

**TESIS**  
**KLASIFIKASI RADIO SIARAN FM BERDASARKAN DATA**  
**IQ MENGGUNAKAN CONVOLUTIONAL NEURAL**  
**NETWORKS**



disusun oleh

**AGUS SUKARNO**

**22.55.2329**

**Konsentrasi : Business Intelligence**

**FAKULTAS ILMU KOMPUTER**  
**UNIVERSITAS AMIKOM YOGYAKARTA**  
**YOGYAKARTA**

**2026**

**TESIS**  
**KLASIFIKASI RADIO SIARAN FM BERDASARKAN DATA**  
**IQ MENGGUNAKAN CONVOLUTIONAL NEURAL**  
**NETWORKS**

**CLASSIFICATION OF FM BROADCAST SIGNALS FROM IQ**  
**DATA USING CONVOLUTIONAL NEURAL NETWORKS**

Diajukan untuk memenuhi salah satu syarat mencapai derajat Pascasarjana  
Program Studi S2 PJJ Informatika



disusun oleh  
**AGUS SUKARNO**  
**22.55.2329**

**Konsentrasi** : Business Intelligence

**FAKULTAS ILMU KOMPUTER**  
**UNIVERSITAS AMIKOM YOGYAKARTA**  
**YOGYAKARTA**

**2026**

**HALAMAN PERSETUJUAN**

**KLASIFIKASI RADIO SIARAN FM BERDASARKAN DATA IQ  
MENGUNAKAN CONVOLUTIONAL NEURAL NETWORKS**

**CLASSIFICATION OF FM BROADCAST SIGNALS FROM IQ DATA  
USING CONVOLUTIONAL NEURAL NETWORKS**

yang disusun dan diajukan oleh

**Agus Sukarno**

**22.55.2329**

telah disetujui oleh Dosen Pembimbing Tesis  
pada tanggal 1 Desember 2025

**Dosen Pembimbing,**



**Prof. Arief Setyanto, S.Si., M.T., Ph.D.**  
**NIK. 190302036**

**HALAMAN PENGESAHAN**

**KLASIFIKASI RADIO SIARAN FM BERDASARKAN DATA IQ  
MENGUNAKAN CONVOLUTIONAL NEURAL NETWORKS**

**CLASSIFICATION OF FM BROADCAST SIGNALS FROM IQ DATA  
USING CONVOLUTIONAL NEURAL NETWORKS**

yang disusun dan diajukan oleh

**Agus Sukarno**

**22.55.2329**

Telah dipertahankan di depan Dewan Penguji  
pada tanggal 1 Desember 2025.

**Susunan Dewan Penguji**

**Nama Penguji**

**Tanda Tangan**

**Robert Marco, S.T., M.T., Ph.D.**  
**NIK. 190302228**



**Dr. Ferry Wahyu Wibowo, S.Si., M.Cs.**  
**NIK. 190302235**



**Prof. Arief Setyanto, S.Si., M.T., Ph.D.**  
**NIK. 190302036**



Tesis ini telah diterima sebagai salah satu persyaratan  
untuk memperoleh gelar Magister Komputer  
Tanggal 1 Desember 2025

**DEKAN FAKULTAS ILMU KOMPUTER**



**Prof. Dr. Kusrini, M.Kom.**  
**NIK. 190302106**

## HALAMAN PERNYATAAN KEASLIAN TESIS

Yang bertandatangan di bawah ini,

Nama mahasiswa : Agus Sukarno  
NIM : 22.55.2329  
Konsentrasi : Business Intelligence

Menyatakan bahwa Tesis dengan judul berikut:  
**Klasifikasi Radio Siaran FM Berdasarkan Data IQ Menggunakan  
Convolutional Neural Networks**

Dosen Pembimbing Utama : Prof. Arief Setyanto, S.Si., M.T., Ph.D.  
Dosen Pembimbing Pendamping : Drs. Asro Nasiri, M.Kom.

1. Karya tulis ini adalah benar-benar ASLI dan BELUM PERNAH diajukan untuk mendapatkan gelar akademik, baik di Universitas AMIKOM Yogyakarta maupun di Perguruan Tinggi lainnya
2. Karya tulis ini merupakan gagasan, rumusan dan penelitian SAYA sendiri, tanpa bantuan pihak lain kecuali arahan dari Tim Dosen Pembimbing
3. Dalam karya tulis ini tidak terdapat karya atau pendapat orang lain, kecuali secara tertulis dengan jelas dicantumkan sebagai acuan dalam naskah dengan disebutkan nama pengarang dan disebutkan dalam Daftar Pustaka pada karya tulis ini
4. Perangkat lunak yang digunakan dalam penelitian ini sepenuhnya menjadi tanggung jawab SAYA, bukan tanggung jawab Universitas AMIKOM Yogyakarta
5. Pernyataan ini SAYA buat dengan sesungguhnya, apabila di kemudian hari terdapat penyimpangan dan ketidakbenaran dalam pernyataan ini, maka SAYA bersedia menerima SANKSI AKADEMIK dengan pencabutan gelar yang sudah diperoleh, serta sanksi lainnya sesuai dengan norma yang berlaku di Perguruan Tinggi

Yogyakarta, 1 Desember 2025

Yang Menyatakan,



Agus Sukarno

## HALAMAN PERSEMBAHAN

Segala Puji bagi Allah SWT yang telah melimpahkan Rahmat dan hidayah-Nya dan shalawat serta salam semoga terlimpah bagi Nabi Muhammad SAW, kepada keluarga, sahabat dan para pengikutnya sampai akhir zaman.

Alhamdulillah atas dukungan serta do'a dari orang-orang tercinta, akhirnya tesis ini dapat terselesaikan dengan baik. Oleh karena itu dengan rasa bangga dan suka cita, tesis ini saya persembahkan untuk:

1. Ibu (Martinah), Istri (Nur Aisyah Ariyanti), Anak (Najla Ahfazhani Hanifa, Hafizh Aqil Muniba, Khansa Ridha Naila, Rasyid Sabil Haq, Muhammad Rizqi Ramadhan) yang telah memberikan dukungan moral maupun materi serta do'a, motivasi dan kesabaran kepada saya.
2. Bapak Prof. Arief Setyanto, S.Si., M.T., Ph.D. dan Bapak Drs. Asro Nasiri, M.Kom. selaku dosen pembimbing saya yang telah memberikan arahan, motivasi serta bimbingan positif dalam menyelesaikan tesis ini sehingga tesis ini dapat terselesaikan dengan baik.
3. Sahabat seperjuangan saya mas Gustian, mbak Maimi Herawati yang telah memberikan semangat, dukungan dan do'a untuk tidak menyerah dalam berjuang.
4. Keluarga besar MTI Amikom, khususnya kelas PJJ MTI Konsentrasi Business Intelligence yang telah menjadi teman-teman saya semasa kuliah pasca sarjana. Serta semua pihak yang telah membantu dan mendukung saya yang tidak saya sebutkan satu per satu.

## KATA PENGANTAR

Segala puji bagi Allah SWT, yang telah memberikan segala nikmat yang tak terhingga kepada hamba-Nya. Alhamdulillah dengan izin Allah SWT, penulis dapat menyelesaikan tesis ini yang berjudul "KLASIFIKASI RADIO SIARAN FM BERDASARKAN DATA IQ MENGGUNAKAN CONVOLUTIONAL NEURAL NETWORKS". Penyusunan tugas akhir ini merupakan salah satu persyaratan kelulusan untuk memperoleh gelar Magister Informatika di Universitas AMIKOM Yogyakarta.

Dengan tersusunnya tesis ini, penulis mengucapkan terima kasih banyak kepada:

1. Orang tua, saudara, anak dan istri tercinta.
2. Bapak Prof Dr. M. Suyanto, M.M. selaku Rektor Universitas AMIKOM Yogyakarta.
3. Ibu Prof. Dr. Kusriani, M.Kom. selaku Ketua Pascasarjana dan Ketua Program S2 Teknik Informatika Universitas AMIKOM Yogyakarta.
4. Bapak Prof. Arief Setyanto, S.Si., M.T., Ph.D. selaku Dosen Pembimbing Pertama.
5. Bapak Drs. Asro Nasiri, M.Kom. selaku Dosen Pembimbing Kedua.
6. Seluruh Dosen Pascasarjana PJJ MTI Universitas AMIKOM Yogyakarta.
7. Teman-teman seperjuangan PJJ MTI Universitas AMIKOM Yogyakarta.
8. Zuhaidir Hamid, temen di Kementerian Komunikasi dan Digital yang telah memberikan dataset radio siaran sehingga penelitian ini dapat terlaksana dengan baik.

Penulis menyadari jika tesis ini masih jauh dari kata sempurna, semoga tesis ini bermanfaat tidak hanya bagi penulis melainkan juga bermanfaat bagi pembaca dan dapat memberikan inspirasi untuk pengembangan penelitian selanjutnya.

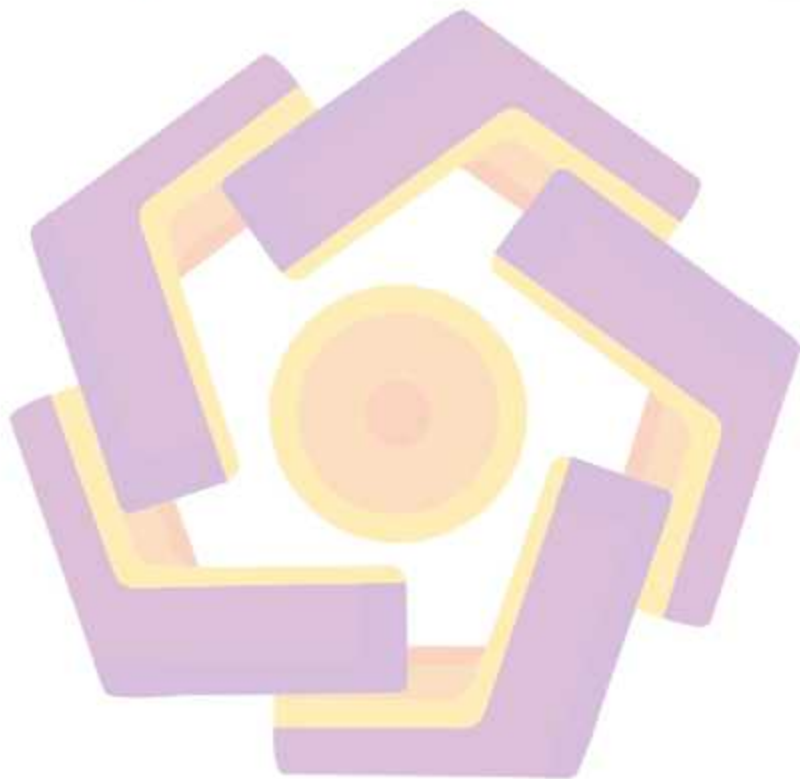
Yogyakarta, 1 Desember 2025

Penulis

## DAFTAR ISI

HALAMAN JUDUL .....	i
HALAMAN PERSETUJUAN .....	ii
HALAMAN PENGESAHAN .....	iii
HALAMAN PERNYATAAN KEASLIAN TESIS .....	iv
HALAMAN PERSEMBAHAN .....	v
KATA PENGANTAR .....	vi
DAFTAR ISI .....	vii
DAFTAR TABEL .....	ix
DAFTAR GAMBAR .....	x
DAFTAR ISTILAH .....	xi
INTISARI .....	xii
<i>ABSTRACT</i> .....	xiii
<b>BAB 1 PENDAHULUAN</b> .....	1
1.1 Latar Belakang .....	1
1.2 Rumusan Masalah .....	4
1.3 Batasan Masalah .....	5
1.4 Tujuan Penelitian .....	6
1.5 Manfaat Penelitian .....	6
<b>BAB 2 TINJAUAN PUSTAKA</b> .....	8
2.1 Tinjauan Pustaka .....	8
2.2 Keaslian Penelitian .....	12
2.3 Landasan Teori .....	18
<b>BAB 3 METODE PENELITIAN</b> .....	39
3.1 Jenis, Sifat, dan Pendekatan Penelitian .....	39
3.2 Metode Pengumpulan Data .....	40
3.3 Metode Analisis Data .....	44
3.4 Alur Penelitian .....	46
<b>BAB 4 HASIL PENELITIAN DAN PEMBAHASAN</b> .....	51
4.1 Persiapan Data .....	51
4.2 Pengembangan Model .....	59
4.3 Pelatihan Model .....	78
4.4 Evaluasi Model .....	90
4.5 Pembahasan Hasil Penelitian .....	104
4.6 Pemilihan Model Terbaik .....	114

4.6.1.	Kelebihan dan Kekurangan CNN-BiLSTM Hasil Penelitian .....	115
4.6.2.	Kelebihan dan Kekurangan CNN-Transformer Hasil Penelitian ..	115
4.6.3.	Kelebihan dan Kekurangan CNN 5-Layers Hasil Penelitian .....	116
BAB 5	PENUTUP .....	117
5.1	Kesimpulan .....	117
5.2	Saran .....	118
DAFTAR PUSTAKA	.....	120



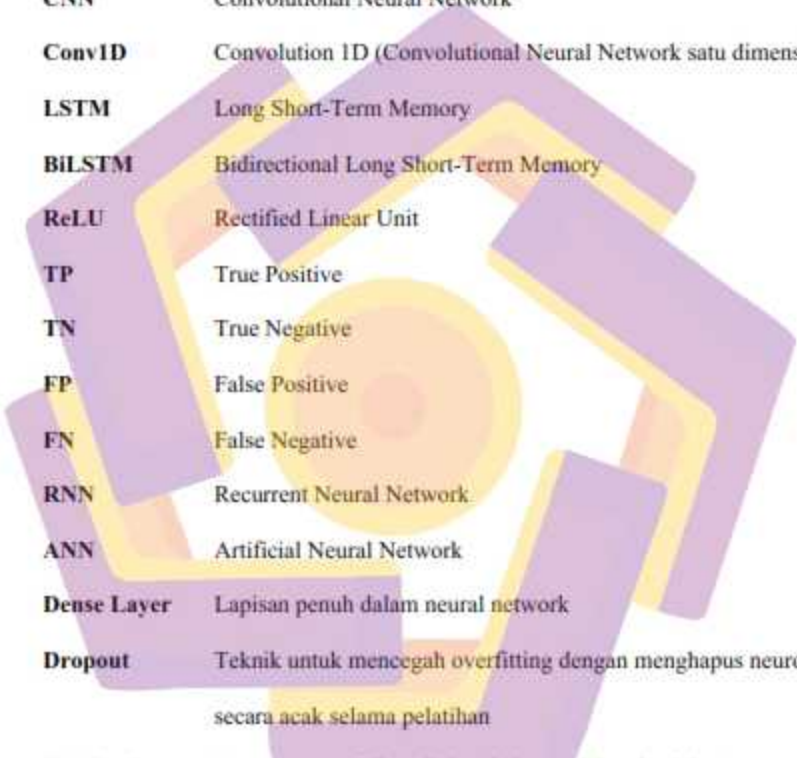
## DAFTAR TABEL

Tabel 2.1. Matriks literatur review dan posisi penelitian .....	12
Tabel 2.2. Confusion Matrix.....	37
Tabel 3.1. Pemancar Radio Penelitian.....	42
Tabel 4.1. Proses Penyeimbangan Data.....	52
Tabel 4.2. Jumlah Parameter CNN-BiLSTM .....	65
Tabel 4.3. Jumlah Parameter CNN-Transformer.....	70
Tabel 4.4. Jumlah Parameter CNN 5-Layers.....	76
Tabel 4.5. Ringkasan Hasil Pelatihan CNN-BiLSTM.....	80
Tabel 4.6. Ringkasan Hasil Pelatihan CNN-Transformer.....	84
Tabel 4.7. Ringkasan Hasil Pelatihan CNN 5-Layers.....	87
Tabel 4.8. Classification Report CNN-BiLSTM.....	92
Tabel 4.9. Classification Report CNN-Transformer.....	96
Tabel 4.10. Classification Report CNN 5-Layers.....	100
Tabel 4.11. Perbandingan Hasil Penelitian Model.....	105
Tabel 4.12. Perbandingan Waktu Klasifikasi per Batch.....	110

## DAFTAR GAMBAR

Gambar 2.1 Sinyal IQ dimodulasi dalam gelombang sinus .....	19
Gambar 2.2. Blok Diagram RTL-SDR .....	20
Gambar 2.3. CNN 1D .....	23
Gambar 2.4. CNN 2D .....	24
Gambar 2.5. Arsitektur LSTM .....	27
Gambar 2.6. Arsitektur dasar Conv1D-LSTM .....	31
Gambar 2.7. Arsitektur Transformer .....	32
Gambar 3.1. Flowchart Pengumpulan Data IQ .....	41
Gambar 3.2. Alur Penelitian .....	46
Gambar 4.1. Visualisasi karakteristik unik data IQ dari sampling empat (4) pemancar FM .....	58
Gambar 4.2. Arsitektur Lapisan CNN untuk Klasifikasi Data IQ .....	59
Gambar 4.3. Plot-model CNN-BiLSTM .....	62
Gambar 4.4. Plot-model CNN-Transformer .....	68
Gambar 4.5. Plot-model CNN 5-Layers .....	74
Gambar 4.6. Grafik Akurasi dan Loss CNN-BiLSTM .....	82
Gambar 4.7. Grafik Akurasi dan Loss CNN-Transformer .....	86
Gambar 4.8. Grafik Akurasi dan Loss CNN 5-Layers .....	89
Gambar 4.9. Confusion Matrix CNN-BiLSTM .....	94
Gambar 4.10. Confusion Matrix CNN-Transformer .....	98
Gambar 4.11. Confusion Matrix CNN 5-Layers .....	103

## DAFTAR ISTILAH



<b>IQ</b>	In-Phase & Quadrature
<b>CNN</b>	Convolutional Neural Network
<b>Conv1D</b>	Convolution 1D (Convolutional Neural Network satu dimensi)
<b>LSTM</b>	Long Short-Term Memory
<b>BiLSTM</b>	Bidirectional Long Short-Term Memory
<b>ReLU</b>	Rectified Linear Unit
<b>TP</b>	True Positive
<b>TN</b>	True Negative
<b>FP</b>	False Positive
<b>FN</b>	False Negative
<b>RNN</b>	Recurrent Neural Network
<b>ANN</b>	Artificial Neural Network
<b>Dense Layer</b>	Lapisan penuh dalam neural network
<b>Dropout</b>	Teknik untuk mencegah overfitting dengan menghapus neuron secara acak selama pelatihan
<b>MaxPooling</b>	Proses pengambilan nilai maksimum dalam jendela tertentu untuk menyederhanakan data
<b>Sigmoid</b>	Fungsi aktivasi untuk klasifikasi biner

## INTISARI

Pengawasan spektrum siaran FM secara real-time memerlukan teknik canggih; pendekatan berbasis klasifikasi sinyal telah terbukti meningkatkan ketepatan deteksi dibandingkan metode manual. Penelitian ini mengembangkan dan membandingkan tiga arsitektur deep learning - CNN 5-Layers, CNN-BiLSTM, dan CNN-Transformer - untuk mengklasifikasikan pengguna siaran radio FM berdasarkan data IQ. Data sinyal dikumpulkan dari 16 pemancar FM menggunakan SDR dan diolah menjadi 80.000 sampel seimbang. Model-model ini dievaluasi berdasarkan akurasi klasifikasi dan waktu inferensi. Hasil eksperimen menunjukkan bahwa CNN-BiLSTM memberikan akurasi tertinggi sebesar 98,96% namun dengan waktu inferensi relatif lama sekitar 62 detik. Sementara itu, CNN 5-Layers memiliki waktu klasifikasi tercepat sekitar 10 detik dengan akurasi tinggi sebesar 98,18%, dan CNN-Transformer paling lambat sekitar 120 detik dengan akurasi sebesar 97,72%. Mengingat waktu klasifikasi per batch harus lebih pendek daripada laju pengambilan data sekitar 2 milidetik per sampel, hanya CNN 5-Layers yang memenuhi persyaratan pemantauan spektrum secara real-time.

**Kata kunci:** Pengawasan Spektrum Frekuensi Radio; CNN; BiLSTM; Transformer; Data IQ; Pemantauan Real-time.

## ABSTRACT

*Real-time monitoring of FM broadcast spectrum requires advanced techniques; signal classification-based approaches have been proven to enhance detection accuracy compared to manual methods. This study develops and compares three deep learning architectures—CNN 5-Layers, CNN-BiLSTM, and CNN-Transformer—for classifying FM broadcast users based on IQ data. Signal data were collected from 16 FM transmitters using an SDR and processed into 80,000 balanced samples. The models were evaluated based on classification accuracy and inference time. Experimental results show that CNN-BiLSTM achieved the highest accuracy of 98.96%, but with a relatively long inference time of approximately 62 seconds. Meanwhile, CNN 5-Layers recorded the fastest classification time of around 10 seconds with a high accuracy of 98.18%, while CNN-Transformer was the slowest at about 120 seconds with an accuracy of 97.72%. Considering that the classification time per batch must be shorter than the data acquisition rate of around 2 ms per sample, only CNN 5-Layers meets the requirements for real-time spectrum monitoring.*

**Keyword:** Radio Frequency Spectrum Monitoring; CNN; BiLSTM; Transformer; IQ Data; Real-time Monitoring.

# BAB 1

## PENDAHULUAN

### 1.1 Latar Belakang

Komunikasi merupakan kebutuhan dasar manusia dalam kehidupan sosial. Seiring dengan perkembangan zaman, teknologi frekuensi radio telah memberikan perubahan signifikan dalam cara manusia berkomunikasi, khususnya dalam penyampaian informasi secara real-time meskipun terdapat hambatan geografis. Teknologi ini memungkinkan informasi ditransmisikan dengan cepat dan luas, sehingga menjadi bagian vital dalam mendukung berbagai sektor kehidupan, seperti pertahanan, transportasi, penyiaran, dan layanan darurat.

Namun demikian, seiring dengan meningkatnya kebutuhan terhadap layanan komunikasi, permintaan terhadap spektrum frekuensi radio pun turut meningkat. Hal ini menjadi tantangan tersendiri karena spektrum frekuensi radio merupakan sumber daya alam yang bersifat terbatas dan tidak dapat diperbaharui. Untuk mencegah terjadinya gangguan dan tumpang tindih penggunaan, pengelolaan spektrum yang tertib dan terkoordinasi menjadi sangat penting, baik secara nasional maupun internasional.

Secara global, pengelolaan spektrum frekuensi radio diatur oleh International Telecommunication Union (ITU), sebuah badan di bawah naungan Perserikatan Bangsa-Bangsa (PBB) yang bertanggung jawab dalam bidang telekomunikasi. ITU menetapkan regulasi dan pedoman yang harus diadopsi oleh negara-negara anggotanya dalam menyusun kebijakan nasional pengelolaan

spektrum. Dengan demikian, pemanfaatan spektrum dapat diselaraskan untuk mendukung komunikasi yang efektif antarnegara.

Di Indonesia, pengelolaan spektrum frekuensi radio dilaksanakan oleh Kementerian Komunikasi dan Digital, melalui Direktorat Jenderal Infrastruktur Digital. Mengingat luasnya wilayah geografis serta keragaman topografi Indonesia, pengawasan spektrum tidak dapat dilakukan secara terpusat. Oleh karena itu, dibentuklah 35 Unit Pelaksana Teknis (UPT) Monitoring Spektrum Frekuensi Radio yang tersebar di seluruh wilayah Indonesia. UPT ini bertugas melakukan pengawasan terhadap pemanfaatan spektrum untuk menjamin ketertiban, efisiensi, serta bebas dari interferensi.

Ruang lingkup pengawasan yang dilakukan meliputi observasi dan identifikasi penggunaan frekuensi, pengukuran parameter teknis stasiun radio, serta inspeksi lapangan. Dalam menjalankan fungsinya, UPT Monitoring juga memiliki kewenangan dalam melakukan pembinaan, termasuk penerapan sanksi administratif maupun pidana bagi pelanggaran yang ditemukan.

Data kinerja pengawasan tahun 2023 menunjukkan bahwa dari total 438.161 frekuensi berizin, hanya sekitar 244.716 yang berhasil dimonitor, dengan 5.672 di antaranya ditertibkan karena pelanggaran dan 443 frekuensi mengalami gangguan [1]. Artinya, tingkat pengawasan hanya mencapai 55,85%. Fakta ini menunjukkan adanya kebutuhan untuk meningkatkan efektivitas pengawasan melalui pendekatan teknologi yang lebih adaptif dan efisien [2].

Pengawasan secara manual memiliki keterbatasan, baik dari segi efisiensi waktu, akurasi, maupun kontinuitas operasional. Untuk mengatasi keterbatasan ini,

penerapan kecerdasan buatan (Artificial Intelligence/AI) menjadi solusi inovatif yang menjanjikan. Berbagai penelitian menunjukkan bahwa deep learning, khususnya Convolutional Neural Network (CNN), efektif untuk mengenali pola kompleks dalam sinyal radio tanpa memerlukan fitur [3]. Dengan memanfaatkan CNN, diharapkan proses identifikasi pemancar siaran FM dapat dilakukan secara otomatis dan akurat, bahkan dalam kondisi spektrum yang padat interferensi [4]. Dalam konteks pengawasan spektrum, CNN dapat digunakan untuk mengidentifikasi karakteristik unik dari setiap pemancar berdasarkan pola sinyal In-phase (I) dan Quadrature (Q) atau data IQ, yang merupakan representasi kompleks sinyal radio dalam domain waktu [5].

Namun, penerapan CNN untuk pengawasan spektrum di Indonesia masih terbatas pada analisis spektral, belum sampai pada klasifikasi otomatis sumber sinyal berdasarkan data IQ. Padahal, sinyal IQ menyimpan ciri khas unik yang memungkinkan identifikasi otomatis setiap pemancar radio. Inilah yang menjadi isu ilmiah utama dalam penelitian ini, yakni kebutuhan untuk mengembangkan metode klasifikasi sinyal berbasis CNN yang dapat mengenali pemancar radio FM secara otomatis dari data IQ yang dihasilkan perangkat SDR (Software Defined Radio).

Selain itu, meskipun CNN terbukti kuat dalam klasifikasi sinyal digital, belum banyak penelitian yang secara spesifik mengevaluasi performa CNN pada sinyal analog FM, khususnya dalam konteks pengawasan spektrum radio siaran. Oleh karena itu, perlu dilakukan pengujian terhadap akurasi, presisi, dan efisiensi

waktu inferensi model CNN untuk memastikan kelayakannya digunakan dalam sistem monitoring yang menuntut kecepatan dan keandalan tinggi.

Lebih jauh, penelitian ini tidak hanya berfokus pada pengembangan arsitektur CNN murni, tetapi juga membandingkannya dengan model hybrid seperti CNN-BiLSTM dan CNN-Transformer. CNN-BiLSTM menggabungkan keunggulan CNN dalam mengekstrak fitur spasial dan LSTM dalam memahami urutan temporal sinyal, sedangkan CNN-Transformer mengombinasikan CNN dengan mekanisme self-attention untuk menangkap hubungan global antar fitur. Melalui perbandingan ini, diharapkan dapat diketahui model yang paling optimal dalam hal akurasi dan efisiensi waktu klasifikasi, sehingga sesuai untuk diterapkan dalam pengawasan spektrum secara real-time.

## 1.2 Rumusan Masalah

Berdasarkan latar belakang yang telah diuraikan, rumusan masalah dalam penelitian ini dapat dirumuskan sebagai berikut:

1. Bagaimana model berbasis CNN dan hybrid (CNN-BiLSTM dan CNN-Transformer) dapat mengklasifikasikan pemancar siaran radio FM berdasarkan data sinyal IQ sesuai kebutuhan proses monitoring secara real-time?
2. Bagaimana perbandingan akurasi model dalam mengklasifikasikan pemancar siaran FM?
3. Apakah model dapat diterapkan dalam sistem pengawasan spektrum frekuensi radio secara real-time, ditinjau dari keseimbangan antara akurasi

klasifikasi dan waktu inferensi dibawah waktu yang dibutuhkan RTL-SDR untuk mengumpulkan satu batch data IQ?

### 1.3 Batasan Masalah

Agar penelitian ini terfokus dan dapat diselesaikan secara terarah, maka ruang lingkup penelitian dibatasi pada beberapa hal berikut:

1. Penelitian ini difokuskan pada pengawasan dan identifikasi pemancar radio siaran FM yang beroperasi pada rentang frekuensi 87,5 MHz hingga 108 MHz, sesuai dengan pita frekuensi layanan siaran FM di Indonesia.
2. Data yang digunakan dalam penelitian ini berupa data IQ yang diperoleh dari hasil perekaman spektrum frekuensi radio menggunakan perangkat Software Defined Radio (SDR). Data IQ ini direpresentasikan dalam bentuk data kompleks dalam domain waktu.
3. Metode klasifikasi yang digunakan dalam penelitian ini dibatasi pada pendekatan CNN dan hybrid (CNN-BiLSTM, CNN-Transformer, dan CNN 5-Layers) dalam bentuk arsitektur satu dimensi (1D CNN). Pendekatan lain seperti Recurrent Neural Network (RNN), Support Vector Machine (SVM), atau Reinforcement Learning tidak dibahas dalam penelitian ini.
4. Identifikasi yang dilakukan bertujuan untuk mengenali identitas pemancar atau karakteristik unik dari sumber sinyal FM berdasarkan pola-pola yang terdapat pada data IQ. Penelitian tidak membahas isi atau konten audio dari siaran.
5. Evaluasi performa model CNN dibatasi pada penggunaan metrik evaluasi akurasi, presisi, recall, F1-score, dan confusion matrix. Evaluasi terhadap

efisiensi komputasi atau waktu proses dilakukan secara terbatas sebagai bahan diskusi tambahan, namun bukan fokus utama penelitian.

6. Penelitian dilakukan dalam lingkungan simulasi atau pengujian terbatas dengan dataset hasil perekaman spektrum di wilayah tertentu, sehingga generalisasi model ke seluruh wilayah Indonesia akan menjadi ruang lingkup penelitian lanjutan.

