

Progress *Automatic Modulation Classification* Menggunakan CNN+LSTM Standart (Serial) dan Parallel

July 30, 2025

I. Konfigurasi Dataset

A. Skema Modulasi Target

Sistem dirancang untuk mengklasifikasikan **9 jenis modulasi digital** yang berbeda. Setiap jenis modulasi memiliki karakteristik unik dalam representasi sinyal I/Q yang memungkinkan model untuk membedakan antara satu jenis dengan jenis lainnya.

Table 1: Daftar Skema Modulasi Target

Indeks Kelas	Modulasi	Deskripsi
0	OOK	On-Off Keying
1	4ASK	4-ary Amplitude Shift Keying
2	8ASK	8-ary Amplitude Shift Keying
3	BPSK	Binary Phase Shift Keying
4	QPSK	Quadrature Phase Shift Keying
5	8PSK	8-ary Phase Shift Keying
6	16QAM	16-ary Quadrature Amplitude Modulation
7	64QAM	64-ary Quadrature Amplitude Modulation
8	OQPSK	Offset Quadrature Phase Shift Keying

Pemilihan modulasi ini mencakup tiga kategori utama:

- **Amplitude Shift Keying (ASK):** OOK, 4ASK, 8ASK
- **Phase Shift Keying (PSK):** BPSK, QPSK, 8PSK, OQPSK
- **Quadrature Amplitude Modulation (QAM):** 16QAM, 64QAM

B. Karakteristik Data

Dataset yang digunakan memiliki spesifikasi teknis sebagai berikut:

Table 2: Spesifikasi Teknis Dataset

Parameter	Nilai	Keterangan
Jumlah Kelas	9	Jenis modulasi yang diklasifikasi
Kanal Input	2	Komponen I dan Q
Panjang Sekuens	1024	Sampel per sinyal
Format Sinyal	I/Q kompleks	Representasi baseband
Ukuran Batch	512	Sampel per batch training
Jumlah Epoch	300	Iterasi training
Pembagian Data	70-15-15	Train-Validation-Test (%)

note : Range SNR yang digunakan dalam dataset ini adalah -20 dB hingga +30 dB sesuai dengan Karakteristik Dataset RadioML2018.01A

C. Struktur Data I/Q

Setiap sampel data terdiri dari:

- **Komponen I (In-phase):** Bagian real dari sinyal kompleks
- **Komponen Q (Quadrature):** Bagian imajiner dari sinyal kompleks
- **Panjang Sinyal:** 1024 sampel untuk setiap komponen
- **Label Kelas:** Indeks numerik (0-8) untuk jenis modulasi

D. Pembagian Dataset

Dataset dibagi menggunakan strategi **70-15-15** untuk memastikan evaluasi yang robust:

Table 3: Pembagian Dataset per Batch

Subset	Persentase	Sampel per Batch	Tujuan
Training	70%	358	Pembelajaran model
Validation	15%	77	Tuning hyperparameter
Testing	15%	77	Evaluasi akhir
Total	100%	512	Satu batch lengkap

II. Pipeline Pemrosesan Data

A. Representasi Sinyal I/Q

Sinyal modulasi digital dalam sistem komunikasi umumnya direpresentasikan sebagai sinyal kompleks dalam bentuk baseband. Sinyal kompleks $s(t)$ dapat dinyatakan sebagai:

$$s(t) = I(t) + jQ(t) \quad (1)$$

dimana:

- $I(t)$ adalah komponen in-phase (bagian real)
- $Q(t)$ adalah komponen quadrature (bagian imajiner)
- j adalah unit imajiner ($j = \sqrt{-1}$)

B. Ekstraksi Fitur Amplitude dan Phase

Untuk meningkatkan kemampuan model dalam membedakan karakteristik modulasi, sinyal I/Q kompleks dikonversi menjadi representasi amplitude dan phase. Transformasi ini memberikan informasi yang lebih eksplisit tentang karakteristik modulasi.

a. Perhitungan Amplitude

Amplitude sinyal kompleks dihitung menggunakan magnitude Euclidean:

$$A[n] = \sqrt{I[n]^2 + Q[n]^2} \quad (2)$$

dimana $A[n]$ adalah amplitude pada sampel ke- n .

b. Perhitungan Phase

Phase sinyal kompleks dihitung menggunakan fungsi arctangent dua argumen:

$$\phi[n] = \arctan 2(Q[n], I[n]) \quad (3)$$

dimana $\phi[n]$ adalah phase pada sampel ke- n dalam rentang $[-\pi, \pi]$.

C. Transformasi ke Representasi 2D

Untuk memungkinkan pemrosesan menggunakan CNN, sinyal 1D dengan panjang 1024 sampel dikonversi menjadi representasi 2D. Proses transformasi ini dapat dinyatakan sebagai:

$$\mathbf{X}_{2D} = \text{reshape}(\mathbf{x}_{1D}, (H, W)) \quad (4)$$

dimana:

- $\mathbf{x}_{1D} \in \mathbb{R}^{1024}$ adalah sinyal 1D
- $\mathbf{X}_{2D} \in \mathbb{R}^{H \times W}$ adalah representasi 2D
- $H \times W = 1024$ (umumnya $H = W = 32$)

Alasan Pemilihan Pendekatan Amplitude-Phase:

Konversi ke domain amplitude-phase memberikan representasi yang lebih sesuai untuk klasifikasi modulasi karena:

1. **Separasi Informasi Eksplisit:** Amplitude dan phase merupakan parameter fundamental yang membedakan jenis modulasi. Modulasi ASK (Amplitude Shift Keying) terutama mengkodekan informasi pada amplitude, PSK (Phase Shift Keying) pada phase, dan QAM pada kombinasi keduanya. Transformasi ini membuat fitur-fitur pembeda ini menjadi eksplisit dan mudah dipelajari oleh CNN.
2. **Invariansi Rotasi:** Komponen amplitude tidak terpengaruh oleh rotasi phase konstanta yang sering terjadi akibat error sinkronisasi atau offset frekuensi dalam sistem komunikasi nyata. Hal ini membuat model lebih robust terhadap kondisi channel yang bervariasi.
3. **Reduksi Korelasi Antar Fitur:** Dalam representasi I/Q mentah, kedua komponen sering berkorelasi tinggi. Transformasi amplitude-phase menghasilkan fitur yang lebih independen, memudahkan CNN dalam mempelajari pola yang unik untuk setiap modulasi.
4. **Konsistensi Magnitude:** Normalisasi amplitude memberikan stabilitas numerik yang lebih baik dibandingkan raw I/Q yang dapat memiliki variasi magnitude yang besar tergantung pada kondisi channel.

