan *healthy* 30 (FP). Penurunan performa sekitar 0.31% dari *validation* ke *testing* menunjukkan generalisasi yang sangat baik.

4.2.2.8 Skenario 2-CA-Y

Uji coba skenario 2-CA-Y menggunakan model *neural network* dengan 2 *hidden layers* (256→64 neuron) dan *learning rate* 0.05. *Training* dilakukan menggunakan *5-Fold Cross-Validation* pada 70% data, menghasilkan rata-rata *validation accuracy* $96.11\% \pm 0.76\%$ dengan *loss* 0.0042 ± 0.0004 sebagaimana yang tertera pada Tabel 4.39.

Tabel 4. 39 Rangkuman Hasil *Training* Model 2-CA-Y

<i>Fold</i>	<i>Final Loss</i>	<i>Final Accuracy</i>	<i>Validation Accuracy</i>
1	0.004223	1.000000	0.957278
2	0.004037	1.000000	0.949367
3	0.004289	1.000000	0.965190
4	0.004899	1.000000	0.965135
5	0.003734	1.000000	0.968304

Model terbaik kemudian dievaluasi pada *test set* menggunakan *5-Fold Cross-Validation*. Hasil *testing* pada Tabel 4.40 menunjukkan *mean accuracy* $96.09\% \pm 0.50\%$, *precision* $96.06\% \pm 0.85\%$, *recall* $96.71\% \pm 0.80\%$, dan *F1-score* $96.38\% \pm 0.45\%$.

Tabel 4. 40 Rangkuman Performa Model 2-CA-Y

<i>Fold</i>	<i>TN</i>	<i>FP</i>	<i>FN</i>	<i>TP</i>	<i>Accuracy</i>	<i>Precision</i>	<i>Recall</i>	<i>F1-score</i>
1	121	5	6	140	0.959559	0.965517	0.958904	0.962199
2	122	4	5	140	0.966790	0.972222	0.965517	0.968858
3	119	6	3	143	0.966790	0.959732	0.979452	0.969492
4	119	6	6	140	0.955720	0.958904	0.958904	0.958904
5	117	8	4	142	0.955720	0.946667	0.972603	0.959459
<i>Mean</i>					0.960915	0.960608	0.967076	0.963783
<i>Standart Deviation</i>					0.004997	0.008450	0.007996	0.004546

Secara kumulatif, model berhasil mengklasifikasikan tumor 705 (TP) dan *healthy* 598 (TN), serta gagal memprediksi kelas tumor 24 (FN) dan *healthy* 29 (FP). Penurunan performa sekitar 0.02% dari *validation* ke *testing* menunjukkan generalisasi yang hampir sempurna.

4.2.2.9 Skenario 2-CA-Z

Uji coba skenario 2-CA-Z menggunakan model *neural network* dengan 2 *hidden layers* (256→64 neuron) dan *learning rate* 0.01. *Training* dilakukan menggunakan *5-Fold Cross-Validation* pada 70% data, menghasilkan rata-rata *validation accuracy* $94.43\% \pm 1.32\%$ dengan *loss* 0.0894 ± 0.0052 sebagaimana yang tertera pada Tabel 4.41.

Tabel 4. 41 Rangkuman Hasil Training Model 2-CA-Z

<i>Fold</i>	<i>Final Loss</i>	<i>Final Accuracy</i>	<i>Validation Accuracy</i>
1	0.097451	0.977435	0.924051
2	0.092237	0.990103	0.931962
3	0.083968	0.986936	0.957278
4	0.090136	0.984171	0.950872
5	0.083574	0.983380	0.957211

Model terbaik kemudian dievaluasi pada *test set* menggunakan *5-Fold Cross-Validation*. Hasil *testing* pada Tabel 4.42 menunjukkan *mean accuracy* $94.10\% \pm 0.81\%$, *precision* $93.13\% \pm 1.54\%$, *recall* $96.16\% \pm 1.11\%$, dan *F1-score* $94.61\% \pm 0.71\%$.

Tabel 4. 42 Rangkuman Performa Model 2-CA-Z

<i>Fold</i>	<i>TN</i>	<i>FP</i>	<i>FN</i>	<i>TP</i>	<i>Accuracy</i>	<i>Precision</i>	<i>Recall</i>	<i>F1-score</i>
1	112	14	3	143	0.937500	0.910828	0.979452	0.943894
2	118	8	6	139	0.948339	0.945578	0.958621	0.952055
3	114	11	8	138	0.929889	0.926174	0.945205	0.935593
4	118	7	6	140	0.952030	0.952381	0.958904	0.955631
5	113	12	5	141	0.937269	0.921569	0.965753	0.943144
<i>Mean</i>					0.941006	0.931306	0.961587	0.946064
<i>Standart Deviation</i>					0.008064	0.015417	0.011143	0.007075

Secara kumulatif, model berhasil mengklasifikasikan tumor 701 (TP) dan *healthy* 575 (TN), serta gagal memprediksi kelas tumor 28 (FN) dan *healthy* 52 (FP). Penurunan performa sekitar 0.33% dari *validation* ke *testing* menunjukkan generalisasi yang baik, meskipun *learning rate* yang rendah menghasilkan akurasi lebih rendah dibanding 2-CA-X dan 2-CA-Y.

4.3 Pembahasan

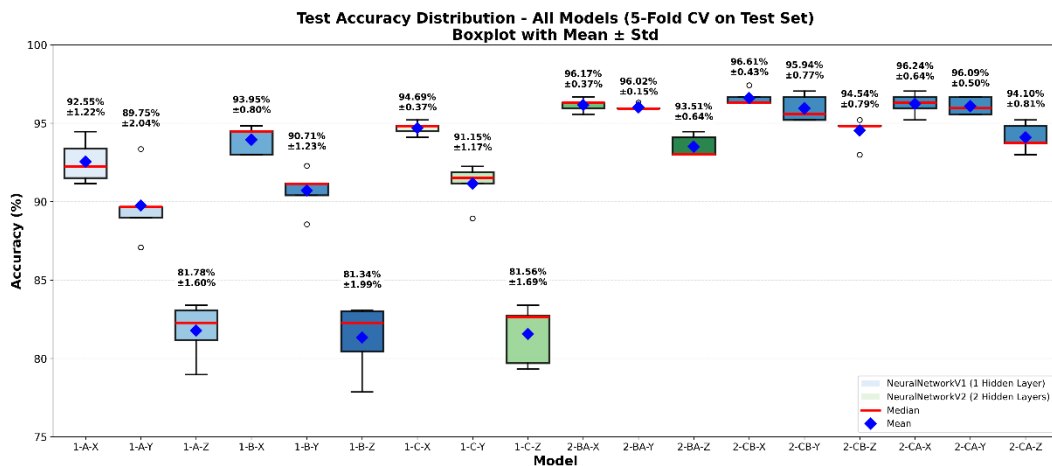
Pengujian terhadap 18 skenario konfigurasi *neural network* dalam penelitian ini menghasilkan temuan komprehensif terkait pengaruh kedalaman arsitektur dan *hyperparameter* terhadap performa deteksi tumor otak pada citra MRI. Evaluasi performa seluruh model telah dilakukan menggunakan empat metrik utama yaitu *accuracy*, *precision*, *recall*, dan *F1-score*. Hasil pengujian secara sistematis untuk *setiap* metrik evaluasi ditampilkan melalui boxplot untuk memvisualisasikan variabilitas dan konsistensi model di seluruh *fold*.

Berdasarkan analisis komprehensif terhadap seluruh metrik evaluasi, dilakukan validasi signifikansi statistik untuk mengidentifikasi model optimal secara objektif dan memastikan bahwa pemilihan model terbaik tidak hanya didasarkan pada performa rata-rata semata, tetapi juga mempertimbangkan signifikansi statistik perbedaan antar model. Untuk itu, dilakukan uji *T-Test* untuk membandingkan model terbaik pada masing-masing terhadap model lainnya.

Terdapat dua hipotesis pada pengujian *T-Test* ini, yaitu hipotesis nol (H_0) menyatakan bahwa tidak terdapat perbedaan signifikan antara rata-rata metrik evaluasi dari dua model yang dibandingkan ($\mu_1 = \mu_2$), sedangkan hipotesis alternatif (H_1) menyatakan bahwa terdapat perbedaan signifikan antara kedua model ($\mu_1 \neq$

μ_2). Tingkat signifikansi yang digunakan adalah $\alpha = 0.05$, di mana jika $p\text{-value} < 0.05$ maka hipotesis nol ditolak dan perbedaan antar model dianggap signifikan secara statistik (Di Leo & Sardanelli, 2020).

Gambar 4.4 berikut menampilkan boxplot distribusi *accuracy* dari 18 model yang diuji menggunakan *5-Fold Cross-Validation* pada *test set*. Secara umum, terlihat perbedaan performa yang signifikan antara arsitektur *NeuralNetworkV1* (1 *hidden layer*) dan *NeuralNetworkV2* (2 *hidden layers*). Model dengan 1 *hidden layer* menunjukkan *accuracy* dalam rentang 81.34%-94.69%, sementara model dengan 2 *hidden layers* mencapai rentang yang lebih tinggi yaitu 93.51%-96.61%.