#### **1.4 Tujuan Penelitian**

Penelitian ini bertujuan untuk:

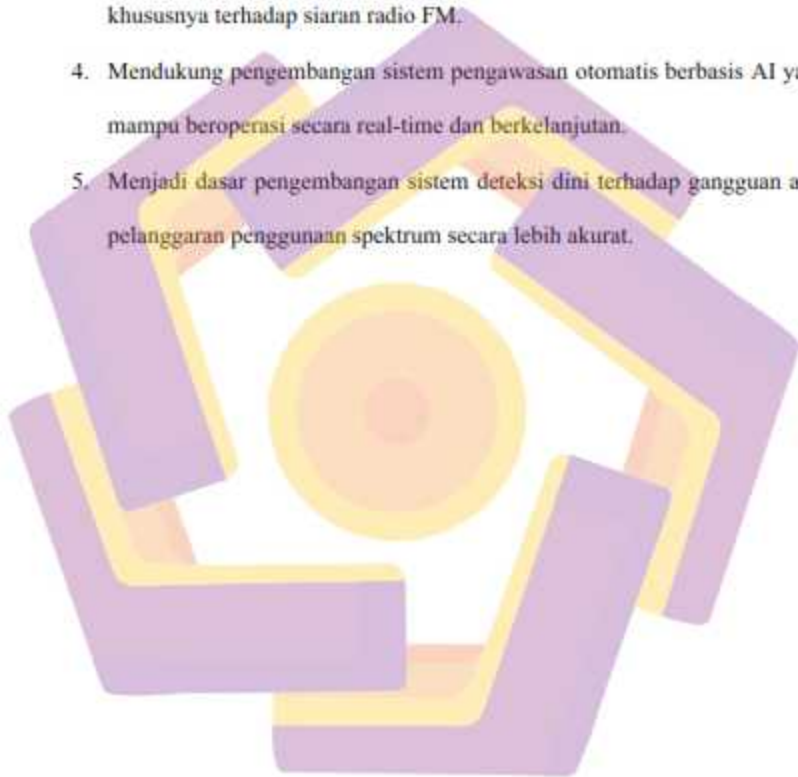
1. Menerapkan metode deep learning berbasis Convolutional Neural Network (CNN) dalam proses identifikasi pemancar siaran radio FM menggunakan data IQ, yang direkam melalui perangkat Software Defined Radio (SDR).
2. Menganalisis dan membandingkan performa tiga arsitektur model deep learning, yaitu CNN 5-Layers, CNN-BiLSTM, dan CNN-Transformer, dalam hal akurasi klasifikasi, stabilitas generalisasi, dan efisiensi komputasi pada data IQ.
3. Menentukan model yang paling optimal untuk implementasi real-time monitoring dalam pengawasan spektrum frekuensi radio, berdasarkan keseimbangan antara akurasi tinggi dan waktu inferensi yang efisien.

#### **1.5 Manfaat Penelitian**

Penelitian ini diharapkan dapat memberikan manfaat sebagai berikut:

1. Memberikan kontribusi dalam pengembangan kajian ilmiah terkait pemanfaatan deep learning, khususnya CNN, dalam analisis sinyal radio.

2. Menambah referensi ilmiah tentang pemanfaatan data IQ sebagai fitur penting dalam proses klasifikasi sinyal radio.
3. Memberikan solusi alternatif bagi instansi pengelola spektrum frekuensi radio dalam meningkatkan efektivitas dan efisiensi pengawasan spektrum, khususnya terhadap siaran radio FM.
4. Mendukung pengembangan sistem pengawasan otomatis berbasis AI yang mampu beroperasi secara real-time dan berkelanjutan.
5. Menjadi dasar pengembangan sistem deteksi dini terhadap gangguan atau pelanggaran penggunaan spektrum secara lebih akurat.



## BAB 2

### TINJAUAN PUSTAKA

#### 2.1 Tinjauan Pustaka

Kebutuhan terhadap pemanfaatan spektrum frekuensi radio oleh berbagai layanan komunikasi meningkat secara signifikan seiring dengan kemajuan teknologi. Hal ini menuntut adanya sistem pengawasan spektrum yang lebih akurat, efisien, dan real-time, terutama untuk memastikan tidak terjadinya interferensi antar pengguna. Kompleksitas sinyal yang ditransmisikan di spektrum radio, seperti modulasi FM, variasi frekuensi, noise lingkungan, serta interferensi antar kanal, menimbulkan tantangan tersendiri dalam proses identifikasi dan pengawasan pemancar secara konvensional.

Dalam konteks ini, kemunculan teknologi berbasis kecerdasan buatan (AI), khususnya deep learning, telah membuka peluang baru dalam sistem pengawasan spektrum yang otomatis dan adaptif. AI tidak hanya mampu memproses volume data besar secara efisien, tetapi juga dapat mengenali pola-pola kompleks yang sulit dianalisis secara manual.

Metode CNN dikenal efektif dalam mengekstraksi fitur spasial dari sinyal dan telah digunakan secara luas untuk klasifikasi gambar, suara, dan sinyal spektrum. CNN juga terbukti mampu mendeteksi serta menganalisis sinyal kompleks dalam domain frekuensi radio. Dalam konteks pengawasan spektrum, CNN digunakan untuk menganalisis data IQ – representasi kompleks dari sinyal

radio dalam domain waktu – yang mencerminkan karakteristik unik dari masing-masing pemancar [6].

Beberapa penelitian menunjukkan bahwa CNN mampu melakukan identifikasi pemancar secara end-to-end, tanpa perlu proses demodulasi, dengan memanfaatkan fingerprint sinyal seperti IQ imbalance. [6] berhasil mengidentifikasi pemancar berdasarkan parameter IQ imbalance, yang mencerminkan ketidaksempurnaan perangkat keras pemancar. [7] menggunakan 3D CNN untuk mempelajari fitur intrinsik dari sinyal IQ mentah, dan menunjukkan hasil klasifikasi dengan akurasi mendekati 99%, bahkan terhadap perangkat yang hampir identik secara fisik.

Lebih lanjut, [8] menggunakan CNN 1D untuk mengklasifikasikan sinyal dari perangkat drone. Pendekatan ini menunjukkan bahwa CNN dapat diimplementasikan secara ringan dan efisien dalam lingkungan spektrum yang padat dan dinamis tanpa fitur buatan (hand-crafted), menjadikannya ideal untuk aplikasi pengawasan spektrum secara real-time.

Arsitektur CNN 5-Layers merupakan salah satu struktur jaringan yang banyak digunakan karena kesederhanaannya, namun tetap mampu mencapai performa yang kompetitif. [9] berhasil menerapkan CNN 5-layer dalam deteksi penyakit Alzheimer menggunakan citra MRI, dengan hasil akurasi sebesar 94,39%. Studi lain oleh [10] menunjukkan bahwa CNN 5-layer mampu mengungguli model populer seperti VGG16 dalam klasifikasi citra kompleks, dengan akurasi mencapai 78,2% dibandingkan 58,4%.

Selain CNN murni, arsitektur hybrid seperti CNN-BiLSTM menggabungkan kekuatan CNN dalam mengekstraksi fitur spasial dengan kemampuan BiLSTM dalam menangkap hubungan temporal jangka pendek maupun panjang. [11] menerapkan CNN-BiLSTM dalam peramalan lalu lintas di Madrid dan menunjukkan peningkatan metrik MAE dan RMSE secara signifikan. [12] juga membuktikan efektivitas CNN-BiLSTM untuk prediksi penyebaran kebakaran hutan secara near-real-time dengan nilai F1-score sebesar 0,73.

Penggabungan CNN dengan Transformer juga memberikan pendekatan baru yang kuat dalam klasifikasi sinyal. CNN unggul dalam menangkap fitur lokal, sementara Transformer efektif dalam memahami hubungan spasial global melalui mekanisme self-attention. [13] mengembangkan Efficient Hybrid Network (EHNet) untuk super-resolusi citra penginderaan jauh, sedangkan [14] memperkenalkan SCDUNet++ untuk pemetaan tanah longsor, keduanya menggabungkan CNN dan Transformer dalam satu arsitektur untuk meningkatkan akurasi klasifikasi pada data kompleks dan terbatas.

Monitoring spektrum frekuensi secara real-time memerlukan sistem yang tidak hanya akurat, tetapi juga efisien dari sisi komputasi. Inferensi cepat menjadi krusial untuk mendukung proses pengambilan keputusan otomatis di lapangan. [15] mengembangkan sistem klasifikasi teknologi radio menggunakan LSTM Autoencoder untuk mengenali teknologi komunikasi dari sinyal RF secara cepat dan akurat. [16] menggunakan CNN dalam sistem monitoring tanda vital real-time menggunakan radar UWB, yang dapat memantau detak jantung dan laju pernapasan secara non-kontak. [17] memperkenalkan sistem klasifikasi berbasis radar dengan

akselerasi GPU hingga 200 MSps, yang mempertahankan akurasi tinggi dalam kondisi interferensi tinggi. Penelitian-penelitian tersebut menunjukkan bahwa kombinasi akurasi tinggi dan efisiensi inferensi menjadi syarat utama dalam desain sistem monitoring spektrum masa kini.

Berdasarkan literatur yang telah dibahas, pengembangan sistem klasifikasi untuk identifikasi pemancar radio siaran FM berbasis data IQ sangat relevan dan memungkinkan secara teknis. Data IQ memiliki potensi sebagai fingerprint digital yang khas untuk setiap pemancar, dan dapat dimanfaatkan oleh CNN secara efektif. CNN 5-Layers menjadi arsitektur ringan yang ideal untuk pengawasan real-time, sedangkan arsitektur hybrid seperti CNN-BiLSTM dan CNN-Transformer menawarkan peluang untuk meningkatkan akurasi dengan menangkap aspek spasial-temporal maupun hubungan global dalam data IQ.

Penelitian ini mengembangkan dan membandingkan tiga arsitektur CNN 5-Layers, CNN-BiLSTM, dan CNN-Transformer dalam mengklasifikasikan pemancar siaran FM berdasarkan data IQ. Evaluasi dilakukan tidak hanya pada akurasi, tetapi juga pada efisiensi waktu inferensi, sebagai syarat utama sistem monitoring spektrum real-time. Dataset dikumpulkan dari 16 pemancar radio FM di wilayah Indonesia untuk merepresentasikan karakteristik sinyal di lapangan. Dengan demikian, penelitian ini memperluas pendekatan identifikasi berbasis deep learning untuk pengawasan spektrum radio yang lebih modern, cepat, dan andal.

## 2.2 Keaslian Penelitian

Tabel 2.1. Matriks literatur review dan posisi penelitian  
Klasifikasi Radio Siaran FM Berdasarkan Data IQ Menggunakan Convolutional Neural Networks

No	Judul Penelitian	Nama Peneliti, Tahun, Index	Metode Penelitian	Hasil	Keunggulan dan Kelemahan	Perbandingan
1	Specific Emitter Identification Using Convolutional Neural Network-Based IQ Imbalance Estimators	Wong, H. Y., Headley, W. C., & Michaels, A. J. – Sensors, 2019, Q1	Mengembangkan metode Specific Emitter Identification (SEI) berbasis CNN untuk mengestimasi parameter IQ imbalance dari sinyal modulasi digital tanpa preprocessing kompleks.	CNN efektif untuk mengestimasi gain imbalance, namun kurang optimal pada phase imbalance. Akurasi meningkat pada SNR tinggi dan lebih unggul dibandingkan metode feature-based (SVM) terutama ketika menggunakan multiple capture.	Kelemahannya adalah performa estimasi phase imbalance masih rendah, serta metode diuji pada data simulasi sehingga perlu validasi pada data nyata.	Penelitian ini berfokus pada fingerprinting emitter melalui impairment hardware (IQ imbalance), sedangkan penelitian kita berfokus pada klasifikasi pemancar FM berbasis data IQ menggunakan CNN, CNN-BiLSTM, dan CNN-Transformer. Selain itu, penelitian kita menekankan pada evaluasi waktu komputasi untuk real-time monitoring, bukan hanya akurasi estimasi parameter.

2	CNN-BiLSTM: A Novel Deep Learning Model for Near-Real-Time Daily Wildfire Spread Prediction	Mohammad Marjani, Masoud Mahdianpari, & Fariba Mohammadinsh. Remote Sensing, MDPL, 2024, Q1	Mengembangkan model deep learning hibrida CNN-BiLSTM untuk memprediksi penyebaran kebakaran hutan secara near-real-time dengan mengintegrasikan data spasial dan temporal.	Model CNN-BiLSTM terbukti lebih unggul dibanding LSTM dan CNN-LSTM dalam memprediksi penyebaran kebakaran, dengan nilai presisi, recall, F1, dan IoU lebih baik. Model mampu menangkap pola spasial dan temporal sehingga meningkatkan akurasi prediksi.	Kelemahan utama terletak pada ketergantungan terhadap data awal (burn map) dari satelit VIIRS serta potensi kesalahan akibat pendekatan berbasis patch yang menghasilkan FP/FN. Penelitian lanjutan perlu mengintegrasikan teknik deep learning baru seperti attention mechanism dan augmentasi data historis.	Inovasi yang ditawarkan bukan hanya pada penggunaan CNN-BiLSTM, tetapi juga membandingkan dengan arsitektur lain (CNN-Transformer dan CNN 5-Layers) untuk menemukan model terbaik. Dengan demikian, kompleksitas permasalahan lebih menekankan pada stabilitas generalisasi model dan kecepatan inferensi, bukan semata akurasi spasial-temporal.
3	An Efficient Hybrid CNN-Transformer Approach for Remote Sensing Super-Resolution	W. Zhang, Y. Li, Z. Tang, J. Liu, B. Zhou, Q. Liu. Remote	Mengembangkan model hibrida CNN-Transformer (EHNet) untuk meningkatkan kualitas super-	EHNet yang menggabungkan Lightweight Feature Extraction Block (LFEB) dengan Swin Transformer	Model cenderung kurang optimal pada dataset besar dengan tekstur rendah atau gradien kecil, serta performa kurang stabil pada faktor SR = 3.	Penelitian ini menggabungkan CNN dan Swin Transformer untuk menyeimbangkan efisiensi dan akurasi, sedangkan penelitian saya lebih fokus

		Sensing, 2024, Q1	resolution (SR) pada citra penginderaan jauh dengan tetap menjaga efisiensi komputasi.	dan modul upsampling berbasis sekuens terbukti meningkatkan hasil rekonstruksi super-resolution pada dataset AID dan UCMerced, dengan kinerja lebih baik dibanding metode lain, meskipun hanya memiliki 2,64 juta parameter .	Penelitian selanjutnya disarankan fokus pada perbaikan generalisasi untuk dataset kompleks .	pada klasifikasi sinyal IQ radio siaran menggunakan CNN-Transformer. Kompleksitas permasalahan juga berbeda: penelitian ini menekankan rekonstruksi citra resolusi tinggi, sementara penelitian saya menitikberatkan pada identifikasi pola spektrum untuk klasifikasi sinyal.
4	Rice Leaf Disease Classification Using CNN	Pallapothala Tejaswini, Priyanshi Singh, Monica Ramchandani, Yogesh Kumar Rathore, Rekh Ram Janghel, IOP	Mengembangkan metode deteksi penyakit daun padi (Brown spot, Hispa, Leaf Blast, Healthy)	Hasil penelitian menunjukkan bahwa model CNN 5-layer buatan sendiri memiliki akurasi terbaik (78,2%) dibandingkan dengan model standar seperti	Keterbatasan penelitian adalah dataset kecil (1600 gambar) dan hanya mencakup empat jenis penyakit daun padi. Penelitian selanjutnya dapat memperluas dataset, menambah variasi penyakit, dan	Dari sisi metode, sama-sama menggunakan CNN, tetapi penelitian ini membandingkan banyak arsitektur (VGG, ResNet, Xception, CNN 5-layer), sedangkan penelitian saya mengarah pada kombinasi CNN dengan

		Conference Series: Earth and Environmental Science; 2022, H-Index 58	VGG16, VGG19, ResNet, Xception, 5-layer CNN) dan membandingkan performa masing-masing model.	VGG16 (58,4%) atau ResNet50 (72,2%). CNN dengan arsitektur sederhana lebih efektif pada dataset terbatas.	menggunakan algoritma yang lebih kompleks.	BiLSTM/Transformer. Kompleksitas masalah juga berbeda: penelitian ini fokus pada citra statis tanaman, sedangkan penelitian saya pada sinyal dinamis dalam domain waktu yang lebih menantang untuk real-time.
5	Exploiting Spatio-Temporal Properties of I/Q Signal Data Using 3D Convolution for RF Transmitter Identification	Muthukumar Ramasubramanian, Chaity Banerjee, Debashri Roy, Eduardo Pasiliao, dan Tathagata Mukherjee, IEEE Journal Of Radio Frequency	Mengembangkan sistem identifikasi pemancar yang menggunakan konvolusi 3D untuk mempelajari fitur intrinsik pemancar untuk membedakan pemancar yang berbeda secara efisien	Penelitian ini berhasil menunjukkan bahwa sistem berbasis konvolusi 3D dapat secara efektif mengidentifikasi pemancar dengan mempelajari fitur intrinsik dari data RF mentah, meskipun ada tantangan dalam menentukan	Keefektifan sistem 3D CNN bergantung pada parameter "jendela" $\delta$ , yang mengontrol interval waktu yang dipertimbangkan untuk inferensi. Ukuran jendela kecil cenderung menyebabkan overfitting pada dataset homogen, ukuran jendela yang lebih besar seperti $\delta = 150$	Metode 3D CNN memerlukan pemahaman mendalam tentang hubungan antara dimensi waktu dan fitur spasial namun metode yang kami kembangkan tidak memerlukan analisis spatio-temporal yang kompleks

		Identification, 2021, Q1		parameter optimal secara objektif.	bekerja dengan baik untuk dataset yang digunakan	
6	Real-Time Radar Classification Based on Software-Defined Radio Platforms: Enhancing Processing Speed and Accuracy with Graphics Processing Unit Acceleration	Seckin Oncu, Mehmet Karakaya, Yaser Dalveren, Ali Kara, & Mohammad Derawi, Sensor, 2024, Q1	Mengembangkan sistem klasifikasi radar real-time berbasis Software Defined Radio (SDR) dengan akselerasi GPU untuk meningkatkan kecepatan pemrosesan dan akurasi.	Hasil menunjukkan bahwa pemanfaatan GPU dapat meningkatkan kecepatan pemrosesan secara signifikan tanpa mengorbankan akurasi klasifikasi. Sistem ini efektif untuk implementasi real-time radar signal classification.	Penelitian masih terbatas pada arsitektur GPU tertentu, sehingga skalabilitas ke berbagai perangkat keras lain perlu diuji. Selain itu, fokus masih pada radar, belum pada jenis sinyal lain.	Penelitian ini menekankan pada optimasi kecepatan melalui akselerasi GPU dalam klasifikasi radar, sedangkan penelitian tesis ini tidak menggunakan akselerasi GPU. Sebaliknya, penelitian ini mengembangkan arsitektur CNN, CNN-BiLSTM, dan CNN-Transformer untuk mengukur serta menganalisis waktu komputasi model agar dapat diterapkan secara efisien pada real-time monitoring pemancar radio FM berbasis data IQ.
7	Convolutional Neural Networks for the Real-	Sang Ho Choi dan Heenam	Mengembangkan sistem berbasis	CNN mampu mendeteksi tanda	Keterbatasan penelitian terletak pada lingkungan	Penelitian ini menggunakan CNN untuk

<p>Time Monitoring of Vital Signs Based on Impulse Radio Ultrawide-Band Radar during Sleep</p>	<p>Yoon, Sensors, 2023, Q1</p>	<p>CNN untuk monitoring tanda vital (pernapasan dan detak jantung) secara real-time menggunakan radar UWB selama tidur, tanpa kontak langsung dengan subjek.</p>	<p>vital dengan akurasi tinggi secara real-time menggunakan data radar UWB. Pendekatan ini menunjukkan potensi kuat untuk pemantauan kesehatan non-invasif.</p>	<p>eksperimen yang masih terbatas (setting tidur terkendali). Pengujian lebih lanjut di kondisi nyata dan populasi yang lebih beragam masih diperlukan.</p>	<p>real-time monitoring sinyal, penelitian kita membandingkan tiga arsitektur model untuk efisiensi real-time, bukan hanya satu pendekatan CNN.</p>
--	--------------------------------	--	---	---	---

## 2.3 Landasan Teori

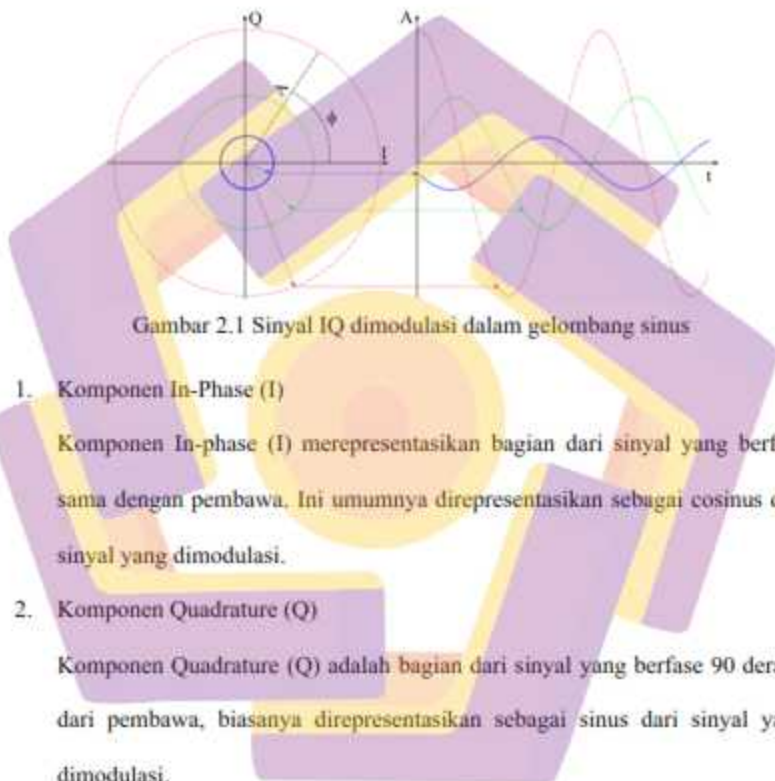
### 2.3.1. Sinyal IQ

Pemancar radio merupakan perangkat yang mengubah informasi audio menjadi gelombang radio yang dapat ditransmisikan melalui udara. Proses modulasi FM (Frequency Modulation) adalah salah satu teknik yang paling umum digunakan dalam siaran radio, di mana frekuensi pembawa sinyal berubah sesuai dengan amplitudo sinyal informasi. Sinyal IQ adalah representasi komprehensif dari sinyal radio, di mana komponen I dan Q memungkinkan representasi yang lebih efisien dan analisis yang lebih mendalam terhadap sinyal tersebut.

Menurut buku "Principles of Communication Systems" oleh Taub dan Schilling, sinyal IQ memecah sinyal kompleks menjadi dua komponen ortogonal yang dapat dianalisis secara terpisah untuk mendapatkan informasi yang lebih detail tentang karakteristik sinyal [5]. Sinyal IQ menawarkan representasi sinyal radio yang kaya dan detail dalam situasi ini, yang dapat digunakan untuk mengidentifikasi pemancar. Ini memberikan keuntungan dalam pemrosesan sinyal digital, di mana teknik-teknik lanjutan seperti CNN dapat digunakan untuk mengenali pola dan fitur unik dari sinyal tersebut. Karena kemampuan CNN untuk mengekstraksi fitur kompleks dari data sekuensial, menggunakan CNN satu dimensi (1D) adalah metode yang efektif untuk menganalisis sinyal IQ.

Sinyal IQ adalah representasi dari sinyal kompleks yang digunakan dalam komunikasi nirkabel dan pemrosesan sinyal digital. Sinyal ini terdiri dari dua komponen orthogonal: komponen In-phase (I) dan komponen Quadrature (Q) ditunjukkan pada gambar 2.1. Kedua komponen ini dapat dikombinasikan untuk

merepresentasikan amplitudo dan fase dari sinyal asli, memberikan informasi lengkap mengenai sinyal yang ditransmisikan. Sinyal IQ digunakan dalam berbagai aplikasi komunikasi, seperti modulasi digital, demodulasi, dan analisis spektrum [18].



Gambar 2.1 Sinyal IQ dimodulasi dalam gelombang sinus

#### 1. Komponen In-Phase (I)

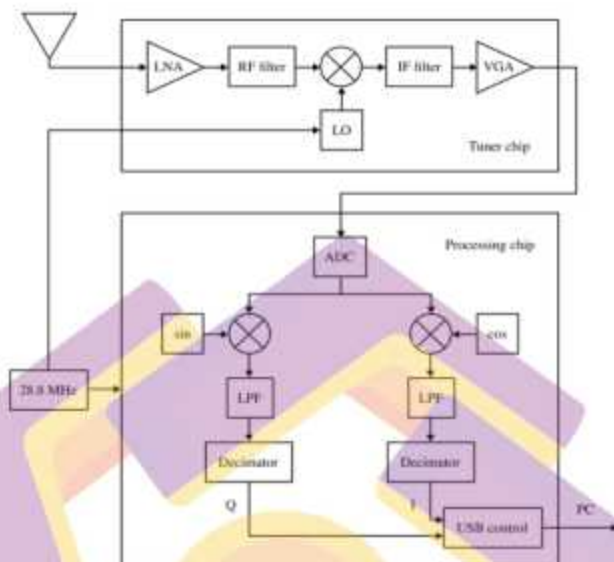
Komponen In-phase (I) merepresentasikan bagian dari sinyal yang berfase sama dengan pembawa. Ini umumnya direpresentasikan sebagai cosinus dari sinyal yang dimodulasi.

#### 2. Komponen Quadrature (Q)

Komponen Quadrature (Q) adalah bagian dari sinyal yang berfase 90 derajat dari pembawa, biasanya direpresentasikan sebagai sinus dari sinyal yang dimodulasi.

### 2.3.2. Software Define Radio (SDR) dan RTL-SDR

Software Defined Radio (SDR) adalah teknologi yang memungkinkan pemrosesan sinyal radio menggunakan perangkat lunak yang berjalan pada perangkat keras yang dapat diprogram.



Gambar 2.1. Blok Diagram RTL-SDR

RTL-SDR adalah perangkat penerima SDR berbasis chipset Realtek RTL2832U, yang mampu menerima sinyal radio dalam rentang frekuensi 24 MHz hingga 1.7 GHz dan mengeluarkan sinyal dalam bentuk data IQ.

Komponen-komponen RTL-SDR pada gambar 2.2. terdiri dari:

#### 1. Antena

Antena akan menerima sinyal radio dari udara dan mengubah sinyal elektromagnetik menjadi sinyal listrik yang dapat diproses oleh komponen elektronis pada tahap selanjutnya.

#### 2. *Low Noise Amplifier* (LNA)

Sinyal dari antena biasanya lemah sehingga memerlukan penguatan sebelum diproses selanjutnya. Hal ini penting dilakukan untuk menjaga agar kualitas sinyal.

### 3. Tuner

Sinyal yang diterima akan diproses oleh tuner untuk menjadi frekuensi menengah (IF) yang lebih mudah diproses. RTL-SDR menggunakan chip R820T sebagai tuner. Chip ini dikembangkan oleh Raphael dengan prinsip superheterodyne, dimana sinyal yang masuk akan dicampur dengan sinyal dari *oscillator* local untuk menghasilkan frekuensi IF.

### 4. Bandpass Filter

Filter ini berfungsi untuk menyaring sinyal yang dikehendaki saja yang akan diteruskan untuk diproses selanjutnya dengan membuang frekuensi yang tidak diinginkan seperti *noise* dan sinyal interferensi.

### 5. Signal Processing

Dalam blok ini terdapat chip RTL2832U yang memiliki fungsi ADC (*Analog-to-Digital Converter*), dimana sinyal yang masih analog akan dirubah menjadi sinyal digital. Chip akan mengubah sinyal analog menjadi digital dengan melakukan *sampling* menggunakan laju pengambilan sampel 28,8 MHz untuk menghasilkan sinyal digital 8-bit. Kemudian sinyal digital dari ADC akan diturunkan ke frekuensi rendah yang lebih mudah diproses lebih lanjut oleh DDC (*Digital Down Converter*) menjadi baseband. DDC juga mengurangi laju *sampling* (*decimation*) sehingga sinyal yang dihasilkan dapat ditangani secara lebih efisien.

### 6. USB Interface

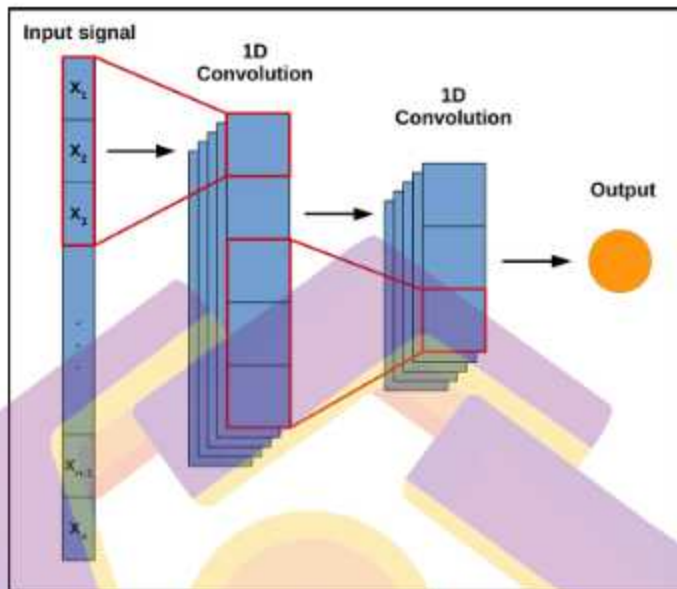
USB *Interface* berfungsi sebagai media pengiriman data digital dari RTL\_SDR ke komputer dan memungkinkan perangkat lunak untuk menerima dan memproses data digital tersebut.

RTL-SDR telah digunakan secara luas dalam penelitian dan aplikasi praktis karena harganya yang terjangkau dan fleksibilitasnya dalam menangani berbagai jenis sinyal radio [19].

### 2.3.3. Convolutional Neural Network

Convolutional Neural Networks (CNN) adalah jenis jaringan saraf tiruan yang dirancang untuk mengenali pola dalam data. CNN terdiri dari beberapa lapisan konvolusi, pooling, dan fully connected. Lapisan konvolusi mengekstraksi fitur penting dari data input, lapisan pooling mereduksi dimensi data untuk mengurangi kompleksitas, dan lapisan fully connected melakukan klasifikasi berdasarkan fitur yang diekstraksi. CNN telah terbukti efektif dalam menangkap fitur spasial dan temporal dari data yang terstruktur [20].

CNN 1 dimensi (1D) digunakan untuk menangani data yang disusun dalam bentuk deret waktu atau satu dimensi, seperti sinyal IQ. Pada CNN 1D, konvolusi dilakukan dengan kernel (filter) yang bergerak sepanjang satu sumbu. CNN 1D mampu mengekstraksi fitur temporal atau sekuensial dari data, seperti pola berulang, tren, dan transisi antar waktu. Menurut buku "Deep Learning" oleh Ian Goodfellow, Yoshua Bengio, dan Aaron Courville, CNN mampu menangkap fitur lokal dari data input melalui operasi konvolusi, yang membuatnya sangat cocok untuk analisis sinyal yang memiliki korelasi temporal atau spasial [21].



