

Klasifikasi Sinyal EEG pada Sistem BCI Pergerakan Jari Manusia Menggunakan Convolutional Neural Network

EEG Signal Classification in Human Finger Movement BCI System Using Convolutional Neural Network

Rahmat Widadi¹, Bongga Arif Widodo², Dodi Zulherman³

^{1,2,3}Fakultas Teknik Telekomunikasi dan Elektro, Institut Teknologi Telkom Purwokerto
E-mail: ¹rahmat@ittelkom-pwt.ac.id, ²bongga@ittelkom-pwt.ac.id, ³zulherman.dodi@ittelkom-pwt.ac.id

Abstrak

Pemanfaatan sistem *Brain-Computer Interface* (BCI) sebagai penghubung pikiran manusia dengan peralatan eksternal sangat bergantung pada keakuratan pengklasifikasian dan pengidentifikasian sinyal EEG khususnya gerak *motor imagery*. Kesuksesan *deep learning*, sebagai contoh *Convolutional Neural Network* (CNN), dalam proses klasifikasi pada berbagai bidang berpeluang untuk diimplementasikan pada klasifikasi gerak *motor imagery*. Pengimplementasian CNN untuk klasifikasi sinyal EEG *motor imagery* (MI-EEG) gerakan jari tangan diperkenalkan dalam tulisan ini. Rancangan sistem klasifikasi terdiri dari dua bagian yaitu *convolution layer* dan *multilayer perceptron* yang diimplementasikan menggunakan Python 3.7 dengan library TensorFlow 2.0 (Keras). Pengujian rancangan sistem dilakukan terhadap lima subjek dari data MI-EEG 5F dengan frekuensi pencuplikan 200 Hz. Pengujian melibatkan *Kfold-cross validation* dan analisis pada *confusion matrix*. Berdasarkan hasil pengujian, peningkatan ukuran kernel menghasilkan peningkatan rata-rata akurasi sistem. Sistem dengan akurasi terbaik diperoleh pada rancangan dengan jumlah kernel 50 sebesar 51,711%. Rancangan sistem menghasilkan kinerja yang melebihi hasil penelitian yang menjadi rujukan utama.

Kata kunci: klasifikasi, EEG, *motor imagery*, lima jari manusia, 1D-CNN

Abstract

A Brain-computer interface provides a new communication bridge between the human mind and devices, depending on the accurate classification and identification of EEG signals. The deep learning approach such as Convolutional Neural Network (CNN) for classification has successfully used in many fields, which is potentially used to motor imagery classification. This paper proposes a method that uses CNN for EEG-based motor imagery (MI-EEG) classification. The proposed method consisting of convolution layer and multilayer perceptron was implemented by using TensorFlow 2.0 library (Keras) in Python 3.7. We used MI-EEG 5F from five subjects with a sampling rate of 200 Hz. Testing involves Kfold-cross validation and confusion matrix analysis. Based on the result, kernel size increasing affected accuracy improvement. The best accuracy for the 50 kernels size reaches 51,711%. The proposed method gets better accuracy than the primary reference.

Keywords: classification, EEG, *motor imagery*, five human fingers, CNN

1. PENDAHULUAN

Implementasi pengolahan bio-sinyal seperti *electromyography* (EMG), *electrocardiography* (ECG), atau *electroencephalography* (EEG) pada dekade terakhir memberikan manfaat dalam bidang kesehatan. Pengolahan sinyal ECG umum dipergunakan sebagai metode dalam tahap diagnosis untuk mengamati kelainan fisiologi jantung maupun kelainan psikis yang berdampak terhadap aktivitas jantung. Sedangkan pengolahan sinyal EMG sebagai pengontrol devais eksternal dengan memanfaatkan sinyal listrik yang terdapat pada otot, umum dipergunakan dalam tahap terapi seperti tangan protestik [1]. Sinyal EEG juga umum dipergunakan dalam tahapan diagnosis namun penggunaan perangkat terapeutik berbasis EEG dalam beberapa tahun terakhir juga mulai diteliti sebagai contoh implementasi *motor imagery* untuk rehabilitasi pasien dengan gangguan motorik neural (stroke). Pembacaan aktivitas otak pengguna memanfaatkan sinyal EEG yang merupakan interpretasi dari perubahan potensial pada korteks serebral akibat dari interaksi berbagai neuron yang saling berhubungan. Pengolahan sinyal EEG merupakan salah satu metode penting untuk mendapatkan informasi dari otak manusia [2]. Di Indonesia sendiri terdapat sekitar 2% sampai dengan 3% penduduk penyandang disabilitas yang disebabkan oleh banyak faktor, salah satunya amputasi [3]. Pemanfaatan alat gerak prostetik berbasis sinyal EMG menjadi salah satu metode yang banyak dipergunakan dalam tahap rehabilitasi bagi pasien teramputasi. Namun, pasien dengan titik amputasi di atas sikut atau pasien dengan gangguan syaraf fungsi motorik tidak dapat menghasilkan perekaman sinyal EMG. Penggunaan sinyal EEG, sebagai pengendali gerak organ eksternal berbasis neural, menjadi solusi untuk pasien dengan kondisi tersebut [4].

