

# Klasifikasi Penyakit Jantung Berdasarkan Nilai Amplitudo ECG Signal Menggunakan Deep Learning

1<sup>st</sup> Karsten Errando Winoto  
Tenik Informatika

Telkom University Purwokerto  
Purwokerto, Indonesia

karstenerr@student.telkomuniversity.ac.id

2<sup>nd</sup> Nicolaus Euclides W  
Tenik Informatika

Telkom University Purwokerto  
Purwokerto, Indonesia

nicolausn@telkomuniversity.ac.id

3<sup>rd</sup> Andi Prademone Y  
Tenik Informatika

Telkom University Purwokerto  
Purwokerto, Indonesia

andiay@telkomuniversity.ac.id

**Abstrak** — Penyakit kardiovaskular merupakan salah satu penyebab utama kematian secara global, dengan jumlah korban mencapai jutaan jiwa setiap tahunnya. Salah satu metode deteksi dini penyakit jantung adalah melalui analisis sinyal ECG (Electrocardiogram). Namun, analisis manual terhadap sinyal ECG masih memiliki keterbatasan dalam hal efisiensi dan akurasi. Penelitian ini bertujuan untuk mengembangkan model klasifikasi penyakit jantung berdasarkan nilai amplitudo sinyal ECG menggunakan metode deep learning. Empat arsitektur model deep learning yang digunakan adalah CNN, RNN, LSTM, dan GRU. Dataset yang digunakan berasal dari MIT-BIH Arrhythmia yang terdiri dari lima kelas jenis detak jantung. Proses pelatihan dan pengujian dilakukan menggunakan parameter yang sama, serta evaluasi model dilakukan menggunakan metrik akurasi, precision, recall, dan F1-score. Hasil pengujian menunjukkan bahwa model CNN dan GRU memiliki performa terbaik dengan akurasi sebesar 0.98, sementara RNN menunjukkan performa terendah dengan akurasi 0.83. Hasil penelitian ini menunjukkan bahwa penggunaan model deep learning, khususnya CNN dan GRU, efektif dalam mengklasifikasikan sinyal ECG untuk deteksi penyakit jantung secara otomatis.

**Kata kunci**—Deep learning, CNN, RNN, LSTM, GRU, Klasifikasi

## I. PENDAHULUAN

Penyakit kardiovaskular merupakan penyakit yang berbahaya, dimana penyakit ini merupakan penyebab kematian secara global. Data dari WHO diperkirakan sekitar 17,9 juta manusia meninggal setiap tahunnya [1]. WHO memperkirakan empat dari lima kematian kardiovaskular disebabkan oleh serangan jantung dan stroke, dan sepertiga dari kematian berusia dibawah 70 tahun [1]. Data dari kemenkes kematian yang disebabkan oleh penyakit kardiovaskular di Indonesia sendiri mencapai 651.481 penduduk pertahunnya [2].

ECG (*electrocardiogram*) merupakan alat untuk memantau kesehatan jantung. Penganalisis sinyal ECG masih dilakukan secara manual, hal ini dapat menyebabkan kesalahan dan memakan waktu. Penyakit jantung seperti gangguan irama jantung (aritmia), pengidentifikasi penyakit ini masih dilakukan dengan pengamatan langsung terhadap

signal ECG. Metode ini sering kali menyebabkan identifikasi tidak efisien, dan rentan terhadap kesalahan diagnosis [3]. Oleh karena itu, diperlukan teknologi dalam melakukan identifikasi akan penyakit dengan melakukan klasifikasi.

Metode yang digunakan dalam penelitian ini adalah *deep learning*, dimana *deep learning* merupakan bagian dari *machine learning* yang algoritmanya sering digunakan dikarenakan struktur dan cara kerjanya seperti struktur otak manusia atau lebih dikenal sebagai jaringan syaraf tiruan ANN (Artificial Neural Network) [4]. Beberapa model *deep learning* yang digunakan dalam penelitian meliputi CNN, RNN, LSTM, dan GRU.

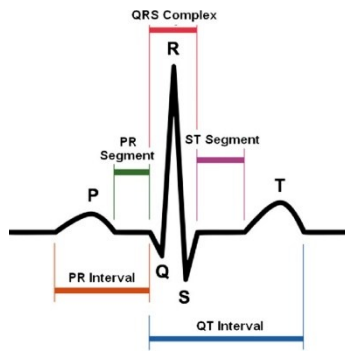
Penelitian ini bertujuan untuk menerapkan 4 model deep learning dalam melakukan klasifikasi akan penyakit jantung. Menggunakan dataset dari MIT-BIH Arrhythmia diambil dari *Kaggle* dengan judul “ECG Heartbeat Categorization Dataset” dan ditulis dalam penelitian yang berjudul “*ECG Heartbeat Classification: A Deep Transferable Representation*” yang ditulis oleh M.Kachuee *et al* [3]. Penelitian ini diharapkan dapat memberikan kontribusi atau manfaat signifikan untuk perkembangan teknologi, sebagai referensi untuk analisis performa model *deep learning*.

## II. KAJIAN TEORI

Kajian teori bertujuan untuk menjelaskan landasan konseptual dan teknis yang relevan dengan penelitian mengenai Klasifikasi Penyakit Jantung Berdasarkan Nilai Amplitudo ECG Signal Menggunakan Deep Learning. Pembahasan teori disusun berdasarkan variabel utama dalam penelitian ini.