Perbandingan dengan Pendekatan Raw I/Q:

Jika data I/Q mentah langsung di-reshape ke 2D tanpa transformasi, beberapa masalah dapat timbul:

- **Informasi Tersembunyi:** Karakteristik modulasi yang terkodekan dalam magnitude dan phase menjadi implisit dan sulit diekstrak oleh CNN
- **Sensitivitas Noise:** Raw I/Q lebih sensitif terhadap noise dan interferensi
- **Ketidakstabilan Training:** Variasi dinamis range yang besar pada raw I/Q dapat menyebabkan ketidakstabilan selama proses training
- **Kompleksitas Model:** CNN harus belajar melakukan transformasi amplitude-phase secara implisit, meningkatkan kompleksitas dan kebutuhan data training

D. Pipeline Pemrosesan Lengkap

Algoritme pemrosesan data secara lengkap dapat dirangkum dalam langkah-langkah berikut:

Algorithm 1 Pipeline Pemrosesan Sinyal I/Q

```

1: Input:  $\mathbf{I} \in \mathbb{R}^{1024}$ ,  $\mathbf{Q} \in \mathbb{R}^{1024}$ 
2: Output:  $\mathbf{I}_{2D}, \mathbf{Q}_{2D} \in \mathbb{R}^{1 \times H \times W}$ 
3:
4: for  $n = 1$  to  $1024$  do
5:    $A[n] \leftarrow \sqrt{I[n]^2 + Q[n]^2}$  {Hitung amplitude}
6:    $\phi[n] \leftarrow \arctan 2(Q[n], I[n])$  {Hitung phase}
7: end for
8:
9:  $\mathbf{I}_{2D} \leftarrow \text{reshape}(\mathbf{A}, (1, H, W))$  {Amplitude ke 2D}
10:  $\mathbf{Q}_{2D} \leftarrow \text{reshape}(\phi, (1, H, W))$  {Phase ke 2D}
11:
12: return  $\mathbf{I}_{2D}, \mathbf{Q}_{2D}$ 

```

E. Karakteristik Transformasi

Transformasi amplitude-phase memberikan beberapa keunggulan:

Table 4: Perbandingan Representasi Sinyal

Aspek	I/Q Original	Amplitude-Phase
Informasi Magnitude	Implisit	Eksplisit ($\sqrt{I^2 + Q^2}$)
Informasi Phase	Implisit	Eksplisit ($\arctan 2(Q, I)$)
Invariansi Rotasi	Tidak	Ya (pada amplitude)
Separabilitas Fitur	Rendah	Tinggi
Kompatibilitas CNN	Memerlukan preprocessing	Langsung kompatibel

III. *Architecture* Model yang Digunakan

A. Model Standar (Serial)

Model yang diusulkan dalam penelitian ini adalah arsitektur hybrid yang menggabungkan Convolutional Neural Network (CNN) dan Long Short-Term Memory (LSTM) dengan mekanisme Attention. Arsitektur ini dirancang khusus untuk tugas klasifikasi sinyal dengan memanfaatkan dua komponen sinyal, yaitu In-phase (I) dan Quadrature (Q). Secara konseptual, model ini menggunakan dua cabang CNN yang identik untuk memproses sinyal I dan Q secara paralel, mengekstraksi fitur spasial dari masing-masing sinyal. Fitur-fitur ini kemudian diproses lebih lanjut oleh lapisan LSTM untuk menangkap dependensi temporal. Berikut adalah rincian dari setiap tahapan dalam arsitektur model:

- **CNN Branches:** Dua cabang CNN untuk sinyal I dan Q.
- **LSTM Layers:** Dua layer LSTM untuk menangkap dependensi temporal.
- **Attention Mechanism:** Layer perhatian untuk menggabungkan fitur dari kedua cabang.
- **Feature Fusion:** Penggabungan fitur dari cabang I dan Q.
- **Classifier:** Beberapa layer fully connected untuk klasifikasi akhir.

Table 5: Rincian Arsitektur Model CNN-LSTM Serial

Lapisan (Layer)	Dimensi Keluaran (Output Dimension)	Keterangan
		— Cabang CNN (untuk I dan Q) —
Masukan (Input Signals I/Q)	(1, 32, 32)	Representasi sinyal 2D
Convolution2D (64 filter, 3x3)	(64, 32, 32)	Ekstraksi fitur awal
Convolution2D (64 filter, 3x3)	(64, 32, 32)	Ekstraksi fitur lanjutan
MaxPool2D (2x2) + Dropout	(64, 16, 16)	Reduksi dimensi & regularisasi
Convolution2D (128 filter, 3x3)	(128, 16, 16)	Ekstraksi fitur kompleks
Convolution2D (128 filter, 3x3)	(128, 16, 16)	Ekstraksi fitur kompleks
MaxPool2D (2x2) + Dropout	(128, 8, 8)	Reduksi dimensi & regularisasi
AdaptiveAvgPool2D	(128, 8, 8)	Penyesuaian dimensi spasial
— Pemrosesan Sekuensial —		
Reshape untuk LSTM	(64, 128)	Mengubah fitur menjadi sekuens
LSTM (100 unit)	(64, 100)	Mempelajari dependensi temporal
LSTM (50 unit)	(64, 50)	Mempelajari dependensi temporal
Global Attention	50	Pembobotan fitur temporal
— Fusi dan Klasifikasi —		
Fusi Fitur (Penjumlahan I + Q)	50	Menggabungkan fitur dari I dan Q
Dense + LeakyReLU	256	Lapisan terhubung penuh pertama
Dropout	256	Regularisasi
Dense + LeakyReLU	128	Lapisan terhubung penuh kedua
Dropout	128	Regularisasi
Dense + LeakyReLU	64	Lapisan terhubung penuh ketiga
Dense + Softmax (Output)	8	Lapisan klasifikasi akhir

Pada Figure 1 menunjukkan arsitektur model deep learning berbasis sinyal I/Q yang dirancang untuk tugas klasifikasi. Model ini menggunakan pendekatan dual-branch, di mana sinyal In-phase (I) dan Quadrature (Q) diproses secara terpisah melalui dua cabang jaringan yang identik. Masing-masing cabang menerima input berupa citra dua dimensi berukuran 32×32 dan memprosesnya melalui blok ekstraksi fitur berbasis CNN yang terdiri dari dua lapisan Conv2D dengan kernel 3×3 , diikuti oleh operasi Batch Normalization, fungsi aktivasi LeakyReLU, MaxPooling, dan Dropout untuk mengurangi overfitting. Output dari CNN kemudian diubah menjadi bentuk sekuensial dan dimasukkan ke dalam blok LSTM yang terdiri dari dua lapisan LSTM berukuran 100 dan 50 unit. Mekanisme Dropout kembali diterapkan untuk regularisasi. Selanjutnya, diterapkan mekanisme Global Attention untuk memberikan bobot pada bagian penting dari sekuens fitur yang dihasilkan oleh LSTM. Setiap cabang menghasilkan output berupa vektor fitur berdimensi 50.

Keluaran dari kedua cabang kemudian digabungkan melalui operasi penjumlahan elemen demi elemen (element-wise addition). Hasil dari proses fusi ini selanjutnya dimasukkan ke dalam blok classifier yang terdiri dari beberapa lapisan fully connected (Linear), masing-masing berukuran $50 \rightarrow 256$, $256 \rightarrow 128$, $128 \rightarrow 64$, dan terakhir $64 \rightarrow$ Jumlah Kelas. Proses ini menghasilkan output akhir berupa logits dengan dimensi sesuai jumlah kelas target. Arsitektur ini dirancang untuk menggabungkan keunggulan dari ekstraksi fitur spasial oleh CNN, pemahaman konteks temporal

melalui LSTM, serta peningkatan fokus terhadap fitur penting melalui attention mechanism, sehingga diharapkan mampu menghasilkan performa klasifikasi yang lebih akurat terhadap sinyal I/Q.