Otak manusia merupakan sistem komputasi cerdas alami yang memiliki kemampuan untuk mengendalikan semua aktivitas saraf manusia seperti perilaku, pikiran, dan emosi termasuk pergerakan tubuh [5]. *Brain-Computer Interface* (BCI) berbasis sinyal EEG merupakan sistem yang menerjemahkan pola aktivitas otak pengguna menjadi pesan atau perintah sehingga dapat digunakan untuk mengendalikan komputer dan perangkat-perangkat eksternal [6]. BCI merupakan perangkat komunikasi yang menghubungkan otak manusia secara langsung kepada perangkat eksternal tanpa melibatkan sistem syaraf tepi dan jaringan otot [6]. Perkembangan terbaru dalam bidang organ buatan berbasis BCI yang memungkinkan pengendalian perangkat eksternal melalui otak secara langsung, memberikan kesempatan peningkatan taraf hidup penyandang disabilitas khususnya bagi pasien yang mengalami gangguan fungsi organ gerak [7]. Dua tipe sistem BCI yang telah diimplementasikan yaitu tipe imajiner dan tipe sensorik. Tipe imajiner salah satu contohnya implementasi *motor imagery*.

Dalam beberapa tahun terakhir, pengembangan sistem terapi berbasis sinyal EEG dilakukan oleh banyak peneliti, sebagai contoh pengembangan sistem rehabilitasi berbasis *motor imagery* (MI). Identifikasi atau klasifikasi sinyal EEG perlu dilakukan guna mengetahui perbedaan pola sinyal dari gerakan tubuh manusia seperti gerakan tangan dan kaki. Penelitian [8] mengembangkan sekaligus mengklasifikasikan enam belas BCI berbasis sinyal EEG yang terdiri dari gerakan tangan kiri, tangan kanan, kaki kiri, kaki kanan, dan lidah. Data sinyal EEG diubah menjadi representasi matrik fitur. Eksperimen dilakukan dengan berbagai fitur representasi sinyal EEG termasuk *Power Spectral Density* (PSD), daya kanal EEG, *Fourier Transform Amplitudes* (FTA), dan fitur ranah waktu. Matrik fitur yang telah diperoleh selanjutnya digunakan sebagai masukan *Support Vector Machine* (SVM) dan *Linear Discriminant Analysis* (LDA). Akurasi yang diperoleh 80-90% dengan enam kelas BCI dalam pengaturan *offline* sedangkan pengaturan *online* belum dapat dilaksanakan. Selain itu, akuisisi data pada pergerakan tangan belum memperhatikan pergerakan masing-masing jari meskipun pergerakan tangan juga mengikutkan pergerakan jari.

Deteksi pergerakan jari menggunakan BCI telah dilakukan oleh beberapa peneliti [4], [9], [10], dan [11]. Penelitian [10] menggunakan nilai *Power Spectral Density* (PSD) sebagai fitur dalam mengidentifikasi pergerakan jari manusia berdasarkan sinyal EEG. Fitur ini selanjutnya digunakan sebagai masukan tahap klasifikasi menggunakan metode *Support Vector*