### A. ECG (Electrocardiogram)

Electrocardiogram (ECG) adalah alat yang digunakan untuk melakukan diagnostik yang merekam akan aktivitas listrik jantung, menampilkan akan perubahan potensial listrik dalam bentuk gelombang P, QRS, dan T [5], [6], [7]. Gelombang PQRST dapat dilihat pada Gambar 1 (A).



GAMBAR 1  
(A) GELOMBANG PQRST [8]

## B. Arrhythmia

*Arrhythmia* atau aritmia adalah gangguan ritme pada jantung yang menyebabkan irama jantung dari penderita penyakit ini menjadi terlalu cepat, terlalu lambat, atau tidak teratur. Kondisi ini disebabkan oleh kelainan dari pembentukan maupun penghantaran impuls listrik di sistem konduksi jantung [9], [10]. Jenis-jenis dari aritmia diantaranya: *Normal*, *Artial Premature Contraction*, *Premature Ventricular Contraction*, *Fusion of Ventricular and Normal*, *Fusion of paced and Normal*.

### a) Normal

*Normal* adalah dimana, gelombang irama jantung memiliki irama yang teratur, dan frekuensi dari 60-100 kali per menit. Gelombang P dan QRS pada sinyal ECG normal memiliki interval PR 120-200 ms [11], [12].

### b) Artial premature contraction (PAC)

*Artial premature contraction* (PAC) adalah detak jantung ekstra yang terjadi lebih awal, berasal dari atrium, pada sinyal ECG terdapat gelombang P abnormal [13], [14], [15].

### c) Premature Ventricular Contraction (PVC)

*Premature ventricular contraction* (PVC) adalah detak jantung ekstra yang berasal dari ventrikel. Pada sinyal ecg tampak QRS lebar dan bentuknya abnormal, muncul sebelum gelombang detak normal [16], [17], [18].

### d) Fusion of Ventricular and Normal

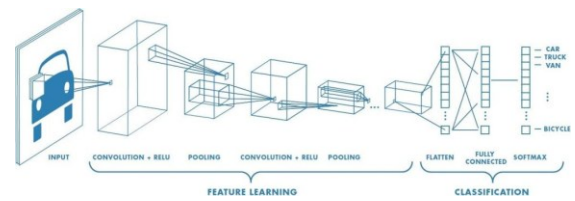
*Fusion of ventricular and normal* adalah detak jantung yang berbentuk dari gabungan antara implus dari nodus normal (SA) dan dari ventrikel. Pada sinyal ECG, memiliki QRS yang kompleks dimana QRS itu berbentuk *hybrid* antara normal dan premature [19].

### e) Fusion of Paced and Normal

*Fusion of paced and normal* adalah fusion beat yang terjadi pada pasien dengan alat pacu jantung (*pacemaker*). Detak alami dan detak dari pacemaker menstimulasi ventrikel hampir bersamaan. Sehingga menghasilkan bentuk QRS gabungan (*hybrid*) antara *normal* dan *paced* [20], [21].

## C. Convolutional Neural Network (CNN)

*Convolutional neural network* (CNN) model yang termasuk metode dari deep learning yang menggunakan jaringan syaraf tiruan untuk mengimplementasikan permasalahan yang mempelajari data-data yang ada [22]. Arsitektur CNN dapat dilihat pada Gambar 2 (C).

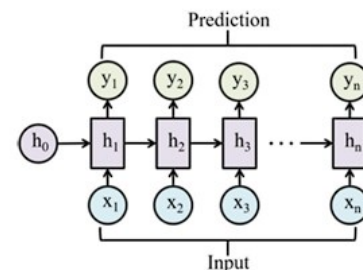


GAMBAR 2  
(C) ARSITEKTUR CNN [23]

*Convolutional neural network* (CNN) memiliki bagian-bagian seperti, *input layer*, *convolutional layer*, *pooling layer*, *activation function*, *fully connected layer*, *loss function*. Bagian ini merupakan isi dari arsitektur CNN itu sendiri yang berguna untuk melakukan klasifikasi akan data [24].

## D. Recurrent Neural Network (RNN)

*Recurrent neural network* (RNN) adalah jaringan saraf tiruan yang sangat cocok pada data berurutan. RNN memiliki koneksi berulang yang memungkinkannya mempertahankan status internal yang secara efektif memungkinkan jaringan untuk menyimpan memori masukkan sebelumnya [25]. Arsitektur RNN bisa dilihat pada Gambar 3 (D).



GAMBAR 3  
(D) ARSITEKTUR RNN [26]

Arsitektur dasar RNN terdiri dari *input layer* ( $x_t$ ), *hidden layer* ( $h_t$ ) dan *output layer* ( $y_t$ ). *Input layer* ( $x_t$ ) digunakan untuk menerima *input* urutan setiap waktu. *Hidden layer* digunakan untuk menyimpan dan memperbarui memori ( $h_t$ ). *Output layer* digunakan untuk menghasilkan *output* ( $y_t$ ) pada setiap waktu [26]. Persamaan rumus pada model RNN dituliskan pada persamaan (1), (2).

$$h_t = \sigma_h(W_{x_h x_t} + W_{h_h h_{t-1}} - b_h) \quad (1)$$

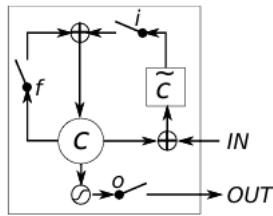
$$y_t = \sigma_y(W_{h_y h_t} + b_y) \quad (2)$$

Keterangan :

$h_t$	: Hidden State
$\sigma$	: Activation Function Sigmoid
$W$	: Weight
$y_t$	: Output
$b$	: Bias

## E. Long Short Term Memory (LSTM)

*Long short term memory* (LSTM) adalah jenis *recurrent neural network* (RNN) yang dirancang untuk menangkap ketergantungan jangka panjang dalam data berurutan [27]. Arsitektur LSTM dapat dilihat pada Gambar 4 (E).