Gambar 4. 4 Boxplot Distribusi Nilai *Accuracy* Semua Model

Model dengan *accuracy* terbaik adalah 2-CB-X dengan *mean accuracy* 96.61% $\pm$ 0.43%, diikuti oleh 2-CA-X (96.24% $\pm$ 0.64%) dan 2-BA-X (96.17% $\pm$ 0.37%). Ketiga model ini menggunakan *learning rate* 0.1, mengindikasikan bahwa *learning rate* yang lebih tinggi mampu mengoptimalkan konvergensi pada arsitektur 2 *hidden layers*. Standar deviasi yang rendah pada model-model terbaik (0.37%-0.64%) juga menunjukkan konsistensi performa yang tinggi di seluruh *fold*.

Sebaliknya, model dengan *learning rate* 0.01 (skenario Z) menunjukkan performa terendah di masing-masing kelompok arsitektur. Model 1-C-Z, 1-B-Z, dan 1-A-Z mencapai *accuracy* hanya sekitar 81-82%, dengan standar deviasi yang lebih tinggi (1.60%-1.99%), mengindikasikan konvergensi yang buruk akibat *learning rate* yang terlalu kecil. Pada arsitektur 2 *hidden layers*, *learning rate* 0.01 juga menghasilkan penurunan performa yang signifikan, meskipun tetap lebih baik dibanding arsitektur 1 *hidden layer*.

Variabilitas performa juga terlihat dari distribusi boxplot. Model-model dengan performa rendah (kelompok Z) menunjukkan rentang *interquartile* yang lebih lebar dan terdapat *outlier* pada beberapa *fold*, mengindikasikan ketidakstabilan model. Sementara itu, model-model terbaik (2-CB-X, 2-CA-X, 2-BA-X) menunjukkan boxplot yang kompak dengan whisker yang pendek, mencerminkan konsistensi tinggi di seluruh *fold testing*. Untuk memvalidasi signifikansi perbedaan performa *accuracy* antar model secara statistik, dilakukan uji T-Test dengan membandingkan model terbaik yaitu 2-CB-X terhadap 17 model lainnya. Hasil perbandingan tersebut disajikan pada Tabel 4.43 berikut.

Tabel 4. 43 Hasil Uji T-Test *Accuracy* Model 2-CB-X terhadap Model Lainnya

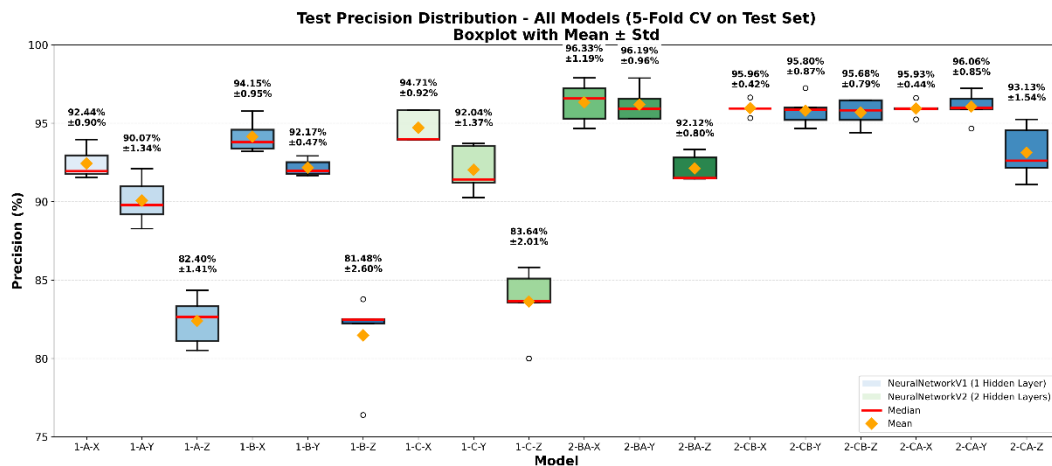
Model	Mean Diff	t-statistic	p-value	Tingkat Signifikansi
Model 1-A-Z	14.8242	14.6729	0.000126	*** (p<0.001)
Model 1-C-Z	15.0453	14.5133	0.000131	*** (p<0.001)
Model 1-B-Z	15.267	13.6167	0.000168	*** (p<0.001)
Model 1-C-X	1.9175	8.9066	0.000878	*** (p<0.001)
Model 1-B-Y	5.9011	7.3607	0.001815	** (p<0.01)
Model 1-C-Y	5.4575	6.9379	0.002266	** (p<0.01)
Model 2-BA-Z	3.0972	6.5183	0.00286	** (p<0.01)
Model 1-A-Y	6.858	5.8336	0.004303	** (p<0.01)
Model 1-A-X	4.0569	5.1776	0.006617	** (p<0.01)
Model 1-B-X	2.6555	4.9146	0.00796	** (p<0.01)
Model 2-CA-Z	2.5073	4.5405	0.010493	* (p<0.05)
Model 2-CB-Z	2.0653	3.4965	0.024974	* (p<0.05)
Model 2-CA-Y	0.5163	3.4965	0.024974	* (p<0.05)

Model	Mean Diff	t-statistic	p-value	Tingkat Signifikansi
Model 2-BA-Y	0.5901	2.6646	0.056119	ns ($p \geq 0.05$)
Model 2-CB-Y	0.6637	2.091	0.104718	ns ($p \geq 0.05$)
Model 2-BA-X	0.4425	2.0565	0.108884	ns ($p \geq 0.05$)
Model 2-CA-X	0.3693	1.5838	0.188421	ns ($p \geq 0.05$)

Berdasarkan Tabel 4.43, hasil uji T-Test menunjukkan pola yang sangat jelas dalam signifikansi perbedaan *accuracy* antar kelompok model. Sebanyak 13 dari 17 model menunjukkan perbedaan yang signifikan ($p < 0.05$) terhadap 2-CB-X, sementara 4 model lainnya tidak signifikan. Seluruh model *NeuralNetworkV1* dan model skenario Z menunjukkan perbedaan yang sangat signifikan, di mana model 1-A-Z, 1-C-Z, dan 1-B-Z menghasilkan p-value mendekati nol ($p < 0.001$) dengan selisih *accuracy* mencapai 14–15%, mengkonfirmasi bahwa *learning rate* 0.01 terlalu rendah untuk mencapai konvergensi optimal. Perbandingan 2-CB-X terhadap 1-C-X sebagai model terbaik *NeuralNetworkV1* juga menghasilkan perbedaan yang signifikan ($p = 0.0009$), menunjukkan bahwa keunggulan arsitektur 2 *hidden layers* bukan sekadar perbedaan numerik. Di sisi lain, perbandingan 2-CB-X terhadap sesama model *NeuralNetworkV2* berkinerja tinggi seperti 2-BA-Y ($p = 0.0561$), 2-CB-Y ($p = 0.1047$), 2-BA-X ($p = 0.1089$), dan 2-CA-X ($p = 0.1884$) semuanya tidak signifikan, mengindikasikan bahwa model-model ini berada pada level *accuracy* yang setara secara statistik.

Selain *accuracy*, evaluasi terhadap metrik *precision* memberikan perspektif yang berbeda tentang kemampuan model dalam meminimalkan false alarm. *Precision* mengukur proporsi prediksi positif (tumor) yang benar dari keseluruhan prediksi positif yang dilakukan model. Metrik ini sangat penting dalam konteks medis untuk meminimalkan *false alarm* (*False Positive*) yang dapat menyebabkan

kecemasan pasien dan tindakan medis yang tidak perlu. Distribusi nilai *precision* tersebut sebagaimana tertera pada Gambar 4.5.



Gambar 4. 5 Boxplot Distribusi Nilai *Precision* Semua Model

Model dengan *precision* tertinggi adalah 2-BA-X dengan *mean* 96.33% ± 1.19%, diikuti oleh 2-CA-Y (96.06% ± 0.85%) dan 2-CB-X (95.96% ± 0.42%). Meskipun 2-CB-X memiliki *accuracy* tertinggi, *precision*-nya sedikit lebih rendah dibanding 2-BA-X, mengindikasikan *trade-off* antara deteksi tumor yang maksimal dengan meminimalkan *false alarm*. Model 2-BA-X menunjukkan keseimbangan terbaik dengan *precision* tinggi dan standar deviasi yang masih dalam batas wajar.

Arsitektur *NeuralNetworkV1* menunjukkan *precision* yang lebih bervariasi, dengan rentang 81.48%-94.71%. Model 1-C-X mencapai *precision* tertinggi di kelompok ini (94.71% ± 0.92%), menunjukkan bahwa peningkatan neuron pada *hidden layer* (256 neuron) mampu meningkatkan kemampuan diskriminasi model terhadap kelas tumor. Namun, model dengan *learning rate* rendah (kelompok Z) kembali menunjukkan performa yang buruk, dengan 1-B-Z mencatat *precision* terendah 81.48% ± 2.60%. Selanjutnya, uji *T-Test precision* dilakukan dengan

membandingkan model terbaik yaitu 2-BA-X terhadap 17 model lainnya, dengan hasil yang disajikan pada Tabel 4.44 berikut.