Machine (SVM) dengan fungsi kernel basis radial. Akurasi tertinggi yang diperoleh pada penelitian ini sebesar 77,11% dari keseluruhan pasangan jari dalam satu tangan. Penelitian ini hanya membandingkan dua kelas yaitu pasangan setiap jari sehingga klasifikasi lima kelas secara langsung masih berpeluang untuk diteliti. Penelitian [11] mengklasifikasikan sinyal menjadi lima kelas yaitu ibu jari, telunjuk, jari tengah, jari manis, dan kelingking. Fitur yang digunakan yaitu nilai spektral *Principal Component Analysis* (PCA) *decomposition*, fitur kanal frekuensi individual, dan data temporal. Akurasi yang diperoleh sebesar 44,2% menggunakan *Support Vector Machine* (SVM). Akurasi yang diperoleh dari penelitian ini perlu ditingkatkan. Penelitian [4] mengembangkan pengklasifikasian gerak lima jari dengan menggunakan *Common Spatial Patter* (CSP) sebagai penyeleksi fitur. Klasifikasi pergerakan menggunakan *Autonomous Deep Learning* (ADL) dan *Random Forest* (RF). Peneliti [4] menggunakan empat subjek dengan frekuensi pencuplikan sebesar 200 Hz dan 1000 Hz. Peneliti [9] melakukan pengembangan klasifikasi untuk 13 subjek dengan frekuensi pencuplikan 200 dan 1000 Hz. Tingkat akurasi rata-rata yang dihasilkan sebesar 43% dengan standar deviasi 10%. Upaya peningkatan akurasi untuk klasifikasi pergerakan lima jari dari lima subjek berbeda dengan frekuensi pencuplikan 200 Hz menjadi fokus dalam tulisan ini. Kinerja rancangan sistem dibandingkan dengan hasil penelitian [9] sebagai rujukan utama penelitian.

Tulisan ini menyajikan sistem klasifikasi pergerakan lima jari tangan. Sistem klasifikasi yang dikembangkan dalam tulisan ini mempergunakan data EEG *motor imagery* (MI) dengan lima pergerakan jari tangan pada frekuensi pencuplikan 200Hz. Sistem klasifikasi menggunakan CNN yang terdiri dari *Convolution Layer* (CL) dan *Multilayer Perceptron* (MLP). Pengujian sistem dibagi menjadi dua yaitu tahap pelatihan dan tahap pengujian. *K-Fold cross validation* dan *confusion matrix* dipergunakan dalam analisis sistem klasifikasi. Penyajian tulisan dalam beberapa bagian yaitu metode penelitian, hasil dan pembahasan, kesimpulan dan saran. Metode penelitian menjelaskan alur kerja rancangan sistem klasifikasi, hasil dan pembahasan membahas hasil pengujian rancangan sistem dan simpulan disajikan pada kesimpulan dan saran