Gambar 2.2. CNN 1D

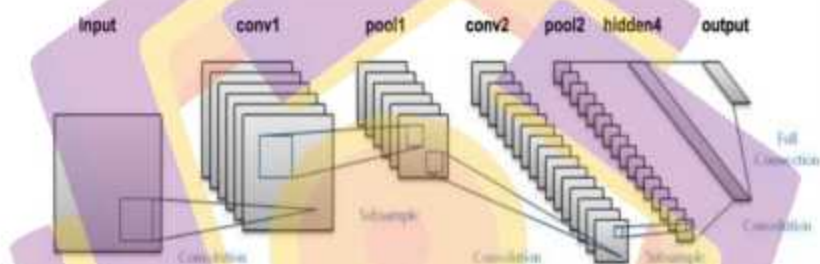
Komponen-komponen CNN 1D pada gambar 2.3. terdiri dari:

1. Input  
Data 1D seperti sinyal audio atau deret waktu dimasukkan dalam bentuk vector.
2. Convolutional  
Layer ini menggunakan filter 1D yang bergerak melintasi input, menghitung produk titik (*dot product*) antara filter dan bagian dari input yang dilaluinya. Hasilnya adalah fitur map 1D.
3. Pooling

Pooling 1D digunakan untuk mengurangi dimensi fitur map, biasanya dengan mengambil nilai maksimum (*max pooling*) atau rata-rata (*average pooling*) dari segmen data.

#### 4. Fully Connected

Layer ini menghubungkan semua neuron dari layer sebelumnya ke semua neuron di layer berikutnya, seperti dalam neural network konvensional, untuk pengambilan keputusan atau klasifikasi.



Gambar 2.3. CNN 2D

CNN 2 dimensi (2D) merupakan jenis CNN yang paling umum dan biasanya digunakan untuk pengolahan citra. Pada CNN 2D, filter bergerak di sepanjang dua sumbu (lebar dan tinggi) untuk mengekstraksi fitur dari gambar, seperti tepi, sudut, dan pola tekstur. CNN 2D sangat efektif dalam memahami struktur spasial dari gambar [20].

Komponen-komponen CNN 2D pada gambar 2.4. terdiri dari:

##### 1. Input

Data input berupa gambar dua dimensi (misalnya, 28x28 piksel untuk gambar grayscale).

##### 2. Convolutional

Layer ini menggunakan filter 2D yang bergerak melintasi seluruh gambar, menghitung produk titik antara filter dan bagian dari gambar untuk menghasilkan fitur map 2D.

### 3. Pooling

Pooling 2D, seperti max pooling, digunakan untuk mengurangi dimensi fitur map sambil menjaga informasi penting.

### 4. Fully Connected

Layer ini menghubungkan semua neuron dari layer sebelumnya ke semua neuron di layer berikutnya, seperti dalam neural network konvensional, untuk pengambilan keputusan atau klasifikasi.

CNN 3D digunakan untuk data volumetrik, seperti video atau gambar medis 3D (misalnya, MRI atau CT scan). Pada CNN 3D, filter bergerak melintasi tiga sumbu (panjang, lebar, dan kedalaman) untuk mengekstraksi fitur spasial dan temporal secara bersamaan.

### 1. Input

Data input berupa tensor tiga dimensi, seperti volume voxel dalam gambar MRI atau urutan frame dalam video.

### 2. Convolutional

Filter 3D bergerak melintasi seluruh volume, menghitung produk titik di sepanjang ketiga sumbu, untuk menghasilkan fitur map 3D.

### 3. Pooling

Pooling 3D digunakan untuk mengurangi dimensi fitur map 3D sambil menjaga informasi penting di seluruh volume.

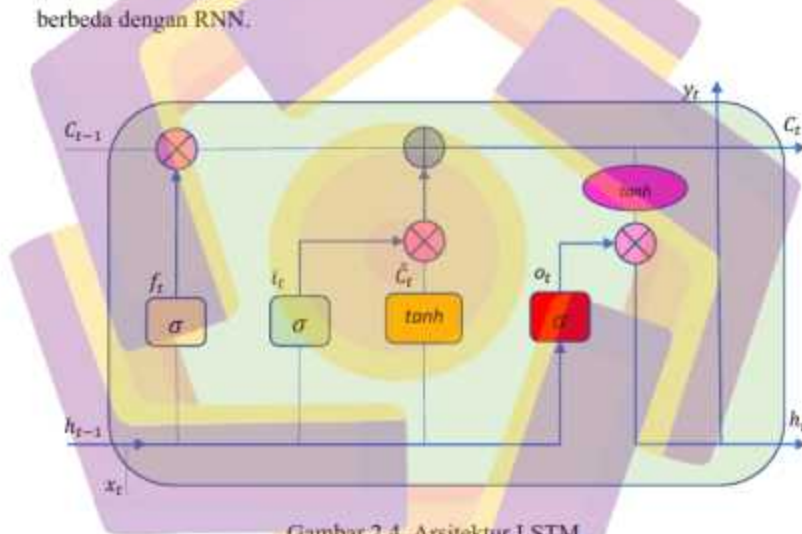
#### 4. Fully Connected

Layer ini menghubungkan semua neuron dari layer sebelumnya ke semua neuron di layer berikutnya, seperti dalam neural network konvensional, untuk pengambilan keputusan atau klasifikasi.

#### 2.3.4. BiLSTM dan CNN-BiLSTM

Umumnya, model seperti Long Short-Term Memory (LSTM) dirancang untuk data berurutan, tetapi penelitian ini mengadaptasinya untuk data tabular. Sebuah tantangan penerapan model deep learning (DL) pada data tabular yang tidak memiliki struktur sekuensial alami seperti data time-series. Proses preprocessing data mencakup beberapa tahap, seperti ekstraksi fitur, penanganan data hilang, normalisasi menggunakan MinMaxScaler untuk meratakan skala ke rentang 0 hingga 1, vektorisasi dan encoding data kategorikal menjadi representasi numerik, serta deteksi dan penanganan outlier. Data tabular kemudian diubah menjadi tensor 3D dengan format (number\_of\_samples, 1, number\_of\_features), di mana setiap sampel memiliki satu time-step. Arsitektur model terdiri dari lapisan input yang menerima tensor 3D, tiga lapisan tersembunyi berupa BiLSTM bertumpuk, dan lapisan dense yang menghasilkan output berupa kelas biner (0 atau 1). Jumlah node pada tiap lapisan bertambah untuk meningkatkan kapasitas model dalam menangkap pola kompleks [22]. LSTM merupakan salah satu jenis dari Recurrent Neural Network Dimana dilakukan modifikasi pada RNN dengan menambahkan memory cell yang dapat menyimpan informasi untuk jangka panjang [23]. LSTM diusulkan sebagai solusi untuk mengatasi terjadinya vanishing gradient pada RNN saat memproses data sequential yang panjang. Permasalahan vanishing gradients

ini mengakibatkan RNN gagal dalam menangkap long term dependencies, sehingga mengurangi akurasi dari suatu prediksi pada RNN. Mengingat informasi jangka panjang merupakan perilaku bawaan LSTM oleh karena itu LSTM cocok digunakan untuk masalah yang memiliki ketergantungan jangka panjang. LSTM memiliki struktur rantai yang hampir sama dengan struktur RNN, perbedaannya terletak pada struktur model pengulangannya [24]. Sebagai gantinya, LSTM memiliki empat lapisan jaringan syaraf tunggal yang berinteraksi dengan cara yang berbeda dengan RNN.



Gambar 2.4. Arsitektur LSTM

Gambar 2.5. menggambarkan arsitektur dari LSTM. Dalam LSTM terdapat 3 gate yaitu input gate, forget gate dan output gate [25]. Proses komputasi pada LSTM dilakukan dengan tahapan berikut:

1. Nilai dari suatu input hanya dapat disimpan kedalam *cell state* hanya jika diijinkan oleh *input gate*. Perhitungan dari nilai pada *input gate* dan kandidat

dari *cell state* dilakukan dengan menggunakan persamaan 1 dan persamaan 2 berikut:

$$i_t = \sigma(W_i x_t + U_i h_{t-1} + b_i)$$

Dimana  $i_t$  adalah nilai dari *input gate*,  $W_i$  adalah bobot nilai input pada waktu ke  $t$ ,  $x_t$  adalah nilai input pada waktu ke  $t$ ,  $U_i$  adalah bobot untuk nilai dari waktu ke  $t - 1$ ,  $h_{t-1}$  adalah nilai *output* dari waktu ke  $t - 1$  dan  $b$  adalah bias pada *input gate* dan  $b_i$  adalah fungsi *sigmoid*.

$$\hat{C}_t = \tanh(W_c x_t + U_c h_{t-1} + b_c)$$

Dimana  $\hat{C}_t$  adalah nilai kandidat *cell state*,  $W_c$  adalah bobot untuk nilai input pada *cell* ke  $c$ ,  $x_t$  adalah nilai *input* pada waktu ke  $t$ ,  $U_c$  adalah bobot untuk nilai *output* dari *cell* ke  $c - 1$ ,  $h_{c-1}$  adalah nilai *output* dari *cell* ke  $c - 1$ , dan  $b_c$  adalah bias pada *cell* ke  $c$  dan  $\tanh$  adalah fungsi *hyperbolic tangent*.

2. Kemudian nilai dari *forget gate* dihitung dengan menggunakan persamaan 3.

$$f_t = \sigma(W_f x_t + U_f h_{t-1} + b_f)$$

Dimana  $f_t$  adalah nilai dari *forget gate*,  $W_f$  adalah bobot untuk nilai input pada waktu ke  $t$ ,  $x_t$  adalah nilai *input* pada waktu ke  $t$ ,  $U_f$  adalah bobot untuk nilai *output* dari waktu ke  $t - 1$ ,  $h_{t-1}$  adalah nilai *output* dari waktu ke  $t - 1$ , dan  $b_f$  adalah bias pada *forget gate* dan  $\sigma$  adalah fungsi *sigmoid*.

3. Selanjutnya *memory cell state* dihitung menggunakan persamaan 4

$$C_t = i_t * \hat{C}_t + f_t * C_{t-1}$$

Dimana  $C_t$  adalah nilai *memory cell state*,  $i_t$  adalah nilai dari *input gate*,  $\hat{C}_t$  adalah nilai kandidat *memory cell state*,  $f_t$  adalah nilai *forget gate* dan  $C_{t-1}$  adalah nilai *memory cell state* pada *cell* sebelumnya.

4. Setelah dihasilkan *memory cell state* yang baru, nilai dari *output gate* dapat dihitung dengan menggunakan persamaan 5.

$$o_t = \sigma(W_o x_t + U_o h_{t-1} + b_o)$$

Dimana  $o_t$  adalah nilai dari *output gate*,  $W_o$  adalah bobot untuk nilai *input* pada waktu ke  $t$ ,  $x_t$  adalah nilai *input* pada waktu ke  $t$ ,  $U_o$  adalah bobot untuk nilai *output* dari waktu ke  $t-1$ ,  $h_{t-1}$  adalah nilai *output* dari waktu ke  $t-1$  dan  $b_o$  adalah bias pada *output gate* dan  $\sigma$  adalah fungsi *sigmoid*.

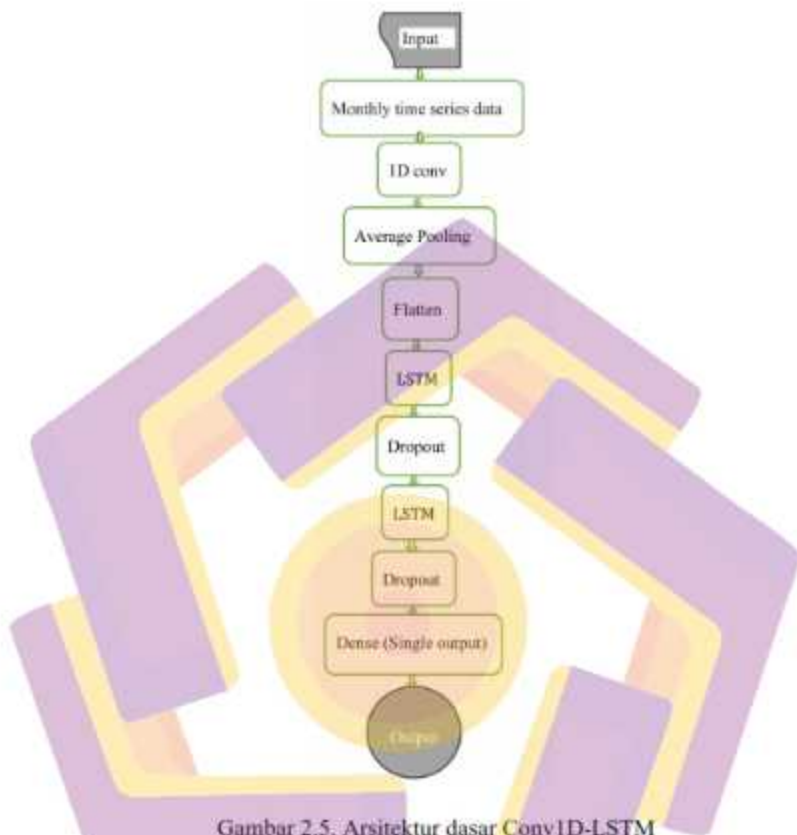
5. Nilai *output final* dihitung dengan menggunakan persamaan 6.

$$h_t = o_t * \tanh(C_t)$$

Dimana  $h_t$  adalah *output final*,  $o_t$  adalah nilai *output gate*,  $C_t$  adalah nilai *memory cell state* yang baru dan  $\tanh$  adalah fungsi *hyperbolic tangent*.

CNN (Convolutional Neural Network) diakui sebagai salah satu model deep learning paling berhasil, terutama berkat efektivitasnya dalam ekstraksi fitur. Arsitektur jaringannya mencakup Conv1D, Conv2D, dan Conv3D. Secara umum, struktur CNN terdiri atas lapisan konvolusi, lapisan pooling, dan lapisan fully connected. Dalam CNN, lapisan konvolusi dan pooling berfungsi sebagai blok pembangun utama. Kedua lapisan ini mengekstraksi berbagai fitur dari lapisan masukan serta mengurangi dimensinya dengan melakukan operasi konvolusi pada lapisan masukan dan menggabungkan keluaran dari sekumpulan neuron menjadi satu neuron. Mekanisme pooling berperan penting dalam mengurangi jumlah parameter pada jaringan, sehingga meningkatkan efisiensi, kemudahan, dan kecepatan tahap pelatihan CNN dibandingkan Artificial Neural Network (ANN) konvensional. Conv1D umumnya digunakan untuk memproses data berurutan

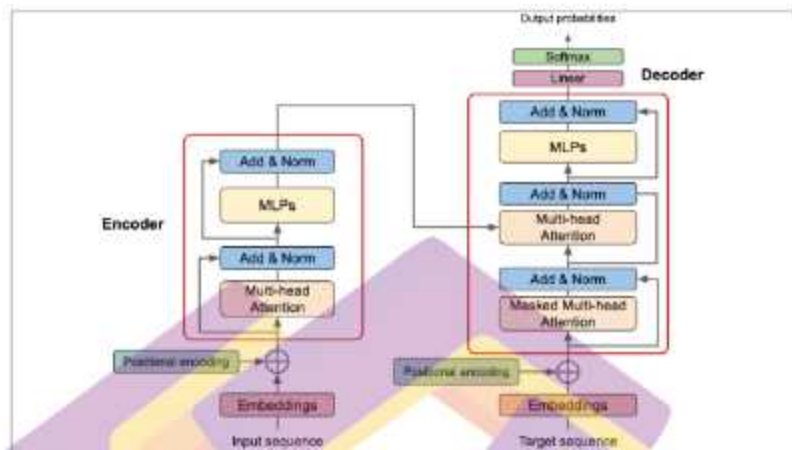
(sequence data). Conv2D biasanya dimanfaatkan untuk pengenalan teks dan citra, sementara Conv3D digunakan untuk pemodelan citra medis dan pengenalan data video. Output dari lapisan CNN digunakan sebagai masukan (input) bagi lapisan LSTM untuk menangkap ketergantungan jangka pendek dan panjang [26]. Bidirectional LSTM (BiLSTM) dapat mengekstraksi informasi kontekstual dari rangkaian fitur dengan mempertimbangkan ketergantungan baik ke arah depan maupun belakang. Pendekatan ini memanfaatkan forward LSTM yang memproses rangkaian secara kronologis dan backward LSTM yang memproses rangkaian secara terbalik, sehingga memungkinkan model untuk melakukan "peramalan ke depan." Hasil keluaran kemudian diperoleh dengan menggabungkan state LSTM dari arah depan dan belakang. Model Conv1D-LSTM tersusun atas dua bagian utama; bagian pertama terdiri dari lapisan konvolusi dan pooling, yang kemudian diikuti oleh lapisan flatten untuk mempersiapkan data sebelum memasuki LSTM [27]. Bagian kedua memanfaatkan lapisan LSTM dan dense guna memproses fitur, dengan menyertakan dropout untuk mencegah overfitting seperti pada Gambar 2.6.



Gambar 2.5. Arsitektur dasar Conv1D-LSTM

### 2.3.5. Transformer dan CNN-Transformer

Arsitektur Transformer awalnya diperkenalkan dalam bidang NLP [28] sebagai pengganti RNN/LSTM dengan mekanisme self-attention yang memungkinkan model menangkap relasi jarak panjang (long-range dependencies) secara paralel dan efisien. Model ini terbukti sangat kuat dalam mempelajari konteks global tanpa ketergantungan sekuensial.



Gambar 2.6. Arsitektur Transformer

Dari gambar 2.7. arsitektur Transformer terdiri dari beberapa komponen utama, yaitu Encoder dan Decoder. Encoder merupakan salah satu blok utama dalam arsitektur Transformer yang terletak tepat pada bagian input dari urutan masukan. Encoder berfungsi untuk mentransformasi urutan input menjadi representasi terkompresi. Pada arsitektur Transformer asli, encoder diulang sebanyak 6 kali (jumlah ini bergantung pada ukuran keseluruhan arsitektur, sehingga dapat diubah sesuai kebutuhan). Setiap blok encoder memiliki 3 lapisan utama, yaitu multi-head attention (MHA), layer normalization, dan MLP (multi-layer perceptrons atau disebut juga feedforward sesuai dengan istilah dalam makalah asli).

Multi-head attention dan MLP disebut sebagai sub-layer dalam makalah Transformer. Di antara setiap sub-layer, terdapat mekanisme layer normalization, dropout, dan residual connection (hubungan residual) agar aliran informasi tetap stabil (lihat diagram untuk alur lapisan yang benar).

Jumlah lapisan encoder pada arsitektur asli adalah 6. Semakin banyak jumlah lapisan encoder, semakin besar ukuran model, dan semakin baik kemampuan model untuk menangkap konteks global dari urutan input. Dengan demikian, hal ini dapat menghasilkan generalisasi yang lebih baik terhadap tugas yang dikerjakan.

Decoder pada dasarnya memiliki struktur yang hampir sama dengan encoder, kecuali terdapat tambahan multi-head attention yang bekerja di atas keluaran dari encoder. Tujuan utama decoder adalah menggabungkan keluaran encoder dengan urutan target, lalu menghasilkan prediksi (atau memprediksi token berikutnya dalam urutan).

Pada decoder, attention yang mengambil urutan target dibuat masked (tertutup) untuk mencegah token saat ini (yang sedang diproses) memperhatikan token berikutnya dalam urutan target. Jika decoder memiliki akses penuh terhadap keseluruhan urutan target, hal ini akan dianggap sebagai "kecurangan" dan dapat menghasilkan model yang tidak mampu melakukan generalisasi di luar data latih. Decoder umumnya juga diulang sebanyak jumlah lapisan encoder. Pada arsitektur Transformer asli, jumlah blok decoder juga sebanyak 6 blok.

Proses komputasi pada Transformer dilakukan dengan tahapan berikut:

1. Scaled Dot-Product Attention yang bekerja dengan memanfaatkan tiga vektor, yaitu Query (Q), Key (K), dan Value (V). Dengan representasi matematika sebagai berikut :

$$\text{Attention}(Q, K, V) = \text{Softmax}\left(\frac{QK^T}{\sqrt{d_k}}\right)V$$

Hubungan antara query dan key dihitung melalui operasi dot product, kemudian dinormalisasi dengan faktor  $\sqrt{d_k}$  agar stabil, dilanjutkan dengan fungsi softmax untuk menghasilkan bobot perhatian. Bobot ini digunakan untuk melakukan kombinasi linear terhadap value sehingga menghasilkan representasi kontekstual.

2. Multi-Head Attention menggunakan beberapa “kepala” attention secara paralel. Setiap kepala menerima proyeksi linier (linear projection) dari Q, K, dan V, melaksanakan attention secara independen, lalu hasilnya dikombinasikan (dengan concatenation dan proyeksi linier akhir). Memungkinkan model untuk secara paralel mempelajari hubungan antar elemen input pada sub-ruang representasi berbeda. Dengan demikian, berbagai pola lokal maupun global dapat ditangkap secara simultan.
3. Karena arsitektur Transformer tidak secara intrinsik mempertahankan urutan (order) seperti RNN, maka ada proses embedding untuk mengubah token (atau elemen input) menjadi vektor berdimensi tetap, dan positional encoding ditambahkan agar informasi posisi setiap elemen dalam urutan tetap diketahui oleh model.
4. Setelah attention, ada lapisan MLP (multi-layer perceptron) atau feed-forward network yang diaplikasikan pada setiap posisi secara independen. Biasanya terdiri dari dua lapis linear dengan aktivasi nonlinier (misalnya ReLU) di antaranya.
5. Terakhir terdapat proses Residual connections untuk mempermudah aliran gradien selama training dan membantu stabilitas pelatihan. Layer

normalization untuk menormalkan aktivasi dalam layer agar distribusi aktivasi lebih stabil terutama saat training. Serta dropout untuk regularisasi agar model tidak overfitting.

Keunggulan utama Transformer terletak pada tiga aspek. Pertama, kemampuannya menangkap hubungan jangka panjang (long-range dependencies) karena setiap elemen input dapat secara langsung berinteraksi dengan seluruh elemen lain dalam urutan. Kedua, arsitektur ini sangat efisien pada perangkat keras modern karena mendukung paralelisasi penuh selama pelatihan, tidak seperti RNN yang sekuensial. Ketiga, mekanisme attention memberikan interpretabilitas yang lebih tinggi, karena memungkinkan peneliti atau praktisi untuk memvisualisasikan bagian mana dari input yang paling memengaruhi prediksi model.

Namun demikian, Transformer juga memiliki keterbatasan. Mekanisme self-attention memiliki kompleksitas kuadratik terhadap panjang urutan input, sehingga memerlukan sumber daya komputasi dan memori yang besar. Selain itu, model ini cenderung membutuhkan data latih dalam jumlah besar agar dapat mencapai kinerja optimal, dan rentan mengalami instabilitas apabila regularisasi serta konfigurasi arsitektur tidak dirancang dengan tepat.

Dalam domain computer vision, Vision Transformers (ViT) mengadaptasi model Transformer dengan membagi citra menjadi patch—seperti token dalam teks—dan menerapkan self-attention antar patch. Pendekatan ini mampu mencapai performa setara atau bahkan melampaui CNN, terutama jika dilatih dengan data besar dan pretraining yang memadai [29] [30].

Namun, Transformer sering kali kurang efisien dalam menangkap fitur lokal seperti tepi atau tekstur, karena tidak memiliki inductive bias lokal seperti pada CNN. Hal ini mendorong munculnya model hybrid yang memadukan CNN dan Transformer untuk memaksimalkan keunggulan keduanya [31].

### 2.3.6. Klasifikasi Radio Siaran Dengan CNN

Identifikasi dan klasifikasi stasiun radio siaran berbasis data sinyal IQ merupakan salah satu penerapan penting dari teknologi deep learning pada bidang pengawasan spektrum. Data IQ yang direkam dari perangkat Software Defined Radio (SDR) merepresentasikan sinyal kompleks dalam domain waktu yang memuat informasi amplitudo dan fase. Karakteristik ini memungkinkan analisis pola modulasi, perbedaan perangkat keras pemancar, hingga sidik jari unik (device fingerprint) dari setiap stasiun radio.

Convolutional Neural Networks (CNN) telah terbukti efektif dalam mengklasifikasikan sinyal komunikasi, termasuk sinyal radio siaran, karena kemampuannya mengekstraksi fitur spasial dan pola berulang secara otomatis tanpa memerlukan teknik ekstraksi manual seperti FFT atau transformasi wavelet [3]. Pada data IQ, lapisan konvolusi dapat mendeteksi pola lokal seperti variasi amplitudo atau pergeseran fase, sedangkan lapisan yang lebih dalam menangkap representasi abstrak yang berkaitan dengan identitas pemancar. Penelitian terdahulu menunjukkan bahwa CNN dapat mencapai akurasi klasifikasi tinggi dalam identifikasi sinyal komunikasi bahkan pada kondisi SNR rendah [32]. Oleh karena itu, CNN menjadi pendekatan yang rasional dan relevan dalam penelitian ini untuk klasifikasi stasiun radio siaran FM berdasarkan data IQ.

### 2.3.7. Confusion Matrix

Confusion matrix merupakan metode yang sering digunakan untuk mengukur dan mengevaluasi performa model [33]. Confusion matrix terdiri dari baris data uji yang diprediksi benar dan tidak benar suatu data oleh model klasifikasi. Tabel 2.1. menunjukkan confusion matrix untuk pengklasifikasian kedalam dua kelas.

Tabel 2.2. Confusion Matrix

Aktual/Prediksi	Positif	Negatif	Total
Positif	TP	FN	Total true
Negatif	FP	TN	Total Predictes

$$Akurasi = \frac{(TP + TN)}{(TP + TN + FP + FN)}$$

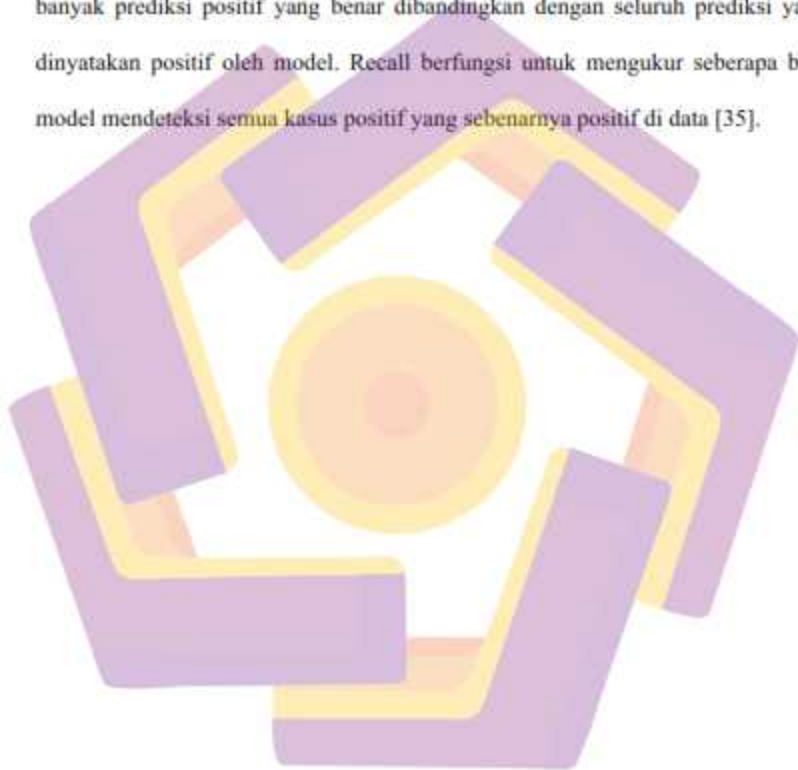
$$Presisi = \frac{TP}{(TP + FP)}$$

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

$$F1 - Score = 2 \times \frac{presisi \times recall}{presisi + recall}$$

True Positive (TP) yaitu jumlah kasus di mana model memprediksi kelas positif dan data aktualnya juga positif. False Negative (FN) yaitu jumlah kasus di mana model memprediksi kelas negatif, tetapi data aktualnya positif. False Positive (FP) yaitu jumlah kasus di mana model memprediksi kelas positif, tetapi data aktualnya negatif. True Negative (TN) yaitu jumlah kasus di mana model memprediksi kelas negatif dan data aktualnya juga negatif [34]. Confusion matrix

memberikan pandangan mendetail mengenai kesalahan prediksi model seperti akurasi, presisi, recall, F-1 score [33]. Akurasi digunakan untuk mengukur seberapa sering model membuat prediksi yang benar, baik untuk kelas positif maupun kelas negatif, dari seluruh prediksi yang dilakukan. Presisi berfungsi mengukur seberapa banyak prediksi positif yang benar dibandingkan dengan seluruh prediksi yang dinyatakan positif oleh model. Recall berfungsi untuk mengukur seberapa baik model mendeteksi semua kasus positif yang sebenarnya positif di data [35].



## **BAB 3**

### **METODE PENELITIAN**

#### **3.1 Jenis, Sifat, dan Pendekatan Penelitian**

Penelitian ini termasuk dalam jenis penelitian terapan karena bertujuan untuk memecahkan permasalahan praktis dalam bidang pengawasan spektrum frekuensi radio, khususnya dalam hal klasifikasi pemancar siaran FM secara otomatis dan efisien menggunakan pendekatan teknologi berbasis deep learning. Penelitian ini diharapkan dapat memberikan kontribusi langsung terhadap peningkatan efektivitas sistem monitoring spektrum frekuensi radio, terutama dalam konteks *real-time* dan lingkungan yang padat interferensi.

Berdasarkan sifatnya, penelitian ini merupakan penelitian kuantitatif karena melibatkan proses pengumpulan dan analisis data numerik, berupa data IQ dalam bentuk sinyal digital, yang kemudian dianalisis secara statistik menggunakan metrik evaluasi model, seperti akurasi, presisi, recall, F1-score dan waktu inferensi. Seluruh pengambilan kesimpulan dilakukan berdasarkan hasil pengukuran yang objektif dan dapat diuji secara empiris.

Ditinjau dari pendekatan penelitian, penelitian ini menggunakan pendekatan eksperimental, dimana beberapa model deep learning, yaitu: CNN-BiLSTM, CNN-Transformer, CNN 5-Layers, dikembangkan, dilatih dan diuji terhadap data IQ dari 16 pemancar siaran FM. Proses eksperimen dilakukan dalam lingkungan terkontrol dengan parameter pelatihan yang seragam untuk memastikan validitas

perbandingan performa antar model. Data dikumpulkan melalui proses pelatihan dan pengujian model secara sistematis.

Dengan karakteristik tersebut, penelitian ini dapat dikategorikan sebagai penelitian kuantitatif terapan dengan pendekatan eksperimental, yang bertujuan menghasilkan solusi praktis berbasis teknologi untuk mendukung sistem pengawasan spektrum frekuensi radio secara otomatis, akurat dan efisien.