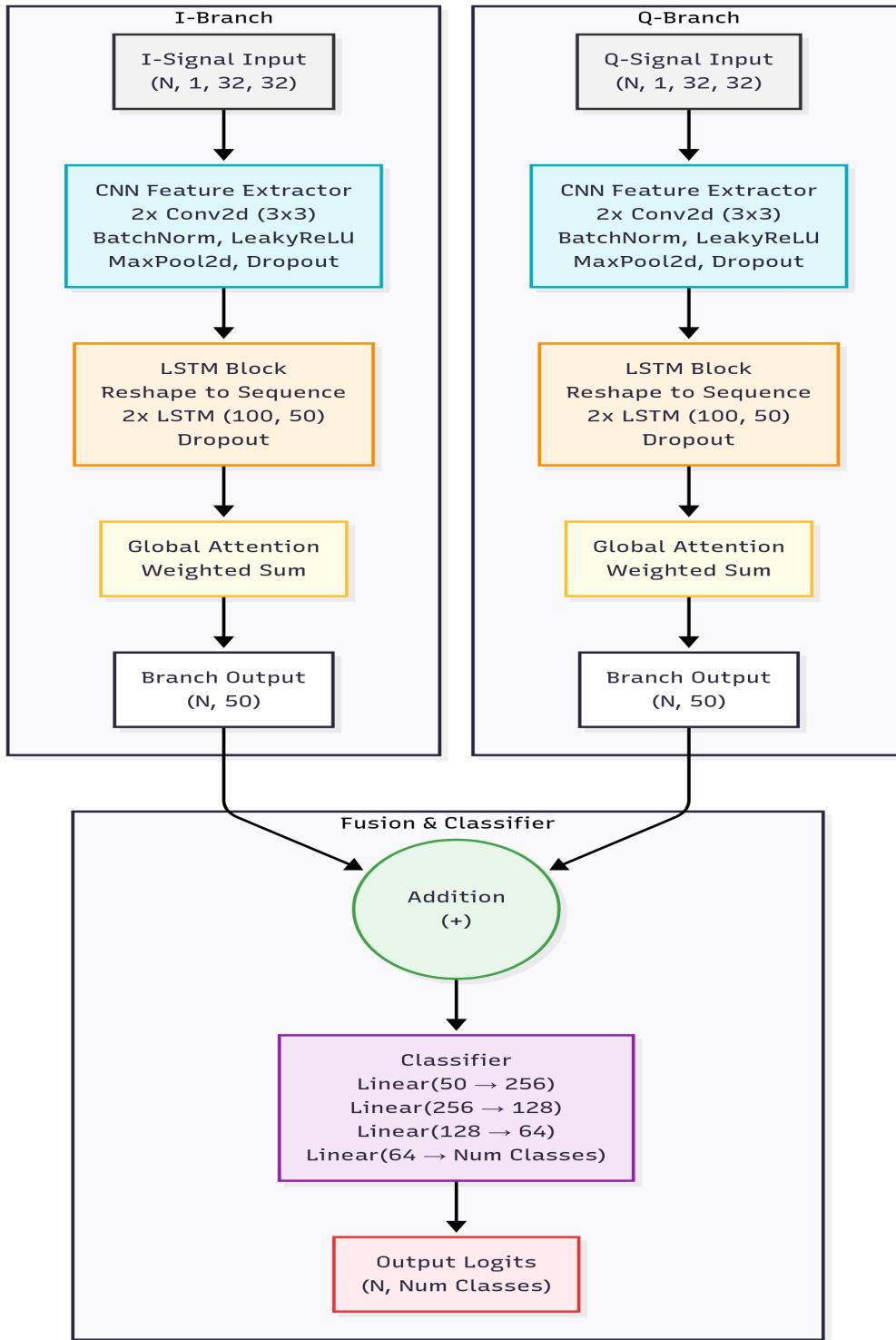


Figure 1: Arsitektur model dual-branch CNN-LSTM dengan mekanisme global attention untuk pemrosesan sinyal I/Q. Model terdiri dari dua cabang paralel (I-Branch dan Q-Branch) yang masing-masing mengolah sinyal spasial dan temporal, kemudian digabungkan dan diklasifikasikan melalui fully connected layers.

B. Model Paralel CNN-LSTM

Model yang digunakan adalah model CNN-LSTM dengan arsitektur paralel, paralel yang dimaksud adalah membagi proses feature extraction menjadi dua cabang CNN yang berbeda yaitu deep dan shallow ini dapat di lihat dari Figure 2. arsitektur-paralel memiki komponen utama sebagai berikut:

- **Cabang CNN Paralel:** Dua jalur CNN dengan kedalaman berbeda - CNN_1 (shallow) dan CNN_2 (deep) untuk mengekstrak fitur multi-skala dari sinyal I dan Q.
- **CNN_1 (Shallow):** Jalur CNN dangkal dengan 2 layer konvolusi ($64 \rightarrow 128$ filter) untuk menangkap fitur dasar.
- **CNN_2 (Deep):** Jalur CNN dalam dengan 4 layer konvolusi ($64 \rightarrow 128 \rightarrow 256 \rightarrow 256$ filter) untuk menangkap fitur kompleks.
- **Feature Concatenation:** Penggabungan fitur dari kedua cabang CNN menggunakan metode concatenation ($128 + 256 = 384$ fitur).
- **LSTM Layers:** Dua layer LSTM ($384 \rightarrow 256 \rightarrow 128$) untuk menangkap dependensi temporal dari fitur yang telah digabungkan.
- **Dual Branch Processing:** Setiap sinyal I dan Q diproses secara terpisah melalui arsitektur yang sama.
- **Average Fusion:** Penggabungan hasil klasifikasi dari cabang I dan Q menggunakan metode rata-rata.
- **Classifier:** Layer dense untuk klasifikasi akhir dengan 9 kelas output.