2. METODE PENELITIAN

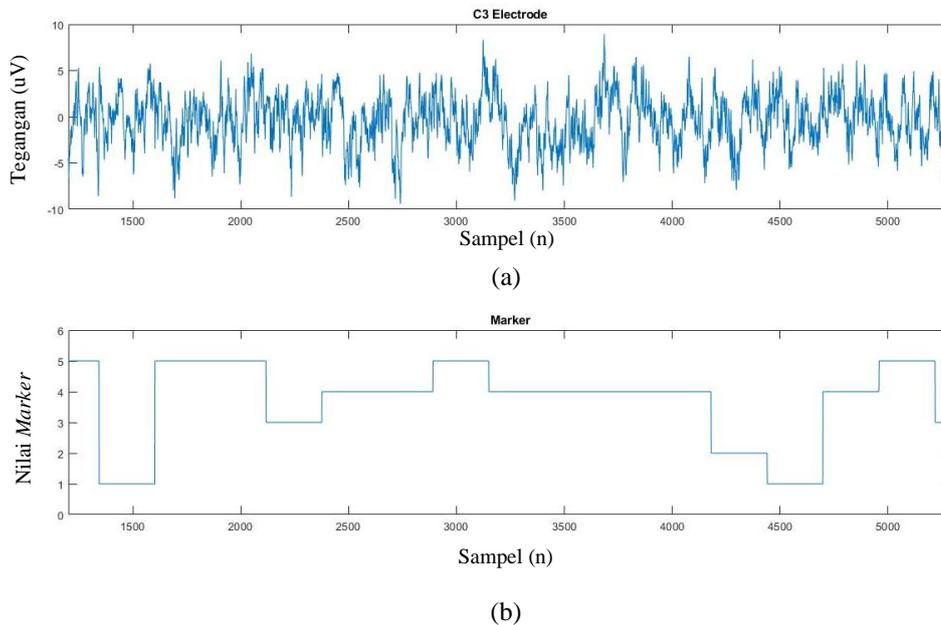
2.1 Dataset

Penelitian ini menggunakan dataset sinyal EEG yang diperoleh dari [9]. Alat perekaman menggunakan sistem EEG-1200 JE-921A EEG (Nihon Kohden, Jepang) dengan 22 elektrode: Fp1, Fp2, F3, F4, C3, C4, P3, P4, O1, O2, A1, A2, F7, F8, T3, T4, T5, T6, Fz, Cz, Pz, X5. Elektrode A1 dan A2 merupakan referensi serta X5 merupakan elektrode sinkronisasi. Tiga elektrode ini tidak digunakan sebagai fitur pada sistem klasifikasi. Secara keseluruhan terdapat 13 subjek perekaman. Subjek diperintahkan untuk membayangkan pergerakan tubuhnya sesuai dengan instruksi pada layar. Pada penelitian ini menggunakan data dengan label 5F (*5 Fingers*) dari 5 subjek. Instruksi pada layar pada paradigma 5F meliputi membayangkan pergerakan (*motor imagery*) lima jari yaitu: ibu jari, jari telunjuk, jari tengah, jari manis, dan jari kelingking. Sinyal EEG yang diperoleh dari perekaman ini disimpan dan ditandai menggunakan sinyal *marker* sesuai dengan instruksi yang diberikan. Tabel 1 memperlihatkan nilai marker disertai dengan keterangannya.

Tabel 1 Keterangan Nilai *Marker*

Marker	Keterangan
1	Ibu Jari
2	Jari Telunjuk
3	Jari Tengah
4	Jari Manis
5	Jari Kelingking

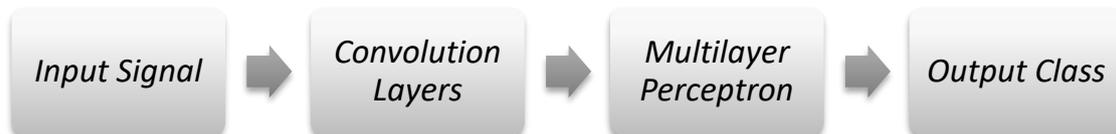
Sinyal EEG direkam dengan frekuensi sampling sebesar 200 Hz. *Bandpass filter* 0,53-70 Hz diterapkan pada sinyal EEG menggunakan perangkat lunak Neurofax. Selain itu, pada perangkat keras EEG-1200 sudah terdapat *notch filter* untuk mengurangi derau jala-jala listrik 50 Hz. Gambar 1 merupakan contoh sinyal EEG pada elektrode C3 yang disertai nilai *marker* untuk menandai kelas sinyal. Pada gambar ini terlihat sinyal *marker* memiliki panjang yang berbeda. Hal ini menandakan proses imajinasi pergerakan dari lima jari memiliki waktu yang tidak sama.



Gambar 1 Contoh (a) Sinyal EEG pada Elektroda C3 dan (b) Sinyal *Marker*

2.2 Convolutional Neural Network

Convolutional Neural Network (CNN) merupakan salah satu jenis *Deep Learning* yang menarik perhatian pada penelitian-penelitian terbaru. CNN secara otomatis dapat mengekstrak fitur dari data yang diberikan. Bahkan fitur yang diperoleh ini lebih merepresentasikan data dibandingkan dengan ekstraksi fitur manual yang dirancang manusia [12]. CNN dibagi berdasarkan dimensi data yang hendak diklasifikasikan. Apabila data berupa sinyal 2 dimensi (citra) maka dinamakan 2D-CNN. Sedangkan 1D-CNN digunakan untuk mengklasifikasi sinyal 1 dimensi. Pada penelitian ini digunakan 1D-CNN karena sinyal EEG merupakan sinyal 1 dimensi. Secara garis besar, struktur 1D-CNN terdiri dari dua bagian utama yaitu *Convolution Layers* (CL) dan *Multilayer Perceptron* (MLP) seperti yang ditunjukkan pada Gambar 2. Pada penelitian ini, masukan 1D-CNN berupa sinyal EEG *multichannel*. Sedangkan luarannya berupa klasifikasi jenis kelas dari sinyal masukan tersebut.



Gambar 2 Struktur 1D-CNN

Pada CL terjadi proses konvolusi, dimana sebuah kernel dikalikan dan digeser pada sinyal masukan. Proses ini juga dinamakan *shift-compute procedure*. Apabila sinyal masukan $x(n)$ memiliki panjang n dan kernel $h(k)$ dengan panjang k . Proses konvolusi antara $x(n)$ dengan $h(k)$ didefinisikan sebagai:

$$y(n) = \begin{cases} \sum_{i=0}^k x(n+i)h(i), & \text{Jika } n = 0 \\ \sum_{i=0}^k x(n+i)(s-1)h(i), & \text{yang lainnya} \end{cases} \quad (1)$$