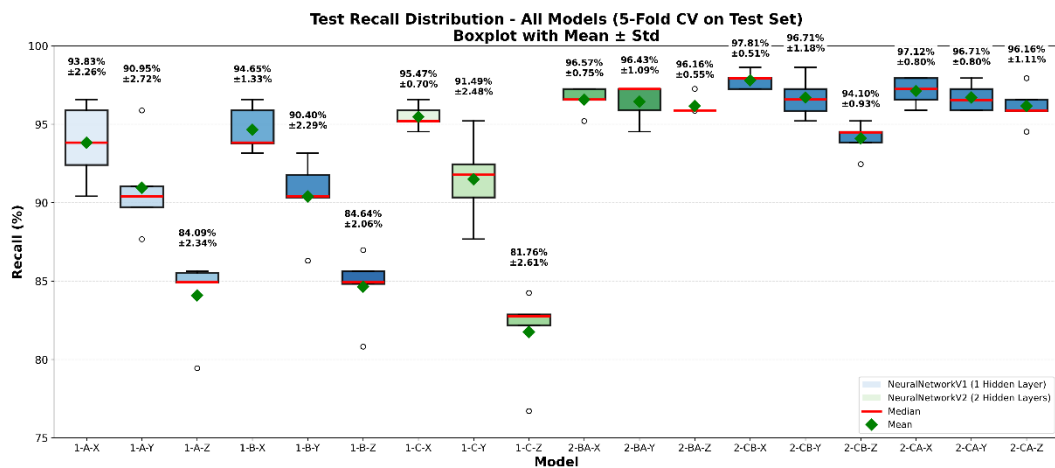
Tabel 4. 44 Hasil Uji *T-Test Precision* Model 2-BA-X terhadap Model Lainnya

Model	Mean Diff	t-statistic	p-value	Tingkat Signifikansi
Model 1-C-Y	4.2943	17.4601	6.30E-05	*** (p<0.001)
Model 1-A-Z	13.9323	14.7902	0.000122	*** (p<0.001)
Model 1-B-Z	14.8501	9.7077	0.00063	*** (p<0.001)
Model 1-C-Z	12.6952	9.3967	0.000715	*** (p<0.001)
Model 2-CA-Z	3.2001	8.7424	0.000943	*** (p<0.001)
Model 1-B-Y	4.1621	6.5883	0.002749	** (p<0.01)
Model 1-A-Y	6.263	6.3185	0.003209	** (p<0.01)
Model 2-BA-Z	4.2112	6.2708	0.003301	** (p<0.01)
Model 1-A-X	3.8901	4.2393	0.013271	* (p<0.05)
Model 1-B-X	2.1833	3.0656	0.037455	* (p<0.05)
Model 1-C-X	1.6217	2.6795	0.055255	ns (p>0.05)
Model 2-CB-Z	0.6503	1.1407	0.317671	ns (p>0.05)
Model 2-CA-X	0.3968	0.752	0.493896	ns (p>0.05)
Model 2-CB-X	0.3676	0.6517	0.550154	ns (p>0.05)
Model 2-CB-Y	0.5356	0.6379	0.558215	ns (p>0.05)
Model 2-CA-Y	0.2699	0.5124	0.63536	ns (p>0.05)
Model 2-BA-Y	0.1414	0.2777	0.795007	ns (p>0.05)

Berdasarkan Tabel 4.44, hasil uji *T-Test precision* menunjukkan pola signifikansi yang sedikit berbeda dibanding *accuracy*. Sebanyak 10 dari 17 model menunjukkan perbedaan yang signifikan ($p < 0.05$) terhadap 2-BA-X, sementara 7 model lainnya tidak signifikan. Model-model dengan *learning rate* rendah dan seluruh arsitektur *NeuralNetworkV1* kecuali 1-C-X menunjukkan perbedaan yang signifikan, di mana 1-C-Y menghasilkan t-statistic tertinggi (17.46) dengan p-value 0.000063, diikuti oleh kelompok skenario Z yang juga sangat signifikan ($p < 0.001$). Menariknya, model 1-C-X sebagai model terbaik *NeuralNetworkV1* berada di ambang signifikansi ($p = 0.0553$), mengindikasikan bahwa gap *precision* antara arsitektur terbaik V1 dan V2 lebih kecil dibandingkan gap pada metrik *accuracy*. Perbandingan 2-BA-X terhadap sesama model *NeuralNetworkV2* berkinerja tinggi

seperti 2-CB-X ($p = 0.5502$), 2-CA-X ($p = 0.4939$), 2-CA-Y ($p = 0.6354$), 2-CB-Y ($p = 0.5582$), dan 2-BA-Y ($p = 0.7950$) semuanya tidak signifikan, mengkonfirmasi bahwa model-model top tier berada pada level *precision* yang setara secara statistik.

Selain metrik *accuracy* dan *precision*, terdapat juga metrik *recall* yang mengukur proporsi kasus tumor yang berhasil dideteksi dari keseluruhan kasus tumor yang sebenarnya ada. Metrik ini sangat krusial dalam aplikasi medis karena kegagalan mendeteksi tumor (*False Negative*) dapat berakibat fatal bagi pasien dengan menunda diagnosis dan pengobatan. Distribusi nilai *Recall* semua model sebagaimana tertera pada Gambar 4.6.



Gambar 4. 6 Boxplot Distribusi Nilai *Recall* Semua Model

Model dengan *recall* tertinggi adalah 2-CB-X dengan *mean* 97.81% ± 0.51%, menunjukkan kemampuan luar biasa dalam mendeteksi hampir semua kasus tumor pada *test set*. Hal ini mengindikasikan bahwa model 2-CB-X sangat sensitif terhadap karakteristik tumor dan meminimalkan *missed detection*. Model 2-CA-X (97.12% ± 0.80%) dan 2-CB-Y (96.71% ± 1.18%) juga menunjukkan

recall yang sangat tinggi, menjadikan arsitektur dengan konfigurasi neuron yang lebih besar (256 neuron pada *layer* pertama) lebih efektif dalam deteksi tumor.

Menariknya, beberapa model dengan *accuracy* sedang menunjukkan *recall* yang sangat baik. Model 2-BA-Z mencapai *recall* $96.16\% \pm 0.55\%$ meskipun *accuracy*-nya hanya 93.51% , mengindikasikan bahwa model ini lebih cenderung memprediksi tumor (*high sensitivity*) dengan *trade-off precision* yang lebih rendah (92.12%). Pola ini juga terlihat pada 1-B-Z (*recall* 84.64% vs *precision* 81.48%), menunjukkan bahwa *learning rate* rendah membuat model lebih konservatif dalam memprediksi kelas *healthy*.

Distribusi boxplot *recall* menunjukkan pola yang berbeda dibanding metrik lain. Model-model terbaik (kelompok 2-C dan 2-CA) menunjukkan boxplot yang sangat kompak di area atas 96% , dengan *whisker* yang pendek dan tanpa outlier, mengindikasikan konsistensi tinggi dalam mendeteksi tumor di seluruh *fold*. Sebaliknya, model dengan performa rendah seperti 1-C-Z ($81.76\% \pm 2.61\%$) menunjukkan distribusi yang lebih lebar dengan beberapa *fold* menghasilkan *recall* di bawah 80% , yang sangat berbahaya dalam konteks medis karena lebih dari 1 dari 5 kasus tumor tidak terdeteksi. Uji *T-Test recall* dilakukan dengan membandingkan model terbaik yaitu 2-CB-X terhadap 17 model lainnya, dengan hasil yang disajikan pada Tabel 4.45 berikut.

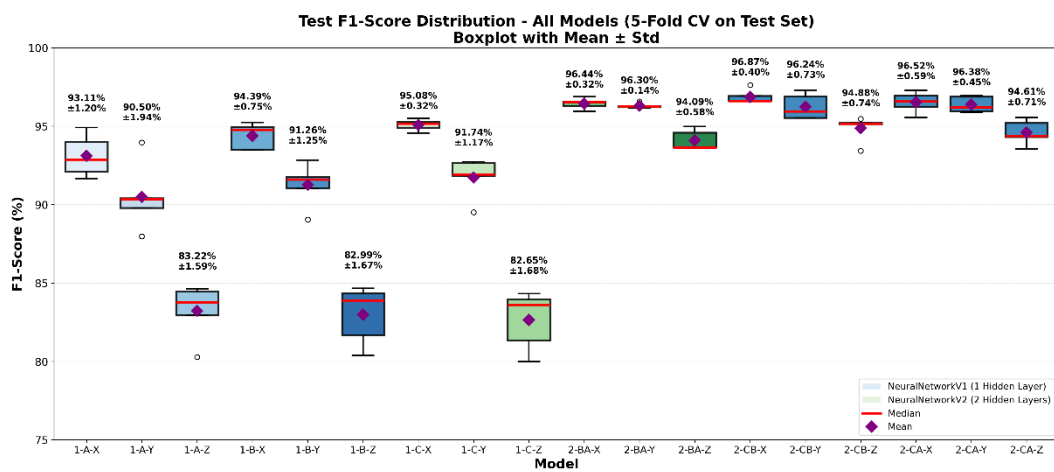
Tabel 4. 45 Hasil Uji T-Test Recall Model 2-CB-X terhadap Model Lainnya