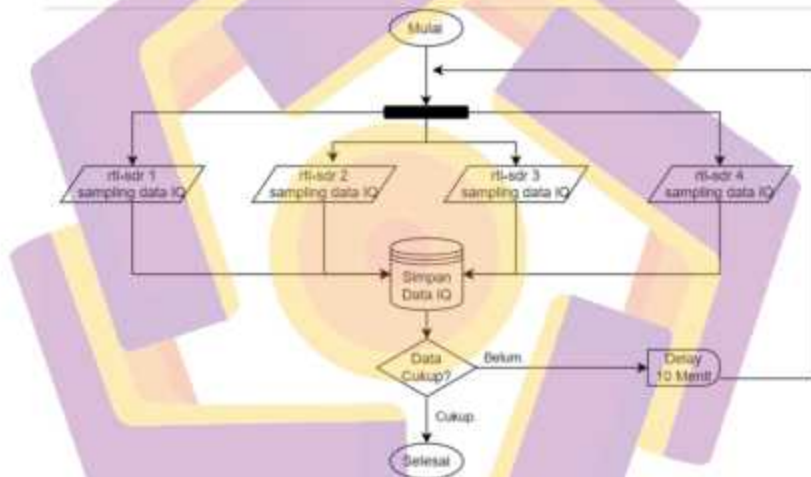
### **3.2 Metode Pengumpulan Data**

Pengumpulan data dalam penelitian ini dilakukan melalui metode eksperimen dan observasi teknis, yang keduanya memiliki peran penting dalam menjamin validitas dan kualitas data IQ yang digunakan untuk pelatihan model klasifikasi.

Metode eksperimen dalam penelitian ini merujuk pada proses perancangan dan pelaksanaan perekaman data IQ secara terkontrol menggunakan perangkat RTL-SDR dan antena penerima FM. Peneliti secara aktif melakukan konfigurasi parameter seperti frekuensi tengah (center frequency), laju sampling, serta waktu perekaman untuk masing-masing pemancar radio FM. Variabel-variabel teknis seperti lokasi pengambilan data dan jenis antena divariasikan secara sengaja untuk mengevaluasi pengaruh lingkungan dan perangkat terhadap karakteristik sinyal IQ yang terekam. Pendekatan eksperimental ini memungkinkan peneliti untuk mengendalikan kondisi perekaman, sehingga data yang dihasilkan memiliki kualitas yang representatif dan bervariasi.

Sementara itu, metode observasi teknis digunakan untuk memastikan kondisi lingkungan dan perangkat saat proses perekaman berlangsung, serta untuk

memverifikasi bahwa sinyal yang diterima memang berasal dari pemancar yang dituju. Observasi teknis dilakukan dengan memantau spektrum sinyal, memvalidasi identitas pemancar melalui informasi metadata (frekuensi, nama siaran, waktu), serta mencatat adanya gangguan atau interferensi yang dapat mempengaruhi hasil perekaman. Observasi ini juga mencakup penggantian antena secara berkala dan pemilihan lokasi pengambilan yang tersebar, untuk mengumpulkan variasi sinyal dari berbagai sudut penerimaan.



Gambar 3.1. Flowchart Pengumpulan Data IQ

Data dikumpulkan secara otomatis menggunakan aplikasi yang dikembangkan dengan bahasa pemrograman Python. Aplikasi ini dirancang untuk merekam data IQ terhadap 16 stasiun radio siaran FM yang mengudara di wilayah Kota Gorontalo menggunakan perangkat RTL-SDR sebanyak 4 unit. Dari gambar 3.1 di atas memperlihatkan diagram alir proses pengumpulan data IQ dari stasiun-stasiun radio FM. Masing-masing dari perangkat RTL-SDR melakukan proses

sampling terhadap sinyal IQ dari frekuensi-frekuensi yang telah ditentukan. Setelah semua data dari RTL-SDR masuk ke repositori penyimpanan, sistem akan mengevaluasi apakah jumlah data yang dikumpulkan sudah mencukupi untuk kebutuhan pelatihan model. Proses pengumpulan dilakukan setiap 10 menit, mulai pukul 10.00 hingga 18.00 WITA, selama beberapa hari. Masing-masing stasiun memiliki frekuensi siaran yang berbeda dan mewakili keragaman spektrum siaran FM yang tersedia secara komersial maupun publik. Setiap stasiun diperlakukan sebagai satu kelas tersendiri dalam proses klasifikasi berbasis data IQ. Tujuan dari pemilihan ini adalah untuk memastikan bahwa model yang dikembangkan mampu mengenali karakteristik sinyal dari masing-masing pemancar secara akurat, meskipun sinyal memiliki modulasi dan kekuatan pancaran yang serupa.

Adapun daftar stasiun radio dan frekuensi siaran yang diamati dalam penelitian ini disajikan pada Tabel 3.1 berikut:

Tabel 3.1. Pemancar Radio Penelitian

NO	NAMA STASIUN RADIO	FREKUENSI (MHz)
1	Relay RRI Pro 3	88.8
2	Memora	89.2
3	Kharisma	90
4	Nada	90.8
5	RRI Pro 2	92.4
6	Be FM	94.3
7	RRI Pro 4	95.1
8	RRI Pro 3	96.7
9	Insania	97.9
10	SK FM	99.1
11	Radio Hulondhalo	99.9
12	Celebes	101
13	RRI Pro 1	101.8

NO	NAMA STASIUN RADIO	FREKUENSI (MHz)
14	RAL	102.6
15	Poliyama	104.2
16	Noise Floor	87.5

Label dan metadata disimpan secara paralel menggunakan NumPy array `train_labels.npy` dan `train_sample_names.npy`, yang dikombinasikan secara iteratif selama proses perekaman berlangsung. Tujuan dari proses ini adalah untuk memastikan bahwa setiap sampel IQ yang dikumpulkan memiliki label kelas yang benar sesuai dengan identitas pemancar, serta dapat ditelusuri kembali melalui informasi nama siaran FM.

Proses pengumpulan dengan cara mengiterasi seluruh daftar nama pemancar radio (disimpan dalam variabel `devices`) dan menghasilkan label numerik yang sesuai dengan indeks pemancar tersebut. Proses ini dilakukan dalam dua loop bersarang yang merepresentasikan struktur pengulangan pengambilan data: loop pertama (`for x in range(1)`) sebagai kontrol jumlah sesi utama, dan loop kedua (`for y in range(50)`) sebagai representasi jumlah batch dalam satu sesi. Dengan demikian, untuk setiap pemancar (`for index in range(len(devices))`), ditambahkan satu label numerik (`all_train_labels`) dan nama pemancar (`all_train_names`).

Setelah proses penandaan selesai, skrip melanjutkan dengan memuat file `.npy` yang sebelumnya berisi label (`train_labels.npy`) dan nama-nama sampel (`train_sample_names.npy`) yang telah disimpan. Kemudian, label dan nama baru yang telah dikumpulkan selama sesi loop digabungkan ke dalam data lama

menggunakan fungsi `np.concatenate()`. Proses ini memungkinkan pelabelan data berlangsung secara inkremental seiring proses perekaman berlangsung.

Sebagai tindakan pencegahan terhadap kemungkinan kegagalan saat penyimpanan file (misalnya karena file terkunci atau rusak), digunakan blok `try-except` untuk menangani potensi exception saat menyimpan data. Di akhir proses, skrip dijeda selama 30 detik (`time.sleep(30)`) untuk memberi waktu sebelum sesi selanjutnya, dan perangkat RTL-SDR ditutup menggunakan `sdr.close()` untuk menjaga stabilitas perangkat keras.

Proses ini merupakan bagian penting dari pipeline pengumpulan data karena memastikan keterkaitan antara sampel IQ yang terekam dengan label kelas yang sesuai dan informasi metadata siaran. Dengan pendekatan ini, seluruh dataset tersusun secara sistematis kemudian digunakan dalam proses pra-pemrosesan dan pelatihan model.

### 3.3 Metode Analisis Data

Metode analisis data yang digunakan adalah pendekatan supervised classification menggunakan algoritma deep learning dalam mengklasifikasikan sinyal IQ dari pemancar siaran FM. Analisis dilakukan secara kuantitatif dengan mengukur performa model berdasarkan metrik klasifikasi, serta secara eksperimental untuk membandingkan tiga arsitektur model berbeda, yaitu CNN-BiLSTM, CNN-Transformer dan CNN 5-Layers.

Analisis data pada penelitian ini dilakukan menggunakan perangkat lunak berikut:

- Python 3.10

Merupakan bahasa utama untuk pemrosesan data dan pelatihan model

- TensorFlow 2.11 dan Keras

Digunakan untuk membangun dan melatih model CNN, BiLSTM dan Transformer

- NumPy dan Pandas

Digunakan untuk mengelola *array* data IQ dan struktur label

- Matplotlib dan Seaborn

Digunakan untuk visualisasi akurasi, *loss* dan *confusion matrix*

- Scikit-Learn

Digunakan untuk metrik evaluasi (akurasi, presisi, *recall* dan *F1-score*)

Dataset terdiri dari 80.000 sampel yang telah discimbangkan (balanced) untuk 16 pemancar siaran FM, masing-masing dengan 5.000 sampel. Setiap sampel direpresentasikan dalam format array dua dimensi berukuran (4096, 2).

Data dibagi menjadi:

- 64% untuk data pelatihan (51.200 sampel),
- 16% untuk data validasi (12.800 sampel),
- 20% untuk data pengujian (16.000 sampel).

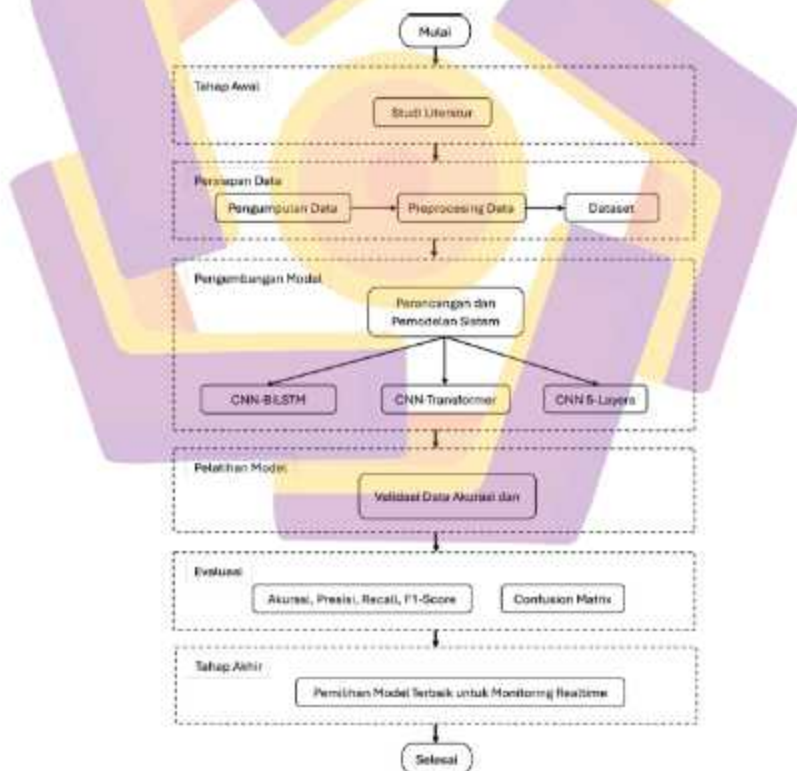
Model dilatih menggunakan arsitektur CNN 5-Layers, CNN-BiLSTM, dan CNN-Transformer. Pelatihan dilakukan menggunakan optimizer Adam, dengan learning rate 0.0003 dan fungsi loss sparse categorical crossentropy. Mekanisme regulasi berupa EarlyStopping dan ReduceLROnPlateau diterapkan untuk mencegah overfitting.

Evaluasi dilakukan terhadap data pengujian menggunakan metrik akurasi, presisi, recall, F1-score, serta visualisasi confusion matrix. Model terbaik ditentukan berdasarkan keseimbangan antara akurasi dan kecepatan inferensi.

### 3.4 Alur Penelitian

Alur penelitian disini menjelaskan mengenai tahapan-tahapan atau prosedur penelitian yang akan dilaksanakan secara urut dan sistematis agar pelaksanaan penelitian dapat berjalan dengan baik.

Adapun alur penelitian ini dapat dilihat pada gambar 3.2 sebagai berikut:



Gambar 3.2. Alur Penelitian

## 1. Tahap Awal

Tahap awal diawali dengan identifikasi permasalahan yang dihadapi dalam proses pengawasan spektrum frekuensi radio yang masih dilakukan secara manual dan belum efisien. Kegiatan ini mencakup analisis terhadap tantangan pengawasan, seperti keterbatasan sumber daya manusia, waktu, serta potensi kesalahan identifikasi sinyal pada lingkungan spektrum yang padat interferensi.

Selanjutnya dilakukan studi literatur yang mendalam mengenai penerapan kecerdasan buatan, khususnya deep learning, dalam pengolahan sinyal radio. Literatur yang dikaji mencakup teori dasar sinyal IQ (In-phase dan Quadrature), metode ekstraksi fitur menggunakan CNN (Convolutional Neural Network), pengolahan urutan temporal menggunakan BiLSTM (Bidirectional Long Short-Term Memory), serta penerapan Transformer dengan mekanisme self-attention dalam data sekuensial.

Output dari tahap ini adalah perumusan pendekatan penelitian dan desain arsitektur model yang akan dikembangkan untuk mengidentifikasi pemancar siaran FM berbasis data IQ.

## 2. Persiapan Data

Pada tahap ini menggunakan data sinyal IQ yang sudah dikumpulkan oleh rekan peneliti di UPT Gorontalo menggunakan perangkat Software Defined Radio (RTL-SDR) yang dihubungkan ke komputer melalui aplikasi berbasis Python. Setiap pengukuran menghasilkan berkas data IQ yang menyimpan nilai real dan imajiner sinyal hasil demodulasi.

Data yang terkumpul kemudian melalui proses preprocessing, yang meliputi:

- Penyeimbangan dataset antar kelas untuk menghindari bias klasifikasi,
- Pelabelan berdasarkan identitas pemancar dan frekuensi siaran FM,
- Validasi visualisasi data IQ menggunakan diagram konstelasi guna memastikan karakteristik sinyal sesuai dengan tipe modulasi FM.

Output tahap ini adalah dataset siap latih yang terstruktur dan telah terverifikasi kualitasnya.

### 3. Pengembangan Model

Pada tahap ini dilakukan perancangan dan pembangunan tiga model *deep learning* dengan pendekatan arsitektur berbeda, yaitu:

- CNN 5-Layers: digunakan sebagai model dasar dengan lima lapisan konvolusi yang bertujuan mengekstraksi fitur lokal dari data IQ secara efisien,
- CNN-BiLSTM: merupakan model hybrid yang menggabungkan kemampuan CNN dalam mengekstraksi fitur spasial dengan BiLSTM yang menangkap dependensi temporal dua arah dari urutan sinyal IQ,
- CNN-Transformer: memanfaatkan lapisan self-attention untuk memahami hubungan global antar fitur sinyal yang tidak bisa ditangkap oleh CNN atau BiLSTM secara penuh.

Setiap model dibangun menggunakan TensorFlow/Keras dengan penyesuaian jumlah filter, kernel size, dan unit sesuai hasil eksperimen awal.

Output tahap ini adalah tiga arsitektur model siap latih yang akan dibandingkan performanya.

#### 4. Pelatihan Model

Model yang telah dirancang kemudian dilatih menggunakan dataset hasil preprocessing.

Proses pelatihan dilakukan dengan konfigurasi hyperparameter tertentu, meliputi jumlah epoch, ukuran batch size, learning rate, fungsi aktivasi, serta algoritma optimizer (Adam). Selama proses pelatihan, metrik akurasi dan loss diamati pada dua subset data: training set dan validation set.

Pemantauan ini dilakukan untuk:

- Menilai kestabilan proses pembelajaran,
- Mendeteksi tanda-tanda overfitting,
- Menentukan epoch optimal di mana model mencapai keseimbangan antara akurasi tinggi dan generalisasi baik

Teknik Batch Normalization dan Dropout diterapkan untuk meningkatkan stabilitas dan mencegah overfitting.

Output tahap ini adalah model terlatih yang performanya diverifikasi melalui hasil akurasi pelatihan dan validasi.

#### 5. Evaluasi

Pada tahap evaluasi, model diuji menggunakan data uji (testing set) yang tidak pernah digunakan selama pelatihan. Evaluasi dilakukan menggunakan beberapa metrik performa, yaitu:

- Akurasi untuk menilai ketepatan klasifikasi keseluruhan,

- Presisi, Recall, dan F1-score untuk mengukur kinerja per kelas, dan
- Confusion Matrix untuk memvisualisasikan distribusi klasifikasi benar dan salah antar kelas pemancar.

Selain itu, diukur pula waktu inferensi (klasifikasi) untuk menilai efisiensi komputasi model, khususnya dalam konteks penerapan real-time monitoring.

Output tahap ini adalah perbandingan kuantitatif performa ketiga model yang menjadi dasar dalam pemilihan model terbaik.

## **6. Tahap Akhir**

Tahap akhir merupakan sintesis hasil keseluruhan penelitian. Model terbaik ditentukan berdasarkan kombinasi antara akurasi tertinggi dan waktu inferensi tercepat, dengan mempertimbangkan juga kompleksitas arsitektur dan jumlah parameternya. Hasil analisis ini kemudian digunakan untuk menyusun kesimpulan mengenai efektivitas metode CNN dan variasinya dalam mengidentifikasi pemancar radio FM.

Selain itu, dilakukan perumusan rekomendasi teknis untuk pengembangan sistem monitoring spektrum berbasis AI yang dapat diimplementasikan secara real-time di lingkungan kerja Unit Pelaksana Teknis (UPT) Monitoring Frekuensi Radio.

Output akhir penelitian berupa model CNN optimal yang dapat dijadikan dasar sistem identifikasi pemancar otomatis dan efisien.

## **BAB 4**

### **HASIL PENELITIAN DAN PEMBAHASAN**

#### **4.1 Persiapan Data**

Hasil dari pengumpulan data menghasilkan sekitar 480.000 sampel IQ, disimpan dalam format .npy. Setiap file berisi sinyal IQ berdimensi (4096, 2), dengan dua kolom yang mewakili komponen In-phase (I) dan Quadrature (Q). Penamaan file dirancang dengan format yang mencerminkan metadata: ID pemancar, ID perangkat, jenis antena, frekuensi siaran, nama stasiun, dan timestamp perekaman. Informasi ini kemudian digunakan untuk memberikan label dan nama kelas pada masing-masing sampel.

##### **4.1.1. Preprocessing Data**

Data yang digunakan dalam penelitian ini terdiri atas data utama berupa sinyal IQ hasil perekaman spektrum radio FM, serta data pendukung (metadata) yang berfungsi untuk memastikan konsistensi, kesetaraan, dan keseimbangan dataset antar kelas. Metadata tersebut mencakup informasi penting seperti ID pemancar, nama stasiun radio, frekuensi siaran, waktu pengambilan data, serta ID perangkat dan antena yang digunakan selama proses perekaman.

Pada tahap pra-pemrosesan, dilakukan langkah krusial berupa penyeimbangan jumlah sampel per kelas (class balancing) untuk menghindari bias klasifikasi yang mungkin timbul apabila model deep learning dilatih menggunakan data yang tidak seimbang antar kelas. Ketidakseimbangan semacam ini berpotensi menyebabkan model lebih cenderung mempelajari pola dari kelas mayoritas dan

mengabaikan kelas minoritas, yang pada akhirnya dapat menurunkan kemampuan generalisasi model.

Untuk memastikan representasi yang adil bagi setiap pemancar, dilakukan penyeragaman jumlah data per kelas dengan menetapkan 5000 sampel untuk masing-masing pemancar. Dengan jumlah total 16 pemancar radio FM yang digunakan sebagai objek klasifikasi, diperoleh 80.000 sampel IQ yang siap digunakan dalam proses pelatihan model. Dalam kondisi tertentu di mana hasil perekaman menghasilkan jumlah sampel berlebih, diterapkan proses down-sampling secara acak untuk menyesuaikan jumlah data agar tetap seimbang di seluruh kelas.

Tabel 4.1. Proses Penyeimbangan Data

ID Pemancar	ID Perangkat	ID Antena	Frek.	Nama Radio	Tanggal	Jumlah Data
0	0, 1, 2, 3	0, 1	88.8	Relay RR1 Pro 3	2020-12-05 05, 2020-12-05 06, 2020-12-05 07, 2020-12-05 08	5000
1	0, 1, 2, 3	0, 1	89.2	Memora	2020-12-05 05, 2020-12-05 06, 2020-12-05 07, 2020-12-05 08	5000
2	0, 1, 2, 3	0, 1	90.0	Kharisma	2020-12-05 05, 2020-12-05 06, 2020-12-05 07, 2020-12-05 08	5000
3	0, 1, 2, 3	0, 1	90.8	Nada	2020-12-05 05, 2020-12-05 06, 2020-12-05 07, 2020-12-05 08	5000

4	0, 1, 2, 3	0, 1	92.4	RRI Pro 2	2020-12-05 05, 2020-12-05 06, 2020-12-05 07, 2020-12-05 08	5000
5	0, 1, 2, 3	0, 1	94.3	Be FM	2020-12-05 05, 2020-12-05 06, 2020-12-05 07, 2020-12-05 08	5000
6	0, 1, 2, 3	0, 1	95.1	RRI Pro 4	2020-12-05 05, 2020-12-05 06, 2020-12-05 07, 2020-12-05 08	5000
7	0, 1, 2, 3	0, 1	96.7	RRI Pro 3	2020-12-05 05, 2020-12-05 06, 2020-12-05 07, 2020-12-05 08	5000
8	0, 1, 2, 3	0, 1	97.9	Insania	2020-12-05 05, 2020-12-05 06, 2020-12-05 07, 2020-12-05 08	5000
9	0, 1, 2, 3	0, 1	99.1	SK FM	2020-12-05 05, 2020-12-05 06, 2020-12-05 07, 2020-12-05 08	5000
10	0, 1, 2, 3	0, 1	99.9	Radio Hufondhalo	2020-12-05 05, 2020-12-05 06, 2020-12-05 07, 2020-12-05 08	5000
11	0, 1, 2, 3	0, 1	101.0	Celebes	2020-12-05 05, 2020-12-05 06, 2020-12-05 07, 2020-12-05 08	5000
12	0, 1, 2, 3	0, 1	101.8	RRI Pro 1	2020-12-05 05, 2020-12-05 06, 2020-12-05 07, 2020-12-05 08	5000

13	0, 1, 2, 3	0, 1	102.6	RAL	2020-12-05 05, 2020-12-05 06, 2020-12-05 07, 2020-12-05 08	5000
14	0, 1, 2, 3	0, 1	104.2	Poliyama	2020-12-05 05, 2020-12-05 06, 2020-12-05 07, 2020-12-05 08	5000
15	0, 1, 2, 3	0, 1	87.5	Noise Floor	2020-12-05 05, 2020-12-05 06, 2020-12-05 07, 2020-12-05 08	5000

Sementara itu, metadata tidak digunakan secara langsung dalam proses pelatihan model. Perannya terbatas sebagai pengendali mutu data (data quality control) yang memastikan seluruh kelas direkam dalam kondisi dan konfigurasi perangkat yang seragam. Pendekatan ini bertujuan agar performa model sepenuhnya merefleksikan kemampuan CNN dalam mengenali karakteristik sinyal IQ, bukan dipengaruhi oleh faktor eksternal seperti jenis antenna, waktu perekaman, atau perangkat pengukuran yang digunakan.

#### 4.1.2. Pelabelan Data

Tahap pelabelan data dilakukan untuk memastikan setiap data sinyal IQ memiliki pasangan label yang merepresentasikan identitas pemancar radio FM secara tepat. Proses ini penting agar model pembelajaran mendalam (deep learning) dapat mengenali pola sinyal yang berbeda berdasarkan karakteristik pemancar yang diamati.

Setiap hasil pengukuran menghasilkan berkas berformat .npy yang berisi deret waktu dua komponen sinyal, yaitu:

- Komponen In-phase (I): merepresentasikan bagian real dari sinyal kompleks.
- Komponen Quadrature (Q): merepresentasikan bagian imajiner dari sinyal kompleks.

Kedua komponen ini disusun secara paralel sehingga membentuk data berdimensi dua (I dan Q) untuk setiap sampel. Setiap sampel data IQ dalam array ini merepresentasikan satu segmen sinyal yang direkam dari satu pemancar radio FM tertentu.

Dalam konteks penelitian ini, fitur (X) dan label (y) didefinisikan secara jelas sebagai berikut:

- Fitur (X): berisi data sinyal IQ yang mencakup nilai amplitudo dari komponen In-phase (I) dan Quadrature (Q) dalam bentuk array tiga dimensi dengan bentuk (80000, 4096, 2). Dimensi pertama 80.000 menunjukkan jumlah total sampel, dimensi kedua 4096 menunjukkan panjang setiap segmen sinyal dan dimensi ketiga 2 menunjukkan dua komponen I dan Q. Fitur ini menggambarkan karakteristik spektrum, pola modulasi, dan dinamika sinyal yang berbeda antar pemancar. Fitur X digunakan sebagai masukan (input) ke dalam model CNN untuk proses klasifikasi.

Karena ukuran dataset yang besar (mencapai 80.000 sampel dengan panjang sinyal 4096 titik dan dua kanal) membuat data tidak ditampilkan secara keseluruhan untuk menghindari overload tampilan atau crash pada IDE maka contoh data X, diambil 5 data pertama sebagai berikut:

```
[[[-0.90588235  0.23137255]
  [-0.63921569  0.67843137]
  [-0.2         0.89803922]
```

```

...
[ 1.          -0.09019608]
[-0.25490196 -0.99215686]
[-0.96862745  0.35686275]

[[-0.12156863 -0.78039216]
 [ 0.34117647 -0.71764706]
 [ 0.68627451 -0.40392157]

...
[ 0.03529412 -0.81176471]
[-0.18431373 -0.78823529]
[-0.34117647 -0.71764706]

[[-0.19215686  0.01176471]
 [-0.12941176 -0.12156863]
 [ 0.05882353 -0.18431373]

...
[-0.01960784  0.20784314]
 [ 0.1372549  0.15294118]
 [ 0.20784314  0.01176471]

[[-0.05882353 -0.30988392]
 [-0.14509804 -0.21568627]
 [-0.27058824 -0.14509804]

...
[-0.24705882  0.16078431]
 [ 0.01960784  0.28627451]
 [ 0.24705882  0.18431373]

[[ 1.          -0.82745098]
 [ 1.          -1.          ]
 [ 0.76470588 -1.          ]

...
[ 1.          0.50588235]
[ 1.          0.41960784]
[ 1.          0.29411765]]]

```

- **Label (y):** merupakan identitas pemancar radio FM yang bersesuaian dengan sinyal IQ pada fitur X. Label ini diberikan dalam bentuk angka kelas (integer) yang mengacu pada identitas stasiun siaran berdasarkan frekuensi siaran dan nama pemancar yang terdaftar. Label didapatkan dari setiap nama file, dimana penamaan file .npy memiliki format :

[ID Pemancar]\_[ID Perangkat]\_[ID Antena]\_[Frek.]\_[Nama Radio]\_[Tanggal]

Setiap komponen informasi dipisahkan oleh tanda garis bawah (\_). Pada format ini, segmen pertama dari nama file (yaitu nilai sebelum garis bawah pertama) merepresentasikan ID Pemancar, yang digunakan sebagai label

kelas dalam proses klasifikasi. Dengan demikian, penentuan label tidak membutuhkan berkas anotasi terpisah, karena label sudah tertanam langsung pada nama file.

Proses ekstraksi label dilakukan dengan cara memisahkan nama berkas berdasarkan karakter “\_”, kemudian mengambil segmen pertama dan mengonversinya menjadi bilangan bulat. Mekanisme ini dapat dilihat pada kode berikut:

```
label = int(filename.split("_")[0])
```

Contoh data  $y$ , diambil 5 data pertama:

```
[13 13 5 5 7]
```

Dataset yang digunakan dalam penelitian ini terdiri atas 16 kelas, yang masing-masing mewakili identitas pemancar radio FM berbeda. Pemberian label ( $y$ ) dilakukan berdasarkan metadata hasil pengukuran lapangan, yang mencatat frekuensi siaran dan identitas pemancar saat proses akuisisi data dilakukan. Setiap berkas data IQ dikaitkan langsung dengan metadata tersebut sehingga dapat diberi label sesuai pemancar asalnya. Setiap kelas memiliki jumlah data yang seimbang, yaitu sebanyak 5.000 sampel per kelas.

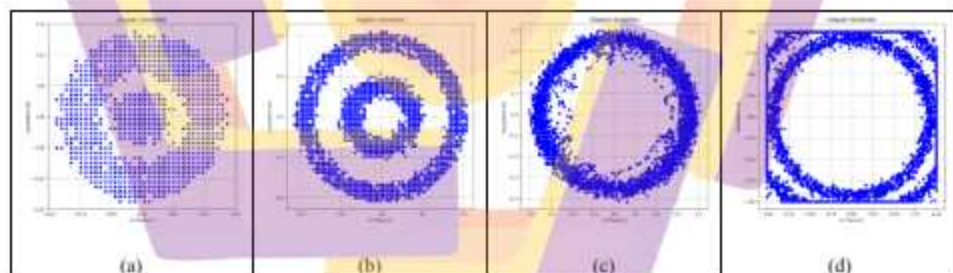
#### 4.1.3. Validasi Data

Sebagai bagian dari upaya untuk memastikan kualitas dan validitas setiap sampel data IQ yang digunakan dalam pelatihan model, dilakukan proses pemeriksaan visual terhadap konstelasi sinyal IQ. Langkah ini bertujuan untuk memastikan bahwa setiap sampel benar-benar memuat informasi sinyal yang valid dan merepresentasikan karakteristik pemancar yang dapat diekstraksi oleh model.

Pemeriksaan dilakukan dengan mengambil sejumlah sampel secara acak dari setiap kelas, kemudian memvisualisasikan komponen I dan Q dalam bentuk diagram konstelasi, yaitu plot sebar dua dimensi yang merepresentasikan sekuens sinyal dalam domain kompleks. Konstelasi yang baik umumnya menunjukkan pola sebaran titik yang konsisten sesuai karakteristik modulasi sinyal FM, meskipun dalam domain IQ FM, titik-titik biasanya membentuk sebaran spiral atau melingkar akibat bentuk frekuensi yang kontinu.

Visualisasi ini membantu mendeteksi adanya:

- Sampel kosong atau noise murni (konstelasi tersebar acak tanpa pola),
- Artefak rekaman akibat kegagalan SDR saat capture (titik terkonsentrasi di satu sumbu),
- Data duplikat atau stagnan.