dengan s merupakan jumlah langkah pada konvolusi. Keluaran proses konvolusi, $y(n)$, memiliki panjang data yang berbeda dengan masukan $x(n)$. Hal ini biasa dinamakan *valid padding*. Secara matematis, panjang luaran $y(n)$ yaitu:

$$m = \left\lfloor \frac{(n-k)}{s} \right\rfloor + 1 \quad (2)$$

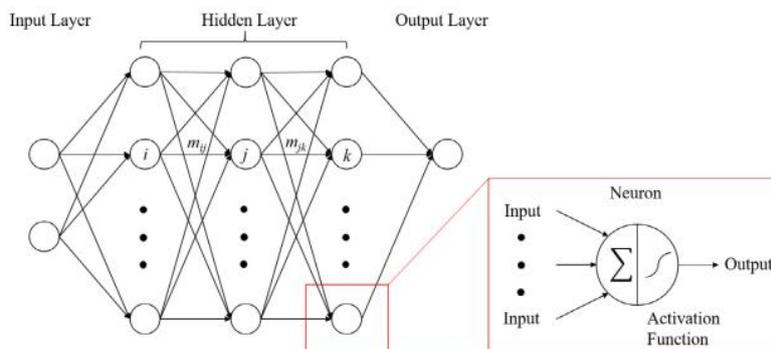
dengan m merupakan panjang $y(n)$.

Nilai panjang data luaran proses konvolusi, m , masih terlalu tinggi sehingga perlu direduksi. Proses reduksi ini dinamakan *pooling*. Metode *pooling* yang paling terkenal yaitu *max-pooling* dan *average-pooling*. *Max-pooling* merupakan metode pengambilan nilai maksimum dari sekumpulan data (*window*) berukuran f . *Window* ini digeser pada $y(n)$ dengan panjang langkah sebesar τ . Sedangkan pada *average-pooling*, nilai yang diambil dari *window* merupakan nilai rata-rata. Luaran proses *pooling* selanjutnya disatukan sehingga menjadi 1 dimensi. Proses ini dinamakan *flatten*. *Flatten* menjadikan luaran tahap *convolution layers* dapat digunakan sebagai masukan *multilayer perceptron*.

Tahap MLP atau *fully connected layer* terdiri dari *input layer*, *hidden layer* dan *output layer*. *Input layer* menerima masukan dari tahap *convolution layers* kemudian mengirimkannya menuju *hidden layer*. Bobot digunakan untuk menghubungkan antar *layer* pada *multilayer perceptron*. Sedangkan prediksi kelas data ditunjukkan oleh *output layer*. Pada setiap *layer* terdapat sejumlah neuron. Gambar 3 merupakan contoh struktur MLP dengan 3 *hidden layer*, m_{ij} merupakan bobot antara neuron ke- i di *hidden layer* 1 dengan neuron ke- j pada *hidden layer* 2. Pada model neuron terjadi kombinasi dari penjumlahan linier input dengan bobot. Kemudian luaran neuron ini dimasukan ke fungsi aktivasi. Salah satu fungsi aktivasi yang sering digunakan pada MLP yaitu *Rectified Linear Unit* (ReLU). Fungsi aktivasi ini memberikan waktu komputasi yang lebih cepat dibandingkan dengan fungsi aktivasi *sigmoid* dan *tanh* [13]. Fungsi aktivasi ReLU didefinisikan pada persamaan berikut:

$$f(x) = \max(0, x) \quad (3)$$

dengan x merupakan hasil kombinasi bobot dengan input neuron, sedangkan $f(x)$ merupakan luaran neuron yang telah mengalami aktivasi. Pembaruan bobot pada propagasi mundur menggunakan fungsi optimisasi untuk memperbaiki proses pelatihan. Fungsi optimisasi Adam digunakan pada penelitian ini [14].



Gambar 3 Struktur *Multilayer Perceptron* [15]

2.3 Pengujian Sistem

Pelatihan dan pengujian sistem 1D-CNN untuk klasifikasi sinyal EEG gerak lima jari *motor imagery* menggunakan prosedur *cross validation*. Prosedur ini menghasilkan bias yang lebih kecil dibandingkan prosedur *holdout* [16]. Pada *cross validation*, data dibagi menjadi k bagian terpisah dengan ukuran yang relatif sama sehingga sering dinamakan *Kfold-cross validation*. Pada penelitian ini nilai k diatur sama dengan 10. Secara bergantian, 9 bagian digunakan sebagai data latih dan 1 bagian lain sebagai data uji. Model dibangun pada data latih kemudian diterapkan untuk memprediksi data uji. Hasil prediksi setiap perulangan k dapat ditampilkan pada *confusion matrix*. Seluruh hasil prediksi kelas kemudian digabungkan untuk dihitung nilai akurasi. Perhitungan akurasi 1D-CNN menggunakan persamaan berikut:

$$\text{Akurasi} = \frac{\text{Jumlah data terprediksi benar}}{\text{Jumlah seluruh data}} \times 100 \% \quad (4)$$