GAMBAR 4  
(E) ARSITEKTUR LSTM [28]

Arsitektur dasar LSTM terdiri dari 2 *State* yaitu cell state dan hidden State dan 3 gates yaitu input gate, output gate, dan forget gate [28]. Persamaan rumus pada model LSTM dituliskan pada persamaan (3), (4), (5), (6), (7), (8).

$$h_t = o_t \cdot \tanh(c_t) \quad (3)$$

$$o_t = \sigma(W_o x_t + U_o h_{t-1} + b_o) \quad (4)$$

$$c_t = f_t \cdot c_{t-1} + i_t \cdot \tilde{c}_t \quad (5)$$

$$\tilde{c}_t = \tanh(W_c x_t + U_c h_{t-1} + b_{\tilde{c}_t}) \quad (6)$$

$$f_t = \sigma(W_f x_t + U_f h_{t-1} + b_f) \quad (7)$$

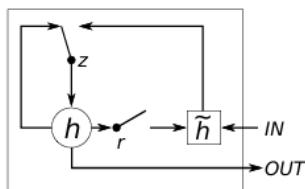
$$i_t = \sigma(W_i x_t + U_i h_{t-1} + b_i) \quad (8)$$

Keterangan :

$h_t$	: Hidden State
$\tilde{c}_t$	: New Memory Content
$c_t$	: Cell State
$\tanh$	: Activation Function Tanh
$\sigma$	: Activation Function Sigmoid
$W$	: Weight
$U$	: Weight
$o_t$	: Output Gate
$f_t$	: Forget Gate
$i_t$	: Input Gate
$b$	: Bias
$h_{t-1}$	: Hidden state sebelumnya
$x_t$	: Input pada waktu ke-t

#### F. Gated Recurrent Units (GRU)

*Gated recurrent units* (GRU) varian LSTM yang disederhanakan yang secara efisien menangkap ketergantungan di berbagai skala waktu [27]. Arsitektur GRU dapat dilihat pada Gambar 5 (F).



GAMBAR 5  
(F) ARSITEKTUR GRU [28]

Arsitektur GRU tidak seperti LSTM yang memiliki tiga *gates* (*input gate*, *output gate*, *forget gate*) dan 2 *state* (*hidden state* dan *cell state*), GRU hanya memiliki *hidden state* dan 2 *gates* utama antara lain *update gate* dan *reset gate* [28].

Persamaan rumus pada model GRU dituliskan pada persamaan (9), (10), (11), (12).

$$h_t = (1 - z_t) \cdot h_{t-1} + z_t \cdot \tilde{h}_t \quad (9)$$

$$z_t = \sigma(W_z x_t + U_z h_{t-1} + b_z) \quad (10)$$

$$\tilde{h}_t = \tanh(W x_t + U(r_t \cdot h_{t-1}) + b_h) \quad (11)$$

$$r_t = \sigma(W_r x_t + U_r h_{t-1} + b_r) \quad (12)$$

Keterangan :

$h_t$	: Hidden State
$\tilde{h}_t$	: New Candidate Hidden State
$\tanh$	: Activation Function Tanh
$\sigma$	: Activation Function Sigmoid
$W$	: Weight
$U$	: Weight
$z_t$	: Update Gate
$r_t$	: Reset Gate
$b$	: Bias
$h_{t-1}$	: Hidden state sebelumnya
$x_t$	: Input pada waktu ke-t

#### G. Confusion Matrix

*Confusion matrix* adalah sebuah tabel yang berfungsi untuk menilai performa model atau sistem klasifikasi terhadap data yang label kelasnya sudah diketahui. *Confusion matrix* mencakup empat nilai, yaitu *true positive* (TP) yang merupakan jumlah data yang tepat diklasifikasikan sebagai positif oleh model, *false positive* (FP) yang mewakili jumlah data negatif yang keliru diidentifikasi sebagai positif oleh model klasifikasi, *false negative* (FN) yang menunjukkan jumlah data positif yang salah ditandai sebagai negatif oleh model klasifikasi, dan *true negative* (TN) yang adalah jumlah data negatif yang berhasil diklasifikasikan dengan tepat oleh model klasifikasi. Dalam penilaian model, metrik utama yang diterapkan meliputi *accuracy*, *presicion*, *recall*, *f1-score*, dan . [29]. *Confusion Matrix* ditunjukkan pada tabel 1 (a).