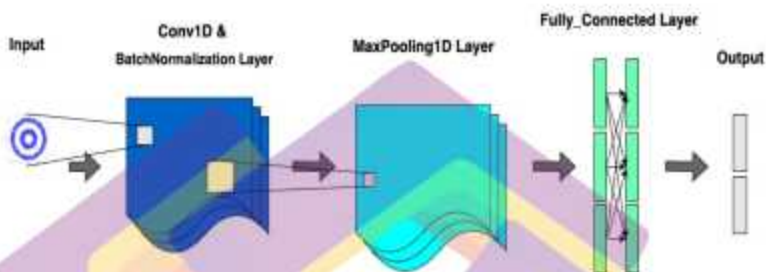


Gambar 4.1. Visualisasi karakteristik unik data IQ dari sampling empat (4) pemancar FM : (a) RRI Pro 3, (b) Memora, (c) Kharisma dan (d) Nada menggunakan diagram konstelasi

Hasil visualisasi pada gambar 4.1 dari pemeriksaan ini menunjukkan bahwa seluruh sampel yang digunakan dalam dataset memiliki representasi konstelasi IQ yang valid dan menunjukkan keberadaan sinyal aktif pada masing-masing frekuensi. Dengan demikian, dapat dipastikan bahwa setiap data yang dimasukkan

ke dalam model pelatihan memiliki fitur sinyal yang layak untuk diproses oleh jaringan saraf konvolusional.

## 4.2 Pengembangan Model



Gambar 4.2. Arsitektur Lapisan CNN untuk Klasifikasi Data IQ

Metode yang digunakan dalam penelitian ini untuk menganalisis data IQ dari pemancar FM adalah pengembangan model CNN, dipilih karena kemampuannya dalam menangkap pola spasial dan temporal dari data. Model ini dirancang untuk mengklasifikasikan data IQ dari 16 pemancar FM dengan akurasi tinggi berdasarkan karakteristik unik dari setiap sinyal pemancar.

Dari gambar 4.2 menunjukkan pemrosesan data IQ pada model CNN yang digunakan dalam penelitian ini. Berikut penjelasan lapisan-lapisan yang digunakan dalam pengembangan model:

- Input: Berbentuk matriks dengan 2 dimensi ( $16 \times 256, 2$ ), di mana 16 adalah jumlah blok data, 256 adalah panjang sampling setiap blok, dan 2 adalah jumlah channel yang merepresentasikan komponen IQ
- Konvolusi 1D (Conv1D): Menggunakan lapisan konvolusi 1 dimensi (Conv1D) untuk memproses data yang memiliki 1 dimensi dari data spasial, dan kernel\_size adalah 3 untuk menangkap pola yang lebih luas dan

menggunakan fungsi aktivasi ReLU (Rectified Linear Unit) untuk membuat nilai menjadi positif sehingga membantu dalam mempercepat konvergensi dan mengurangi masalah *vanishing gradient*. Batch Normalization: Menstabilkan pelatihan dengan menormalkan output dari lapisan konvolusi sebelumnya, sebab selama pelatihan distribusi input ke setiap lapisan dapat berubah, kemudian dengan distribusi data tetap stabil model dapat belajar lebih cepat dan memungkinkan untuk menggunakan learning rate yang lebih tinggi. Pada setiap batch fungsi akan menghitung rata-rata dan deviasi standar output dari lapisan sebelumnya dan menormalkan output tersebut dengan menggunakan nilai rata-rata dan deviasi standar serta fungsi ini menerapkan 2 parameter yang dapat dilatih yaitu skala ( $\gamma$ ) dan geser ( $\beta$ ) untuk mempelajari kembali distribusi yang optimal jika diperlukan.

- Max Pooling 1D: Menggunakan ukuran pool adalah 2 dimana lapisan ini akan mengambil nilai maksimum dari setiap dua elemen berturut-turut dari inputnya, hal ini mengurangi dimensi data dengan cara mengambil nilai maksimum dari setiap segmen yang ditentukan dan kompleksitas model. Manfaatnya mencegah overfitting dan membuat model focus pada informasi penting dan mengabaikan noise serta memberikan invariant terhadap perubahan kecil dalam input sehingga jika ada sedikit pergeseran data hasilnya akan tetap sama yang membantu model lebih robust.
- Fully Connected: keluaran dari pooling diluruskan (Flatten) menjadi vektor 1D lalu diproses oleh beberapa lapisan Dense (fully connected) untuk melakukan proses klasifikasi berdasarkan fitur yang diekstraksi. Lapisan-lapisan ini

menangkap hubungan non-linear antara fitur dan menghasilkan representasi yang siap diklasifikasikan. Pada lapisan ini ditambahkan Dropout sehingga mencegah overfitting dengan menonaktifkan secara acak 50% neuron dalam lapisan selama pelatihan. Dengan menonaktifkan neuron, model tidak dapat mengandalkan neuron tertentu dan harus belajar untuk mendistribusikan informasi di antara neuron yang tersisa dan membantu model untuk lebih baik dalam mengenali data yang belum pernah dilihat sebelumnya.

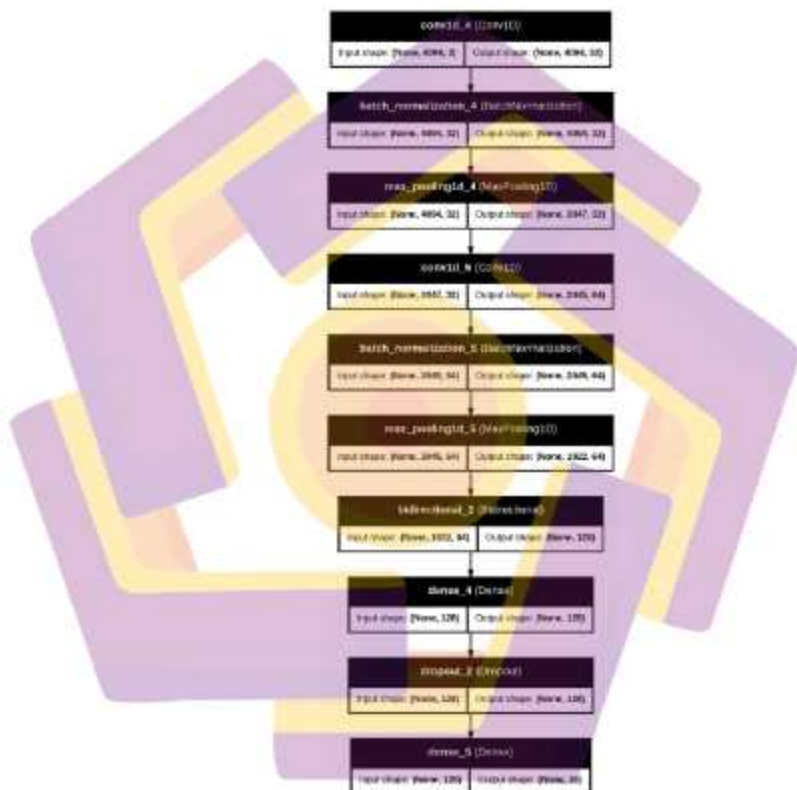
- Output: Menghasilkan probabilitas untuk setiap kelas yang mungkin dari masing-masing kelas pemancar FM (16 kelas) menggunakan fungsi Softmax, ini berarti bahwa output dari setiap neuron akan berada dalam rentang 0 atau 1 dan jumlah dan jumlah total output akan sama dengan 1.

#### 4.2.1. CNN-BiLSTM

Model CNN-BiLSTM yang digunakan dalam penelitian ini dirancang untuk menggabungkan keunggulan ekstraksi fitur lokal melalui jaringan konvolusional dan pemahaman ketergantungan temporal dari data sinyal IQ melalui jaringan *Long Short-Term Memory* (LSTM) dua arah. Pendekatan hibrida ini dipilih untuk meningkatkan kemampuan klasifikasi sinyal IQ dari berbagai pemancar radio FM dengan memanfaatkan karakteristik lokal dan sekuensial yang terkandung dalam data.

Gambar 4.3 menggambarkan alur arsitektur model secara visual, dari input hingga output. Arsitektur model diawali dengan dua buah lapisan 1D Convolutional (Conv1D) berturut-turut, masing-masing dengan 32 dan 64 filter berukuran kernel 3. Pemilihan parameter-parameter ini didasarkan pada pertimbangan efisiensi

komputasi dan validasi empiris dari berbagai studi sebelumnya yang relevan dengan klasifikasi sinyal dan pengolahan data sekuensial. Lapisan ini berfungsi untuk mengekstraksi fitur lokal dari sinyal IQ, yaitu pola-pola khas yang muncul dalam bentuk nilai amplitudo I dan Q sepanjang waktu.



Gambar 4.3. Plot-model CNN-BiLSTM

Penggunaan 32 filter pada lapisan konvolusi pertama bertujuan untuk mengekstraksi fitur-fitur dasar yang bersifat lokal dari sinyal IQ. Sinyal IQ yang merupakan data kompleks dua dimensi (komponen I dan Q) mengandung informasi penting dalam bentuk perubahan amplitudo dan fasa terhadap waktu. Filter dalam

jumlah sedang seperti 32 ini terbukti cukup efektif dalam menangkap pola-pola dasar tersebut, terutama pada sinyal dengan karakteristik temporal yang halus.

Selanjutnya, 64 filter digunakan pada lapisan konvolusional kedua untuk menangkap representasi fitur yang lebih kompleks dan abstrak. Filter pada lapisan ini bertugas untuk memperkuat pemodelan terhadap struktur sinyal yang lebih tinggi, seperti pola frekuensi yang berulang atau anomali temporal tertentu. Meskipun jumlah parameter bertambah, penggunaan 64 filter dinilai tidak membebani sistem secara signifikan, namun memberikan peningkatan dalam kapasitas generalisasi model.

Adapun pemilihan ukuran kernel sebesar 3 merupakan praktik umum dalam desain arsitektur CNN modern, seperti VGGNet. Kernel kecil berukuran 3 memungkinkan penerapan konvolusi bertingkat yang mampu membentuk receptive field yang luas tanpa meningkatkan jumlah parameter secara drastis. Strategi ini tidak hanya efisien dalam hal komputasi, tetapi juga memberikan performa yang stabil dalam berbagai tugas klasifikasi, termasuk pengolahan citra dan sinyal.

Secara keseluruhan, kombinasi filter 32 dan 64 dengan ukuran kernel 3 dalam penelitian ini dipilih berdasarkan tiga pertimbangan utama, yaitu:

- Efisiensi komputasi dan jumlah parameter yang relatif moderat,
- Kapasitas dalam menangkap fitur lokal dan pola kompleks secara bertahap,
- Validasi empiris dari berbagai penelitian sebelumnya dalam domain klasifikasi sinyal dan deep learning.

Setelah masing-masing lapisan konvolusi, ditambahkan Batch Normalization untuk menstabilkan distribusi aktivasi selama pelatihan, serta

MaxPooling1D untuk mengurangi dimensi spasial data dan menangkap fitur yang paling menonjol. Kombinasi antara Batch Normalization dan MaxPooling1D pada setiap tahap ekstraksi fitur konvolusional memberikan keuntungan baik dari sisi stabilitas pelatihan maupun efisiensi model. Hal ini sangat penting dalam pemrosesan data sinyal IQ yang memiliki dinamika temporal tinggi dan variasi antar sumber pemancar yang kompleks.

Selanjutnya, output dari lapisan konvolusi diteruskan ke lapisan Bidirectional LSTM dengan 64 unit di setiap arah. Lapisan ini memungkinkan pemodelan urutan data dalam dua arah—maju dan mundur—yang penting dalam mendeteksi pola temporal yang mungkin muncul di kedua sisi waktu sinyal IQ. Pendekatan dua arah (bidirectional) terbukti bermanfaat dalam banyak kasus pemrosesan sinyal sekuensial karena memungkinkan model menangkap konteks informasi yang lebih luas.

Setelah lapisan BiLSTM, arsitektur dilanjutkan dengan fully connected layer (Dense) berukuran 128 unit dengan aktivasi ReLU yang bertugas mengintegrasikan fitur yang telah dipelajari sebelumnya ke dalam representasi yang lebih abstrak. Untuk mencegah overfitting, digunakan mekanisme Dropout sebesar 0.3 sebelum menuju ke lapisan output. Lapisan output menggunakan fungsi aktivasi softmax dengan 16 unit, yang mewakili jumlah total kelas pemancar FM dalam penelitian ini.

Model ini dirancang agar mampu mengenali perbedaan karakteristik sinyal IQ dari berbagai pemancar, baik dari sisi pola lokal maupun struktur temporalnya.

Pemilihan struktur CNN-BiLSTM ini juga mempertimbangkan efisiensi pelatihan dan performa generalisasi terhadap data baru.

Tabel 4.2. Jumlah Parameter CNN-BiLSTM

Komponen Arsitektur	Jumlah Parameter
Conv1D - 32 filter	224
Conv1D - 64 filter	6.208
Bidirectional LSTM (2×64)	66.048
Dense (128 neuron)	16.512
Dense (16 output kelas)	2.064
<b>Total Parameter</b>	<b>91.440</b>

Model CNN-BiLSTM yang dikembangkan memiliki total 91.440 parameter, yang terdiri dari kontribusi beberapa lapisan utama, yaitu:

#### 1. Lapisan Conv1D

- Lapisan pertama Conv1D memiliki 32 filter dengan ukuran kernel 3 dan menerima input 2 channel (I dan Q), sehingga jumlah parameternya dihitung sebagai:

$$(3 \times 2 + 1) \times 32 = 224 \text{ parameter}$$

- Lapisan Conv1D kedua memiliki 64 filter dan menerima input dari 32 channel sebelumnya, dengan parameter:

$$(3 \times 32 + 1) \times 64 = 6.208 \text{ parameter}$$

#### 2. Lapisan Bidirectional LSTM

- Setiap arah LSTM memiliki 64 unit, sehingga total unit adalah 128 (64 forward + 64 backward).
- Jumlah parameter untuk satu arah LSTM dihitung dengan input dimensi dari CNN sebelumnya adalah 64, maka:

$$4 \times (64 + 64 + 1) \times 64 = 33.024 \text{ parameter per arah}$$

- Sehingga total parameter untuk BiLSTM adalah:

$$2 \times 33.024 = 66.048 \text{ parameter}$$

### 3. Lapisan Dense (Fully Connected)

- Lapisan Dense pertama memiliki 128 neuron dan menerima input dari 128 unit (output BiLSTM dua arah), maka:

$$128 \times 128 + 128 = 16.512 \text{ parameter}$$

- Lapisan output memiliki 16 unit kelas dan menerima input dari 128 unit, maka:

$$128 \times 16 + 16 = 2.064 \text{ parameter}$$

Dengan komposisi seperti di atas, terlihat bahwa lapisan BiLSTM menyumbang proporsi terbesar dari parameter total model, yakni sebesar  $\approx 72\%$  dari total parameter. Hal ini dapat dimaklumi mengingat karakteristik LSTM yang memerlukan banyak parameter untuk mengelola informasi sekuensial secara efisien, terutama ketika bekerja dalam konfigurasi dua arah (*bidirectional*).

Pemahaman terhadap jumlah parameter ini penting untuk mempertimbangkan kebutuhan sumber daya komputasi, risiko *overfitting*, dan waktu pelatihan, terutama jika model akan diimplementasikan dalam sistem pengawasan spektrum frekuensi secara *real-time*.

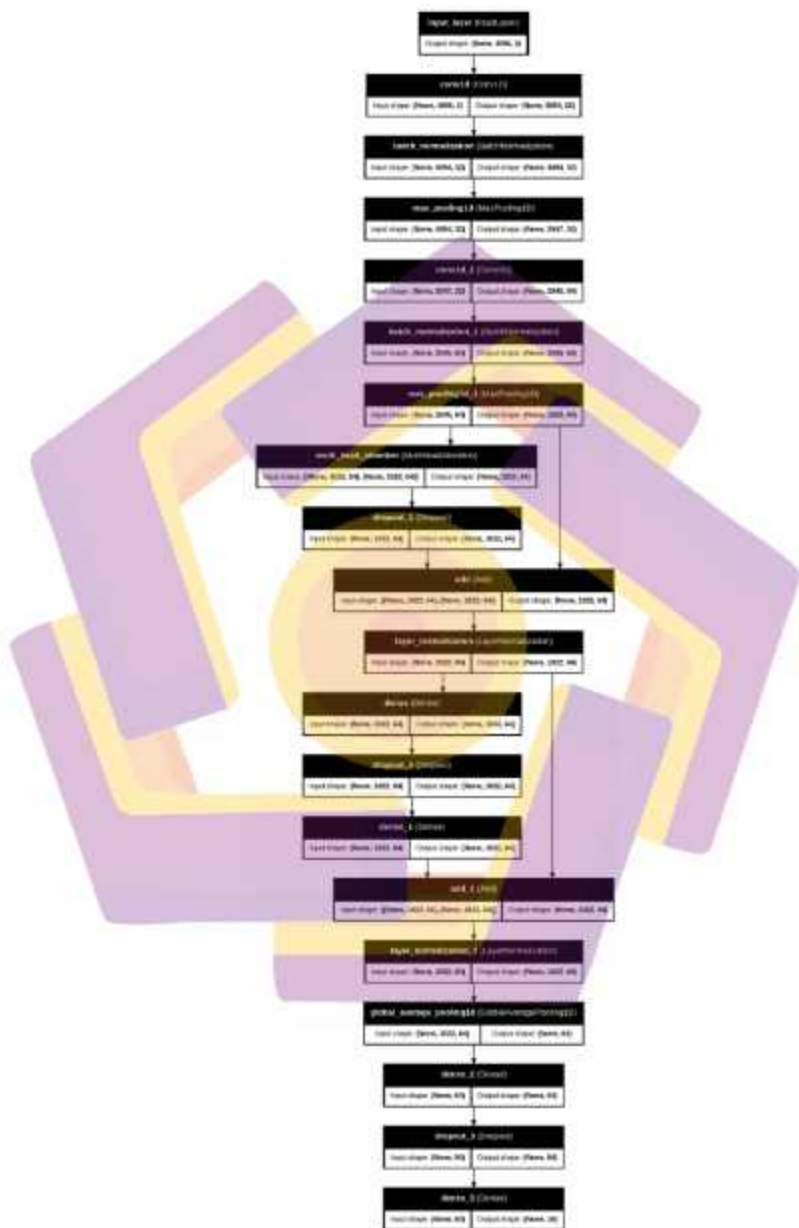
#### 4.2.2. CNN-Transformer

Sementara model CNN-Transformer (*hybrid*) memanfaatkan CNN di bagian awal untuk mengekstrak fitur spasial, yang kemudian diproses oleh blok *Transformer* melalui mekanisme *multi-head attention*. Untuk menjembatani

keterbatasan pendekatan konvolusi yang hanya mampu menangkap hubungan lokal dalam sinyal IQ, serta keterbatasan metode rekuren dalam efisiensi komputasi dan penangkapan relasi jangka panjang, maka pendekatan hibrida dikembangkan. Dalam hal ini, arsitektur CNN digunakan di tahap awal untuk mereduksi kompleksitas data dan mengekstraksi pola spasial atau fitur lokal dari segmen sinyal IQ. Hasil ekstraksi tersebut kemudian menjadi representasi yang lebih padat dan informatif bagi blok Transformer, yang memiliki kemampuan untuk memodelkan ketergantungan global secara paralel melalui mekanisme *multi-head attention*.

Dengan memadukan kedua pendekatan ini, model tidak hanya efisien dalam mendeteksi pola lokal seperti amplitudo atau fase dari sinyal, tetapi juga mampu memahami hubungan kontekstual antar bagian sinyal yang berjauhan secara temporal. Pengembangan model CNN-Transformer dibangun berdasarkan pertimbangan bahwa data IQ merupakan data deret waktu dengan struktur lokal dan pola berulang, tetapi juga mengandung relasi spasial-temporal yang lebih panjang yang sulit ditangkap hanya dengan pendekatan konvolusi atau rekuren secara tunggal.

Dari gambar 4.4 dapat dijelaskan model dimulai dengan dua buah lapisan Conv1D berturut-turut, masing-masing memiliki 32 dan 64 filter dengan kernel size 3. Pemilihan filter 32 dan 64 digunakan untuk meningkatkan kapasitas model dalam mengekstraksi fitur bertingkat dari sinyal IQ. Filter awal berfungsi untuk mendeteksi fitur dasar seperti perubahan amplitudo dan fase yang khas untuk setiap pemancar, sedangkan filter pada lapisan kedua digunakan untuk menangkap pola frekuensi yang lebih kompleks dan abstrak.



Gambar 4.4. Plot-model CNN-Transformer

Penggunaan ukuran kernel 3 mengacu pada prinsip desain CNN yang telah terbukti efektif dalam model VGG, di mana kernel kecil bertumpuk mampu menciptakan receptive field yang cukup luas secara efisien tanpa meningkatkan jumlah parameter secara signifikan. Lapisan-lapisan ini juga dilengkapi dengan BatchNormalization untuk menstabilkan proses pelatihan serta MaxPooling1D untuk mengurangi dimensi data secara bertahap sambil mempertahankan fitur penting.

Setelah fitur lokal terekstraksi, output dari lapisan CNN kemudian diteruskan ke blok *Transformer*. Blok ini terdiri dari lapisan *Multi-Head Attention* (MHA) dengan 2 kepala perhatian (*heads*), dimensi kunci 32, serta jaringan feed-forward dengan dimensi 64. MHA berfungsi untuk memetakan keterkaitan antar posisi dalam data sekuensial, sehingga mampu memahami hubungan antar bagian sinyal yang berjauhan, sesuatu yang tidak bisa dicapai oleh CNN saja. Mekanisme ini dinormalisasi menggunakan *Layer Normalization* dan ditambahkan dengan *residual connection* untuk mempercepat konvergensi dan mencegah degradasi gradien. Blok *Transformer* ini kemudian diikuti oleh *GlobalAveragePooling1D* untuk mereduksi dimensi dan mempertahankan konteks global, serta dua lapisan *Dense* (64 neuron dan 16 output class) dengan aktivasi *ReLU* dan *softmax* secara berurutan.

Pemilihan model hybrid CNN-Transformer dalam penelitian ini tidak hanya didasari oleh efisiensi arsitektural, tetapi juga didukung oleh berbagai studi empiris. Model *Transformer* yang dikombinasikan dengan lapisan konvolusi mampu memberikan hasil klasifikasi yang unggul dalam tugas-tugas *time series*

classification. Selain itu, sinyal spektrum RF (radio frequency) membuktikan bahwa Transformer dapat secara efektif mengatasi noise dan mempertahankan konteks panjang dalam sinyal frekuensi rendah hingga tinggi. Kelebihan utama Transformer dibanding model berbasis LSTM atau BiLSTM terletak pada kemampuannya memproses sekuens data secara paralel, yang pada dasarnya lebih efisien pada skala besar, serta kemampuannya untuk mengakses informasi global tanpa memerlukan propagasi sekuensial yang panjang.

Namun demikian, walaupun jumlah parameter CNN-Transformer (37.232 parameter) lebih sedikit dibandingkan model CNN-BiLSTM yang dikembangkan dalam penelitian ini (91.440 parameter), waktu komputasi model CNN-Transformer cenderung lebih tinggi. Hal ini disebabkan oleh kompleksitas operasi self-attention yang memiliki biaya komputasi kuadratik terhadap panjang sekuens input. Oleh karena itu, pemilihan model ini lebih ditujukan pada kapasitas representasi dan akurasi dalam memahami struktur global sinyal IQ, bukan sekadar efisiensi runtime. Dengan demikian, pengembangan CNN-Transformer dalam penelitian ini diharapkan dapat mengakomodasi keterbatasan pendekatan konvolusi dan rekuren dalam menangkap kompleksitas sinyal IQ multi-pemancar.

Tabel 4.3: Jumlah Parameter CNN-Transformer

<b>Type</b>	<b>Output Shape</b>	<b>Parameter</b>
InputLayer	(None, 4096, 2)	0
Conv1D	(None, 4094, 32)	224
BatchNormalization	(None, 4094, 32)	128
MaxPooling1D	(None, 2047, 32)	0
Conv1D	(None, 2045, 64)	6.208
BatchNormalization	(None, 2045, 64)	256
MaxPooling1D	(None, 1022, 64)	0
MultiHeadAttention	(None, 1022, 64)	16.640

Dropout	(None, 1022, 64)	0
Add	(None, 1022, 64)	0
LayerNormalization	(None, 1022, 64)	128
Dense	(None, 1022, 64)	4.160
Dropout	(None, 1022, 64)	0
Dense	(None, 1022, 64)	4.160
Add	(None, 1022, 64)	0
LayerNormalization	(None, 1022, 64)	128
GlobalAveragePooling1D	(None, 64)	0
Dense	(None, 64)	4.160
Dropout	(None, 64)	0
Dense	(None, 16)	1.040
<b>Total Parameter</b>		<b>37.232</b>

Arsitektur hybrid CNN-Transformer yang diusulkan dalam penelitian ini memiliki total 37.232 parameter, dengan 37.040 parameter trainable dan 192 parameter non-trainable. Model ini dirancang untuk memanfaatkan kekuatan ekstraksi fitur lokal dari CNN dan kemampuan Transformer dalam menangkap ketergantungan global dari data sekuensial, khususnya pada sinyal IQ.

Struktur lapisan model ditunjukkan pada Tabel 4.3 berikut dan dijelaskan sebagai berikut:

#### 1. Input Layer

Input berbentuk matriks berdimensi (4096, 2) yang merepresentasikan 4096 titik sampel IQ (in-phase dan quadrature) dari sinyal yang ditangkap.

#### 2. Lapisan Konvolusional

Dua buah lapisan Conv1D digunakan secara berurutan dengan jumlah filter masing-masing 32 dan 64, serta ukuran kernel sebesar 3. Pemilihan ini didasarkan pada praktik umum dalam arsitektur deep learning modern (misalnya VGGNet) di mana filter kecil seperti kernel 3x1 cukup efektif untuk menangkap fitur lokal

sambil menjaga jumlah parameter tetap rendah. Jumlah parameter pada conv1d pertama adalah 224 dan pada conv1d\_1 sebesar 6.208. Ini dihitung dari:

$$\text{Parameter} = (\text{kernel size} \times \text{input channel} \times \text{filter}) + \text{bias}$$

Untuk conv1d pertama:  $(3 \times 2 \times 32) + 32 = 224$  parameter.

### 3. Batch Normalization dan MaxPooling1D

Setiap lapisan konvolusional diikuti oleh BatchNormalization yang bertujuan untuk menstabilkan proses pelatihan dan mempercepat konvergensi dengan menjaga distribusi aktivasi tetap konsisten melalui 64 parameter gamma dan 64 beta ( $2 \times 32$ ), menghasilkan total 128 parameter. (Ioffe & Szegedy, 2015). Kemudian, MaxPooling1D dengan pool size 2 digunakan untuk mereduksi dimensi spasial secara bertahap, mempertahankan informasi penting sambil mengurangi beban komputasi.

### 4. Transformer Block

Blok Transformer yang diterapkan melibatkan Multi-Head Attention dengan 2 head dan dimensi kunci 32. Layer ini menghasilkan 16.640 parameter, nilai ini diperoleh dari total 4 weight matrix: Q, K, V, dan output (masing-masing  $64 \times 64 = 4.096$ ), ditambah bias, memungkinkan model untuk menangkap ketergantungan jangka panjang (long-range dependencies) dalam sinyal IQ. Selanjutnya dilakukan operasi Dropout dan penjumlahan residual melalui Add, lalu distabilkan dengan LayerNormalization dengan 128 parameter ( $2 \times 64$ ). Arsitektur ini diikuti oleh dua Dense layer sebagai bagian dari feed-forward network, masing-masing memiliki 4.160 parameter, serta LayerNormalization dan residual connection (Add) untuk menjaga stabilitas gradient flow.

#### 5. GlobalAveragePooling1D

Layer ini mereduksi dimensi dari bentuk (sequence\_length, features) menjadi (features) dengan mengambil rata-rata per fitur. Ini mengurangi kompleksitas sekaligus mempertahankan informasi agregat.

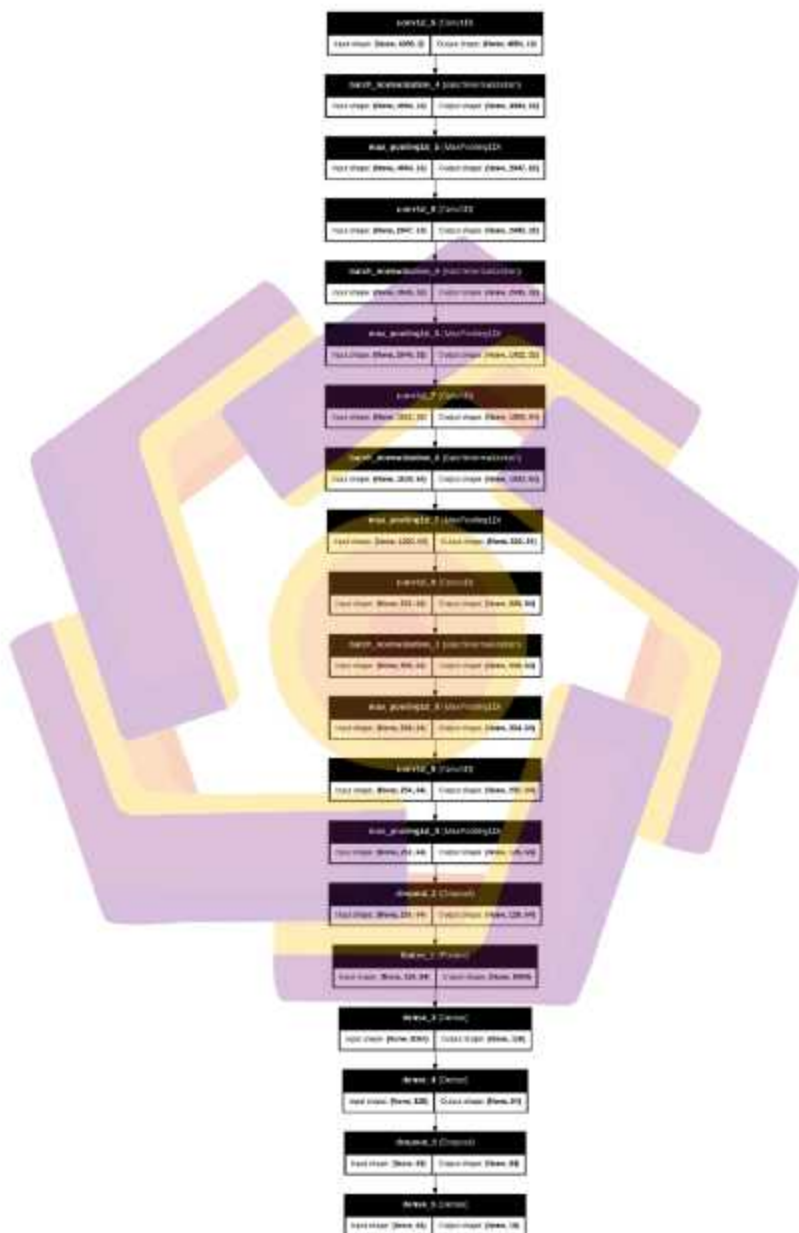
#### 6. Fully Connected Layer

Setelah pooling, terdapat Dense layer dengan 64 unit dan dropout sebesar 0.3 untuk mengurangi overfitting. Output terakhir adalah Dense layer dengan 16 neuron, sesuai dengan jumlah kelas target dalam klasifikasi, menggunakan aktivasi softmax.