2.4 Implementasi Sistem

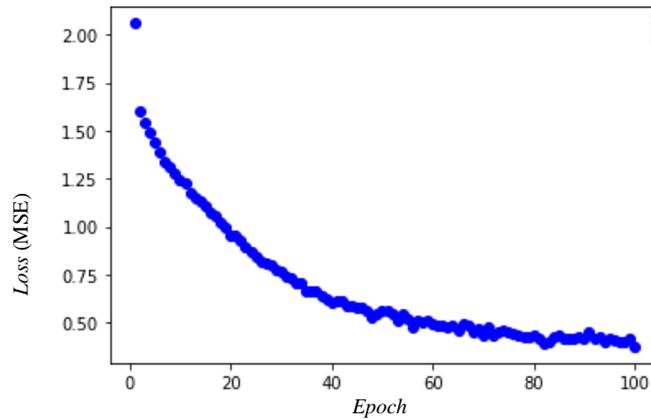
Sistem klasifikasi sinyal EEG diimplementasikan menggunakan bahasa pemrograman Python 3.7. Adapun *library* yang digunakan adalah Keras dengan Tensorflow 2.0 *backend* [17]. Sistem diimplementasikan pada komputer dengan spesifikasi: Windows 10, *proseccor* Core i5-9400, RAM 8 GB, dan GPU Nvidia RTX 2060. Spyder 4.0.1 pada Anaconda Navigator 1.9.12 digunakan sebagai *editor* dan *compiler* dari program. Pada penelitian ini, *Hyperparameter* 1D-CNN diatur sesuai dengan konfigurasi pada Tabel 2. Skenario jumlah kernel dilakukan untuk mengetahui pengaruhnya terhadap nilai akurasi klasifikasi. Jumlah kernel diatur dari 5 sampai 50 dengan perubahan sebanyak 5 kernel.

Tabel 2 Konfigurasi *Hyperparameter* 1D-CNN

<i>Hyperparameter</i>	Konfigurasi
Jumlah Kernel	5-50
Panjang Kernel	5
<i>Pooling</i>	<i>Average Polling</i>
<i>Dense</i>	64
Fungsi Aktivasi	ReLU
Neuron <i>Output layer</i>	5
Optimisasi MLP	Adam

3. HASIL DAN PEMBAHASAN

Hasil perekaman sinyal EEG imajinasi pergerakan lima jari yang telah dibahas pada bagian dataset memiliki panjang sampel yang berbeda. Kondisi ini mengakibatkan masukan 1D-CNN tidak dapat langsung menggunakan dataset hasil perekaman. Proses segmentasi diperlukan agar sinyal memiliki panjang yang sama. Pada penelitian ini, sinyal EEG tersegmentasi memiliki panjang 200 sampel (1 detik) sehingga sebuah sinyal EEG memiliki ukuran 200×19 . Nilai 19 menandakan jumlah elektrode yang digunakan. Secara keseluruhan, data yang digunakan pada proses pelatihan dan pengujian sebanyak 4750 sinyal yang diperoleh dari 5 subjek. Selanjutnya proses *Kfold-Cross Validation* diterapkan. Data dibagi menjadi sepuluh bagian. Secara bergantian sembilan bagian akan menjadi data pelatihan dan satu bagian menjadi data pengujian. Pelatihan 1D-CNN menghasilkan kesalahan klasifikasi yang disebut *training loss*. Gambar 4 merupakan grafik *training loss* pada tahap pelatihan dengan skenario jumlah kernel sebesar 5 dan *Kfold* sebesar 10. Gambar ini menampilkan *Mean Squared Error* (MSE) dari kesalahan klasifikasi pada setiap *epoch*.