Table 6: Rincian Arsitektur Model CNN-LSTM Paralel

Lapisan (Layer)	Dimensi Keluaran (Output Dimension)	Keterangan
— Input Sinyal —		
Masukan (Input Signals I/Q)	(1, 32, 32)	Representasi sinyal 2D
— Cabang CNN_1 (Shallow) —		
Convolution2D (64 filter, 3×3)	(64, 32, 32)	Ekstraksi fitur awal
Convolution2D (128 filter, 3×3)	(128, 32, 32)	Ekstraksi fitur lanjutan
MaxPool2D (2×2) + Dropout	(128, 16, 16)	Reduksi dimensi & regularisasi
AdaptiveAvgPool2D	(128, 1, 1)	Global average pooling
Flatten	128	Linearisasi fitur
— Cabang CNN_2 (Deep) —		
Convolution2D (64 filter, 3×3)	(64, 32, 32)	Ekstraksi fitur awal
Convolution2D (128 filter, 3×3)	(128, 32, 32)	Ekstraksi fitur menengah
MaxPool2D (2×2) + Dropout	(128, 16, 16)	Reduksi dimensi & regularisasi
Convolution2D (256 filter, 3×3)	(256, 16, 16)	Ekstraksi fitur kompleks
Convolution2D (256 filter, 3×3)	(256, 16, 16)	Ekstraksi fitur kompleks
MaxPool2D (2×2) + Dropout	(256, 8, 8)	Reduksi dimensi & regularisasi
AdaptiveAvgPool2D	(256, 1, 1)	Global average pooling
Flatten	256	Linearisasi fitur
— Fusi Fitur & Pemrosesan LSTM —		
Concatenation (CNN_1 + CNN_2)	384	Penggabungan fitur paralel
Reshape untuk LSTM	(1, 384)	Mengubah untuk input LSTM
LSTM (256 unit)	(1, 256)	Mempelajari dependensi temporal
Dropout	(1, 256)	Regularisasi
LSTM (128 unit)	(1, 128)	Mempelajari dependensi temporal
Last Time Step	128	Ekstraksi fitur temporal terakhir
Dense (Output per cabang)	9	Klasifikasi per cabang I/Q
— Fusi Cabang I/Q —		
I-Branch Output	9	Hasil klasifikasi cabang I
Q-Branch Output	9	Hasil klasifikasi cabang Q
Average Fusion	9	Rata-rata dari kedua cabang
Softmax (Output Final)	9	Klasifikasi akhir

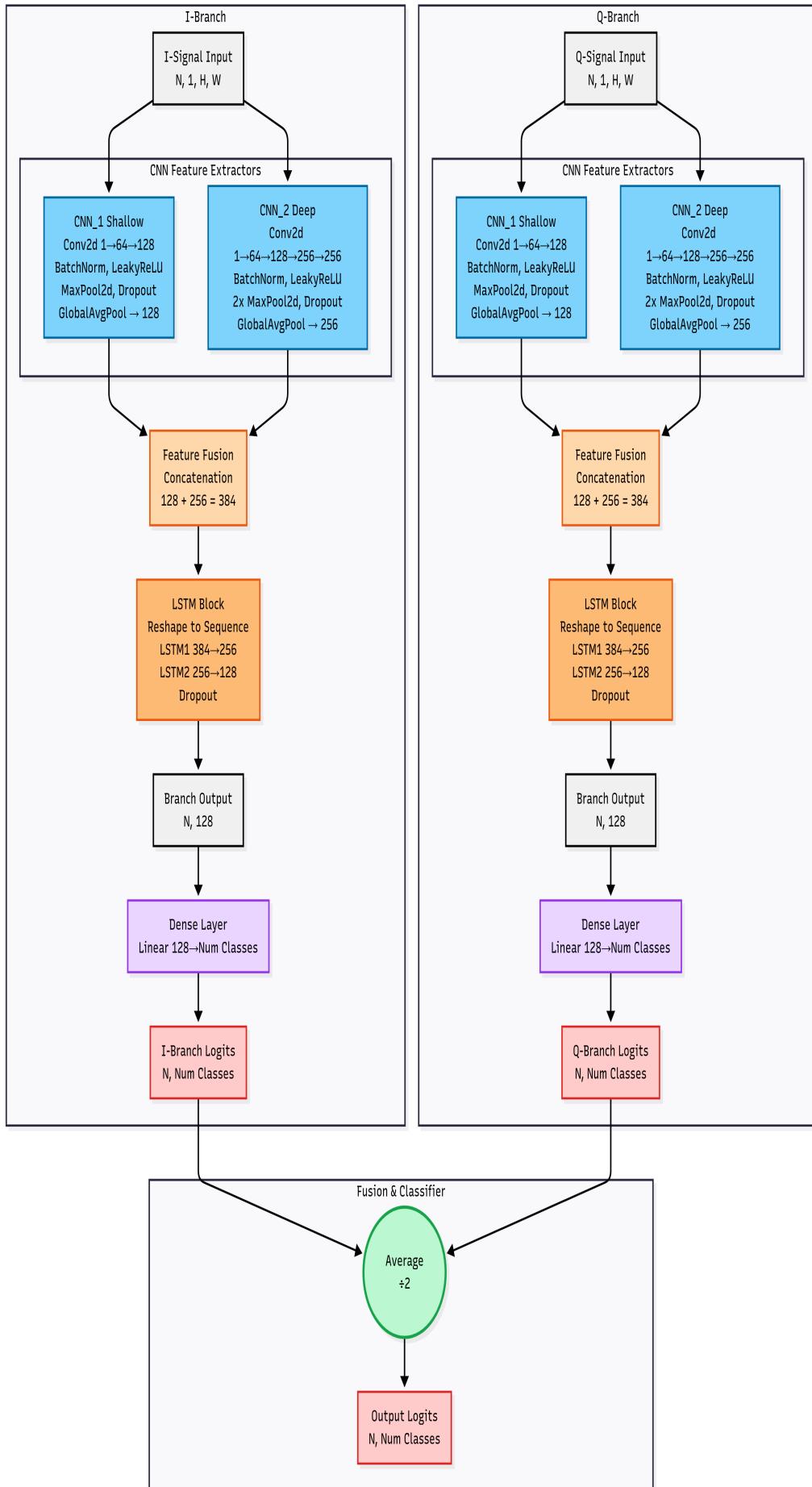


Figure 2: Arsitektur Model Paralel CNN-LSTM yang menggabungkan dua cabang CNN dengan kedalaman berbeda untuk ekstraksi fitur multi-skala, diikuti oleh pemrosesan LSTM dan penggabungan hasil klasifikasi.

Pada Gambar 2 ditampilkan arsitektur dari model CNN-LSTM paralel yang memiliki dua percabangan, yaitu cabang-I dan cabang-Q, di mana keduanya mengikuti jalur pemrosesan yang identik. Model paralel ini dirancang untuk membagi proses ekstraksi fitur menjadi dua jalur CNN yang berbeda, yaitu jalur *shallow* dan *deep*. Setiap cabang menerima input sinyal berdimensi $N \times 1 \times H \times W$, dan menggunakan dua jenis ekstraktor fitur CNN: CNN *shallow* (CNN_1) yang memproses sinyal melalui jalur konvolusi tunggal ($1 \rightarrow 64 \rightarrow 128$ saluran) dengan *Batch Normalization*, aktivasi LeakyReLU, *MaxPooling*, dan *Dropout*, sebelum akhirnya dilakukan *Global Average Pooling* untuk menghasilkan fitur berdimensi 128. Sedangkan CNN *deep* (CNN_2) menggunakan arsitektur konvolusi yang lebih kompleks ($1 \rightarrow 64 \rightarrow 128 \rightarrow 256 \rightarrow 256$ saluran) dengan dua operasi *MaxPooling*, menghasilkan fitur berdimensi 256.

Output dari kedua ekstraktor fitur digabungkan melalui proses konkatenasi, menciptakan vektor fitur berdimensi 384 ($128 + 256$) yang merepresentasikan kombinasi informasi spasial dangkal dan mendalam dari sinyal input. Representasi gabungan ini kemudian diproses melalui blok LSTM, yang terlebih dahulu membentuk ulang data ke format sekuensial, kemudian menerapkan dua lapisan LSTM ($384 \rightarrow 256 \rightarrow 128$) disertai *Dropout* untuk regularisasi, guna menangkap ketergantungan temporal. Hasil akhirnya berupa output cabang berdimensi $N \times 128$.