TABEL 1  
(A) CONFUSION MATRIX

	True	False
True (Positive)	TP (True Positive) Correct Result	FP (False Positive) Unexpected Result
False (Negative)	FN (False Negative) Missing Result	TN (True Negative) Correct Absence of Result

#### a) Accuracy

*Accuracy* menunjukkan jumlah prediksi yang benar dibandingkan dengan seluruh prediksi. Persamaan *accuracy* ditunjukkan pada persamaan (13).

$$Accurary = \frac{TP + TN}{TP + TN + FP + FN} \quad (13)$$

#### b) Precision

*Precision* adalah ukuran yang menunjukkan seberapa banyak prediksi positif yang dibuat benar. Persamaan *precision* ditunjukkan pada persamaan (14).

$$precision = \frac{TP}{TP + FP} \quad (14)$$

#### c) Recall

*Recall* adalah seberapa banyak kasus yang berhasil diidentifikasi sebagai positif jika dibandingkan dengan jumlah total kasus positif dalam data.. Persamaan *recall* ditunjukkan pada persamaan (15).

$$recall = \frac{TP}{TP + FN} \quad (15)$$

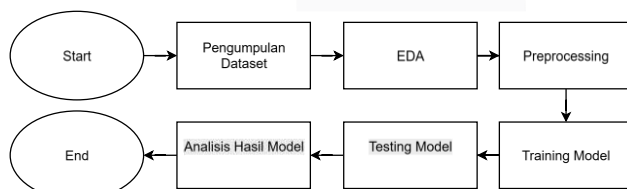
#### d) F1-Score

*F1-score* memperhitungkan kedua rasio *precision* dan *recall* secara seimbang. Persamaan *f1-score* ditunjukkan pada persamaan (16).

$$f1 - score = 2 \times \left( \frac{precision \times recall}{precision + recall} \right) \quad (16)$$

### III. METODE

Gambaran umum alur penelitian pada penelitian ini dapat dilihat pada gambar 6 (III).



GAMBAR 6  
(III) ALUR PENELITIAN

#### A. Pengumpulan Dataset

Data pada penelitian ini dataset diambil dari *Kaggle* dengan judul “*ECG Heartbeat Categorization Dataset*” dan ditulis dalam penelitian yang berjudul “*ECG Heartbeat Classification: A Deep Transferable Representation*” [3][30].

#### B. EDA

Dataset yang digunakan dalam penelitian ini memiliki 2 dataset diantaranya dataset training dan juga dataset testing. Jumlah data yang dimiliki pada dataset training sebesar 87554, untuk dataset testing sebesar 21892. Tiap sampelnya memiliki panjang sample sebesar 1870 ms dan memiliki frekuensi signal sebesar 120 Hz.

Terdapat distribusi kelas tiap data, untuk dataset training sendiri sebanyak, Normal sebesar 72471, Fusion of paced and normal sebesar 6431, Premature ventricular contraction

sebesar 5788, Atrial Premature sebesar 2223, Fusion of ventricular and normal sebesar 641. Sedangkan untuk dataset testing sebanyak, Normal sebesar 18118, Fusion of paced and normal sebesar 1608, Premature ventricular contraction sebesar 1448, Atrial Premature sebesar 556, Fusion of ventricular and normal sebesar 162.

#### C. Preprocessing

Pada tahap preprocessing dilakukannya label encoding, splitting dataset, distribusi kelas setelah dilakukannya splitting dataset.

Label encoding yang digunakan untuk mengubah nilai label menjadi nilai numerik. Normal = 0, Atrial Premature = 1, Premature ventricular contraction = 2, Fusion of ventricular and normal = 3, Fusion of paced and normal = 4.

Dataset training dibagi menjadi 2 dengan perbandingan 80 untuk training dan 20 untuk validation, sedangkan dataset tesing menggunakan dataset testing yang sudah ada.

Terdapat distribusi kelas baru akan tiap data, untuk dataset training sendiri sebanyak, Normal sebesar 57892, Fusion of paced and normal sebesar 5182, Premature ventricular contraction sebesar 4676, Atrial Premature sebesar 1797, Fusion of ventricular and normal sebesar 496. Sedangkan untuk dataset validation sebanyak, Normal sebesar 14579, Fusion of paced and normal sebesar 1249, Premature ventricular contraction sebesar 1112, Atrial Premature sebesar 426, Fusion of ventricular and normal sebesar 145. Sedangkan untuk dataset testing sebanyak, Normal sebesar 18118, Fusion of paced and normal sebesar 1608, Premature ventricular contraction sebesar 1448, Atrial Premature sebesar 556, Fusion of ventricular and normal sebesar 162.

#### D. Training Model

Pada tahap ini, dilakukannya pembuatan 4 model deep learning, diantaranya (CNN, RNN, LSTM dan GRU) untuk melakukan klasifikasi penyakit jantung. Model CNN dibuat menggunakan library torch dengan menggunakan Convolutional 1 dimensi, serta menggunakan kernel size sebesar 3, stride sebesar 1, dan juga padding sebesar 1. Model RNN, LSTM, dan GRU dibuat menggunakan library torch dengan menggunakan input\_size sebesar 1, hidden\_size sebesar 64, layer\_size sebesar 1, serta output\_size sebesar 5.

Setelah dilakukannya pembuatan model deep learning model dilatih menggunakan dataset training dan validation yang telah dilakukan pada tahap pembagian dataset. Pelatihan tiap model dilatih menggunakan parameter epoch sebesar 100, batch size sebesar 64, learning rate sebesar 0,001 dan menggunakan optimizer Adam.