Model	Mean Diff	t-statistic	p-value	Tingkat Signifikansi
Model 1-B-Z	13.1686	10.8533	0.000409	*** (p<0.001)
Model 1-C-Z	16.0482	10.7623	0.000423	*** (p<0.001)
Model 1-A-Z	13.7156	10.0429	0.000553	*** (p<0.001)
Model 2-CB-Z	3.7034	5.8349	0.0043	** (p<0.01)
Model 1-B-Y	7.4077	5.5911	0.005021	** (p<0.01)
Model 1-A-Y	6.8588	4.4341	0.011386	* (p<0.05)
Model 1-C-Y	6.3118	4.3183	0.012464	* (p<0.05)

Model	Mean Diff	t-statistic	p-value	Tingkat Signifikansi
Model 1-B-X	3.1564	4.2679	0.012972	* (p<0.05)
Model 1-C-X	2.3325	4.1866	0.013845	* (p<0.05)
Model 2-BA-Z	1.6467	3.5399	0.024016	* (p<0.05)
Model 2-CA-X	0.6859	3.1666	0.033968	* (p<0.05)
Model 2-CA-Y	1.0978	3.1399	0.034851	* (p<0.05)
Model 2-BA-X	1.2348	3.0904	0.03656	* (p<0.05)
Model 1-A-X	3.9802	3.0407	0.038376	* (p<0.05)
Model 2-BA-Y	1.3708	2.8323	0.047233	* (p<0.05)
Model 2-CB-Y	1.0987	2.6628	0.056225	ns (p=0.05)
Model 2-CA-Z	1.6467	2.3337	0.079926	ns (p=0.05)

Berdasarkan Tabel 4.45, hasil uji *T-Test recall* menunjukkan pola signifikansi yang unik dibanding metrik lainnya. Sebanyak 15 dari 17 model menunjukkan perbedaan yang signifikan ($p < 0.05$) terhadap 2-CB-X, dengan hanya 2-CB-Y ($p = 0.0562$) dan 2-CA-Z ($p = 0.0799$) yang tidak signifikan. Ini merupakan proporsi signifikansi tertinggi dibanding metrik lainnya, mengkonfirmasi bahwa keunggulan 2-CB-X pada *recall* bersifat lebih konsisten dan menyeluruh. Ketiga model skenario Z dari *NeuralNetworkV1* menunjukkan perbedaan yang sangat signifikan ($p < 0.001$) dengan selisih *recall* mencapai 13–16%, menegaskan bahwa *learning rate* 0.01 sangat tidak optimal untuk mencapai sensitivitas yang diperlukan dalam deteksi tumor. Berbeda dengan *precision*, pada metrik *recall* ini bahkan beberapa model *NeuralNetworkV2* berkinerja tinggi pun menunjukkan perbedaan yang signifikan terhadap 2-CB-X, seperti 2-CA-X ($p = 0.0340$), 2-BA-X ($p = 0.0366$), dan 2-BA-Y ($p = 0.0472$), menunjukkan bahwa 2-CB-X memiliki keunggulan *recall* yang terukur bahkan dibanding sesama model top tier. Hanya 2-CB-Y yang berada di ambang signifikansi ($p = 0.0562$), mengindikasikan perbedaan yang marginal antara kedua model tersebut.

Sebagai metrik yang mengintegrasikan *precision* dan *recall* secara bersamaan, *F1-score* memberikan gambaran paling komprehensif tentang keseimbangan performa model secara keseluruhan. Metrik ini sangat relevan untuk evaluasi model medis karena keseimbangan antara kedua aspek tersebut sangat penting dalam praktik klinis. Distribusi nilai *F1-score* pada semua model sebagaimana tertera pada Gambar 4.10.



Gambar 4. 7 Boxplot Distribusi Nilai *F1-Score* Semua Model

Model dengan *F1-score* tertinggi adalah 2-CB-X dengan *mean* 96.87% $\pm$ 0.40%, diikuti oleh 2-CA-X (96.52% $\pm$ 0.59%) dan 2-BA-X (96.44% $\pm$ 0.32%). Ketiga model ini menunjukkan keunggulan konsisten di semua metrik evaluasi, mengkonfirmasi bahwa arsitektur 2 *hidden layers* dengan *learning rate* 0.1 menghasilkan performa optimal yang seimbang. Standar deviasi yang sangat rendah (0.32%-0.59%) menunjukkan bahwa model-model ini menghasilkan prediksi yang konsisten dan dapat diandalkan di berbagai kondisi data.

Model 2-BA-Y mencatat standar deviasi *F1-score* terendah (0.14%), mengindikasikan konsistensi tertinggi di antara semua model meskipun *mean F1-*

score-nya sedikit lebih rendah (96.30%) dibanding 2-CB-X. Hal ini menunjukkan bahwa *learning rate* 0.05 pada arsitektur 2-BA menghasilkan konvergensi yang lebih stabil dengan variabilitas minimal di seluruh *fold*.

Arsitektur *NeuralNetworkV1* menunjukkan *F1-score* dalam rentang 82.65%-95.08%, dengan model 1-C-X mencapai performa terbaik (95.08% $\pm$ 0.32%) di kelompok ini. Performa ini mendekati beberapa model *NeuralNetworkV2* kelas bawah seperti 2-BA-Z (94.09%) dan 2-CB-Z (94.88%), menunjukkan bahwa dengan konfigurasi optimal (256 neuron, *learning rate* 0.1), arsitektur 1 *hidden layer* masih dapat menghasilkan performa yang kompetitif. Namun, *ceiling effect* tetap terlihat jelas dengan tidak ada model *NeuralNetworkV1* yang mampu melampaui 95.5%.

Distribusi boxplot *F1-score* menunjukkan pola yang sangat informatif tentang trade-off *precision-recall*. Model seperti 2-CB-X menunjukkan boxplot yang sangat kompak di area atas dengan median yang hampir identik dengan *mean*, mengindikasikan distribusi yang sangat simetris dan performa yang konsisten tinggi. Sebaliknya, model dengan *learning rate* rendah (kelompok Z) menunjukkan boxplot yang lebih lebar dengan beberapa outlier, terutama pada 1-A-Z (83.22% $\pm$ 1.59%) dan 1-C-Z (82.65% $\pm$ 1.68%), mengkonfirmasi bahwa *learning rate* 0.01 tidak cukup untuk mencapai konvergensi optimal pada dataset ini.

Variabilitas *F1-score* juga mencerminkan keseimbangan *precision-recall* yang berbeda di setiap model. Model dengan *recall* tinggi tapi *precision* sedang (seperti 2-BA-Z dengan *recall* 96.16% dan *precision* 92.12%) menunjukkan *F1-score* 94.09% yang lebih rendah dibanding model dengan keseimbangan lebih baik

seperti 2-CA-Y (*recall* 96.71%, *precision* 96.06%, F1 96.38%). Hal ini mengindikasikan bahwa keseimbangan antara kedua metrik lebih penting daripada memaksimalkan salah satunya untuk mencapai *F1-score* yang optimal. Selanjutnya, uji *T-Test F1-score* dilakukan dengan membandingkan model terbaik yaitu 2-CB-X terhadap 17 model lainnya, dengan hasil yang disajikan pada Tabel 4.46 berikut.

Tabel 4. 46 Hasil Uji *T-Test F1-score* Model 2-CB-X terhadap Model Lainnya

Model	Mean Diff	t-statistic	p-value	Tingkat Signifikansi
Model 1-B-Z	13.8817	14.1808	0.000144	*** (p<0.001)
Model 1-C-Z	14.2230	13.8198	0.000159	*** (p<0.001)
Model 1-A-Z	13.6539	13.7989	0.00016	*** (p<0.001)
Model 1-C-X	1.7914	8.6544	0.000981	*** (p<0.001)
Model 1-B-Y	5.6144	7.0021	0.00219	** (p<0.01)
Model 1-C-Y	5.1394	6.6099	0.002716	** (p<0.01)
Model 2-BA-Z	2.7806	6.4312	0.003006	** (p<0.01)
Model 1-A-Y	6.3781	5.7102	0.004651	** (p<0.01)
Model 1-A-X	3.763	4.9407	0.007813	** (p<0.01)
Model 1-B-X	2.4854	4.9249	0.007902	** (p<0.01)
Model 2-CA-Z	2.2685	4.5114	0.010729	* (p<0.05)
Model 2-CA-Y	0.4966	3.9431	0.016915	* (p<0.05)
Model 2-CB-Z	1.9923	3.6069	0.022619	* (p<0.05)
Model 2-BA-Y	0.5739	2.9228	0.043124	* (p<0.05)
Model 2-BA-X	0.4337	2.2831	0.084501	ns (p>0.05)
Model 2-CB-Y	0.6317	2.1359	0.099542	ns (p>0.05)
Model 2-CA-X	0.3524	1.6352	0.177347	ns (p>0.05)

Berdasarkan Tabel 4.46, hasil uji *T-Test F1-score* menunjukkan pola yang paling komprehensif sebagai representasi keseluruhan performa model. Sebanyak 14 dari 17 model menunjukkan perbedaan yang signifikan ($p < 0.05$) terhadap 2-CB-X, dengan hanya 3 model yang tidak signifikan. Seluruh model *NeuralNetworkV1* dan model skenario Z menunjukkan perbedaan yang sangat signifikan, di mana 1-B-Z, 1-C-Z, dan 1-A-Z menghasilkan p-value mendekati nol ($p < 0.001$) dengan selisih *F1-score* lebih dari 13%, dan 1-C-X sebagai model

terbaik V1 pun masih menunjukkan perbedaan yang sangat signifikan ($p = 0.0010$). Pada kelompok *NeuralNetworkV2*, model 2-BA-Z ($p = 0.0030$) dan 2-CB-Z ($p = 0.0226$) menunjukkan perbedaan yang signifikan, mengkonfirmasi bahwa *learning rate* 0.1 memberikan keunggulan yang terukur dibanding 0.01 bahkan pada arsitektur yang sama. Menariknya, 2-CA-Y ($p = 0.0169$) dan 2-BA-Y ($p = 0.0431$) juga signifikan meskipun perbedaan *mean F1-score*-nya hanya sekitar 0.49%–0.57%, yang menunjukkan bahwa standar deviasi 2-CB-X yang sangat rendah (0.40%) membuat perbedaan kecil pun cukup untuk menghasilkan signifikansi statistik. Sebaliknya, 2-BA-X ($p = 0.0845$), 2-CB-Y ($p = 0.0995$), dan 2-CA-X ($p = 0.1773$) tidak signifikan, mengindikasikan bahwa ketiga model ini berada pada level *F1-score* yang setara secara statistik dengan 2-CB-X.