Setiap cabang diakhiri dengan lapisan linier (*dense layer*) yang memetakan vektor fitur berdimensi 128 ke jumlah kelas target, menghasilkan *logits* spesifik cabang. Pada tahap akhir, hasil prediksi dari cabang-I dan cabang-Q digabungkan melalui operasi rata-rata untuk menghasilkan *output logits* akhir. Strategi ini memungkinkan penggabungan informasi pelengkap yang dipelajari secara paralel oleh jalur ekstraksi fitur *shallow* dan *deep*.

IV. Hasil dan Evaluasi Model

A. Parameter yang Digunakan untuk Evaluasi

Parameter yang digunakan ketika mengevaluasi model adalah sebagai berikut:

- **Batch Size:** 512
- **Epoch:** 300
- **Learning Rate:** 0.0015
- **Weight Decay:** 5e-4
- **Scheduler:** Cosine Annealing
- **Loss Function:** Cross Entropy Loss
- **Optimizer:** Adam Weight
- **Metrics:** Accuracy
- **Early Stopping:** 84 epoch dengan akurasi terbaik
- **Best Model:** Model terbaik disimpan berdasarkan akurasi validasi dan testing tertinggi

B. Hasil Akurasi Model

Hasil akurasi model pada dataset RadioML2018.01A adalah sebagai berikut:

a. Matrix Confusion Kedua Model

Pada Figure 3 terlihat bahwa Model Paralel (kanan) menunjukkan peningkatan akurasi pada beberapa kelas modulasi dibandingkan Model Standar (kiri). Gambar tersebut yang digunakan untuk mengevaluasi kinerja dua model klasifikasi yang berbeda (Standar dan Paralel) dalam mengenali berbagai jenis modulasi sinyal digital (OOK, BPSK, 8PSK, dll.).

- **Sumbu Vertikal (True Label):** Label atau kelas yang sebenarnya dari sinyal modulasi.
- **Sumbu Horizontal (Predicted Label):** Label atau kelas yang diprediksi oleh model.

Test Set Confusion Matrices (Percentages)

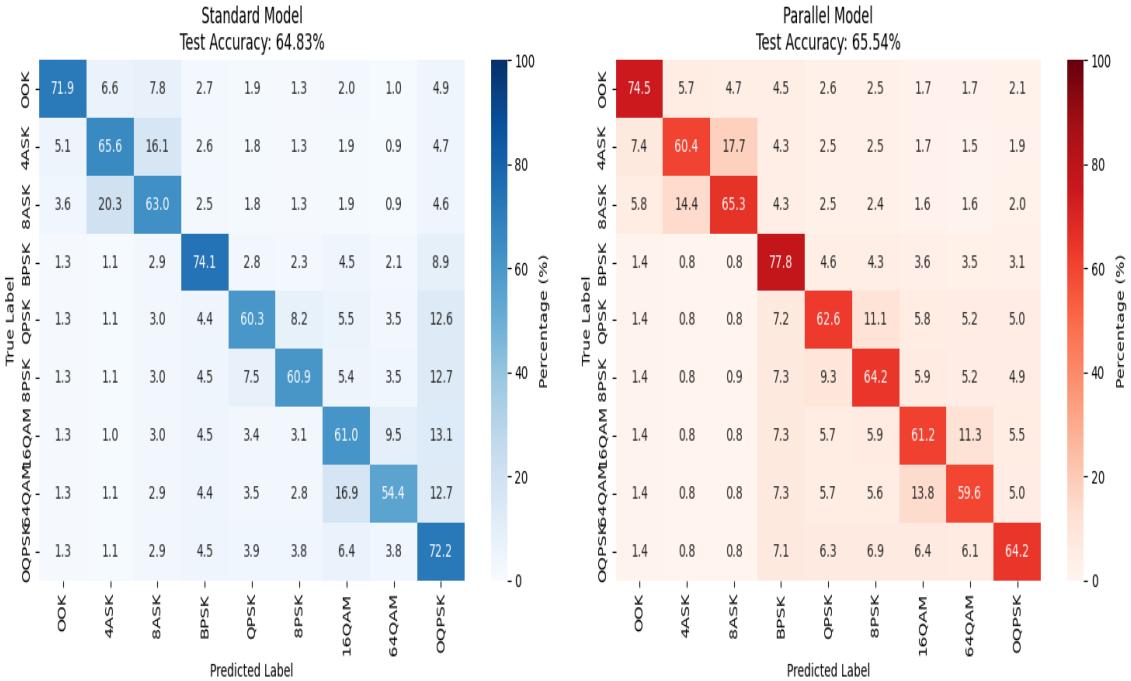


Figure 3: Matriks konfusi yang membandingkan performa Model Standar (kiri) dan Model Paralel (kanan) pada set data uji. Model Paralel mencapai akurasi keseluruhan yang sedikit lebih tinggi sebesar 65,54% dibandingkan Model Standar sebesar 64,83%, dengan peningkatan yang signifikan pada klasifikasi modulasi 8PSK dan 64QAM.

- **Diagonal Utama (Dari sisi Kiri atas ke Kanan Bawah):** Angka-Angka pada diagonal ini menunjukkan persentase prediksi yang benar. Semakin tinggi nilainya, semakin baik model tersebut dalam mengidentifikasi kelas tersebut. Pada Model Standart (Serial), 74.1% sinyal BPSK berhasil diidentifikasi dengan benar sebagai BPSK
- **Nilai di Luar Diagonal:** Angka-angka ini menunjukkan prediksi yang salah atau "kebingungan" model. Angka ini menunjukkan kelas mana yang sering tertukar. Misalnya, pada Model Standar, 20,3% sinyal BASK salah diklasifikasikan sebagai BPSK.

note: Untuk semua range SNR yang digunakan, **Model Paralel menunjukkan performa yang lebih baik dibandingkan Model Standar.** berikut untuk analisis dan perbandingan kinerja kedua model :

- **Akurasi Keseluruhan:** Model Paralel (65.54%) lebih unggul dari pada Model Standar (64.83%). Ini menunjukkan bahwa peningkatan performa secara umum.
- **Performa per-Kelas:**
 - **Kekuatan Model Paralel:** Model paralel menunjukkan kinerja yang jauh lebih baik untuk modulasi yang lebih kompleks seperti 8PSK (64.2% vs 60.4%) dan 64QAM (59.6% vs 54.4%). Ini merupakan keunggulan utama dari model paralel yang dapat menangkap fitur-fitur kompleks dari sinyal modulasi.
 - **Kelemahan Model Paralel:** Namun, performa Model paralel menurun untuk klasifikasi OQPSK (64.2% vs 72.2% pada Model Standar). Ini menunjukkan adanya trade-off antara kompleksitas model dan akurasi pada beberapa kelas modulasi.
- **Konsistensi:** Model Paralel menunjukkan konsistensi yang lebih baik pada berbagai kelas modulasi, dengan peningkatan akurasi yang signifikan pada kelas-kelas yang lebih sulit.
- **Area Kebingungan (Di Luar Diagonal):** Kedua model menunjukkan kebingungan antara jenis modulasi yang mirip, seperti:
 - Antara QPSK dan OQPSK