#### E. Testing Model

Pada tahap testing model, model yang telah dilakukan pelatihan pada tahap training model akan dilakukan testing model menggunakan dataset testing yang belum pernah dilihat selama proses pelatihan dan validasi. Testing model ini bertujuan untuk mengevaluasi kemampuan model dalam melakukan klasifikasi penyakit jantung pada data yang benar-benar baru. Evaluation matrix yang digunakan seperti accuracy model, rata-rata akan precision, recall dan F1-score pada masing-masing kelas guna menilai kinerja model. Hasil dari tahap ini memberikan gambaran yang lebih objektif terkait seberapa baik model dalam melakukan klasifikasi pada dataset testing.



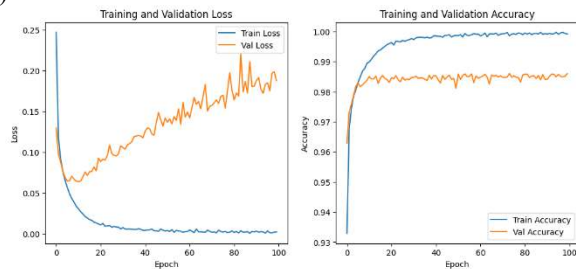
#### IV. HASIL DAN PEMBAHASAN

Dalam penelitian ini, model yang digunakan untuk melakukan klasifikasi penyakit jantung adalah CNN, RNN, LSTM, dan GRU. Untuk mendapatkan hasil evaluasi yang terbaik, dilakukan pelatihan model dengan menggunakan parameter sebagai berikut. Epoch sebesar 100, Batch Size sebesar 64, Learning Rate sebesar 0.001, dan menggunakan Optimizer Adam.

##### A. Training Model

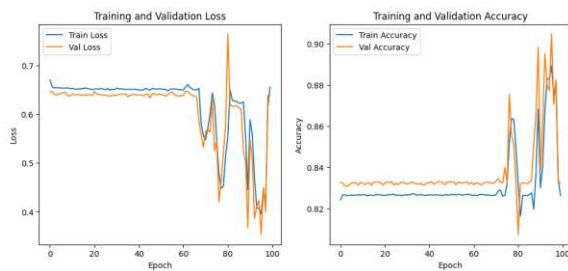
Pada tahap dilakukan training model dengan menggunakan parameter yang telah disebutkan sebelumnya.

Hasil Training model CNN dapat dilihat pada gambar 7 (A).



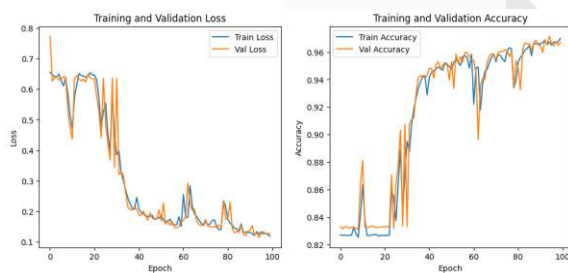
GAMBAR 7  
(A) HASIL TRAIN VALIDATION LOSS CNN

Hasil Training model RNN dapat dilihat pada gambar 8 (A).



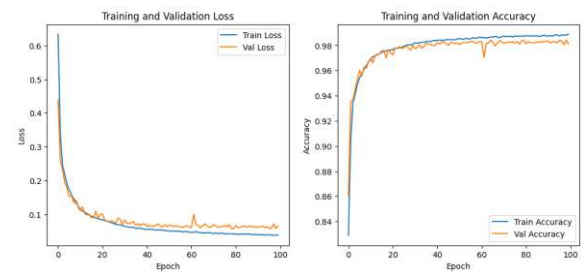
GAMBAR 8  
(A) HASIL TRAIN VALIDATION LOSS RNN

Hasil Training model LSTM dapat dilihat pada gambar 9 (A).



GAMBAR 9  
(A) HASIL TRAIN VALIDATION LOSS LSTM

Hasil Training model GRU dapat dilihat pada gambar 10 (A).



GAMBAR 10  
(A) HASIL TRAIN VALIDATION LOSS GRU

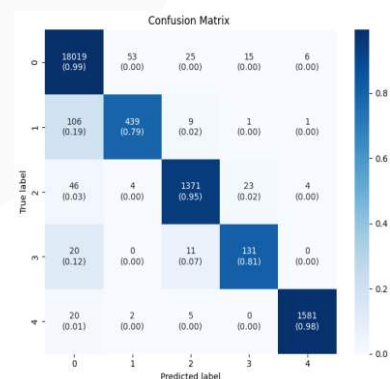
Berdasarkan grafik training dan validation loss pada Gambar 7 hingga 10, terlihat perbedaan performa tiap model dalam proses training. Model CNN pada gambar 7 mengalami overfitting, dengan akurasi training 0.9990 dan validation 0.9862, serta loss masing-masing 0.0038 dan 0.2054. Model RNN pada Gambar 8 menunjukkan training yang tidak stabil, terutama setelah epoch ke-60, dengan akurasi training 0.8265 dan validation 0.8326, serta loss 0.6554 dan 0.6423.

Sebaliknya, model LSTM dan GRU pada Gambar 9 dan 10 menunjukkan proses training yang baik, ditandai dengan penurunan loss yang konsisten pada training dan validation. Pada Gambar 9, akurasi training dan validation masing-masing sebesar 0.9699 dan 0.9669, dengan loss 0.1185 dan 0.1268. Sedangkan pada Gambar 10, diperoleh akurasi 0.9887 dan 0.9811, dengan loss 0.0389 dan 0.0650.