#### 4.2.3. CNN 5-Layers

Model CNN 5-Layers mengekstraksi fitur spasial dari sinyal IQ melalui lima lapisan konvolusi yang dilengkapi normalisasi dan pooling. Dengan prinsip local convolution dan shared weights, CNN mampu mengenali pola spasial dalam data IQ yang memiliki struktur waktu dan keterkaitan lokal yang kuat. Arsitektur ini menghasilkan representasi fitur yang efektif namun tetap efisien secara komputasi karena tidak memerlukan pemrosesan urutan waktu secara eksplisit.

Dari gambar 4.5 model diawali dengan lapisan Conv1D pertama yang memiliki 16 filter dengan ukuran kernel 3, yang berfungsi untuk mengekstraksi fitur dasar dari sinyal. Lapisan ini diikuti oleh BatchNormalization untuk menstabilkan distribusi aktivasi selama pelatihan dan MaxPooling1D untuk mereduksi dimensi spasial. Pola ini diulang pada lapisan berikutnya, dengan peningkatan jumlah filter secara bertahap menjadi 32, 64, hingga tiga lapisan konvolusi terakhir yang konsisten menggunakan 64 filter.



Gambar 4.5. Plot-model CNN 5-Layers

Lapisan BatchNormalization disisipkan setelah setiap lapisan konvolusi. Selain itu, MaxPooling1D digunakan setiap lapisan konvolusi untuk mengekstrak fitur dominan dan mengurangi dimensi temporal, sehingga membantu mengurangi kompleksitas komputasi. Pada tahap akhir ekstraksi fitur, digunakan lapisan Dropout dengan tingkat 0.5 untuk mencegah.

Setelah fitur spasial terekstraksi, hasilnya diratakan dengan Flatten dan diproses oleh tiga lapisan Dense secara berturut-turut, dengan jumlah neuron 128, 64, dan 16 (kelas keluaran). Lapisan-lapisan ini bertanggung jawab atas klasifikasi akhir ke dalam salah satu dari 16 pemancar radio yang diamati dalam penelitian. Fungsi aktivasi ReLU digunakan pada seluruh lapisan Dense kecuali output, yang menggunakan softmax untuk menghasilkan distribusi probabilitas antar kelas. Seluruh model dikompilasi dengan optimizer Adam, yang dikenal memiliki performa pelatihan yang stabil dan adaptif terhadap gradien, serta loss function `sparse_categorical_crossentropy` yang sesuai untuk tugas klasifikasi multi-kelas dengan label berupa integer.

Struktur CNN 5-Layers ini dipilih karena kesederhanaannya namun mampu memberikan akurasi yang kompetitif. Beberapa studi telah menunjukkan bahwa CNN dengan lima lapisan konvolusi mampu mengungguli arsitektur yang lebih kompleks dalam klasifikasi pola visual, terutama pada data dengan volume sedang hingga besar. Selain itu, CNN 5-Layers juga telah berhasil digunakan dalam domain sinyal biomedis dan pengenalan suara, menunjukkan kemampuannya dalam menangani data sekuensial dengan struktur spasial yang jelas.

Tabel 4.4. Jumlah Parameter CNN 5-Layers

Layer	Output Shape	Parameter
Conv1D (16 filter)	(None, 4094, 16)	112
BatchNormalization (16 unit)	(None, 4094, 16)	64
MaxPooling1D	(None, 2047, 16)	0
Conv1D (32 filter)	(None, 2045, 32)	1.568
BatchNormalization (32 unit)	(None, 2045, 32)	128
MaxPooling1D	(None, 1022, 32)	0
Conv1D (64 filter)	(None, 1020, 64)	6.208
BatchNormalization (64 unit)	(None, 1020, 64)	256
MaxPooling1D	(None, 510, 64)	0
Conv1D (64 filter)	(None, 508, 64)	12.352
BatchNormalization (64 unit)	(None, 508, 64)	256
MaxPooling1D	(None, 254, 64)	0
Conv1D (64 filter)	(None, 252, 64)	12.352
MaxPooling1D	(None, 126, 64)	0
Dropout	(None, 126, 64)	0
Flatten	(None, 8064)	0
Dense (128)	(None, 128)	1.032.320
Dense (64)	(None, 64)	8.256
Dropout	(None, 64)	0
Dense (Output - 16 class)	(None, 16)	1.040
<b>Total Parameter</b>		<b>1.074.912</b>

Model CNN 5-Layers yang dikembangkan dalam penelitian ini terdiri dari 20 lapisan, yang disusun untuk secara bertahap mengekstraksi fitur spasial dari data sinyal IQ dan melakukan klasifikasi akhir ke dalam 16 kelas pemancar. Proses perhitungan jumlah parameter dilakukan berdasarkan formula standar dari arsitektur jaringan saraf konvolusional. Misalnya, pada lapisan pertama Conv1D dengan 16 filter dan kernel size 3, jumlah parameternya dihitung dengan formula  $(\text{kernel\_size} \times \text{input\_channels} + \text{bias}) \times \text{output\_channels}$ , yaitu  $(3 \times 2 + 1) \times 16 = 112$ . Proses ini berulang pada lapisan konvolusi berikutnya dengan peningkatan jumlah

filter, sehingga Conv1D kedua dengan 32 filter memiliki 1.568 parameter, dan seterusnya hingga mencapai 12.352 parameter pada lapisan keempat dan kelima.

Lapisan BatchNormalization menambahkan dua parameter untuk setiap output ( $\gamma$  dan  $\beta$ ), sehingga ketika digunakan setelah Conv1D dengan 64 filter, menghasilkan 128 parameter. MaxPooling tidak menambah parameter karena hanya melakukan pengambilan nilai maksimum. Setelah proses ekstraksi fitur selesai, data diratakan menggunakan Flatten untuk menghasilkan vektor berdimensi 8.064 (dari dimensi sebelumnya:  $126 \text{ waktu} \times 64 \text{ channel}$ ).

Vektor ini kemudian diproses oleh tiga lapisan fully connected (Dense), masing-masing dengan 128, 64, dan 16 unit output. Lapisan pertama Dense memiliki jumlah parameter terbesar yaitu 1.032.320, diperoleh dari  $8064 \times 128 + 128$ . Dua lapisan berikutnya memiliki 8.256 dan 1.040 parameter secara berturut-turut. Lapisan Dropout ditambahkan untuk meningkatkan generalisasi model, namun tidak menambah parameter.

Total parameter dari seluruh model ini adalah 1.074.912, dengan sebagian besar kontribusi berasal dari lapisan Dense. Ini menunjukkan bahwa kompleksitas utama model bukan terletak pada bagian ekstraksi fitur CNN, tetapi pada bagian klasifikasi akhir, yang menyimpan bobot terbesar. Meskipun demikian, struktur ini tetap dipertahankan karena kedalaman CNN dapat secara efektif mempelajari representasi spasial yang kompleks dari sinyal IQ sebelum dipetakan ke dalam ruang klasifikasi.

### 4.3 Pelatihan Model

Seluruh model deep learning dalam penelitian ini dilatih dengan menggunakan strategi pembagian data terstruktur guna menjaga objektivitas proses pelatihan serta memastikan akurasi evaluasi model terhadap data yang benar-benar belum pernah dilihat sebelumnya. Dari total 80.000 sampel data IQ, data dibagi menjadi dua bagian utama; 80% data pelatihan dan validasi, yaitu sebanyak 64.000 sampel dan 20% data pengujian, yaitu sebanyak 16.000 sampel

Selanjutnya, dari 64.000 data pelatihan tersebut, 20% (sebanyak 12.800 sampel) dialokasikan secara acak sebagai data validasi selama proses pelatihan, sehingga komposisi akhir menjadi: Training set: 64% (51.200 sampel), Validation set: 16% (12.800 sampel) serta Testing set: 20% (16.000 sampel).

Pemilihan data dilakukan secara acak (random shuffle) dan dengan distribusi label yang merata (stratified sampling) untuk memastikan bahwa semua kelas (pemancar FM) terwakili secara seimbang dalam setiap subset. Strategi ini dirancang untuk menjaga validitas internal proses pelatihan model, sekaligus meminimalkan risiko data leakage, karena data pengujian sama sekali tidak digunakan selama pelatihan.

Proses pelatihan model dilakukan menggunakan optimizer Adam (Adaptive Moment Estimation), yang dikenal memiliki performa stabil dan efisien dalam menangani pembaruan bobot pada tugas pembelajaran mendalam dengan data besar dan bervariasi. Adam menggabungkan keunggulan dari algoritma RMSProp dan momentum, dengan kemampuan untuk secara otomatis menyesuaikan learning rate dari setiap parameter berdasarkan estimasi momen pertama dan kedua dari gradien.

Untuk menjaga kestabilan proses pelatihan dan menghindari pembaruan bobot yang terlalu agresif, learning rate ditetapkan sebesar 0.0003, yang tergolong kecil namun stabil. Nilai ini dipilih agar model memiliki laju pembelajaran yang moderat, memungkinkan proses konvergensi yang lebih hati-hati, terutama ketika model cukup dalam atau ketika data memiliki variasi noise yang tinggi seperti dalam sinyal IQ.

Fungsi loss yang digunakan adalah SparseCategoricalCrossentropy, karena masalah klasifikasi yang dihadapi adalah klasifikasi multi-kelas dengan label dalam bentuk integer, bukan one-hot encoded vector. Fungsi loss ini efektif untuk menghitung probabilitas prediksi terhadap lebih dari dua kelas dan sangat umum digunakan pada skenario klasifikasi dengan banyak kategori yang saling eksklusif.

Untuk mencegah terjadinya overfitting, yaitu kondisi ketika model terlalu menghafal data pelatihan dan gagal menggeneralisasi pada data baru, digunakan dua teknik regulasi, yaitu EarlyStopping dan ReduceLROnPlateau. Earlystopping akan otomatis menghentikan pelatihan jika akurasi validasi tidak meningkat setelah sejumlah epoch berturut-turut. Hal ini mencegah pelatihan berlangsung terlalu lama ketika model sudah mencapai titik optimum pada data validasi. ReduceLROnPlateau akan aktif ketika akurasi validasi stagnan atau tidak menunjukkan peningkatan, learning rate akan diturunkan secara otomatis, memungkinkan model melakukan pembelajaran lebih hati-hati di fase akhir pelatihan.

#### 4.3.1. Pelatihan Model CNN-BiLSTM

Tabel 4.5. Ringkasan Hasil Pelatihan CNN-BiLSTM

Epoch	Akurasi Pelatihan (%)	Loss Pelatihan	Akurasi Validasi (%)	Loss Validasi
1	36.29	1.7809	58.44	1.1243
5	67.33	0.8634	73.79	0.6865
10	82.04	0.4904	86.04	0.3856
20	92.05	0.2283	92.55	0.2161
30	95.75	0.1255	95.1	0.1392
40	97.69	0.0691	94.48	0.1637
50	98.87	0.0331	98.29	0.0504
70	99.75	0.0088	98.77	0.0446
90	99.8	0.0066	98.91	0.042
91	99.86	0.0054	98.9	0.0421

Berdasarkan log epoch pelatihan model CNN-BiLSTM pada tabel 4.5, perkembangan kinerja model menunjukkan pola peningkatan yang konsisten pada fase awal hingga pertengahan epoch, diikuti oleh fase stabilisasi dan konvergensi pada epoch-epoch akhir. Pada epoch pertama, akurasi pelatihan tercatat sebesar 36,29% dengan validation accuracy sebesar 58,44%, yang menunjukkan bahwa meskipun model belum terlatih optimal, kemampuannya dalam generalisasi awal cukup baik. Hal ini didukung oleh nilai validation loss sebesar 1,1243 yang lebih rendah dari training loss (1,7809), indikasi bahwa model belum mengalami overfitting pada tahap awal. Memasuki epoch ke-5, akurasi pelatihan telah meningkat menjadi 67,33% dan validation accuracy melonjak ke 73,79%, disertai penurunan validation loss signifikan ke 0,6865. Tren ini berlanjut hingga sekitar epoch ke-20, di mana akurasi pelatihan mencapai 92,05% dan validation accuracy

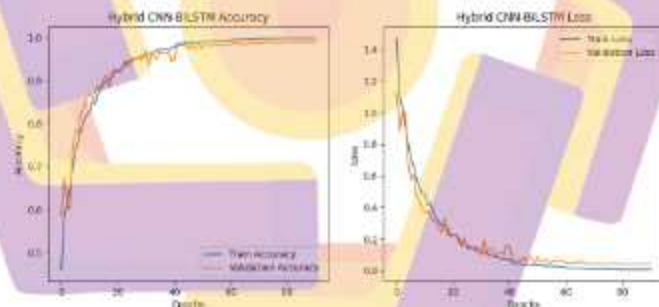
92,55%, menandakan bahwa model telah mempelajari representasi fitur yang relevan dari data IQ FM yang dilatihkan.

Pada fase pertengahan (sekitar epoch 21–40), model terus menunjukkan peningkatan akurasi secara gradual dengan penurunan loss yang konsisten, dan mencapai validation accuracy di atas 95% pada epoch ke-28. Menariknya, pada epoch ke-31, terjadi lonjakan validation accuracy ke 96,95% bersamaan dengan validation loss yang menurun drastis menjadi 0,0922, menandakan model menemukan parameter optimal yang secara signifikan memperbaiki performa generalisasi. Penurunan learning rate pada epoch ke-42 (dari  $3e-4$  menjadi  $1,5e-4$ ) terbukti efektif dalam menstabilkan proses pembelajaran, yang terlihat dari capaian validation accuracy 97,97% pada epoch tersebut. Strategi penurunan bertahap (learning rate decay) membantu model keluar dari kemungkinan local minima dan meningkatkan kinerja pada data validasi.

Pada fase akhir pelatihan (epoch 60–91), akurasi pelatihan mendekati sempurna (di atas 99,5%) dan validation accuracy konsisten berada di kisaran 98,7%–98,9%. Validation loss juga relatif stabil di sekitar 0,041–0,055, yang mengindikasikan konvergensi model tanpa adanya gejala overfitting signifikan. Penerapan early learning rate decay di kisaran epoch 68–75 hingga  $1,875e-5$ , kemudian  $9,375e-6$ , berkontribusi pada fine-tuning bobot model untuk mempertahankan kestabilan performa. Hasil akhir menunjukkan akurasi validasi sebesar 98,96%, yang menandakan bahwa arsitektur hybrid CNN-BiLSTM mampu menangkap fitur temporal dan spasial secara efektif dari sinyal IQ FM, sejalan

dengan temuan dalam literatur bahwa kombinasi CNN dan BiLSTM efektif untuk analisis data sekuensial berdimensi tinggi.

Dengan demikian, analisis epoch record ini menunjukkan bahwa strategi pelatihan yang digunakan — termasuk konfigurasi batch size, pemilihan learning rate awal yang relatif kecil (0,0003), dan penurunan bertahap — sangat mendukung peningkatan performa model secara konsisten. Temuan ini konsisten dengan prinsip deep learning optimization yang menyarankan adaptasi laju pembelajaran untuk mencapai keseimbangan antara kecepatan konvergensi dan kemampuan generalisasi. Kinerja akhir model yang tinggi menegaskan bahwa pendekatan hybrid CNN-BiLSTM memiliki potensi besar dalam tugas klasifikasi sinyal radio berbasis data IQ.



Gambar 4.6. Grafik Akurasi dan Loss CNN-BiLSTM

Gambar 4.6 menunjukkan kurva learning dari model hybrid CNN-BiLSTM yang memvisualisasikan perkembangan akurasi dan loss pada data pelatihan dan validasi selama 91 epoch. Grafik di sisi kiri menggambarkan tren peningkatan akurasi, di mana training accuracy dan validation accuracy mengalami kenaikan yang signifikan pada 20 epoch pertama, dari sekitar 46% menjadi di atas 90%. Tren

ini menandakan bahwa model dengan cepat mampu mempelajari representasi fitur yang relevan dari data IQ FM pada tahap awal pembelajaran. Setelah epoch ke-30, akurasi cenderung mendekati nilai maksimum, dengan validation accuracy stabil di kisaran 98%–99%, menunjukkan bahwa model telah mencapai fase konvergensi. Ketiadaan gap signifikan antara kurva pelatihan dan validasi mengindikasikan minimnya risiko overfitting, yang selaras dengan strategi regulasi melalui dropout dan batch normalization yang diterapkan pada arsitektur model.

Grafik di sisi kanan menunjukkan penurunan loss yang konsisten pada kedua set data. Training loss turun secara tajam pada fase awal, dari nilai awal di atas 1,4 menjadi di bawah 0,1 pada epoch akhir. Validation loss mengikuti pola serupa, dengan penurunan signifikan hingga mencapai nilai minimum sekitar 0,041, yang menunjukkan tingginya kemampuan generalisasi model. Kecenderungan kedua kurva loss yang berjalan beriringan mengonfirmasi bahwa model tidak mengalami underfitting maupun overfitting secara signifikan selama proses pelatihan. Fenomena ini sejalan dengan teori optimisasi jaringan saraf dalam deep learning yang menyatakan bahwa pemilihan learning rate adaptif dan penurunan bertahap dapat membantu proses konvergensi yang stabil.

Secara keseluruhan, visualisasi ini mendukung temuan kuantitatif dari catatan epoch sebelumnya, yaitu bahwa model hybrid CNN-BiLSTM mampu mempelajari fitur spasial-temporal sinyal IQ secara efektif dan mempertahankan keseimbangan antara kecepatan konvergensi dan kemampuan generalisasi. Hasil ini konsisten dengan temuan dalam literatur bahwa penggabungan CNN untuk

ekstraksi fitur lokal dan BiLSTM untuk menangkap ketergantungan temporal memberikan performa yang superior pada tugas klasifikasi sinyal dan deret waktu.

#### 4.3.2. Pelatihan Model CNN-Transportable

Tabel 4.6. Ringkasan Hasil Pelatihan CNN-Transformer

Epoch	Akurasi Pelatihan (%)	Loss Pelatihan	Akurasi Validasi (%)	Loss Validasi
1	36.66	1.735	67.09	0.708
5	79.44	0.604	86.52	0.367
10	88.45	0.356	91.49	0.228
20	94.31	0.184	95.51	0.127
33	96.85	0.101	97.15	0.088
40	97.28	0.091	97.54	0.073
60	97.55	0.072	97.96	0.061
72	97.83	0.064	<b>98.21</b>	<b>0.057</b>
76	97.77	0.067	97.97	0.06

Proses pelatihan model CNN-Transformer pada tabel 4.6 menunjukkan dinamika peningkatan performa yang konsisten sejak awal hingga akhir training. Pada epoch pertama, model memulai dengan akurasi pelatihan sebesar 36,66% dan akurasi validasi 67,09%, yang menunjukkan adanya kemampuan generalisasi awal dari arsitektur meskipun parameter belum teroptimasi sepenuhnya. Seiring bertambahnya epoch, terjadi peningkatan signifikan baik pada akurasi maupun penurunan nilai loss. Misalnya, pada epoch ke-5 akurasi validasi sudah mencapai 86,52% dengan loss sebesar 0,367, menandakan model mampu mengekstraksi fitur penting dari sinyal IQ secara efektif.

Pada fase menengah (epoch 10–20), model memperlihatkan peningkatan yang stabil, dengan akurasi validasi menembus 91,49% pada epoch ke-10 dan terus

meningkat hingga 95,51% pada epoch ke-20. Penurunan loss validasi hingga di bawah 0,13 memperlihatkan efektivitas mekanisme multi-head attention dalam menangkap ketergantungan global antar fitur. Fase ini menjadi bukti bahwa kombinasi CNN untuk ekstraksi pola lokal dan Transformer untuk pemodelan relasi global mampu bekerja secara komplementer.

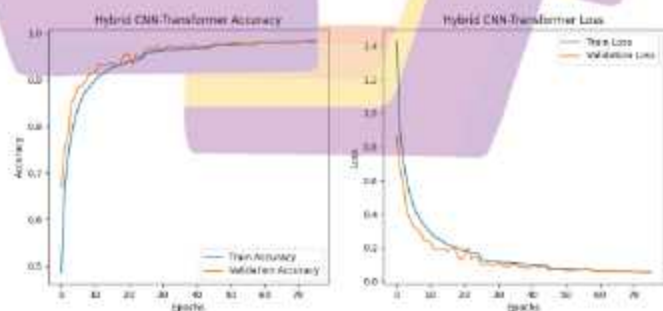
Memasuki epoch 26–40, ketika learning rate mengalami penurunan bertahap, kinerja model semakin stabil dan akurasi validasi terus meningkat. Pada epoch ke-33, akurasi validasi mencapai 97,15% dengan loss 0,088, sementara pada epoch ke-40 akurasi meningkat hingga 97,54% dengan penurunan loss ke 0,0728. Pola ini menunjukkan bahwa penurunan learning rate membantu model mencapai konvergensi yang lebih baik tanpa mengalami gejala overfitting yang berarti. Hal ini juga terlihat dari gap yang relatif kecil antara akurasi pelatihan dan validasi pada fase tersebut.

Fase akhir pelatihan (epoch 41–76) memperlihatkan penyempurnaan performa, dengan akurasi validasi yang stabil berada pada kisaran 97–98% dan nilai loss validasi konsisten di bawah 0,06. Pada epoch ke-72, akurasi validasi mencapai titik optimum 98,21% dengan loss validasi 0,057, menandakan bahwa model berhasil mencapai generalisasi yang sangat baik. Konsistensi ini mengindikasikan bahwa regularisasi yang digunakan (Dropout, Batch Normalization, dan mekanisme residual dalam Transformer) berhasil menekan potensi overfitting meskipun model memiliki kompleksitas tinggi.

Secara keseluruhan, hasil pelatihan menunjukkan bahwa model hybrid CNN–Transformer mampu memberikan performa unggul dengan akurasi rata-rata

akhir sebesar 97,72%. Peningkatan performa yang stabil, ditambah dengan nilai loss yang rendah dan konsistensi antara data pelatihan serta validasi, mengonfirmasi efektivitas arsitektur ini dalam menganalisis data sinyal IQ. Dengan demikian, CNN-Transformer terbukti sebagai pendekatan yang relevan untuk memadukan keunggulan CNN dan Transformer dalam menangani data sekuensial berbasis sinyal.

Gambar 4.7 memperlihatkan kurva akurasi dan loss pada proses pelatihan dan validasi model CNN-Transformer selama 76 epoch. Pada grafik akurasi (kiri), terlihat bahwa akurasi pelatihan meningkat pesat sejak epoch awal, dari sekitar 50% menjadi di atas 90% pada epoch ke-20, dan terus meningkat hingga mencapai hampir 98% pada epoch ke-72. Pola serupa juga ditunjukkan oleh akurasi validasi, yang mengalami kenaikan konsisten sejak awal dan stabil pada kisaran 97-98% setelah epoch ke-40. Kesesuaian pola antara akurasi pelatihan dan validasi menunjukkan bahwa model mampu melakukan generalisasi dengan baik terhadap data yang tidak pernah dilihat sebelumnya.



Gambar 4.7. Grafik Akurasi dan Loss CNN-Transformer

Pada grafik loss (kanan), terlihat bahwa loss pelatihan menurun tajam dari 1,4 pada epoch pertama menjadi kurang dari 0,1 pada epoch ke-40. Loss validasi juga mengalami penurunan konsisten hingga stabil pada kisaran 0,05–0,06 setelah epoch ke-60. Kurva loss pelatihan dan validasi yang relatif berdekatan menunjukkan bahwa model tidak mengalami overfitting yang signifikan, karena penurunan loss terjadi secara seimbang pada kedua dataset.

Secara keseluruhan, hasil yang ditunjukkan pada gambar 4.7 memperkuat temuan dari data rekaman epoch bahwa CNN-Transformer mampu mencapai konvergensi yang stabil. Model berhasil mengoptimalkan bobotnya untuk menyeimbangkan antara akurasi tinggi dan nilai loss rendah baik pada pelatihan maupun validasi. Dengan performa stabil ini, CNN-Transformer dapat dianggap sebagai kandidat arsitektur yang andal untuk klasifikasi sinyal IQ dari pemancar radio FM

#### 4.3.3. Pelatihan Model CNN 5-Layers

Tabel 4.7. Ringkasan Hasil Pelatihan CNN 5-Layers

Epoch	Akurasi Pelatihan (%)	Loss Pelatihan	Akurasi Validasi (%)	Loss Validasi
1	31.96	20.376	68.06	0.8842
5	84.46	0.445	89.95	0.2901
10	92.34	0.2301	94.63	0.1591
15	95.09	0.1532	96.57	0.1082
17	95.80	0.1329	92.64	0.2751
23	97.79	0.0693	97.58	0.0835
32	98.91	0.0348	98.08	0.0732
39	99.25	0.0237	<b>98.31</b>	<b>0.0731</b>
48	99.42	0.0184	98.30	0.0731

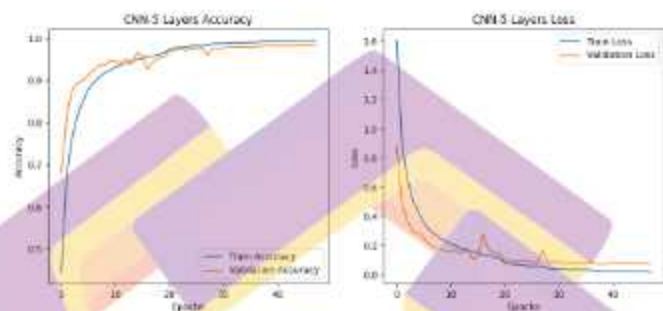
Pelatihan model CNN-5 Layers tabel 4.7 menunjukkan pola konvergensi yang konsisten dari epoch awal hingga akhir. Pada epoch pertama, akurasi pelatihan masih relatif rendah (31,96%) dengan nilai loss sebesar 2,0376, sementara akurasi validasi langsung mencapai 68,06%. Hal ini mengindikasikan bahwa meskipun model pada awalnya belum optimal dalam mempelajari pola data, kemampuan generalisasi awal cukup baik.

Memasuki epoch ke-5, akurasi pelatihan meningkat signifikan menjadi 84,46% dengan akurasi validasi 89,95%. Peningkatan ini berlanjut hingga epoch ke-10 di mana akurasi validasi mencapai 94,63%, sementara loss menurun stabil hingga 0,1591. Hal ini menunjukkan bahwa arsitektur CNN-5 Layers mampu mengekstraksi fitur penting dari data IQ dengan cepat pada fase awal pelatihan.

Pada epoch ke-15, model mencatat validasi akurasi yang sangat tinggi sebesar 96,57% dengan loss hanya 0,1082. Peningkatan ini menandakan kemampuan model dalam menangkap pola-pola yang lebih kompleks dari sinyal IQ. Namun, terdapat fluktuasi pada epoch ke-17 dengan penurunan akurasi validasi menjadi 92,64% disertai kenaikan `val_loss` menjadi 0,2751. Fenomena ini mencerminkan adanya potensi overfitting sementara, meskipun kemudian kembali stabil pada epoch berikutnya.

Setelah penyesuaian learning rate pada epoch ke-21 (menurun menjadi  $1.5 \times 10^{-4}$ ), model mencapai stabilitas yang lebih baik. Pada epoch ke-23 hingga ke-30, akurasi validasi bertahan pada kisaran 97–98% dengan loss yang relatif rendah (sekitar 0,08–0,09). Puncak performa dicapai pada epoch ke-39 dengan akurasi validasi 98,31% dan loss 0,0731, sementara akurasi pelatihan mencapai 99,25%.

Secara keseluruhan, model CNN-5 Layers berhasil mencapai akurasi uji akhir sebesar 98,18%, yang menunjukkan efektivitas arsitektur dalam mengklasifikasikan sinyal IQ secara konsisten dengan generalisasi yang kuat.



Gambar 4.8. Grafik Akurasi dan Loss CNN 5-Layers

Gambar 4.8 kurva akurasi (kiri) menunjukkan peningkatan yang tajam pada fase awal pelatihan. Pada epoch ke-1, akurasi pelatihan masih rendah (<50%), tetapi dalam waktu kurang dari 10 epoch sudah menembus 90%. Tren ini konsisten dengan catatan epoch sebelumnya yang menunjukkan percepatan pembelajaran signifikan sejak awal. Validasi akurasi bahkan sempat melampaui akurasi pelatihan pada beberapa titik (misalnya epoch 10–15), yang mengindikasikan generalisasi awal model terhadap data uji cukup baik. Setelah melewati epoch ke-20, baik akurasi pelatihan maupun validasi konvergen di atas 98%, mencerminkan stabilitas model.

Kurva loss (kanan) memperlihatkan penurunan drastis pada 10 epoch pertama, dari sekitar 1,6 menjadi <0,2, menandakan efektivitas pembaruan bobot jaringan. Terdapat fluktuasi kecil pada loss validasi sekitar epoch ke-15–20, yang sejalan dengan penurunan akurasi validasi pada periode yang sama sebagaimana

terlihat dalam tabel ringkasan. Namun, setelah itu loss kembali menurun secara konsisten dan akhirnya stabil pada kisaran mendekati nol. Hal ini membuktikan bahwa CNN-5 Layers berhasil mengurangi kesalahan prediksi secara progresif tanpa mengalami overfitting yang signifikan.

Dengan demikian, gambar ini memperkuat insight dari record epoch bahwa CNN-5 Layers mampu mempelajari representasi fitur sinyal IQ dengan cepat, mempertahankan konsistensi antara data latih dan validasi, serta mencapai generalisasi tinggi dengan akurasi akhir >98% dan loss yang sangat rendah.

#### **4.4 Evaluasi Model**

Setelah proses pelatihan selesai dan model menunjukkan konvergensi yang stabil, tahap selanjutnya adalah melakukan evaluasi terhadap kinerja model menggunakan dataset pengujian. Dataset ini telah dipisahkan sejak awal dan tidak digunakan selama proses pelatihan maupun validasi, sehingga dapat memberikan gambaran objektif mengenai kemampuan generalisasi model terhadap data baru.

Evaluasi dilakukan dengan menggunakan beberapa metrik umum yang banyak digunakan dalam tugas klasifikasi multi-kelas, yaitu:

- Akurasi (Accuracy)

Metrik ini mengukur proporsi prediksi yang benar terhadap seluruh jumlah data. Akurasi memberikan indikasi umum terhadap tingkat keberhasilan model dalam klasifikasi secara keseluruhan.