Berdasarkan analisis komprehensif terhadap keempat metrik evaluasi dan validasi signifikansi statistik melalui uji *T-Test*, dapat disimpulkan beberapa temuan kunci untuk mengidentifikasi model terbaik dalam deteksi tumor otak menggunakan fitur GLCM pada citra MRI. Hasil analisis menunjukkan konsistensi dari arsitektur *NeuralNetworkV2* dibanding *NeuralNetworkV1* di seluruh metrik evaluasi. Uji *T-Test* mengkonfirmasi bahwa hampir semua perbandingan *cross-architecture* menghasilkan *p-value* kurang dari 0.05, dengan sebagian besar bahkan mencapai *p-value* kurang dari 0.01. *Gap* performa rata-rata antara kedua arsitektur berkisar 3-5% pada *accuracy*, *precision*, dan *F1-score*, serta 2-4% pada *recall*.

Fenomena menarik terungkap dari analisis *T-Test* pada kelompok model-model terbaik, yaitu 2-CB-X, 2-CA-X, 2-BA-X, 2-CA-Y, dan 2-BA-Y. Sebagian besar perbandingan di antara model-model ini menghasilkan *p-value* lebih dari

0.05, mengindikasikan bahwa meskipun terdapat perbedaan numerik kecil pada nilai *mean*, model-model ini berada pada level performa yang setara secara statistik. Hal ini menunjukkan bahwa dengan arsitektur 2 *hidden layers* dan *learning rate* yang optimal pada rentang 0.05 hingga 0.1, performa model telah mencapai *ceiling effect* yang sulit untuk ditingkatkan lebih lanjut hanya dengan mengubah konfigurasi jumlah neuron.

Namun demikian, meskipun secara statistik setara, terdapat perbedaan karakteristik yang penting untuk dipertimbangkan dalam pemilihan model berdasarkan prioritas aplikasi klinis. Model 2-CB-X dengan *accuracy* 96.61%, *precision* 95.96%, *recall* 97.81%, dan *F1-score* 96.87% menunjukkan keunggulan pada metrik *recall* yang signifikan lebih tinggi dari semua model lain dengan *p-value* kurang dari 0.05 pada sebagian besar perbandingan, menjadikannya pilihan terbaik untuk aplikasi yang memprioritaskan minimisasi *missed detection*. Model 2-BA-X dengan *accuracy* 96.17%, *precision* 96.33%, *recall* 96.57%, dan *F1-score* 96.44% unggul pada metrik *precision* meskipun perbedaan dengan 2-CB-X tidak signifikan secara statistik dengan *p-value* 0.5502, namun tetap menjadikannya pilihan yang lebih sesuai untuk aplikasi yang memprioritaskan minimisasi false alarm. Model 2-CA-X dengan *accuracy* 96.24%, *precision* 95.93%, *recall* 97.12%, dan *F1-score* 96.52% menunjukkan keseimbangan terbaik antara semua metrik dengan konsistensi tinggi di seluruh *fold*, menjadikannya pilihan yang cocok untuk aplikasi *general-purpose*. Sementara itu, model 2-BA-Y dengan *accuracy* 96.02%, *precision* 96.19%, *recall* 96.43%, dan *F1-score* 96.30% memiliki standar deviasi terendah berkisar 0.14% hingga 0.15%, mengindikasikan konsistensi tertinggi di

antara semua model dan menjadikannya pilihan ideal untuk deployment yang membutuhkan prediksi sangat stabil di berbagai kondisi data.

Berdasarkan seluruh pertimbangan di atas, model 2-CB-X diidentifikasi sebagai model terbaik secara keseluruhan dengan beberapa justifikasi kuat. Pertama, dari segi performa. Model ini menempati nilai tertinggi pada tiga dari empat metrik evaluasi yaitu *accuracy* dengan nilai 96.61%, *recall* dengan nilai 97.81%, dan *F1-score* dengan nilai 96.87%, serta menempati ranking kedua pada *precision* dengan nilai 95.96%. Kedua, *recall* tertinggi yang dicapai model ini yaitu 97.81% dengan standar deviasi sangat rendah 0.51% terbukti signifikan secara statistik lebih tinggi dibanding semua model lain dengan *p-value* kurang dari 0.05 pada sebagian besar perbandingan, termasuk perbandingan dengan 2-CA-X yang menghasilkan *p-value* 0.0340 dan 2-CB-Y dengan *p-value* 0.0562. Ketiga, konsistensi yang ditunjukkan melalui standar deviasi yang sangat rendah di semua metrik berkisar 0.40% hingga 0.51% mengindikasikan performa yang stabil dan dapat diandalkan di berbagai *subset data*, yang merupakan karakteristik penting untuk aplikasi klinis. Keempat, *F1-score* optimal sebesar 96.87% menunjukkan keseimbangan terbaik antara *precision* dan *recall*, mengkonfirmasi bahwa model tidak hanya sensitif terhadap karakteristik tumor tetapi juga mampu meminimalkan *false alarm* yang dapat menyebabkan kecemasan pasien dan prosedur *follow-up* yang tidak perlu.

Keberhasilan sistem deteksi tumor otak yang dikembangkan dalam penelitian ini merupakan wujud nyata dari ikhtiar manusia untuk mencari sebab-sebab kesembuhan (*asbāb asy-syifā'*) sebagaimana yang diperintahkan oleh syariat,

tanpa mengabaikan hakikat bahwa penyembuh sejati hanyalah Allah SWT. Allah SWT berfirman dalam Surat Asy-Syu'arā' ayat 80:

وَإِذَا مَرِضْتُ فَهُوَ يَشْفِينِ ﴿٨٠﴾

"Apabila aku sakit, Dialah yang menyembuhkanku" (QS. As-Syuara : 80).

Ayat ini mengandung adab mulia seorang hamba terhadap Tuhannya, dimana ketika sakit menimpa, seorang mukmin menyadari bahwa yang berhak menyembuhkan, baik melalui sebab (dokter, obat, teknologi) maupun tanpa sebab, hanyalah Allah semata. Dalam konteks penelitian ini, sistem deteksi bukanlah penyembuh sejati, melainkan alat bantu diagnostik yang Allah sediakan melalui kemampuan akal manusia untuk mengolah pengetahuan. Kesadaran ini melahirkan penghayatan mendalam terhadap nikmat kesembuhan setelah sakit, sekaligus menjadi pengingat akan pentingnya menjaga kesehatan sebagai amanah ilahi, sehingga pengembangan teknologi kecerdasan buatan untuk deteksi dini tumor otak bukanlah bentuk pengingkaran terhadap kekuasaan Allah sebagai Asy-Syāfi, melainkan justru wujud tawakkal yang sempurna dengan berusaha maksimal melalui pemanfaatan ilmu pengetahuan yang telah Allah anugerahkan (Fateha & Mansur, 2025).

Selain itu, keberhasilan sistem deteksi tumor otak berbasis *neural network* dalam penelitian ini juga merupakan salah satu bentuk ikhtiar nyata umat Islam untuk kembali mengamalkan perintah Al-Qur'an dalam menguasai ilmu pengetahuan dan teknologi demi kemaslahatan umat. Allah berfirman dalam Surat Al-Isrā' ayat 82:

وَنُنَزِّلُ مِنَ الْقُرْآنِ مَا هُوَ شِفَاءٌ وَرَحْمَةٌ لِّلْمُؤْمِنِينَ وَلَا يَرْيَدُ الظَّالِمِينَ إِلَّا خَسَارًا ﴿٨٢﴾

"Kami turunkan dari Al-Qur'an sesuatu yang menjadi penawar dan rahmat bagi orang-orang mukmin, sedangkan bagi orang-orang zalim (Al-Qur'an itu) hanya akan menambah kerugian" (QS. Al-Isra : 82).