##### B. Testing Model

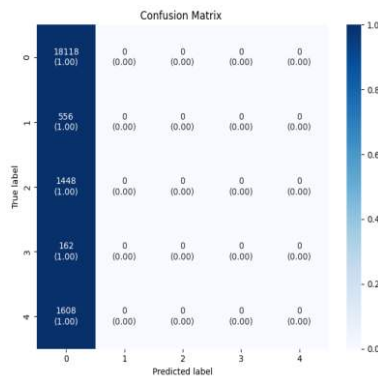
Pada Tahap ini dilakukannya testing akan tiap model dengan menggunakan dataset yang belum pernah di lihat sebelumnya, dengan menggunakan dataset testing sebesar 21892 sample signal. Hasil testing model pun diukur menggunakan evaluation matrix diantaranya precision, recall, F1-Score, accuracy, dan Macro Avg yaitu nilai rata rata dari precision, recall, F1-Score dari tiap distribusi kelas. Serta dibuatnya confusion matrix untuk untuk melihat distribusi prediksi model terhadap kelas yang sebenarnya. Model yang di testing pada tahap ini diantaranya CNN, RNN, LSTM, dan GRU.

Hasil Testing model CNN dapat dilihat pada gambar 11 (B).



GAMBAR 11  
(B) HASIL CONFUSION MATRIX MODEL CNN

Hasil Testing model RNN dapat dilihat pada gambar 12 (B).

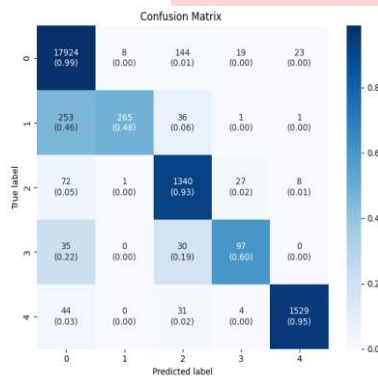


GAMBAR 12

(B) HASIL CONFUSION MATRIX MODEL RNN

Hasil Testing model LSTM dapat dilihat pada gambar 13

(B).

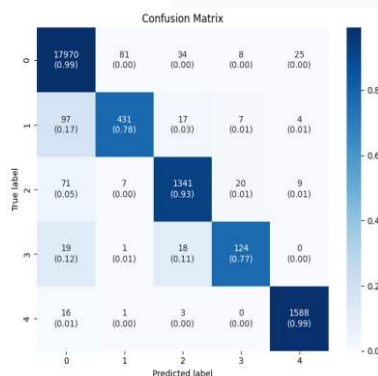


GAMBAR 13

(B) HASIL CONFUSION MATRIX MODEL LSTM

Hasil Testing model GRU dapat dilihat pada gambar 14

(B).



GAMBAR 14

(B) HASIL CONFUSION MATRIX MODEL GRU

### C. Hasil Analisis

Hasil Analisis pada penelitian ini, dilakukannya evaluasi tiap model *deep learning* yang telah melewati tahap *training model* dan *testing model*, model yang telah melewati 2 tahap itu meliputi :

1. CNN
2. RNN
3. LSTM
4. GRU

Hasil evaluasi tiap model, dapat dilihat pada tabel 2 (C).

TABEL 2  
(C) HASIL EVALUASI TIAP MODEL

Model	Accuracy	Precision	Recall	F1-Score
CNN	0.98	0.92	0.90	0.91
RNN	0.83	0.17	0.20	0.18
LSTM	0.97	0.89	0.79	0.82
GRU	0.98	0.90	0.89	0.90

Testing 4 model deep learning didapatkan hasil bahwa:

1. CNN mendapatkan hasil performa yang terbaik pada saat testing, dengan akurasi sebesar 0.98 precision sebesar 0.92, recall sebesar 0.90, dan f1-score sebesar 0.91. Meskipun model mengalami overfitting selama pelatihan dapat dilihat pada gambar 4.1 namun model tetap mampu memberikan generalisasi yang baik pada pengujian.

2. GRU mendapatkan hasil performa yang sama baiknya dengan CNN, dengan akurasi sebesar 0.98 precision sebesar 0.90, recall sebesar 0.89, dan f1-score sebesar 0.90. Model GRU tidak mengalami overfitting selama pelatihan dapat dilihat pada gambar 4.4. Tidak seperti model CNN yang mengalami overfitting.

3. LSTM mendapatkan hasil performa bagus dengan akurasi sebesar 0.97, precision sebesar 0.89, recall sebesar 0.79, dan f1-score sebesar 0.82. model LSTM tidak mengalami overfitting saat pelatihan dapat dilihat pada gambar 4.3. Namun model LSTM kurang mampu dalam mengenali seluruh kelas dengan baik.

4. RNN mendapatkan hasil performa yang paling rendah dimana mendapatkan akurasi sebesar 0.83, precision sebesar 0.17, recall sebesar 0.20, dan f1-score sebesar 0.18. model RNN Masih belum belajar dengan baik, bisa dilihat pada graph train val loss gambar 4.2. Model RNN mengalami underfitting sehingga membuat hasil performa paling rendah dibandingkan yang lain.