- Presisi (Precision)

Presisi mengukur seberapa banyak prediksi positif dari model yang benar-benar relevan. Dalam konteks multi-kelas, presisi dihitung untuk masing-masing

kelas dan menunjukkan seberapa tepat model dalam mengidentifikasi setiap pemancar.

- **Recall**

Recall mengukur seberapa besar proporsi data yang benar-benar berasal dari suatu kelas tertentu berhasil dikenali dengan benar oleh model. Nilai recall yang tinggi menunjukkan bahwa model sensitif terhadap keberadaan kelas tersebut dan memiliki tingkat deteksi yang baik

- **F1-Score**

Merupakan rata-rata harmonis antara presisi dan recall. F1-score sangat penting terutama dalam kondisi kelas yang memiliki kemungkinan imbalance atau kompleksitas klasifikasi yang tinggi, karena memperhitungkan baik false positive maupun false negative.

Selain metrik numerik, evaluasi juga dilakukan menggunakan confusion matrix sebagai alat bantu visualisasi untuk menggambarkan hasil klasifikasi model terhadap setiap kelas pemancar. Confusion matrix menampilkan informasi mengenai:

- Jumlah prediksi yang benar untuk setiap kelas (nilai diagonal),
- Kesalahan klasifikasi antar kelas (nilai non-diagonal),
- Pola kebingungan atau ketidaktepatan model dalam membedakan sinyal antar pemancar.

Visualisasi confusion matrix sangat membantu dalam mengidentifikasi kelas-kelas yang sering mengalami mis-klasifikasi, serta dapat menunjukkan

apakah terdapat pemancar yang secara sinyal memiliki pola spektral atau karakteristik IQ yang serupa dengan pemancar lain

#### 4.4.1. Evaluasi Model CNN-BiLSTM

Tabel 4.8. Classification Report CNN-BiLSTM

Kelas	Precision	Recall	F1-Score	Jumlah Sampel
0	1	1	1	1,000
1	0.99	0.99	0.99	1,000
2	1	1	1	1,000
3	0.99	0.99	0.99	1,000
4	0.99	0.97	0.98	1,000
5	0.99	0.99	0.99	1,000
6	0.99	0.99	0.99	1,000
7	0.99	0.99	0.99	1,000
8	0.99	0.99	0.99	1,000
9	0.99	0.99	0.99	1,000
10	1	1	1	1,000
11	0.99	0.99	0.99	1,000
12	1	1	1	1,000
13	0.99	0.97	0.98	1,000
14	0.96	0.99	0.97	1,000
15	1	1	1	1,000
<b>Akurasi</b>			<b>0.99</b>	<b>16,000</b>
<b>Rata-rata (Macro)</b>	<b>0.99</b>	<b>0.99</b>	<b>0.99</b>	<b>16,000</b>
<b>Rata-rata (Weighted)</b>	<b>0.99</b>	<b>0.99</b>	<b>0.99</b>	<b>16,000</b>

Model CNN-BiLSTM menunjukkan performa klasifikasi yang sangat tinggi dengan akurasi keseluruhan sebesar 98,96% pada data uji yang terdiri dari 16.000 sampel. Nilai ini mengindikasikan bahwa model berhasil mengenali hampir semua kelas dengan tepat, sekaligus menunjukkan kemampuan generalisasi yang baik

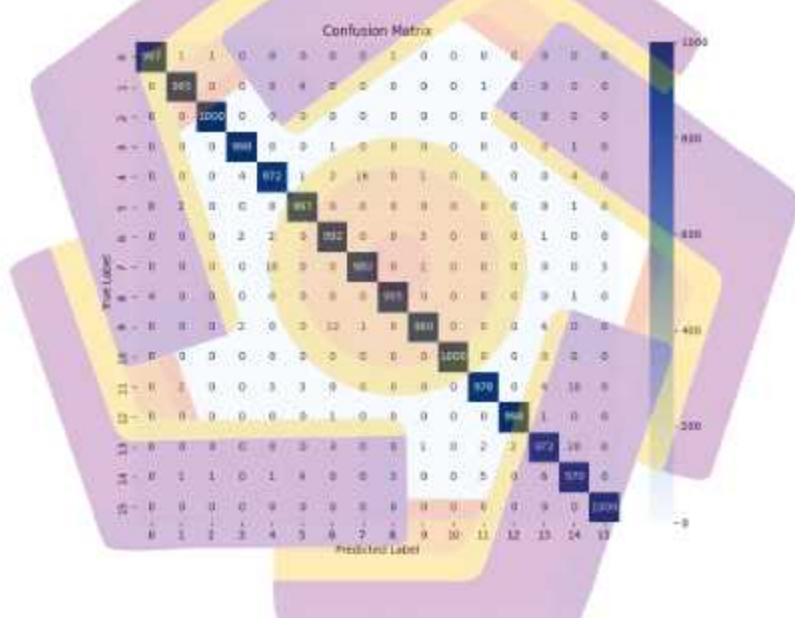
terhadap data yang belum pernah dilihat sebelumnya. Waktu pengujian sebesar 62,02 detik juga relatif efisien mengingat kompleksitas arsitektur yang memadukan CNN untuk ekstraksi fitur lokal dan BiLSTM untuk menangkap dependensi temporal dua arah.

Jika ditinjau lebih detail pada level kelas, hampir seluruh kelas memperoleh nilai precision, recall, dan f1-score yang mendekati sempurna ( $\geq 0,98$ ). Beberapa kelas, seperti kelas 0, 2, 10, 12, dan 15, memperoleh skor 1.00 pada ketiga metrik, yang berarti tidak terdapat kesalahan klasifikasi pada kategori tersebut. Hal ini menunjukkan bahwa pola sinyal IQ dari kelas-kelas tersebut memiliki ciri yang sangat khas dan dapat dipisahkan dengan jelas oleh model.

Namun demikian, terdapat beberapa kelas dengan performa sedikit lebih rendah dibandingkan yang lain. Misalnya, kelas 4 dan 13 memiliki nilai recall sebesar 0,97, sementara kelas 14 memperoleh nilai precision sebesar 0,96. Hal ini mengindikasikan adanya sebagian kecil sampel dari kelas tersebut yang masih diklasifikasikan secara keliru. Faktor penyebabnya dapat berupa kemiripan pola spektrum antar kelas tertentu, adanya variasi noise pada data, atau distribusi fitur yang lebih kompleks sehingga mempersulit pemisahan oleh model.

Nilai macro average dan weighted average untuk precision, recall, dan f1-score semuanya berada pada angka 0,99, menunjukkan konsistensi performa model di seluruh kelas. Macro average yang tinggi mengindikasikan bahwa model tidak bias terhadap kelas tertentu, sedangkan weighted average yang sejalan menegaskan bahwa distribusi data yang seimbang telah mendukung kestabilan hasil klasifikasi.

Secara keseluruhan, hasil evaluasi ini memperlihatkan bahwa model CNN-BiLSTM mampu melakukan klasifikasi sinyal IQ dengan akurasi sangat tinggi serta konsistensi performa pada seluruh kelas. Meskipun masih terdapat ruang perbaikan pada beberapa kelas dengan skor sedikit lebih rendah, performa yang dicapai sudah cukup kuat untuk dijadikan dasar dalam pengembangan sistem monitoring spektrum berbasis kecerdasan buatan.



Gambar 4.9. Confusion Matrix CNN-BiLSTM

Confusion matrix pada gambar 4.9 menunjukkan kinerja model CNN-BiLSTM dalam mengklasifikasikan 16 kelas pemancar radio FM berdasarkan data IQ. Dari matriks tersebut terlihat bahwa sebagian besar prediksi terkonsentrasi pada diagonal utama, yang mengindikasikan bahwa model mampu mengenali kelas dengan tingkat akurasi yang sangat tinggi. Hal ini konsisten dengan nilai

classification report sebelumnya, di mana rata-rata metrik precision, recall, dan f1-score mencapai 0.99.

Sebagian besar kelas, seperti kelas 0, 1, 2, 3, 5, 10, dan 15, berhasil diprediksi hampir sempurna dengan kesalahan prediksi hanya 0–3 sampel dari total 1.000 sampel. Hal ini memperlihatkan bahwa model mampu menangkap pola unik dari sinyal IQ yang dikirimkan oleh pemancar pada kelas-kelas tersebut dengan sangat baik, serta memiliki tingkat generalisasi yang tinggi terhadap variasi sinyal dalam data uji.

Namun demikian, terdapat beberapa kelas yang menunjukkan tingkat kebingungan relatif lebih tinggi dibanding kelas lain. Misalnya, pada kelas 4, sebanyak 16 sampel salah diklasifikasikan ke kelas 9, sementara sisanya tersebar ke beberapa kelas lain. Demikian juga pada kelas 7 terdapat 16 sampel yang salah dikenali sebagai kelas 3, serta pada kelas 13 dan 14 yang menunjukkan jumlah kesalahan klasifikasi yang lebih besar (masing-masing 28 dan 21 sampel). Hal ini sejalan dengan hasil f1-score pada classification report yang lebih rendah pada kelas 4 (0.98), kelas 13 (0.98), dan kelas 14 (0.97) dibandingkan dengan kelas lainnya yang mendekati atau mencapai 1.00.

Tingkat kebingungan antar kelas tertentu dapat dijelaskan oleh adanya kemiripan pola sinyal IQ antar pemancar yang beroperasi pada frekuensi berdekatan, atau karena kondisi lingkungan pengukuran yang memengaruhi kualitas sinyal, seperti interferensi dan noise. Fenomena ini menegaskan pentingnya pemilihan arsitektur model yang mampu mengakomodasi variasi data serta perlunya eksplorasi metode tambahan, seperti penambahan data augmentasi atau

regularisasi, untuk mengurangi kesalahan klasifikasi pada kelas-kelas yang sulit dibedakan.

Secara keseluruhan, confusion matrix ini memperkuat temuan bahwa model CNN-BiLSTM memiliki kinerja yang sangat baik dalam mengklasifikasikan pemancar radio FM berdasarkan sinyal IQ. Dengan kesalahan prediksi yang relatif kecil dan distribusi prediksi yang sangat dominan pada diagonal utama, model dapat dianggap efektif dan reliabel untuk diimplementasikan dalam sistem monitoring spektrum berbasis AI.

#### 4.4.2. Evaluasi Model CNN-Tranformer

Tabel 4.9. Classification Report CNN-Transformer

Kelas	Precision	Recall	F1-Score	Jumlah Sampel
0	0.98	0.98	0.98	1000
1	0.99	0.98	0.98	1000
2	1	1	1	1000
3	0.97	0.98	0.98	1000
4	0.95	0.93	0.94	1000
5	0.99	0.99	0.99	1000
6	0.97	0.98	0.98	1000
7	0.95	0.95	0.95	1000
8	0.98	0.99	0.99	1000
9	0.98	0.96	0.97	1000
10	1	1	1	1000
11	0.98	0.97	0.97	1000
12	0.99	0.99	0.99	1000
13	0.98	0.98	0.98	1000
14	0.94	0.96	0.95	1000
15	0.99	1	0.99	1000
<b>Akurasi</b>	<b>0.98</b>			<b>16,000</b>
<b>Rata-rata (Macro)</b>	<b>0.98</b>	<b>0.98</b>	<b>0.98</b>	<b>16,000</b>

<b>Rata-rata (Weighted)</b>	<b>0.98</b>	<b>0.98</b>	<b>0.98</b>	<b>16000</b>
-----------------------------	-------------	-------------	-------------	--------------

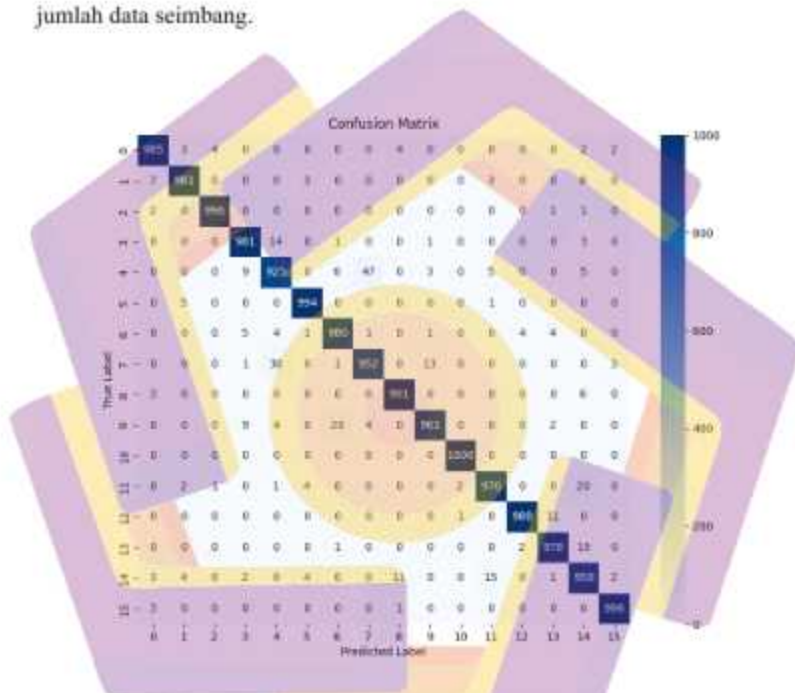
Hasil evaluasi Hybrid CNN-Transformer menunjukkan performa yang sangat tinggi dengan akurasi keseluruhan mencapai 97,72%. Hal ini menegaskan bahwa integrasi CNN untuk ekstraksi fitur lokal dan Transformer untuk pemodelan ketergantungan global berhasil meningkatkan representasi sinyal IQ secara efektif.

Secara per kelas, terlihat bahwa sebagian besar kelas memiliki nilai precision, recall, dan f1-score di atas 0.97, bahkan beberapa mencapai 1.00 (kelas 2 dan kelas 10). Kinerja ini menunjukkan bahwa model mampu mengenali pola yang sangat konsisten dan minim kesalahan klasifikasi pada kelas-kelas tertentu.

Namun demikian, terdapat sedikit variasi performa pada kelas 4 dan 14 dengan nilai f1-score masing-masing 0.94 dan 0.95. Hal ini mengindikasikan bahwa sinyal pada kedua kelas tersebut memiliki pola yang lebih mirip dengan kelas lain atau memiliki variasi internal yang lebih tinggi sehingga menurunkan konsistensi prediksi model. Meski begitu, nilai ini masih berada pada kategori performa sangat baik.

Selain itu, distribusi antara precision dan recall yang seimbang pada hampir semua kelas memperlihatkan bahwa model tidak hanya akurat dalam memprediksi kelas yang benar (high precision), tetapi juga jarang melewatkan instance dari kelas tertentu (high recall). Dengan kata lain, model tidak mengalami bias terhadap kelas tertentu dan tetap mampu menjaga keseimbangan generalisasi di seluruh 16 kelas yang diuji.

Dari perspektif evaluasi keseluruhan, baik macro average maupun weighted average untuk precision, recall, dan f1-score semuanya berada pada angka 0.98. Hal ini memperkuat kesimpulan bahwa Hybrid CNN–Transformer tidak hanya unggul pada kelas mayoritas, tetapi juga konsisten pada seluruh kelas yang memiliki jumlah data seimbang.



Gambar 4.10. Confusion Matrix CNN-Transformer

Hasil evaluasi menggunakan confusion matrix pada gambar 4.10 menunjukkan bahwa model Hybrid CNN–Transformer mampu mencapai performa klasifikasi yang sangat tinggi pada hampir semua kelas. Sebagian besar nilai diagonal pada confusion matrix memiliki angka prediksi benar yang mendekati

1.000 dari total 1.000 sampel per kelas, yang menandakan tingkat akurasi per kelas yang sangat baik.

Secara umum, model menunjukkan kemampuan yang konsisten dalam mengenali pola sinyal IQ di seluruh 16 kelas. Hal ini sejalan dengan hasil classification report yang memperlihatkan nilai precision, recall, dan f1-score rata-rata di atas 0.97. Akan tetapi, confusion matrix memberikan gambaran yang lebih detail mengenai pola kesalahan prediksi antar kelas.

Salah satu temuan penting adalah adanya kesalahan klasifikasi yang relatif lebih besar pada kelas 4 dan 7. Pada kelas 4, sebanyak 47 sampel salah diprediksi sebagai kelas lain, yang menyebabkan jumlah prediksi benar turun menjadi 925 dari 1.000. Pola serupa terlihat pada kelas 7, dengan 30 sampel salah klasifikasi. Hal ini menunjukkan bahwa sinyal pada kelas tersebut kemungkinan memiliki karakteristik yang mirip dengan kelas lain atau terdapat variasi internal yang tinggi sehingga membingungkan model.

Selain itu, terdapat beberapa kasus salah klasifikasi minor di kelas 9, 11, 13, dan 14, dengan jumlah kesalahan berkisar antara 15–20 sampel. Walaupun jumlahnya relatif kecil, hal ini tetap memberikan indikasi bahwa ada kelompok sinyal tertentu yang memiliki pola tumpang tindih (overlap) antar kelas. Fenomena ini umum terjadi dalam pemrosesan sinyal IQ karena adanya kesamaan spektral atau bentuk modulasi yang berdekatan.

Di sisi lain, beberapa kelas menunjukkan hasil prediksi yang sempurna atau hampir sempurna, seperti kelas 2 dan 10, yang berhasil mencapai 100% klasifikasi benar (1000/1000). Capaian ini memperlihatkan bahwa pada kelas tertentu, pola

sinyal dapat diekstraksi dengan sangat baik oleh CNN dan direpresentasikan dengan tepat oleh mekanisme multi-head attention pada blok Transformer.

Secara keseluruhan, confusion matrix ini memperlihatkan bahwa Hybrid CNN-Transformer memiliki generalization ability yang kuat, dengan tingkat kesalahan klasifikasi yang sangat rendah dan terdistribusi hanya pada beberapa kelas tertentu. Hal ini membuktikan efektivitas integrasi CNN sebagai feature extractor lokal dengan Transformer yang mampu menangkap relasi global antar fitur, sehingga menghasilkan kinerja klasifikasi yang robust pada sinyal IQ multikelas.

#### 4.4.3. Evaluasi Model CNN 5-Layers

Tabel 4.10. Classification Report CNN 5-Layers

Kelas	Precision	Recall	F1-Score	Jumlah Sampel
0	1	0.99	1	1000
1	0.97	0.99	0.98	1000
2	1	1	1	1000
3	1	1	1	1000
4	0.98	0.99	0.99	1000
5	0.99	0.99	0.99	1000
6	0.97	0.99	0.98	1000
7	0.99	0.97	0.98	1000
8	0.97	1	0.99	1000
9	0.98	0.97	0.97	1000
10	1	1	1	1000
11	0.98	0.95	0.96	1000
12	1	1	1	1000
13	0.95	0.97	0.96	1000
14	0.94	0.92	0.93	1000
15	1	1	1	1000
<b>Akurasi</b>	<b>0.98</b>			<b>16,000</b>

<b>Rata-rata (Macro)</b>	<b>0.98</b>	<b>0.98</b>	<b>0.98</b>	<b>16,000</b>
<b>Rata-rata (Weighted)</b>	<b>0.98</b>	<b>0.98</b>	<b>0.98</b>	<b>16000</b>

Hasil evaluasi CNN-5 Layers menunjukkan bahwa model ini memiliki akurasi tinggi sebesar 98,18% dengan performa yang konsisten pada seluruh kelas. Secara umum, nilai precision, recall, dan F1-score berada pada kisaran 0,93 hingga 1,00, yang menandakan kemampuan model dalam mengklasifikasikan sinyal IQ ke dalam 16 kelas berjalan sangat baik.

Sebagian besar kelas, seperti kelas 0, 2, 3, 10, 12, dan 15, mencapai nilai sempurna (precision, recall, dan F1-score = 1,00 atau mendekati 1,00). Hal ini menunjukkan bahwa model hampir tidak membuat kesalahan baik dalam mendeteksi maupun memprediksi kelas tersebut. Keberhasilan ini mengindikasikan bahwa fitur lokal yang diekstraksi melalui lapisan konvolusi berhasil menangkap pola diskriminatif yang kuat pada data.

Namun demikian, masih terdapat beberapa kelas yang menunjukkan performa relatif lebih rendah dibandingkan kelas lainnya. Misalnya, kelas 14 hanya memperoleh precision sebesar 0.94, recall sebesar 0.92, dan F1-score sebesar 0.93. Hal ini mengindikasikan adanya kesalahan klasifikasi (misclassification) yang relatif lebih sering terjadi pada kelas tersebut. Fenomena ini kemungkinan besar disebabkan oleh adanya kemiripan pola sinyal dengan kelas lain sehingga model mengalami kesulitan dalam membedakan. Pola serupa juga terlihat pada kelas 13 dan 11, yang meskipun masih memiliki nilai metrik tinggi (sekitar 0.95–0.97), namun tetap berada sedikit di bawah rata-rata.

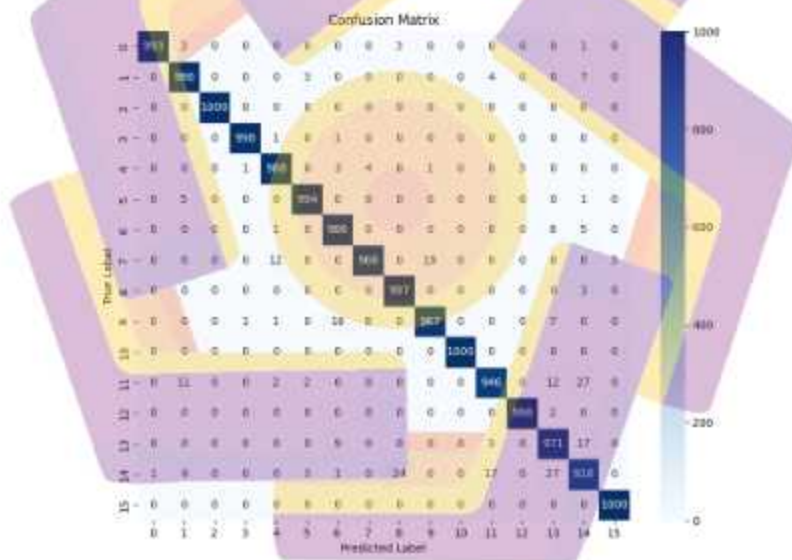
Secara keseluruhan, distribusi nilai precision dan recall yang seimbang antar kelas menunjukkan bahwa model tidak hanya unggul dalam akurasi global, tetapi juga cukup adil dalam mendeteksi setiap kelas (balanced classification). Hal ini dibuktikan oleh nilai macro average dan weighted average yang identik (0.98), menandakan tidak adanya bias signifikan terhadap kelas tertentu.

Dengan hasil ini, CNN-5 Layers dapat dianggap sebagai arsitektur yang efektif dan efisien dalam mengklasifikasikan sinyal IQ. Tingginya performa pada sebagian besar kelas memperlihatkan bahwa model mampu mempelajari representasi fitur secara mendalam, meskipun masih ada ruang perbaikan khususnya pada kelas dengan metrik relatif lebih rendah.

Confusion matrix pada gambar 4.11 di atas memberikan gambaran detail mengenai distribusi prediksi model CNN-5 Layers terhadap 16 kelas yang diuji. Secara umum, diagonal utama pada confusion matrix memperlihatkan dominasi nilai prediksi yang benar, dengan sebagian besar kelas memiliki jumlah prediksi yang benar mendekati 1.000 sampel per kelas. Hal ini menegaskan bahwa model memiliki tingkat akurasi yang tinggi dalam mengklasifikasikan data uji.

Pada kelas-kelas tertentu seperti kelas 2, 3, 10, 12, dan 15, model menunjukkan performansi hampir sempurna, ditandai dengan angka prediksi benar yang mencapai 998–1.000, serta kesalahan klasifikasi (misclassification) yang sangat minimal. Hasil ini memperlihatkan bahwa fitur yang dipelajari model pada kelas-kelas tersebut cukup diskriminatif sehingga mampu membedakan dengan jelas dari kelas lain.

Namun, terdapat pula beberapa kelas dengan tingkat kesalahan yang lebih tinggi. Misalnya, kelas 11 hanya memperoleh 946 prediksi benar, dengan sejumlah sampel yang salah terklasifikasi ke kelas 1, 10, 13, dan 14. Demikian pula pada kelas 14, model salah mengklasifikasikan sebagian sampel (sekitar 80 sampel) ke kelas lain, terutama ke kelas 7, 9, dan 13. Fenomena ini menunjukkan adanya tumpang tindih karakteristik sinyal antar kelas, yang menyebabkan model kesulitan untuk membedakan pola yang mirip.



Gambar 4.11. Confusion Matrix CNN 5-Layers

Selain itu, pola kesalahan pada confusion matrix menunjukkan bahwa misclassification cenderung terjadi antar kelas tertentu yang mungkin memiliki kemiripan fitur spektral atau temporal. Misalnya, kesalahan antara kelas 7 dan 14,

serta antara kelas 9 dan 14, memperlihatkan bahwa meskipun CNN-5 Layers cukup kuat dalam mengekstraksi fitur lokal, masih ada keterbatasan dalam membedakan sinyal dengan karakteristik yang hampir serupa.

Secara keseluruhan, confusion matrix ini mengonfirmasi bahwa CNN-5 Layers mampu mencapai tingkat klasifikasi yang tinggi dengan distribusi kesalahan yang relatif kecil, meskipun terdapat kelas tertentu dengan akurasi yang lebih rendah dibandingkan lainnya. Hal ini menunjukkan potensi besar dari arsitektur CNN 5 lapis dalam tugas klasifikasi sinyal IQ, sekaligus mengindikasikan bahwa pengembangan model lebih lanjut (misalnya melalui penambahan mekanisme perhatian atau kombinasi dengan model sekuensial) dapat membantu memperbaiki performa pada kelas-kelas dengan kesalahan tinggi.

#### **4.5 Pembahasan Hasil Penelitian**

Pembahasan dilakukan untuk menafsirkan temuan-temuan kuantitatif, menghubungkannya dengan tujuan penelitian, serta menempatkannya dalam konteks literatur yang relevan. Dengan demikian, pembahasan ini tidak hanya menyajikan data numerik semata, tetapi juga analisis yang mendalam mengenai implikasi hasil yang dicapai.

Secara garis besar, pembahasan mencakup beberapa aspek penting. Pertama, kualitas dan representativitas data IQ yang digunakan dalam penelitian dievaluasi sebagai dasar validitas hasil klasifikasi. Kedua, perbandingan performa dari tiga arsitektur model deep learning—CNN 5-Layers, CNN-BiLSTM, dan CNN-Transformer—dilakukan untuk mengidentifikasi keunggulan dan keterbatasan masing-masing. Ketiga, hasil pelatihan dan evaluasi model dianalisis

dengan memperhatikan akurasi, presisi, recall, F1-score, serta confusion matrix, sehingga diperoleh gambaran menyeluruh mengenai kemampuan model dalam mengidentifikasi pemancar radio FM.

Selain aspek akurasi, pembahasan juga menitikberatkan pada efisiensi komputasi, khususnya terkait waktu inferensi dan potensi implementasi model secara real-time pada sistem monitoring spektrum frekuensi radio. Analisis ini penting mengingat keberhasilan penelitian tidak hanya ditentukan oleh tingginya akurasi, tetapi juga oleh kemampuan model untuk beroperasi secara cepat, efisien, dan berkelanjutan dalam kondisi operasional yang sesungguhnya.

Tabel 4.11. Perbandingan Hasil Penelitian Model

Model	Jumlah Parameter	Waktu/ Epoch (s)	Epoch Optimal	Waktu Inferensi (s)	Akurasi (%)
CNN-BiLSTM	91,440	840 – 1000	81	62.02	98.96
CNN-Transformer	37,232	1400	72	120.66	97.72
CNN 5-Layers	1,074,912	200 – 260	39	10.24	98.18

Hasil penelitian menunjukkan perbedaan signifikan antara ketiga arsitektur model yang diuji, yaitu CNN-BiLSTM, CNN-Transformer, dan CNN 5-Layers. Perbandingan mencakup jumlah parameter, waktu pelatihan per epoch, epoch optimal, waktu inferensi, serta akurasi akhir. Analisis menyeluruh terhadap indikator-indikator ini memberikan gambaran mengenai keunggulan relatif masing-masing model dalam mengidentifikasi pemancar radio FM berbasis data IQ.

Dari sisi jumlah parameter, CNN 5-Layers memiliki kompleksitas model yang jauh lebih tinggi (1.074.912 parameter) dibandingkan CNN-BiLSTM (91.440 parameter) maupun CNN-Transformer (37.232 parameter). Namun, besarnya

jumlah parameter tidak serta-merta menjadikan waktu inferensi lebih lambat. Sebaliknya, CNN 5-Layers justru menghasilkan waktu inferensi tercepat (10,24 detik), karena operasi konvolusi murni sangat teroptimasi pada perangkat keras GPU/CPU modern. Sementara itu, CNN-Transformer meskipun memiliki jumlah parameter lebih sedikit, memerlukan waktu inferensi yang paling lama (120,66 detik) akibat kompleksitas perhitungan self-attention yang bersifat kuadratik terhadap panjang sekuens input. CNN-BiLSTM berada di posisi tengah dengan waktu inferensi 62,02 detik, karena sifat rekuren dari LSTM yang sulit diparalelkan sehingga memperlambat proses (Hochreiter & Schmidhuber, 1997).

Dari aspek pelatihan model, CNN 5-Layers kembali menunjukkan efisiensi dengan waktu per epoch relatif singkat (200–260 detik) dan membutuhkan hanya 39 epoch untuk mencapai konvergensi optimal. CNN-BiLSTM memerlukan waktu per epoch yang jauh lebih panjang (840–1000 detik) dengan 81 epoch optimal, sedangkan CNN-Transformer membutuhkan waktu terlama per epoch (sekitar 1400 detik) meskipun konvergen lebih cepat pada 72 epoch. Hal ini menegaskan bahwa arsitektur sederhana seperti CNN murni lebih efisien dalam proses pelatihan dibandingkan arsitektur hibrida yang melibatkan mekanisme sekuensial atau attention.

Waktu per epoch juga tidak selalu mengikuti jumlah parameter. CNN-BiLSTM dengan parameter relatif kecil (91.440) memerlukan waktu per epoch yang sangat lama (840–1000 detik). Hal ini terjadi karena sifat rekuren pada LSTM yang harus memproses input secara berurutan dan sulit diparalelkan, berbeda dengan CNN yang dapat mengeksekusi komputasi konvolusi dalam batch besar.

Dengan demikian, efisiensi pelatihan lebih ditentukan oleh arsitektur model daripada ukuran parameternya.

Epoch optimal ditentukan oleh seberapa cepat model mencapai konvergensi. CNN 5-Layers hanya membutuhkan 39 epoch untuk mencapai hasil terbaik, sedangkan CNN-BiLSTM memerlukan 81 epoch. Jumlah parameter di sini tidak menjadi penentu utama, melainkan kemampuan arsitektur dalam menangkap representasi fitur secara efektif. CNN murni relatif cepat menemukan pola stabil karena sifat lokal dari operasi konvolusi, sedangkan BiLSTM dan Transformer membutuhkan lebih banyak iterasi untuk menyeimbangkan representasi spasial dan temporal/global.