Ayat ini menunjukkan bahwa Al-Qur'an adalah *syifā'* (penyembuh) terutama bagi penyakit hati seperti kesyirikan, kekafiran, kemunafikan, kebodohan, dan perpecahan, sekaligus menjadi rahmat yang melimpah bagi kaum mukmin yang mengamalkan kandungannya karena Al-Qur'an membebaskan mereka dari kegelapan jahiliah menuju puncak peradaban (Fateha & Mansur, 2025). Sejarah membuktikan bahwa ketika umat Islam benar-benar berpegang teguh pada Al-Qur'an, mereka menjadi pelopor ilmu pengetahuan, kedokteran, matematika, dan teknologi pada masa Daulah Umayyah dan Abbasiyah, sehingga lahir ulama ahli sains seperti Ibnu Sina, Al-Biruni, dan Al-Khawarizmi. Ayat ini menjadi peringatan keras sekaligus motivasi bahwasanya jika umat Islam hari ini ingin kembali meraih kejayaan umat dan memberikan manfaat bagi manusia, termasuk dalam bidang kedokteran dan teknologi kesehatan, maka kuncinya adalah kembali kepada Al-Qur'an secara utuh, mengamalkan perintahnya untuk menuntut ilmu, bersatu, dan memanfaatkan akal yang telah Allah karuniakan.

Prinsip memanfaatkan karunia akal untuk menemukan *syifā'* (penyembuh) juga didukung oleh ayat-ayat lain, di mana Allah menunjukkan bukti nyata adanya khasiat penyembuhan yang telah Ia tanamkan dalam ciptaan-Nya di alam semesta. Allah berfirman dalam Surah An-Nahl ayat 69:

ثُمَّ كُلِّي مِنْ كُلِّ الثَّمَرَاتِ فَاسْلُكِي سُبُلَ رَبِّكِ ذُلُلًا يَخْرُجُ مِنْ بَطْنِهَا شَرَابٌ مُخْتَلِفٌ أَلْوَانُهُ ۖ فِيهِ شِفَاءٌ
لِلنَّاسِ إِنَّ فِي ذَلِكَ لَآيَةً لِّقَوْمٍ يَتَفَكَّرُونَ ﴿٦٩﴾

“Kemudian, makanlah (wahai lebah) dari segala (macam) buah-buahan lalu tempuhlah jalan-jalan Tuhanmu yang telah dimudahkan (bagimu).” Dari perutnya itu keluar minuman (madu) yang beraneka warnanya. Di dalamnya terdapat obat bagi manusia. Sesungguhnya pada yang demikian itu benar-benar terdapat tanda (kebesaran Allah) bagi kaum yang berpikir” (QS. An-Nahl : 69).

Ayat ini mengandung pelajaran mendalam tentang bagaimana Allah telah menanamkan khasiat penyembuhan di berbagai ciptaan-Nya di alam semesta, baik melalui proses yang kasat mata seperti lebah menghasilkan madu yang terbukti mengandung berbagai senyawa bioaktif (Safarsyah, 2018). Tugas manusia sebagai makhluk yang dianugerahi akal adalah untuk meneliti, menemukan, dan memanfaatkan khasiat serta hukum-hukum tersebut.

Upaya manusia untuk meneliti dan mencari obat ini semakin diperkuat oleh sabda Rasulullah SAW dalam Shahih Muslim:

عَنْ رَسُولِ اللَّهِ ﷺ أَنَّهُ قَالَ لِكُلِّ دَاءٍ دَوَاءٌ ، فَإِذَا أُصِيبَ دَوَاءُ الدَّاءِ بَرَأَ بِإِذْنِ اللَّهِ .

“Dari Rasulullah SAW, bahwa beliau bersabda, Setiap penyakit ada obatnya. Maka apabila obat itu tepat mengenai penyakitnya, ia akan sembuh dengan izin Allah” (HR. Muslim).

Hadits ini memuat semangat optimisme medis dan menjadi landasan filosofis bagi kemajuan kedokteran, karena menanamkan keyakinan bahwa setiap penyakit (*dā'*) pasti memiliki solusi (*dawā'*-nya). Ulama seperti Al-Mazari menegaskan, kegagalan dalam penyembuhan di masa lalu bukan terletak pada ketidakberfungsian obat, melainkan pada minimnya pengetahuan dan kesalahan dalam diagnosa penyakit. Dalam konteks penelitian AI untuk deteksi tumor otak,

hadits ini memberikan legitimasi yang kuat. Tumor otak adalah salah satu *dā'* yang disebutkan, dan sistem AI berperan sebagai *syifā'* dengan memastikan *dawa'u ad-dā'* (obat yang tepat) dapat diusip (mengenai) penyakitnya melalui diagnosis yang cepat dan akurat. Dengan demikian, pengembangan teknologi AI untuk diagnosis medis bukan hanya kontribusi ilmiah, tetapi juga merupakan bentuk ibadah dan pengamalan langsung dari ajaran Rasulullah ﷺ dalam mencari obat bagi umat manusia (Ismiyah & Asy-Syifa, 2024).

Lebih dari itu, penelitian ini juga mencerminkan prinsip *taskhīr* (penundukan alam untuk manusia) yang disebutkan dalam Al-Qur'an. Allah telah menundukkan alam semesta beserta hukum-hukumnya untuk kepentingan manusia, termasuk hukum-hukum yang berkaitan dengan matematika, statistika, dan komputasi yang digunakan dalam membangun *neural network*. Hal ini merupakan salah satu bentuk manifestasi nyata dari firman Allah pada Surat Al-Jatsiyah ayat 13 :

وَسَخَّرَ لَكُم مَّا فِي السَّمٰوٰتِ وَمَا فِي الْاَرْضِ جَمِيعًا مِّنْهُۥ اِنَّ فِيْ ذٰلِكَ لَآيٰتٍ لِّقَوْمٍ يَّتَفَكَّرُوْنَ ﴿١٣﴾

"Dan Dia menundukkan untukmu apa yang ada di langit dan apa yang ada di bumi semuanya, (sebagai rahmat) dari-Nya" (QS. Al-Jātsiyah: 13).

Ayat ini menunjukkan bahwasanya Allah menundukkan segala ciptaan-Nya agar manusia dapat menggunakan dan memanfaatkannya untuk kepentingan mereka dalam melaksanakan tugas sebagai khalifah di bumi, dimana kunci dari semua ini adalah kemauan berusaha dan keinginan mengetahui sebagian dari pengetahuan Allah (Mubarok et al., 2025).

Dengan demikian, penelitian ini tidak hanya memiliki nilai akademis dan praktis, tetapi juga nilai spiritual yang mendalam. Ia menjadi bukti bahwa sains dan agama tidak bertentangan, melainkan saling melengkapi dalam misi bersama untuk meningkatkan kualitas hidup manusia dan memuliakan ciptaan Allah. Semoga penelitian ini dapat memberikan manfaat yang luas bagi umat manusia dan menjadi amal jariyah yang pahalanya terus mengalir.

BAB V

PENUTUP

5.1 Kesimpulan

Berdasarkan hasil pengujian dan evaluasi terhadap delapan belas skenario konfigurasi model *neural network* untuk deteksi tumor otak pada citra MRI menggunakan fitur tekstur GLCM, dapat disimpulkan bahwa kedalaman arsitektur merupakan faktor determinan yang secara signifikan mempengaruhi performa klasifikasi. Model dengan arsitektur dua *hidden layer* secara konsisten menunjukkan superioritas dibandingkan arsitektur satu *hidden layer*. Kelompok model dengan satu *hidden layer* hanya mampu mencapai rentang *accuracy* 81,34%–94,69% dengan variabilitas tinggi dan keberadaan outlier, sedangkan seluruh model dengan dua *hidden layer* secara konsisten mencapai akurasi di atas 93,51% dengan rentang 93,51%–96,61% dan standar deviasi rendah ($\pm 0,15\%$ – $\pm 0,81\%$).

Model 2-CB-X dengan arsitektur *NeuralNetworkV2* (256→128 neuron, aktivasi ReLU→Sigmoid, *learning rate* 0,1) ditetapkan sebagai model optimal berdasarkan evaluasi *5-Fold Stratified Cross-Validation* pada hold-out *test set*. Model ini mencapai *mean* akurasi 96,61% $\pm 0,43\%$, *precision* 95,96% $\pm 0,42\%$, *recall* 97,81% $\pm 0,51\%$, dan *F1-score* 96,87% $\pm 0,40\%$. Validasi statistik melalui uji *T-Test* berpasangan dengan tingkat signifikansi $\alpha = 0,05$ mengonfirmasi bahwa Model 2-CB-X memiliki perbedaan signifikan ($p\text{-value} < 0,05$) dengan seluruh model satu *hidden layer*, dengan selisih *mean* akurasi +1,92% hingga +15,27% dan nilai *t-statistic* tinggi, mengindikasikan keunggulan arsitektur dua *hidden layer*

dalam menangkap pola kompleks pada fitur GLCM. Sebaliknya, perbandingan dengan model lain dalam kelompok dua *hidden layer* menghasilkan perbedaan tidak signifikan ($p\text{-value} > 0,05$), dengan selisih *mean* akurasi hanya +0,02%–+3,10%, mengindikasikan bahwa berbagai konfigurasi dalam arsitektur dua *hidden layer* menghasilkan performa setara secara statistik. Dengan demikian, sistem deteksi tumor otak yang dikembangkan pada penelitian ini berpotensi diimplementasikan sebagai sistem pendukung keputusan diagnostik di fasilitas kesehatan dengan keterbatasan tenaga ahli radiologi.