## V. KESIMPULAN

Berdasarkan hasil penelitian yang telah dilakukan, dapat disimpulkan bahwa dari penerapan 4 model Deep Learning dalam melakukan klasifikasi penyakit jantung dengan dataset berbentuk signal ECG dari dataset MIT-BIH Arrhythmia, menghasilkan beberapa performa yang sangat baik. Model di training menggunakan dataset signal dengan 5 kelas yaitu Normal, Atrial Premature, Premature Ventricular Contraction, Fusion of Ventricular and Normal, Fusion of Paced and Normal. Model yang memiliki performa terbaik pada tahap testing adalah model CNN dimana didapatkan hasil akurasi sebesar 0.98 dengan rata-rata precision sebesar 0.92, rata-rata recall sebesar 0.90 dan rata-rata F1-Score sebesar 0.91. Untuk model yang memiliki performa terbaik kedua pada tahap testing adalah model GRU dimana didapatkan hasil akurasi sebesar 0.98 dengan rata-rata precision 0.90, rata-rata recall sebesar 0.89, dan rata-rata F1-Score sebesar 0.90. Untuk model yang memiliki kinerja performa terburuk pada tahap testing adalah model RNN dimana didapatkan hasil akurasi sebesar 0.83 dengan rata-rata precision sebesar 0.17, rata-rata recall sebesar 0.20, dan rata-rata F1-Score sebesar 0.18. Untuk model LSTM memiliki performa cukup baik, namun model LSTM masih lemah dalam mengklasifikasi pada kelas 1 dan 3, sehingga hasil performa dari model LSTM tidak sebaik dengan model

CNN dan GRU, hasil akurasi yang didapatkan oleh model LSTM sebesar 0.97 dengan rata-rata Precision sebesar 0.89, rata-rata recall sebesar 0.79, dan rata-rata F1-Score sebesar 0.82. Hasil tersebut menunjukkan bahwa model CNN, dan GRU mampu melakukan klasifikasi penyakit jantung pada dataset berbentuk signal ECG dengan baik dan konsisten.

#### REFERENSI

- [1] N. Leyva-López, C. Hernández, C. E. Lizárraga-Velázquez, E. P. Gutiérrez-Grijalva, and J. B. Heredia, "Cardiovascular diseases," in *Phytochemicals from Mexican Medicinal Plants: Potential Biopharmaceuticals against Noncommunicable Diseases*, 2020, pp. 263–325. doi: 10.29309/tpmj/2016.23.11.1770.
- [2] Kemenkes, "Peringatan Hari Jantung Sedunia 2021: Jaga Jantungmu untuk Hidup Lebih Sehat," *Kemenkes*, pp. 1–7, 2021, Accessed: Aug. 04, 2025. [Online]. Available: <https://ayosehat.kemkes.go.id/peringatan-hari-jantung-sedunia-2021-jaga-jantungmu-untuk-hidup-lebih-sehat>
- [3] M. Kachuee, S. Fazeli, and M. Sarrafzadeh, "ECG Heartbeat Classification: A Deep Transferable Representation," *Jul.* 2018, doi: 10.1109/ICHI.2018.00092.
- [4] I. H. Sarker, "Machine learning: Algorithms, real-world applications and research directions," *SN Comput. Sci.*, vol. 2, no. 3, p. 160, 2021.
- [5] M. Rifali and D. Irmawati, "Sistem Cerdas Deteksi Sinyal Elektrokardiogram (EKG) untuk Klasifikasi Jantung Normal dan Abnormal Menggunakan Jaringan Syaraf Tiruan (JST)," *Elinvo (Electronics, Informatics, Vocat. Educ.*, vol. 4, pp. 49–55, Nov. 2019, doi: 10.21831/elinvo.v4i1.28242.
- [6] J. Sihotang, H. Tanudjaja, K. Indriati, and S.-N. Yu, "Analisis Sinyal ECG (Electrocardiogram) pada Aktivitas Duduk dan Berdiri dengan Metode HRV (Heart Rate Variability) pada Domain Waktu," *J. Elektro*, vol. 12, no. 2, pp. 63–74, 2019.
- [7] R. A. Putri, J. Y. Mindara, and S. R. I. Suryaningsih, "Rancang bangun wireless elektrokardiogram (EKG)," *Jiif (Jurnal Ilmu dan Inov. Fis.*, vol. 1, no. 1, pp. 58–64, 2017.
- [8] T. Moller, Y. Georgie, M. Voss, and L. Kaltwasser, "An Arduino Based Heartbeat Detection Device (ArdMob-ECG) for Real-Time ECG Analysis," *2022 IEEE Signal Process. Med. Biol. Symp. SPMB 2022 - Proc.*, pp. 1–26, 2022, doi: 10.1109/SPMB55497.2022.10014819.
- [9] M. P. Lukman, D. Widyaningsih, A. Lawi, and A. Asmila, "Sistem Deteksi Penyakit Aritmia Berdasarkan Jumlah Detak Jantung Berbasis Internet of Things dan Cloud Storage," in *Seminar Nasional Teknik Elektro dan Informatika (SNTEI)*, 2022, pp. 310–315.
- [10] D. K. Widjaja, A. A. Setiawan, and A. Ariosta, "GAMBARAN GANGGUAN Irama Jantung YANG DISEBABKAN KARENA HIPERTIROID," *J. Kedokt. Diponegoro (Diponegoro Med. Journal)*, vol. 6, no. 2, pp. 434–442, 2017.
- [11] E. L. Utari, "Analisa deteksi gelombang qrs untuk menentukan kelainan fungsi kerja jantung," *Teknoin*, vol. 22, no. 1, 2016.
- [12] A. Arifin *et al.*, "PENGENALAN RITME ELEKTROKARDIOGRAFI DALAM MENDETEKSI KELAINAN JANTUNG," *EEPIS Final Proj.*, 2011.
- [13] Primaya, "Penyebab Premature Atrial Complex (PAC) Jantung & Penanganannya - Primaya Hospital," Primaya Hospital. Accessed: Aug. 04, 2025. [Online]. Available: <https://primayahospital.com/jantung/premature-atrial-complex/>
- [14] R. Pranata, "Frequent premature atrial complex: A neglected marker of adverse cardiovascular events," *Int. J. Cardiovasc. Acad.*, 2020.
- [15] R. L. Mampioper, R. Maulana, and B. H. Prasetyo, "Implementasi Sistem Pendeteksi Premature Atrial Contraction (PAC) menggunakan Metode Naive Bayes Classifier," *J. Pengemb. Teknol. Inf. dan Ilmu Komput.*, vol. 6, no. 7, pp. 3291–3300, 2022.
- [16] G. M. Marcus, "Evaluation and management of premature ventricular complexes," *Circulation*, vol. 141, no. 17, pp. 1404–1418, 2020.
- [17] A. R. Pratama, R. Maulana, and D. Syauqy, "Implementasi Sistem Pendeteksi Premature Ventricular Contraction (PVC) Aritmia menggunakan Metode SVM," *J. Pengemb. Teknol. Inf. dan Ilmu Komput.*, vol. 5, no. 2, pp. 542–548, 2021.
- [18] D. M. O. Sitinjak, E. R. Widasari, and F. Utaminingrum, "Sistem Pendeteksi Premature Ventricular Contraction (PVC) berdasarkan Lebar QRS dan Gradien R menggunakan Metode FK-NN," *J. Pengemb. Teknol. Inf. dan Ilmu Komput.*, vol. 7, no. 2, pp. 592–598, 2023.
- [19] A. D. Fakhruddin and P. H. Gunawan, "Arrhythmia Classification Using CNN-SVM from ECG Spectrogram Representation," *Eduvest-Journal Univers. Stud.*, vol. 4, no. 12, pp. 11245–11254, 2024.
- [20] P. H. Waddingham, P. Lambiase, A. Muthumala, E. Rowland, and A. W. C. Chow, "Fusion pacing with biventricular, left ventricular-only and multipoint pacing in cardiac resynchronisation therapy: latest evidence and strategies for use," *Arrhythmia Electrophysiol. Rev.*, vol. 10, no. 2, p. 91, 2021.
- [21] M. Kucio, A. Kułach, T. Skowerski, M. Bałys, M. Skowerski, and G. Smolka, "Acute Effects of Fusion Pacing Versus Standard CRT on Myocardial Function in Heart Failure Patients with LBBB," *J. Clin. Med.*, vol. 14, no. 13, p. 4433, 2025.
- [22] Y. B. E. Purba, N. F. Saragih, A. P. Silalahi, S. Sitepu, and A. Gea, "Perancangan Alat Pendeteksi Kematangan Buah Nanas Dengan Menggunakan Mikrokontroler Dengan Metode Convolutional Neural Network (CNN)," *Methotika J. Ilm. Tek. Inform.*, vol. 2, no. 1, pp. 13–21, 2022.