Gambar 4 Training Loss Pada Skenario Jumlah Kernel = 5 dan Kfold = 10

Model 1D-CNN yang telah mengalami proses pelatihan selanjutnya digunakan pada proses pengujian. Data uji digunakan pada tahap ini. Hasil prediksi 1D-CNN ditunjukkan pada *confusion matrix*. Gambar 5 merupakan *confusion matrix* tahap pengujian dengan skenario jumlah kernel sebesar 5 dan *Kfold* sebesar 10. Baris pada *Confusion matrix* mewakili kelas sebenarnya dan kolom mewakili kelas hasil klasifikasi.

		Kelas Klasifikasi					Kelas Sebenarnya
		Ibu Jari	Jari Telunjuk	Jari Tengah	Jari Manis	Jari Kelingking	
Ibu Jari	39	27	7	3	2		
Jari Telunjuk	14	38	12	7	1		
Jari Tengah	10	23	19	18	9		
Jari Manis	6	8	19	30	13		
Jari Kelingking	4	6	7	22	36		
	Ibu Jari	Jari Telunjuk	Jari Tengah	Jari Manis	Jari Kelingking		

Gambar 5 Confusion Matrix Pada Skenario Jumlah Kernel = 5 dan Kfold = 10

Berdasarkan data yang terdapat pada Gambar 5, perhitungan nilai akurasi dari *confusion matrix* dengan mengacu pada persamaan (4) diuraikan dalam rincian sebagai berikut:

$$\text{Akurasi} = \frac{\text{Jumlah data terprediksi benar}}{\text{Jumlah seluruh data}} \times 100 \% = \frac{162}{380} \times 100 \% = 42,634 \%$$

Proses perhitungan nilai akurasi dalam tahap pengujian dengan memvariasikan *Kfold* dari satu hingga sepuluh ditampilkan pada Tabel 3. Tabel 3 menampilkan nilai akurasi pada setiap nilai *Kfold* dengan skenario jumlah kernel sebesar 5 pada 1D-CNN. Tingkat akurasi terendah terjadi pada *Kfold* bernilai 9 dengan akurasi sebesar 34,999% sedangkan nilai tertinggi pada *Kfold* bernilai 10 dengan akurasi sebesar 42,634%. Perhitungan nilai akurasi untuk variasi ukuran kernel menggunakan pendekatan rata-rata untuk setiap variasi *Kfold*. Rata-rata akurasi sistem dengan variasi ukuran kernel 5 sebesar 38,789% dengan standar deviasi sebesar 2,32%.

Tabel 3 Akurasi Pada Skenario Jumlah Kernel = 5

<i>Kfold</i>	Akurasi (%)
1	38,684
2	39,474
3	40
4	40
5	41,316
6	38,158
7	35,263
8	37,368
9	34,999
10	42,634
Rata-rata	38,789

Pengujian sistem selanjutnya dengan memvariasikan jumlah kernel seperti ditampilkan pada Tabel 4. Peningkatan ukuran kernel meningkatkan rata-rata akurasi sistem. Akurasi terendah pada sistem dengan ukuran kernel 5 dan akurasi tertinggi pada sistem dengan ukuran kernel 50. Berdasarkan hasil penelitian, rata-rata akurasi yang dihasilkan pada sistem dengan ukuran kernel 50 sebesar 51,711% lebih besar dari 43% yang terdapat pada [9].

Tabel 4 Rata-rata Akurasi Pada Keseluruhan Skenario Jumlah Kernel

Jumlah Kernel	Rata-rata Akurasi (%)
5	38,789
10	44,079
15	46,447
20	47,553
25	48,5
30	49,184
35	50,579
40	49,711
45	50,737
50	51,711

4. KESIMPULAN DAN SARAN

Perancangan sistem klasifikasi pergerakan jari tangan berbasis *motor imagery* (MI) EEG menggunakan *Convolutional Neural Network* (CNN) terdiri dari dua bagian yaitu *Convolution Layer* dan *Multilayer Perceptron*. Pengujian sistem terhadap lima subjek menghasilkan rata-rata akurasi yang lebih tinggi dari rujukan utama sebesar 51,711% dibandingkan dengan 43%. Namun, pengembangan lebih lanjut dibutuhkan yaitu menguji rancangan untuk klasifikasi subjek dengan sampling rate 200 Hz dan 1000 Hz.

UCAPAN TERIMA KASIH

Penelitian ini didanai oleh Kementerian Riset dan Teknologi/Badan Riset dan Inovasi (Kemenristek/BRIN) Republik Indonesia pada skema Penelitian Dosen Pemula (PDP).