- Antara **8PSK** dan **QPSK/16QAM**

Ini wajar terjadi karena karakteristik sinyal mereka berdekatan.

b. Kurva Train dan Validation ke Dua Model

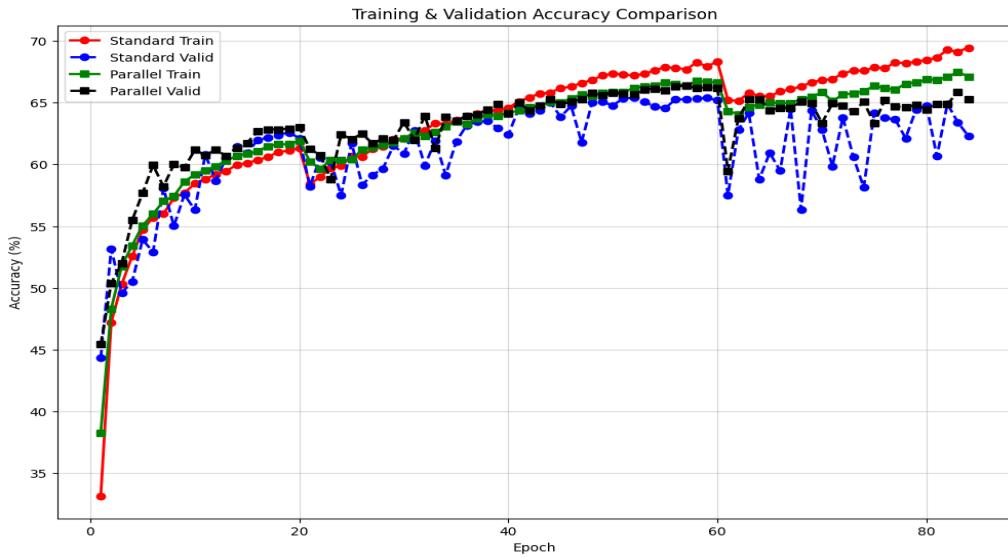


Figure 4: Kurva akurasi dan loss untuk model Standar (Serial) dan Paralel pada set data pelatihan dan validasi. Model Paralel menunjukkan peningkatan akurasi yang lebih baik pada set data pelatihan dan validasi dibandingkan model Standar.

Pada Figure 4 membandingkan keinerja akurasi dari dua model, yaitu Model Standar (Serial) dan Model Paralel, selama proses pelatihan yang diukur dalam satuan EPOCH. Kurva ini memperlihatkan bagaimana kedua model memahami dan belajar dengan kemampuannya untuk melakukan generalisasi terhadap data pelatihan dan validasi. Secara umum, kedua model menunjukkan proses pembelajaran yang efektif, ditandai dengan peningkatan akurasi yang signifikan pada tahap awal sebelum kurva mulai melandai. Namun, ketika membandingkan kemampuan generalisasi pada data baru, yang tercermin pada kurva validasi (garis putus-putus), perbedaan kinerja menjadi sangat jelas.

Kurva validasi Model Paralel (garis hitam) secara konsisten berada di posisi yang lebih tinggi dibandingkan Model Standar (garis biru), yang mengindikasikan bahwa Model Paralel secara superior mampu melakukan generalisasi. Lebih jauh lagi, proses pembelajaran Model Paralel terbukti jauh lebih stabil dan andal, terlihat dari kurvanya yang mulus. Hal ini sangat kontras dengan Model Standar yang menunjukkan fluktuasi tajam, termasuk penurunan drastis di sekitar epoch ke-60. Meskipun kedua model menunjukkan adanya celah antara akurasi pelatihan dan validasi—sebuah tanda overfitting yang wajar—kemampuan Model Paralel untuk mempertahankan akurasi validasi yang lebih tinggi dan stabil menunjukkan bahwa ia mengelola overfitting dengan lebih baik. Dengan demikian, dapat disimpulkan bahwa Model Paralel tidak hanya mencapai akurasi akhir yang lebih tinggi, tetapi juga menawarkan proses pembelajaran yang lebih robust, menjadikannya pilihan yang lebih unggul secara keseluruhan.

c. Kurva Perbandingan Kerja dari Dua Model

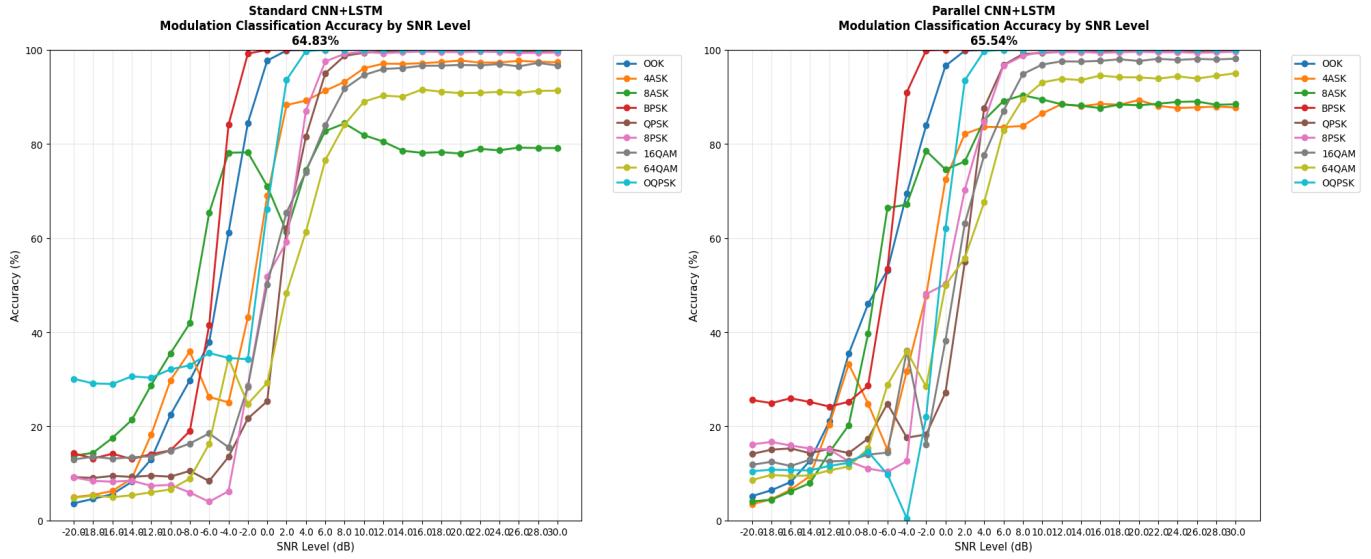


Figure 5: Kinerja klasifikasi modulasi yang menunjukkan keunggulan model Paralel (kanan) dalam mencapai akurasi tinggi pada tingkat SNR yang lebih rendah dibandingkan model Standar (kiri), terutama untuk modulasi kompleks.

Figure 5 menunjukkan seberapa akurat kedua model dalam menebak jenis sinyal pada berbagai tingkat kebisingan (SNR). Semakin ke kanan, sinyal semakin jernih dan akurasi semakin tinggi untuk kedua model. Kinerja Model Paralel lebih baik daripada Model Standar. Model Paralel dapat mencapai tingkat akurasi yang tinggi pada rasio Signal-to-Noise (SNR) yang lebih rendah, yang merupakan keunggulan utamanya. Dengan kata lain, bahkan ketika kualitas sinyal menurun secara signifikan, model ini masih dapat melakukan klasifikasi dengan baik.