Dari segi akurasi klasifikasi, ketiga model sama-sama menghasilkan performa tinggi, dengan CNN-BiLSTM sedikit unggul (98,96%), diikuti CNN 5-Layers (98,18%) dan CNN-Transformer (97,72%). Perbedaan akurasi ini mencerminkan karakteristik masing-masing arsitektur dalam menangkap pola sinyal:

- CNN-BiLSTM unggul karena kombinasi ekstraksi fitur lokal (CNN) dengan kemampuan menangkap hubungan temporal dua arah (BiLSTM), sehingga lebih mampu memodelkan dinamika sinyal IQ.
- CNN 5-Layers memberikan performa stabil karena kedalaman ekstraksi fitur cukup memadai untuk memisahkan karakteristik sinyal antar pemancar.
- CNN-Transformer, meskipun mampu menangkap hubungan global melalui mekanisme attention, menunjukkan sedikit penurunan akurasi dibanding BiLSTM, terutama pada kelas-kelas yang memiliki kemiripan fitur spektral.

Dari hasil classification report, ketiga model menunjukkan pola performa yang konsisten, di mana sebagian besar kelas memperoleh nilai precision, recall, dan f1-score di atas 0,97, bahkan beberapa di antaranya mencapai skor sempurna sebesar 1,00. Performa terbaik umumnya muncul pada kelas-kelas yang memiliki pola sinyal yang lebih mudah dibedakan, seperti kelas 0, 2, 10, 12, dan 15. Sebaliknya, terdapat beberapa kelas yang memperlihatkan penurunan akurasi relatif, khususnya kelas 4, 11, 13, dan 14.

Faktor penyebab penurunan akurasi pada kelas-kelas tersebut dapat berasal dari:

- Kemiripan pola spektral antar pemancar, terutama yang memiliki frekuensi berdekatan.
- Variasi internal sinyal akibat dinamika modulasi atau noise.
- Tumpang tindih fitur yang membuat model sulit membedakan kelas secara konsisten.

Hal ini diperlihatkan secara jelas dalam confusion matrix dimana kelas 4, 13, dan 14 pada semua model menunjukkan tingkat kesalahan prediksi yang relatif lebih tinggi dibanding kelas lainnya.

Nilai macro average dan weighted average pada semua model berada pada interval 0,98–0,99, yang menunjukkan bahwa:

- Model tidak bias terhadap kelas tertentu,
- Performa seragam di seluruh kelas, karena dataset yang digunakan seimbang,
- Model memiliki stabilitas prediksi yang baik meskipun arsitektur berbeda.

CNN-BiLSTM memiliki konsistensi terbaik karena seluruh kelas memiliki skor di atas 0.97 dengan variasi kesalahan yang sangat kecil.

Perbedaan ini relatif kecil sehingga semua model dapat dikategorikan efektif untuk tugas identifikasi pemancar radio FM. Namun, jika ditinjau dari perspektif aplikasi real-time monitoring, trade-off antara akurasi dan kecepatan inferensi menjadi faktor utama dalam pemilihan model. CNN 5-Layers menempati posisi ideal dengan akurasi yang kompetitif dan waktu inferensi yang sangat singkat, menjadikannya kandidat kuat untuk implementasi praktis pada sistem pengawasan spektrum.

Hasil ini sejalan dengan literatur sebelumnya, di mana CNN terbukti unggul dalam mengekstraksi pola spasial sinyal radio secara efisien. CNN-BiLSTM menawarkan kelebihan dalam menangkap dinamika temporal sinyal, namun dengan biaya komputasi yang lebih tinggi. Sementara itu, Transformer memiliki potensi dalam menangkap relasi global antar fitur, tetapi efisiensinya terbatas dalam kasus input berdimensi tinggi seperti data IQ. Dengan demikian, penelitian ini menegaskan pentingnya mempertimbangkan keseimbangan antara akurasi, kompleksitas model, dan kecepatan inferensi dalam pemilihan arsitektur deep learning untuk pengawasan spektrum frekuensi radio.

Dengan demikian, dapat disimpulkan bahwa jumlah parameter, waktu pelatihan, epoch optimal, waktu inferensi, dan akurasi merupakan faktor yang saling melengkapi tetapi tidak selalu berkorelasi langsung.

Dalam rangka mengevaluasi kelayakan implementasi model klasifikasi berbasis deep learning untuk monitoring spektrum frekuensi radio secara real-time,

penting untuk membandingkan waktu klasifikasi model dengan waktu pengambilan data oleh perangkat penerima. Perangkat yang digunakan dalam penelitian ini adalah RTL-SDR, yang memiliki spesifikasi laju pengambilan data sebesar 2.048 mega-samples per second (MSps). Dalam eksperimen, satu batch data IQ disusun dari 16 blok data dengan masing-masing berisi 256 sampel, sehingga total terdapat 4096 sampel per batch.

Waktu yang dibutuhkan RTL-SDR untuk mengumpulkan satu batch data IQ adalah  $2 \times 10^{-3}$  detik (2 ms) dihitung berdasarkan persamaan (1).

$$\text{waktu total pengambilan} = \frac{\text{jumlah sampel}}{\text{sampel\_rate}} \quad (1)$$

Dengan demikian, sistem monitoring real-time menuntut agar proses klasifikasi per batch oleh model tidak melebihi  $2 \times 10^{-3}$  detik agar dapat mengejar kecepatan akuisisi data dan mencegah backlog pada sistem.

$$\text{waktu per batch} = \frac{\text{waktu total}}{\text{jumlah sampel}} \quad (2)$$

Tabel 4.12. Perbandingan Waktu Klasifikasi per Batch

Model	Waktu Klasifikasi per Batch (s)
CNN-BiLSTM	$3,876 \times 10^{-3}$
CNN-Transformer	$7,541 \times 10^{-3}$
CNN 5-Layers	$0,64 \times 10^{-3}$

Hasil pengujian waktu klasifikasi per batch menunjukkan adanya perbedaan signifikan antar model yang diuji. Model CNN-BiLSTM membutuhkan waktu sekitar  $3,876 \times 10^{-3}$  detik per batch, yang relatif efisien untuk model berbasis LSTM mengingat adanya proses komputasi dua arah pada lapisan BiLSTM. Hal ini menunjukkan bahwa meskipun arsitektur BiLSTM cenderung kompleks, optimisasi

dalam implementasi masih memungkinkan model ini digunakan pada skenario klasifikasi dengan kebutuhan waktu tanggap menengah.

Sebaliknya, model CNN-Transformer memiliki waktu klasifikasi per batch yang paling tinggi, yaitu  $7,541 \times 10^{-3}$  detik. Nilai ini hampir dua kali lipat lebih lambat dibandingkan CNN-BiLSTM. Penyebab utama keterlambatan ini adalah kompleksitas mekanisme multi-head attention yang harus menghitung interaksi antar seluruh posisi dalam sekuens input. Dengan demikian, meskipun CNN-Transformer unggul dalam menangkap hubungan global, beban komputasi menjadi faktor pembatas jika model ini diterapkan pada sistem real-time dengan keterbatasan sumber daya komputasi.

Di sisi lain, model CNN 5-Layers menunjukkan kinerja yang sangat efisien dengan waktu klasifikasi per batch hanya  $0,64 \times 10^{-3}$  detik, jauh lebih cepat dibandingkan dua model lainnya. Hal ini menegaskan bahwa arsitektur CNN murni dengan lima lapisan konvolusi dapat menjadi solusi praktis untuk aplikasi yang memerlukan klasifikasi cepat dengan latensi sangat rendah. Walaupun akurasi sedikit di bawah CNN-BiLSTM, keunggulan dari segi kecepatan membuat model ini lebih layak digunakan dalam implementasi nyata, terutama pada perangkat monitoring yang bekerja secara berkelanjutan dan real-time.

Dengan demikian, analisis ini memperlihatkan bahwa dalam konteks pemilihan model, trade-off antara kecepatan dan kompleksitas arsitektur menjadi faktor krusial. CNN-BiLSTM menawarkan akurasi tinggi dengan waktu moderat, CNN-Transformer unggul dalam generalisasi namun terhambat oleh waktu

klasifikasi yang lama, sedangkan CNN 5-Layers lebih efisien dari sisi waktu meskipun dengan sedikit kompromi pada akurasi.

Temuan penelitian ini menunjukkan bahwa pemancar siaran FM memiliki pola karakteristik sinyal IQ yang konsisten dan dapat dikenali secara akurat oleh model deep learning. Hasil tersebut memberikan implikasi penting terhadap mekanisme pengawasan spektrum frekuensi radio:

1. Kemampuan model dalam mengidentifikasi pemancar secara otomatis membuka peluang penerapan sistem monitoring berbasis kecerdasan buatan yang mampu melakukan klasifikasi sinyal secara real-time tanpa ketergantungan pada observasi manual oleh operator. Dengan demikian, proses identifikasi pemancar dapat berlangsung lebih cepat, akurat, dan minim kesalahan subjektif.
2. Model mampu membedakan pemancar yang sah, sistem ini berpotensi digunakan sebagai alat deteksi dini terhadap keberadaan pemancar tidak berizin atau sinyal yang tidak dikenali oleh basis data model. Hal ini dapat meningkatkan efektivitas penertiban serta memperkuat mekanisme pengawasan berbasis bukti digital.
3. Integrasi model ke dalam sistem monitoring nasional memungkinkan peningkatan efisiensi operasional, karena operator tidak lagi harus melakukan identifikasi secara visual, tetapi beralih pada fungsi analitis untuk interpretasi hasil dan tindak lanjut lapangan. Selain itu, performa model yang stabil menunjukkan bahwa pendekatan berbasis pola sinyal IQ dapat menjadi fondasi bagi pengembangan sistem pengawasan spektrum

yang adaptif, termasuk deteksi anomali pemancar, pemetaan fingerprint sinyal, dan pemantauan kualitas siaran dalam jangka panjang.

Selain berbagai capaian positif, penelitian ini juga memiliki sejumlah keterbatasan yang perlu dicermati untuk memahami ruang perbaikan yang masih terbuka. Salah satu kelemahan utama terletak pada keterbatasan variasi data, karena seluruh sinyal IQ diperoleh hanya dari satu lokasi UPT dan menggunakan satu jenis perangkat penerima RTL-SDR.

Kondisi ini membuat model rentan mengalami penurunan performa apabila diterapkan pada lingkungan spektrum yang berbeda, menggunakan perangkat yang berbeda sensitivitasnya, atau menghadapi kondisi kanal yang lebih kompleks. Selain itu, beberapa proses interpretasi internal model—seperti visualisasi feature map CNN, hidden state BiLSTM, dan attention weight Transformer—masih menunjukkan pola yang sulit ditafsirkan secara langsung, sehingga belum mampu memberikan penjelasan causal mengenai keputusan klasifikasi yang dibuat oleh model. Penelitian ini juga belum berhasil mengeksplorasi secara tuntas pengaruh noise lingkungan dan interferensi kuat terhadap stabilitas prediksi, karena tidak semua kondisi spektrum direkam secara representatif.

Dari sisi komputasi, model CNN-Transformer menunjukkan waktu inferensi yang relatif lambat sehingga tidak sepenuhnya memenuhi syarat real-time, dan upaya optimasi seperti pruning atau quantization belum sempat diuji dalam penelitian ini. Kekurangan-kekurangan tersebut terutama dipengaruhi oleh keterbatasan sumber daya perangkat, cakupan data yang terbatas, serta waktu penelitian yang tidak memungkinkan dilakukannya eksperimen tambahan seperti

pengujian cross-device, cross-location, atau integrasi model dalam sistem streaming IQ secara langsung. Dengan memahami faktor-faktor penyebab ini, diharapkan penelitian lanjutan dapat mengatasi keterbatasan tersebut sehingga model yang dikembangkan menjadi lebih robust, interpretatif, dan kompatibel untuk implementasi operasional di lapangan.

#### **4.6 Pemilihan Model Terbaik**

Setelah dilakukan analisis terhadap hasil pelatihan dan evaluasi kinerja ketiga model, langkah selanjutnya adalah membahas pemilihan model yang paling sesuai untuk diimplementasikan dalam sistem monitoring radio FM secara real-time. Pemilihan model tidak hanya didasarkan pada tingkat akurasi, tetapi juga mempertimbangkan efisiensi komputasi, waktu inferensi, serta kemampuan generalisasi model terhadap data uji. Faktor-faktor tersebut sangat penting karena monitoring spektrum frekuensi radio memerlukan sistem yang mampu beroperasi secara cepat, akurat, dan berkelanjutan, sehingga dapat mendeteksi dan mengidentifikasi pemancar radio FM dengan efektif.

Dalam konteks ini, setiap model yang diuji memiliki karakteristik unik dengan kelebihan dan kekurangan masing-masing. CNN 5-Layers unggul dari sisi efisiensi inferensi, CNN-BiLSTM menonjol dalam hal akurasi tertinggi berkat kombinasi fitur spasial dan temporal, sedangkan CNN-Transformer menawarkan keunggulan dalam generalisasi melalui mekanisme self-attention. Dengan membandingkan ketiga model ini secara menyeluruh, akan diperoleh gambaran yang lebih jelas mengenai model yang paling tepat untuk mendukung implementasi monitoring radio FM secara real-time.

#### **4.6.1. Kelebihan dan Kekurangan CNN-BiLSTM Hasil Penelitian**

CNN-BiLSTM berhasil mencapai akurasi tertinggi (98,96%) di antara ketiga model. Hal ini disebabkan kombinasi kekuatan CNN dalam mengekstraksi fitur spasial dari sinyal IQ dan BiLSTM dalam memodelkan dependensi temporal dua arah. Kelebihan ini menjadikan CNN-BiLSTM unggul dalam hal akurasi klasifikasi, terutama ketika sinyal memiliki variasi temporal yang signifikan. Namun, kelemahan utama terletak pada efisiensi komputasi, baik dalam pelatihan maupun inferensi. Waktu per epoch yang tinggi (840–1000 detik) serta waktu inferensi (62,02 detik) menunjukkan bahwa model ini relatif lambat untuk implementasi real-time pada perangkat monitoring dengan sumber daya terbatas.

#### **4.6.2. Kelebihan dan Kekurangan CNN-Transformer Hasil Penelitian**

CNN-Transformer menghadirkan pendekatan modern yang memanfaatkan mekanisme self-attention untuk menangkap relasi global antar fitur. Kelebihannya adalah kemampuan generalisasi yang baik dengan gap akurasi pelatihan dan validasi paling kecil, sehingga model ini lebih tahan terhadap overfitting. Namun, CNN-Transformer memiliki dua kelemahan penting. Pertama, waktu inferensi terlalu lama (120,66 detik), akibat kompleksitas kuadratik dari operasi self-attention terhadap panjang sekuens input. Kedua, meskipun parameter relatif kecil (37.232), akurasinya (97,72%) justru sedikit lebih rendah dibanding dua model lainnya. Kelemahan dalam efisiensi waktu membuat model ini kurang ideal untuk implementasi real-time, tetapi potensial untuk riset lanjutan yang berfokus pada generalisasi.

#### 4.6.3. Kelebihan dan Kekurangan CNN 5-Layers Hasil Penelitian

Model CNN 5-Layers menunjukkan keunggulan dalam hal efisiensi komputasi. Meskipun memiliki jumlah parameter paling besar (1.074.912), model ini mampu memberikan waktu inferensi tercepat (10,24 detik) berkat sifat operasi konvolusi yang sangat teroptimasi pada perangkat keras modern. Dari sisi akurasi, model ini juga kompetitif (98,18%) dan hanya sedikit di bawah CNN-BiLSTM. Kelemahan utama CNN 5-Layers adalah kurangnya kemampuan untuk menangkap dependensi temporal maupun relasi global pada sinyal, karena arsitekturnya murni spasial. Namun demikian, untuk kebutuhan monitoring radio FM secara real-time, efisiensi waktu inferensi membuat model ini menjadi kandidat yang sangat kuat untuk implementasi praktis.

Berdasarkan kelebihan dan kekurangan masing-masing model, CNN 5-Layers merupakan pilihan paling tepat untuk diaplikasikan dalam sistem monitoring radio FM secara real-time. Hal ini karena model ini menawarkan keseimbangan terbaik antara akurasi yang tinggi dan waktu inferensi yang sangat efisien, dua faktor krusial dalam konteks pemantauan spektrum frekuensi radio yang membutuhkan respons cepat dan kontinuitas pengawasan.

## BAB 5 PENUTUP

### 5.1 Kesimpulan

Penelitian ini menunjukkan bahwa klasifikasi pemancar radio FM berbasis data IQ dapat dilakukan secara efektif menggunakan pendekatan deep learning. Tiga model yang diusulkan, yakni CNN 5-Layers, CNN-BiLSTM, dan CNN-Transformer, telah diuji dan dibandingkan berdasarkan kinerja akurasi, F1-score, serta efisiensi waktu inferensi. Hasil penelitian memperlihatkan bahwa ketiga model mampu mencapai tingkat akurasi yang tinggi, dengan variasi kinerja yang ditentukan oleh kompleksitas arsitektur masing-masing.

CNN-BiLSTM mampu menggabungkan kekuatan ekstraksi fitur spasial dari CNN dengan pemodelan dependensi temporal melalui BiLSTM, sehingga memberikan akurasi yang sangat tinggi, meskipun dengan kebutuhan waktu inferensi yang relatif lebih lama. CNN-Transformer menunjukkan performa yang stabil dengan generalisasi yang baik, ditunjukkan oleh gap yang sangat kecil antara akurasi pelatihan dan validasi. Hal ini mengindikasikan bahwa CNN-Transformer cukup tangguh terhadap overfitting, meskipun memerlukan waktu komputasi yang lebih tinggi. Di sisi lain, CNN 5-Layers terbukti paling efisien dari sisi komputasi, dengan waktu inferensi yang jauh lebih cepat, namun tetap kompetitif dalam hal akurasi dan F1-score.

Berdasarkan temuan tersebut, dapat disimpulkan bahwa pemilihan model untuk implementasi nyata harus mempertimbangkan keseimbangan antara akurasi

dan efisiensi. CNN 5-Layers muncul sebagai kandidat terkuat untuk penerapan sistem monitoring spektrum secara real-time, khususnya dalam kondisi keterbatasan sumber daya komputasi. Sementara itu, CNN-BiLSTM dan CNN-Transformer dapat lebih sesuai diterapkan pada skenario di mana akurasi menjadi prioritas utama, meskipun dengan konsekuensi waktu proses yang lebih besar.

## 5.2 Saran

Berdasarkan hasil penelitian ini, terdapat beberapa saran yang dapat menjadi arahan bagi penelitian selanjutnya. Pertama, untuk meningkatkan efisiensi model, riset lanjutan dapat mengadopsi arsitektur deep learning yang lebih ringan seperti MobileNet, EfficientNet, atau model berbasis TinyML yang dirancang khusus untuk perangkat edge. Pendekatan ini dapat memungkinkan penerapan sistem monitoring spektrum pada perangkat dengan kapasitas komputasi terbatas, seperti embedded system atau Internet of Things (IoT).

Kedua, diperlukan eksplorasi terhadap teknik optimisasi model, seperti model pruning, quantization, dan knowledge distillation, yang dapat mengurangi ukuran model sekaligus mempercepat waktu inferensi tanpa mengorbankan akurasi secara signifikan. Hal ini penting untuk memastikan keberlanjutan implementasi sistem dalam skala besar.

Ketiga, penelitian mendatang dapat diperluas dengan menguji kinerja model pada variasi lingkungan spektrum yang lebih kompleks, misalnya dengan memasukkan data dari wilayah berbeda, kondisi interferensi yang lebih tinggi, atau variasi intensitas sinyal. Evaluasi tersebut akan memperkaya pemahaman tentang kemampuan generalisasi model.

Keempat, integrasi metode hybrid antara CNN dan algoritma lain, seperti Reinforcement Learning (RL) untuk manajemen spektrum dinamis, juga dapat menjadi arah penelitian potensial yang mampu memberikan solusi lebih adaptif terhadap dinamika penggunaan frekuensi radio.

Dengan demikian, penelitian ini diharapkan tidak hanya memberikan kontribusi pada pengembangan teknologi monitoring spektrum berbasis deep learning, tetapi juga menjadi pijakan untuk inovasi lebih lanjut dalam membangun sistem yang adaptif, efisien, dan andal untuk kebutuhan nyata di bidang pengelolaan spektrum frekuensi radio.



## DAFTAR PUSTAKA

- [1] Kementerian Komunikasi dan Informatika, *Data Statistik SDPPI 2023 Direktorat Jenderal Sumber Daya dan Perangkat Pos dan Informatika*. Bogor: IPB Press, 2023. [Online]. Available: <https://ipbpress.my.id/product/pasar-papan-apik-sadean-angsalan-rejekeni-60883-73393-61890-84301-48271-56650-39483-76621-78816-15456-35732-55031-30176-33242-15996-20757-2408-85453-86877-47359-59252-13545-15716-72316-5443-44191-95392-99774-81185-67021-11720-61651-2465-83115-58258-16836-54476-65487-3954-62174-54-86008-23254-88440-87350-51107-7465-40381-12192-15063-66122-24589-41115-68032-13080-41215-12709-27882>
- [2] A. D. Gultom, "Evaluasi Sistem Monitoring dan Penertiban Frekuensi dan Perangkat Telekomunikasi Evaluation of Monitoring and Controlling System of Frequency and Telecommunication Devices," *Bul. Pos dan Telekomun.*, vol. 12, no. 1, pp. 15–28, 2014.
- [3] T. C. C. Timothy J O'Shea, Johnathan Corgan, "Convolutional Radio Modulation Recognition Networks," no. Feb, 2016.
- [4] S. Riyaz, K. Sankhe, S. Ioannidis, and K. Chowdhury, "Deep Learning Convolutional Neural Networks for Radio Identification," *IEEE Commun. Mag.*, vol. 56, no. Sept, pp. 146–152, 2018, doi: 10.1109/MCOM.2018.1800153.
- [5] H. Taub and D. L. Schilling, *Principles of Communication Systems*. in McGraw-Hill electrical and electronic engineering series. McGraw-Hill, 1986. [Online]. Available: <https://books.google.co.id/books?id=leFSAAAAMAAJ>
- [6] L. J. Wong, W. C. Headley, and A. J. Michaels, "Specific Emitter Identification Using Convolutional Neural Network-Based IQ Imbalance Estimators," *IEEE Access*, vol. 7, pp. 33544–33555, 2019, doi: 10.1109/ACCESS.2019.2903444.
- [7] M. Ramasubramanian, C. Banerjee, D. Roy, E. Pasilio, and T. Mukherjee, "Exploiting Spatio-Temporal Properties of I/Q Signal Data Using 3D Convolution for RF Transmitter Identification," *IEEE J. Radio Freq. Identif.*, vol. 5, no. 2, pp. 113–127, 2021, doi: 10.1109/JRFID.2021.3051901.

- [8] S. Kunze and B. Saha, "Drone Classification with a Convolutional Neural Network Applied to Raw IQ Data," in *2022 3rd URSI Atlantic and Asia Pacific Radio Science Meeting (AT-AP-RASC)*, 2022, pp. 1–4. doi: 10.23919/AT-AP-RASC54737.2022.9814170.
- [9] S. Gao, "Alzheimer's disease diagnosis via 5-layer Convolutional Neural Network and Data Augmentation," *EAI Endorsed Trans.*, vol. 7, no. 23, pp. 1–13, 2022.
- [10] S. Ghosal and K. Sarkar, "Rice Leaf Diseases Classification Using CNN With Transfer Learning," in *2020 IEEE Calcutta Conference (CALCON)*, 2020, pp. 230–236. doi: 10.1109/CALCON49167.2020.9106423.
- [11] M. Méndez, M. G. Merayo, and M. Núñez, "Long-term traffic flow forecasting using a hybrid CNN-BiLSTM model," *Eng. Appl. Artif. Intell.*, vol. 121, no. December 2022, p. 106041, 2023, doi: 10.1016/j.engappai.2023.106041.
- [12] M. Marjani, M. Mahdianpari, and F. Mohammadimanesht, "CNN-BiLSTM: A Novel Deep Learning Model for Near-Real-Time Daily Wildfire Spread Prediction," *Remote Sens.*, vol. 16, no. 8, p. 1467, 2024, doi: 10.3390/rs16081467.
- [13] W. Zhang, Z. Tan, Q. Lv, J. Li, B. Zhu, and Y. Liu, "An Efficient Hybrid CNN-Transformer Approach for Remote," *Remote Sens.*, vol. 16, no. 5, p. 880, 2024, doi: 10.3390/rs16050880.
- [14] L. Wu *et al.*, "Landslide mapping based on a hybrid CNN-transformer network and deep transfer learning using remote sensing images with topographic and spectral features," *Int. J. Appl. Earth Obs. Geoinf.*, vol. 126, no. February, p. 103612, 2024, doi: 10.1016/j.jag.2023.103612.
- [15] Z. Ke and H. Vikalo, "Real-Time Radio Technology and Modulation Classification via an LSTM Auto-Encoder," *IEEE Trans. Wirel. Commun.*, vol. 21, no. 1, pp. 370–382, 2022, doi: 10.1109/TWC.2021.3095855.
- [16] S. H. Choi and H. Yoon, "Convolutional Neural Networks for the Real-Time Monitoring of Vital Signs Based on Impulse Radio Ultrawide-Band Radar during Sleep," *Sensors (Basel)*, vol. 23, no. 6, Mar. 2023, doi: 10.3390/s23063116.
- [17] S. Oncu, M. Karakaya, Y. Dalveren, A. Kara, and M. Derawi, "Real-Time

Radar Classification Based on Software-Defined Radio Platforms: Enhancing Processing Speed and Accuracy with Graphics Processing Unit Acceleration.," *Sensors (Basel)*, vol. 24, no. 23, Dec. 2024, doi: 10.3390/s24237776.

- [18] J. G. Proakis and D. G. Monolakis, "Digital signal processing: principles, algorithms, and applications," *Pentice Hall*, pp. 1–42, 1996.
- [19] E. G. Sierra and G. A. R. Arroyave, "Low cost SDR spectrum analyzer and analog radio receiver using GNU radio, raspberry Pi2 and SDR-RTL dongle," *2015 7th IEEE Latin-American Conf. Commun. LATINCOM 2015*, pp. 1–6, 2016, doi: 10.1109/LATINCOM.2015.7430125.
- [20] Y. LeCun, Y. Bengio, and G. Hinton, "Deep learning," *Nature*, vol. 521, no. 7553, pp. 436–444, 2015, doi: 10.1038/nature14539.
- [21] I. Goodfellow, Y. Bengio, and A. Courville, *Deep Learning*. MIT Press, 2016.
- [22] A. Gicie, D. Donko, and A. Subasi, "Time Sequence Deep Learning Model for Ubiquitous Tabular Data with Unique 3D Tensors Manipulation," *Entropy*, vol. 26, no. 9, 2024, doi: 10.3390/e26090783.
- [23] L. Wiranda and M. Sadikin, "Penerapan Long Short Term Memory Pada Data Time Series Untuk Memprediksi Penjualan Produk PT . Metiska Farma," *J. Nas. Pendidik. Tek. Inform.*, vol. 8, no. Desember, pp. 184–196, 2019.
- [24] F. Yanti, B. N. Sari, and S. Defiyanti, "Implementasi Algoritma LSTM Pada Peramalan Stok Obat ( Studi Kasus : Puskesmas Beber )," *J. Mhs. Tek. Inform.*, vol. 8, no. 4, pp. 6082–6089, 2024.
- [25] R. Al Kiramy, I. Permana, and A. Marsal, "Comparison of RNN and LSTM Algorithm Performance in Predicting the Number of Umrah Pilgrims at PT . Hajar Aswad Perbandingan Performa Algoritma RNN dan LSTM dalam Prediksi Jumlah Jamaah Umrah pada PT . Hajar Aswad," *MALCOM Indones. J. Mach. Learn. Comput. Sci.*, vol. 4, no. October, pp. 1224–1234, 2024.
- [26] R. Chetana, A. S. Rao, and K. Mahantesh, "Application of Conv-1D and Bi-LSTM to Classify and Detect Epilepsy in EEG Data," *Int. J. Adv. Comput. Sci. Appl.*, vol. 14, no. 6, pp. 253–261, 2023.

- [27] N. B. Mena, "Application of SA-Conv1D-BiGRU model for streamflow prediction in southern Ethiopia," *Hydrol. Res.*, vol. 55, no. 9, pp. 936–957, 2024, doi: 10.2166/nh.2024.074.
- [28] A. Vaswani *et al.*, "Attention Is All You Need," *IEEE Ind. Appl. Mag.*, vol. 8, no. 1, pp. 8–15, 2017, doi: 10.1109/2943.974352.
- [29] S. Bhojanapalli, A. Chakrabarti, D. Glasner, D. Li, T. Unterthiner, and A. Veit, "Understanding Robustness of Transformers for Image Classification," 2021. [Online]. Available: <https://arxiv.org/abs/2103.14586>
- [30] J. Lu, "Convolutional neural network and vision transformer for image classification," in *Signal Processing and Machine Learning*, 2023, pp. 104–108. doi: 10.54254/2755-2721/5/20230542.
- [31] X. Zhang, X. Wang, Z. Zhang, and Z. Wang, "CNN-Transformer for Microseismic Signal Classification," *Electronics*, vol. 12, p. 2468, 2023, doi: 10.3390/electronics12112468.
- [32] S. Rajendran, S. Member, W. Meert, and I. D. Giustiniano, "Deep Learning Models for Wireless Signal Classification with Distributed Low-Cost Spectrum Sensors," vol. 7731, no. c, pp. 1–13, 2018, doi: 10.1109/TCCN.2018.2835460.
- [33] Zhanming Wen, Weizhen Lin, and Hongwei Liu, "Machine-learning-based approach for anonymous online customer purchase intentions using clickstream data," *Systems*, vol. 11, no. 5, pp. 1–14, 2023.
- [34] J. Samarão, A. Moreno, M. B. Gaspar, and M. M. Rufino, "Improving machine learning predictions to estimate fishing effort using vessel's tracking data," *Ecol. Inform.*, vol. 85, no. December 2024, 2025, doi: 10.1016/j.ecoinf.2024.102953.
- [35] and M. T. I. Kurniawan, Abdussomad, M. F. Akbar, D. F. Saepudin, M. S. Azis, "Improving The Effectiveness of Classification Using The Data Level Approach and Feature Selection Techniques in Online Shoppers Purchasing Intention Prediction Improving The Effectiveness of Classification Using The Data Level Approach and Feature Selecti," *J. Phys. Conf. Ser.*, vol. 1641, 2020, doi: 10.1088/1742-6596/1641/1/012083.