DAFTAR PUSTAKA

- [1] T. Triwiyanto, I. P. A. Pawana, and M. H. Purnomo, "An Improved Performance of Deep Learning Based on Convolution Neural Network to Classify the Hand Motion by Evaluating Hyper Parameter," *IEEE Trans. Neural Syst. Rehabil. Eng.*, vol. 28, no. 7, pp. 1678–1688, Jul. 2020.
- [2] D. B. MacDonald, "Electroencephalography: Basic Principles and Applications," in *International Encyclopedia of the Social & Behavioral Sciences (Second Edition)*, Second Edi., J. D. Wright, Ed. Oxford: Elsevier, 2015, pp. 353–363.
- [3] Irwanto, E. R. Kasim, A. Fransiska, M. Lusli, and O. Siradj, *Analisis situasi penyandang disabilitas di Indonesia: sebuah desk-review*. Depok: PUSAT KAJIAN DISABILITAS

- FAKULTAS ILMU-ILMU SOSIAL DAN POLITI, UNIVERSITAS INDONESIA, 2010.
- [4] K. Anam, S. Bukhori, F. S. Hanggara, and M. Pratama, "Subject-independent Classification on Brain-Computer Interface using Autonomous Deep Learning for finger movement recognition," in *2020 42nd Annual International Conference of the IEEE Engineering in Medicine Biology Society (EMBC)*, 2020, pp. 447–450.
 - [5] R. Langner, S. Leiberger, F. Hoffstaedter, and S. B. Eickhoff, "Towards a human self-regulation system: Common and distinct neural signatures of emotional and behavioural control," *Neurosci. Biobehav. Rev.*, vol. 90, pp. 400–410, 2018.
 - [6] H. Chang and J. Yang, "Automated Selection of a Channel Subset Based on the Genetic Algorithm in a Motor Imagery Brain-Computer Interface System," *IEEE Access*, vol. 7, pp. 154180–154191, 2019.
 - [7] R. Chaudhari and H. J. Galiyawala, "A Review on Motor Imagery Signal Classification for BCI," *Signal Process. An Int. J.*, vol. 11, no. 2, pp. 16–34, 2017.
 - [8] Y. Mishchenko, M. Kaya, E. Ozbay, and H. Yanar, "Developing a Three- to Six-State EEG-Based Brain-Computer Interface for a Virtual Robotic Manipulator Control," *IEEE Trans. Biomed. Eng.*, vol. 66, no. 4, pp. 977–987, Apr. 2019.
 - [9] M. Kaya, M. K. Binli, E. Ozbay, H. Yanar, and Y. Mishchenko, "A large electroencephalographic motor imagery dataset for electroencephalographic brain computer interfaces," *Sci. Data*, vol. 5, p. 180211, Oct. 2018.
 - [10] K. Liao, R. Xiao, J. Gonzalez, and L. Ding, "Decoding Individual Finger Movements from One Hand Using Human EEG Signals," *PLoS One*, vol. 9, no. 1, pp. 1–12, 2014.
 - [11] R. Xiao and L. Ding, "Evaluation of EEG Features in Decoding Individual Finger Movements from One Hand," *Comput. Math. Methods Med.*, vol. 2013, pp. 1–10, 2013.
 - [12] S.-H. Feng, J.-Y. Xu, and H.-B. Shen, "Chapter Seven - Artificial intelligence in bioinformatics: Automated methodology development for protein residue contact map prediction," in *Biomedical Information Technology (Second Edition)*, Second Edi., D. D. Feng, Ed. Academic Press, 2020, pp. 217–237.
 - [13] Y. Wang, Y. Li, Y. Song, and X. Rong, "The Influence of the Activation Function in a Convolution Neural Network Model of Facial Expression Recognition," *Appl. Sci.*, vol. 10, no. 5, 2020.
 - [14] D. P. Kingma and J. Ba, "Adam: Method for Stochastic Optimization," in *3rd International Conference on Learning Representations, {ICLR} 2015, San Diego, CA, USA, May 7-9, 2015, Conference Track Proceedings*, 2015.
 - [15] Y. Chen, H. Chang, J. Meng, and D. Zhang, "Ensemble Neural Networks (ENN): A gradient-free stochastic method," *Neural Networks*, vol. 110, pp. 170–185, 2019.
 - [16] Q. Wu and P. Vos, "Chapter 6 - Inference and Prediction," in *Computational Analysis and Understanding of Natural Languages: Principles, Methods and Applications*, vol. 38, V. N. Gudivada and C. R. Rao, Eds. Elsevier, 2018, pp. 111–172.
 - [17] F. Chollet and others, "Keras," 2015. [Online]. Available: <https://keras.io>.