5.2 Saran

Berdasarkan pengujian dan analisis model *Neural Network (NN)* yang telah dilakukan, serta menyadari adanya potensi pengembangan lebih lanjut, berikut adalah saran-saran untuk penelitian berikutnya guna meningkatkan kinerja dan validitas sistem:

1. *Dataset* yang digunakan dalam penelitian ini masih terbatas pada satu sumber dengan jumlah sampel yang relatif kecil. Karena keterbatasan penggunaan *dataset* ini, hasil yang didapatkan mungkin tidak maksimal jika kedepannya diterapkan pada *dataset* lainnya dengan karakteristik berbeda. Oleh karena itu, pada penelitian berikutnya disarankan untuk menambah variasi *dataset* citra MRI dari berbagai sumber seperti BraTS *Dataset* atau *dataset* publik lainnya, serta meningkatkan jumlah sampel data untuk masing-masing kelas guna meningkatkan kemampuan generalisasi model.
2. Metode ekstraksi fitur yang digunakan dalam penelitian ini masih terbatas pada GLCM yang mengekstraksi fitur tekstur secara manual. Penelitian selanjutnya

dapat mengeksplorasi metode ekstraksi fitur yang lebih kompleks seperti *Local Binary Pattern (LBP)*, *Histogram of Oriented Gradients (HOG)*, atau kombinasi dari beberapa metode ekstraksi fitur untuk mendapatkan representasi yang lebih kaya dan informatif dari citra MRI.

3. Arsitektur *Neural Network (NN)* yang digunakan dalam penelitian ini masih merupakan arsitektur Feedforward sederhana. Merujuk pada kesimpulan yang didapatkan bahwa penambahan kedalaman arsitektur dapat meningkatkan kinerja model secara signifikan, maka penelitian berikutnya dapat mengembangkan variasi jumlah *hidden layer*.

DAFTAR PUSTAKA

- Abdalla, H. E. M., & Esmail, M. Y. (2018). Brain Tumor Detection by using Artificial Neural Network. *2018 International Conference on Computer, Control, Electrical, and Electronics Engineering (ICCCEEE)*, 1–6. <https://doi.org/10.1109/ICCCEEE.2018.8515763>
- Afaq, S., & Rao, S. (2020). Significance Of Epochs On Training A Neural Network. *INTERNATIONAL JOURNAL OF SCIENTIFIC & TECHNOLOGY RESEARCH*, 9(06). www.ijstr.org
- Anantharajan, S., Gunasekaran, S., Subramanian, T., & R, V. (2024). MRI brain tumor detection using deep learning and machine learning approaches. *Measurement: Sensors*, 31. <https://doi.org/10.1016/j.measen.2024.101026>
- Asiri, A. A., Soomro, T. A., Shah, A. A., Pogrebna, G., Irfan, M., & Alqahtani, S. (2024). Optimized Brain Tumor Detection: A Dual-Module Approach for MRI Image Enhancement and Tumor Classification. *IEEE Access*, 12, 42868–42887. <https://doi.org/10.1109/ACCESS.2024.3379136>
- Chen, H., Wang, N., Du, X., Mei, K., Zhou, Y., & Cai, G. (2023). Classification Prediction of Breast Cancer Based on Machine Learning. *Computational Intelligence and Neuroscience*, 2023(1). <https://doi.org/10.1155/2023/6530719>
- Chen, X., Geng, Y., Wei, G., He, D., Lv, J., Wen, W., Xiang, F., Tao, K., & Wu, C. (2024). Neural Circuitries between the Brain and Peripheral Solid Tumors. *Cancer Research*, 84(21), 3509–3521. <https://doi.org/10.1158/0008-5472.CAN-24-1779>
- Dheepak, G., J., A. C., & Vaishali, D. (2024). Brain tumor classification: a novel approach integrating GLCM, LBP and composite features. *Frontiers in Oncology*, 13. <https://doi.org/10.3389/fonc.2023.1248452>
- Di Leo, G., & Sardanelli, F. (2020). Statistical significance: p value, 0.05 threshold, and applications to radiomics—reasons for a conservative approach. *European Radiology Experimental*, 4(1), 18. <https://doi.org/10.1186/s41747-020-0145-y>
- Dong, W. (2025). Brain Tumor Classification in MRI Images Based on EfficientNetB0: Impact of Learning Rate, Optimizer and Batch Size. *Science and Technology of Engineering, Chemistry and Environmental Protection*, 1(4). <https://doi.org/10.61173/zccy4f83>

- Faruq, A., Adeyani, B. R., & Syafaah, L. (2024). *Evaluation of learning rate training model on heart disease detection using LSTM*. 040016. <https://doi.org/10.1063/5.0192603>
- Fateha, T. N., & Mansur, S. (2025). Konstruksi Konsep Penyembuhan dalam Al-Qur'an. *Reflection : Islamic Education Journal*, 2(4), 42–52. <https://doi.org/10.61132/reflection.v2i4.1481>
- Ferreira, L. C. L., Costa, A. L. C., & Moreno, A. L. (2025). Classification of brain tumors in magnetic resonance images using a Convolutional Neural Network. *The Journal of Engineering and Exact Sciences*, 11(1), 21317. <https://doi.org/10.18540/jcecvl11iss1pp21317>
- Fikriah, F. K., Ariyanto, A. D. P., & Setyawan, A. F. (2024). KLASIFIKASI HASIL MRI TUMOR OTAK DENGAN EKTRAKSI FITUR GRAY LEVEL CO-OCCURANCE MATRIX (GLCM). *Rabit : Jurnal Teknologi Dan Sistem Informasi Univrab*, 9(2), 343–350. <https://doi.org/10.36341/rabit.v9i2.4793>
- Granzio, D., Zohren, S., & Roberts, S. (2020). Learning Rates as a Function of Batch Size: A Random Matrix Theory Approach to Neural Network Training. *Journal of Machine Learning Research* 1. <http://arxiv.org/abs/2006.09092>
- Handayani, P. W., Dartanto, T., Moeis, F. R., Pinem, A. A., Azzahro, F., Hidayanto, A. N., Denny, & Ayuningtyas, D. (2021). The regional and referral compliance of online healthcare systems by Indonesia National Health Insurance agency and health-seeking behavior in Indonesia. *Heliyon*, 7(9), e08068. <https://doi.org/10.1016/j.heliyon.2021.e08068>
- Ijaz, M., Hasan, I., Aslam, B., Yan, Y., Zeng, W., Gu, J., Jin, J., Zhang, Y., Wang, S., Xing, L., & Guo, B. (2025a). Diagnostics of brain tumor in the early stage: current status and future perspectives. *Biomaterials Science*, 13(10), 2580–2605. <https://doi.org/10.1039/D4BM01503G>
- Ijaz, M., Hasan, I., Aslam, B., Yan, Y., Zeng, W., Gu, J., Jin, J., Zhang, Y., Wang, S., Xing, L., & Guo, B. (2025b). Diagnostics of brain tumor in the early stage: current status and future perspectives. *Biomaterials Science*, 13(10), 2580–2605. <https://doi.org/10.1039/D4BM01503G>
- Ismiyah, N., & Asy-Syifa, A. M. (2024). Telemedicine dan Kesehatan: Memahami Dampak Teknologi Kedokteran di Era Digital dalam Konteks Ajaran Islam. *Jurnal Medika Nusantara*, 2(4), 45–55. <https://doi.org/10.59680/medika.v2i4.1478>

- Jabbar, A., Naseem, S., Mahmood, T., Saba, T., Alamri, F. S., & Rehman, A. (2023). Brain Tumor Detection and Multi-Grade Segmentation Through Hybrid Caps-VGGNet Model. *IEEE Access*, 11, 72518–72536. <https://doi.org/10.1109/ACCESS.2023.3289224>
- Kandel, I., & Castelli, M. (2020). The effect of batch size on the generalizability of the convolutional neural networks on a histopathology dataset. *ICT Express*, 6(4), 312–315. <https://doi.org/10.1016/j.icte.2020.04.010>
- Kang, J., Ullah, Z., & Gwak, J. (2021). MRI-Based Brain Tumor Classification Using Ensemble of Deep Features and Machine Learning Classifiers. *Sensors*, 21(6), 2222. <https://doi.org/10.3390/s21062222>
- Khaliki, M. Z., & Başarslan, M. S. (2024). Brain tumor detection from images and comparison with transfer learning methods and 3-layer CNN. *Scientific Reports*, 14(1), 2664. <https://doi.org/10.1038/s41598-024-52823-9>
- Khan, Md. S. I., Rahman, A., Debnath, T., Karim, Md. R., Nasir, M. K., Band, S. S., Mosavi, A., & Dehzangi, I. (2022a). Accurate brain tumor detection using deep convolutional neural network. *Computational and Structural Biotechnology Journal*, 20, 4733–4745. <https://doi.org/10.1016/j.csbj.2022.08.039>
- Khan, Md. S. I., Rahman, A., Debnath, T., Karim, Md. R., Nasir, M. K., Band, S. S., Mosavi, A., & Dehzangi, I. (2022b). Accurate brain tumor detection using deep convolutional neural network. *Computational and Structural Biotechnology Journal*, 20, 4733–4745. <https://doi.org/10.1016/j.csbj.2022.08.039>
- Kumar, T. (2025). Early Diagnosis of Brain Tumor Using AI and Mathematical Modeling. *INTERNATIONAL JOURNAL OF SCIENTIFIC RESEARCH IN ENGINEERING AND MANAGEMENT*, 09(01), 1–9. <https://doi.org/10.55041/IJSREM41114>
- Li, J. (2023). Asymptotics of K-Fold Cross Validation. *Journal of Artificial Intelligence Research*, 78, 491–526. <https://doi.org/10.1613/jair.1.13974>
- Mahmud, M. I., Mamun, M., & Abdelgawad, A. (2023). A Deep Analysis of Brain Tumor Detection from MR Images Using Deep Learning Networks. *Algorithms*, 16(4), 176. <https://doi.org/10.3390/a16040176>
- Marcot, B. G., & Hanea, A. M. (2021). What is an optimal value of k in k-fold cross-validation in discrete Bayesian network analysis? *Computational Statistics*, 36(3), 2009–2031. <https://doi.org/10.1007/s00180-020-00999-9>