Selain itu, stabilitas proses pelatihan yang lebih baik mendukung model ini. Kurva Learning dari Model Paralel menunjukkan kurva validasi yang lebih konvergen dengan fluktuasi yang lebih kecil. Ini menunjukkan proses learning yang lebih andal. Secara kuantitatif, Model Paralel berkontribusi pada akurasi keseluruhan yang sedikit lebih tinggi sebesar 65,54% dibandingkan Model Standar yang mencapai 64,83%. Oleh karena itu, Model Paralel dapat dianggap sebagai arsitektur yang lebih efisien dan dapat diandalkan untuk tugas klasifikasi modulasi ini.

d. Heatmap Perbandingan Akurasi Per-Kelas

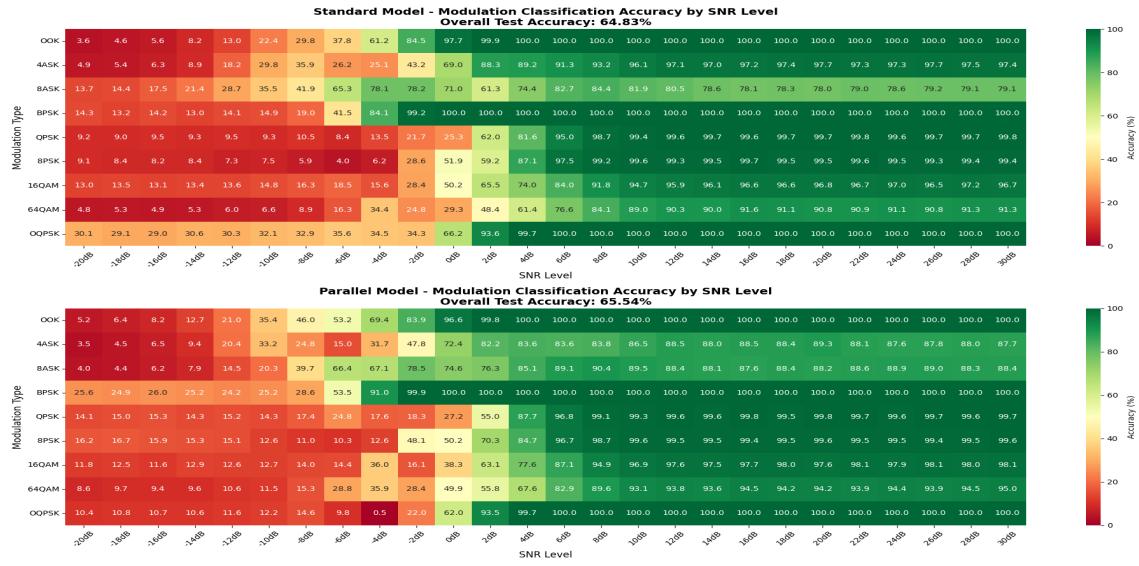


Figure 6: Heatmap perbandingan akurasi klasifikasi antara Model Standar (atas) dan Model Paralel (bawah) untuk setiap jenis modulasi pada berbagai level SNR.

Figure 6 menyajikan perbandingan kuantitatif kinerja akurasi antara Model Standar (atas) dan Model Paralel (bawah) dalam bentuk heatmap. Heatmap tersebut memberikan ilustrasi secara detail performa klasifikasi untuk setiap jenis modulasi pada berbagai level SNR dari -20 dB hingga 30 dB.

Secara umum, kedua model menunjukkan tren yang erwarted, di mana akurasi klasifikasi berbanding lurus dengan peningkatan level SNR. Hal ini terbukti dari transisi warna dari merah (akurasi rendah) ke hijau (akurasi tinggi) seiring meningkatnya kualitas sinyal. Namun, analisis komparatif menunjukkan keunggulan signifikan pada Model Paralel, terutama dalam hal robustitas terhadap derau (noise). Keunggulan ini secara visual terlihat dari transisi warna ke hijau yang terjadi pada level SNR yang lebih rendah dibandingkan dengan Model Standar. Sebagai contoh, pada level SNR 0 dB untuk modulasi 8PSK, Model Paralel telah mencapai akurasi sebesar 70.3%, sementara Model Standar baru mencapai 51.9%. Perbedaan efisiensi ini juga teramat pada modulasi kompleks lainnya seperti 16QAM dan 64QAM.

Dengan demikian, analisis heatmap ini secara kuantitatif mengonfirmasi bahwa arsitektur Model Paralel lebih efektif. Kemampuannya untuk mencapai akurasi tinggi dalam kondisi sinyal yang lebih terdegradasi oleh derau menjadikannya pilihan yang lebih andal dan superior untuk aplikasi klasifikasi modulasi ini.

C. Visualisasi Fitur dengan t-SNE

Untuk memahami mengapa Model Paralel menunjukkan performa yang lebih baik dibandingkan model Standar, dilakukan analisis visualisasi fitur menggunakan t-Distributed Stochastic Neighbor Embedding (t-SNE). Teknik ini memvisualisasikan fitur high-dimensional yang dipelajari oleh kedua model dalam ruang 2D yang dapat diinterpretasikan secara visual.

a. Metodologi t-SNE:

- Ekstraksi fitur dari layer terakhir sebelum klasifikasi
- 2000 sampel random dari test set
- Parameter t-SNE: perplexity=30, max_iter=1000