- [23] Trivusi, "Pengertian dan Cara Kerja Algoritma Convolutional Neural Network (CNN)," Trivusi. Accessed: Aug. 04, 2025. [Online]. Available: <https://www.trivusi.web.id/2022/04/algoritma-cnn.html>
- [24] L. Alzubaidi *et al.*, "Review of deep learning: concepts, CNN architectures, challenges, applications, future directions," *J. Big Data*, vol. 8, no. 1, Dec. 2021, doi: 10.1186/s40537-021-00444-8.
- [25] M. A. Khan and P. Lim, "Predicting cognitive load in immersive driving scenarios with a hybrid CNN-RNN model."
- [26] I. D. Mienye, T. G. Swart, and G. Obaido, "Recurrent Neural Networks: A Comprehensive Review of Architectures, Variants, and Applications," Sep. 01, 2024, *Multidisciplinary Digital Publishing Institute (MDPI)*. doi: 10.3390/info15090517.
- [27] A.-C. Akazan, V. R. Mbingui, G. L. R. N'guessan, and I. Karambal, "Localized Weather Prediction Using Kolmogorov-Arnold Network-Based Models and Deep RNNs," May 2025, [Online]. Available: <http://arxiv.org/abs/2505.22686>
- [28] J. Chung, C. Gulcehre, K. Cho, and Y. Bengio, "Empirical Evaluation of Gated Recurrent Neural Networks on Sequence Modeling," Dec. 2014, [Online]. Available: <http://arxiv.org/abs/1412.3555>
- [29] A. Michael and J. Rusman, "Klasifikasi Cacat Biji Kopi Menggunakan Metode Transfer Learning dengan Hyperparameter Tuning Gridsearch," *J. Teknol. dan Manaj. Inform.*, vol. 9, no. 1, pp. 37–45, 2023.
- [30] "ECG Heartbeat Categorization Dataset." Accessed: Aug. 07, 2025. [Online]. Available: <https://www.kaggle.com/datasets/shayanfazeli/heart-beat>