- Mohammed, F. A., Tune, K. K., Assefa, B. G., Jett, M., & Muhie, S. (2024). Medical Image Classifications Using Convolutional Neural Networks: A Survey of Current Methods and Statistical Modeling of the Literature. *Machine Learning and Knowledge Extraction*, 6(1), 699–736. <https://doi.org/10.3390/make6010033>
- Mubarok, R., Armi, W. L., & SW, O. F. (2025). Kecerdasan Buatan Dalam Perspektif Agama Islam. *Al-Mau'izhoh: Jurnal Pendidikan Agama Islam*, 7. <https://doi.org/10.31949/am.v7i2.16819>
- Naseer, A., Yasir, T., Azhar, A., Shakeel, T., & Zafar, K. (2021). Computer-Aided Brain Tumor Diagnosis: Performance Evaluation of Deep Learner CNN Using Augmented Brain MRI. *International Journal of Biomedical Imaging*, 2021. <https://doi.org/10.1155/2021/5513500>
- Ostrom, Q. T., Price, M., Neff, C., Cioffi, G., Waite, K. A., Kruchko, C., & Barnholtz-Sloan, J. S. (2022). CBTRUS Statistical Report: Primary Brain and Other Central Nervous System Tumors Diagnosed in the United States in 2015–2019. *Neuro-Oncology*, 24(Supplement_5), v1–v95. <https://doi.org/10.1093/neuonc/noac202>
- Park, N., Wang, X., Simoulin, A., Yang, S., Yang, G., Rossi, R., Trivedi, P., & Ahmed, N. (2024). Forward Learning of Graph Neural Networks. *ArXiv, Abs/2403.11004*.
- Pichaivel, M., Anbumani, G., Theivendren, P., & Gopal, M. (2022). An Overview of Brain Tumor. In *Brain Tumors*. IntechOpen. <https://doi.org/10.5772/intechopen.100806>
- Pimpalkar, A., Tembhurne, P., Ingle, A., Gosawi, V., & Patle, P. (2023). BRAIN TUMOR DETECTION AND CLASSIFICATION USING MACHINE LEARNING: A COMPREHENSIVE SURVEY. *International Research Journal of Modernization in Engineering Technology and Science*. <https://doi.org/10.56726/IRJMETs41528>
- Rahman, T., & Islam, M. S. (2023). MRI brain tumor detection and classification using parallel deep convolutional neural networks. *Measurement: Sensors*, 26, 100694. <https://doi.org/10.1016/j.measen.2023.100694>
- Rahmartani, L. D., Budiarso, M. N., Ilmana, G., Muhaimin, R., Susanto, E., Amal, M. Y., Nuryadi, E., Rafli, A., & Gunawan, K. (2024). LMIC-22. TOWARD A COMPREHENSIVE PEDIATRIC BRAIN TUMOR REGISTRY: ESTABLISHING FUNDAMENTAL VALUE IN INDONESIA TERTIARY

HOSPITAL. *Neuro-Oncology*, 26(Supplement_4), 0–0.
<https://doi.org/10.1093/neuonc/noae064.739>

Saeedi, S., Rezayi, S., Keshavarz, H., & R. Niakan Kalhori, S. (2023). MRI-based brain tumor detection using convolutional deep learning methods and chosen machine learning techniques. *BMC Medical Informatics and Decision Making*, 23(1), 16. <https://doi.org/10.1186/s12911-023-02114-6>

Safarsyah, A. I. (2018). HADITS NABI SAW TENTANG OBAT DALAM TINJAUAN ILMU KEDOKTERAN MODERN. *AL-DZIKRA Jurnal Studi Ilmu Al-Qur'an Dan Al-Hadits*, 12(2), 165–188.
<https://doi.org/10.24042/penyakit>

Salehi, A. W., Khan, S., Gupta, G., Alabdullah, B. I., Almjally, A., Alsolai, H., Siddiqui, T., & Mellit, A. (2023). A Study of CNN and Transfer Learning in Medical Imaging: Advantages, Challenges, Future Scope. *Sustainability*, 15(7), 5930. <https://doi.org/10.3390/su15075930>

Santoso, I. B., Utama, S. N., & Supriyono. (2024). A New Voting of Convolutional Neural Networks for Brain Tumor Detection Based on MRI Images. *International Journal of Intelligent Engineering and Systems*, 17(1), 212–227.
<https://doi.org/10.22266/ijies2024.0229.21>

Seidi, E., Kaviari, F., & Miller, S. F. (2025). Hyperparameter Tuning of Artificial Neural Network-Based Machine Learning to Optimize Number of Hidden Layers and Neurons in Metal Forming. *Journal of Manufacturing and Materials Processing*, 9(8). <https://doi.org/10.3390/jmmp9080260>

Shah, H. A., Saeed, F., Yun, S., Park, J. H., Paul, A., & Kang, J. M. (2022). A Robust Approach for Brain Tumor Detection in Magnetic Resonance Images Using Finetuned EfficientNet. *IEEE Access*, 10, 65426–65438.
<https://doi.org/10.1109/ACCESS.2022.3184113>

Shakir, M., Shariq, S. F., Tahir, I., Khowaja, A. H., Irshad, H. A., Rae, A. I., Hamzah, R., Gupta, S., Park, K. B., & Enam, S. A. (2024). Challenges to Early Detection of Brain Tumors in Low- and Middle-Income Countries: A Systematic Review. *World Neurosurgery*, 191, 68–80.
<https://doi.org/10.1016/j.wneu.2024.07.130>

Sharma, M., Sharma, P., Mittal, R., & Gupta, K. (2022). Brain Tumour Detection Using Machine Learning. *Journal of Electronics and Informatics*, 3(4), 298–308. <https://doi.org/10.36548/jei.2021.4.005>

- Takahashi, K., Yamamoto, K., Kuchiba, A., & Koyama, T. (2022). Confidence interval for micro-averaged F1 and macro-averaged F1 scores. *Applied Intelligence*, 52(5), 4961–4972. <https://doi.org/10.1007/s10489-021-02635-5>
- Taylor-Phillips, S., & Stinton, C. (2019). Fatigue in radiology: a fertile area for future research. *The British Journal of Radiology*, 92(1099). <https://doi.org/10.1259/bjr.20190043>
- Torres-Huitzil, C., & Girau, B. (2017). Fault and Error Tolerance in Neural Networks: A Review. *IEEE Access*, 5, 17322–17341. <https://doi.org/10.1109/ACCESS.2017.2742698>
- Vyas, R., Kanumuri, T., Sheoran, G., & Dubey, P. (2017). Co-occurrence features and neural network classification approach for iris recognition. *2017 4th International Conference on Image Information Processing, ICIIP 2017, 2018-January*, 306–311. <https://doi.org/10.1109/ICIIP.2017.8313730>
- Wong, T.-T., & Yeh, P.-Y. (2020). Reliable Accuracy Estimates from k -Fold Cross Validation. *IEEE Transactions on Knowledge and Data Engineering*, 32(8), 1586–1594. <https://doi.org/10.1109/TKDE.2019.2912815>
- Wright, L. G., Onodera, T., Stein, M. M., Wang, T., Schachter, D. T., Hu, Z., & McMahon, P. L. (2022). Deep physical neural networks trained with backpropagation. *Nature*, 601(7894), 549–555. <https://doi.org/10.1038/s41586-021-04223-6>
- Xie, Y., Zaccagna, F., Rundo, L., Testa, C., Agati, R., Lodi, R., Manners, D. N., & Tonon, C. (2022). Convolutional Neural Network Techniques for Brain Tumor Classification (from 2015 to 2022): Review, Challenges, and Future Perspectives. *Diagnostics*, 12(8), 1850. <https://doi.org/10.3390/diagnostics12081850>
- Zhang, Y., Yin, X., Chen, C., Zhao, G., You, Y., Tao, S., & Xiong, W. (2025). Dynamic analysis of Hopfield neural network with Sigmoid-type activation function. *Physica Scripta*, 100(3), 035223. <https://doi.org/10.1088/1402-4896/adb34a>