b. Hasil perbandingan Global

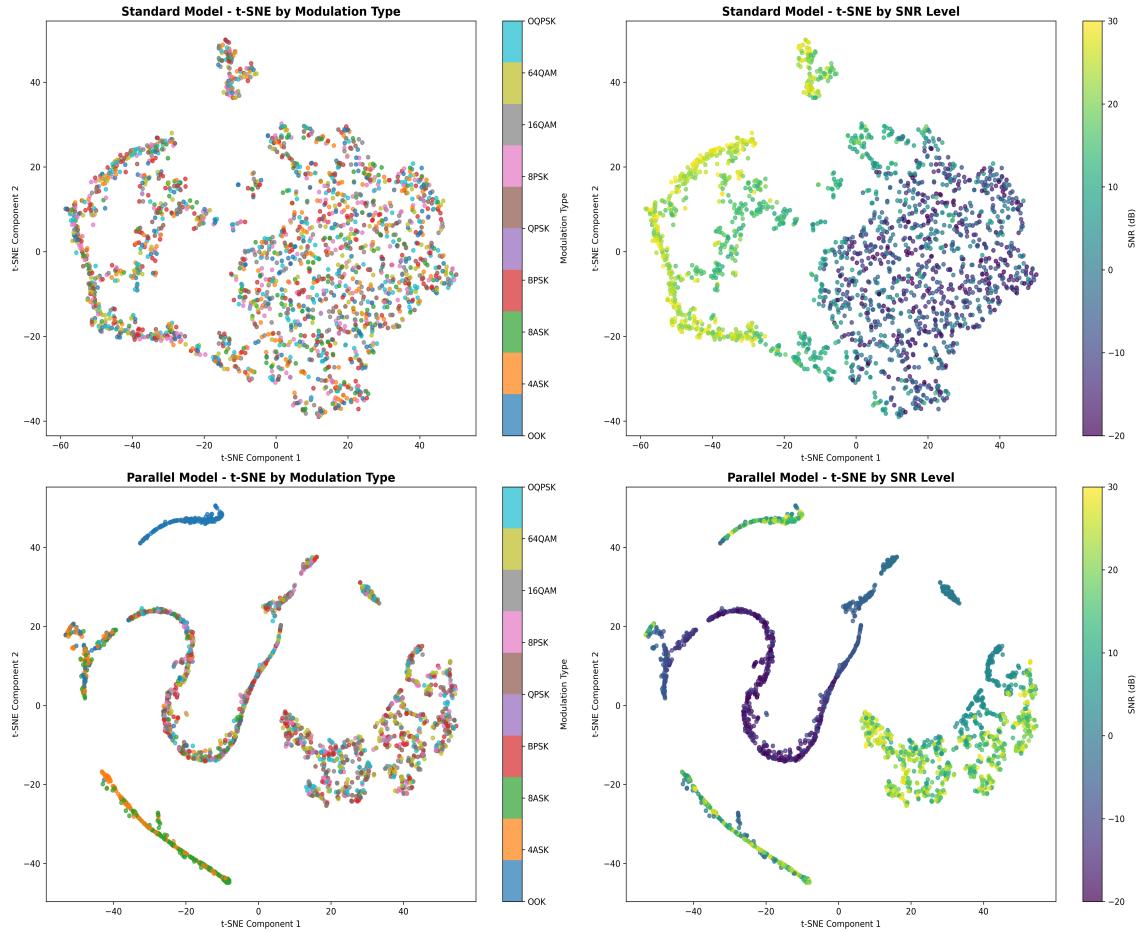


Figure 7: Perbandingan t-SNE antara Model Standar (atas) dan Model Paralel (bawah). Model Paralel menunjukkan clustering yang jauh lebih baik dan separasi kelas yang lebih jelas.

Pada Figure 7, Model Paralel menghasilkan representasi fitur yang dramatically lebih terorganisir dibandingkan Model Standar yang menunjukkan distribusi yang scattered dan overlapping. Visualisasi t-SNE menunjukkan perbandingan separabilitas fitur antara Model Standar (Atas) dan Model Paralel (Bawah). Model Paralel menunjukkan clustering yang lebih jelas dan separasi yang lebih baik antar kelas modulasi, terutama untuk 8PSK, 16QAM, dan 64QAM.

Pada Model Standar, distribusi fitur yang dihasilkan tampak kurang terstruktur. Hal ini terlihat jelas dari beberapa fenomena: pertama, adanya tumpang tindih (overlap) yang signifikan antara modulasi yang secara teoritis mirip, seperti QPSK dan OQPSK. Kedua, terjadi separasi yang buruk dan pengelompokan (clustering) yang tidak jelas untuk modulasi yang lebih kompleks, terutama antara 16QAM dan 64QAM. Secara keseluruhan, representasi fitur yang dipelajari oleh Model Standar tidak cukup diskriminatif.

Sebaliknya, Model Paralel menunjukkan superioritas yang jelas dengan menghasilkan ruang fitur yang jauh lebih terorganisir. Setiap kelas modulasi membentuk cluster yang lebih padat (tight) dan memiliki batas yang tegas, yang mengindikasikan bahwa model berhasil mempelajari fitur-fitur yang informatif. Peningkatan ini secara khusus terlihat pada separasi yang lebih baik antara modulasi-modulasi yang sulit dibedakan. Menurunnya tumpang tindih antar kelas dalam visualisasi ini konsisten dengan hasil kuantitatif yang ditunjukkan pada confusion matrix, yang mengonfirmasi bahwa arsitektur paralel secara fundamental lebih efektif dalam tugas klasifikasi ini.

c. Hasil Perbedangan Per-Kelas

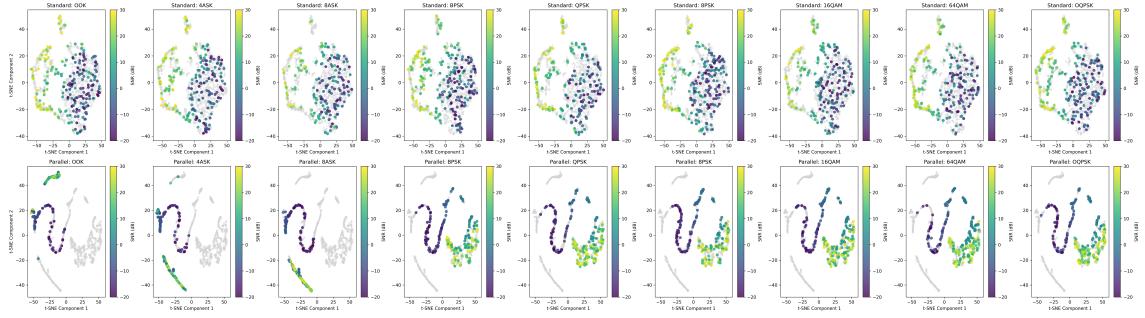


Figure 8: Analisis t-SNE per-kelas menunjukkan setiap jenis modulasi. Model Paralel (bawah) menghasilkan struktur yang organized untuk semua kelas, sementara Model Standar (atas) menunjukkan distribusi yang chaotic.

Figure 8 menyajikan visualisasi t-SNE yang digunakan untuk menganalisis ruang fitur (*feature space*) yang dipelajari oleh kedua model. Visualisasi ini memetakan fitur-fitur yang diekstraksi dari *layer* terakhir sebelum klasifikasi ke dalam ruang 2D untuk mengamati kemampuan separasi kelas dari masing-masing model.

Pada **Model Standar** (baris atas), terlihat bahwa fitur-fitur untuk setiap kelas modulasi cenderung tersebar dan tumpang tindih secara signifikan dengan fitur dari kelas lain (direpresentasikan oleh titik abu-abu). Hal ini mengindikasikan bahwa model tersebut kesulitan membentuk batas keputusan yang jelas antar kelas, sehingga menghasilkan representasi fitur yang kurang diskriminatif.

Sebaliknya, **Model Paralel** (baris bawah) menunjukkan hasil yang jauh lebih superior. Fitur untuk setiap kelas membentuk *cluster* atau manifold yang padat, terstruktur, dan memiliki separasi yang jelas dari kelas lainnya. Lebih lanjut, di dalam setiap *cluster*, teramatid adanya gradasi warna yang mulus sesuai dengan level SNR, dari ungu (SNR rendah) ke kuning (SNR tinggi). Fenomena ini membuktikan bahwa Model Paralel tidak hanya mampu mempelajari fitur yang sangat diskriminatif untuk memisahkan kelas, tetapi juga berhasil menangkap struktur internal data yang berkorelasi dengan kualitas sinyal. Dengan demikian, analisis t-SNE ini secara kualitatif mengonfirmasi bahwa Model Paralel memiliki kemampuan representasi fitur yang jauh lebih efektif dan teratur dibandingkan Model Standar.