**PERANCANGAN DAN IMPLEMENTASI APLIKASI MOBILE IDENTIFIKASI AUTISM SPECTRUM DISORDER (ASD) BERBASIS CONVOLUTIONAL NEURAL NETWORK (CNN)**

**PROPOSAL TUGAS AKHIR**

Program Studi

Teknik Komputer

**Diajukan Oleh :**

**ALYAFI**

**1904111010051**



**JURUSAN TEKNIK ELEKTRO DAN KOMPUTER**

**FAKULTAS TEKNIK UNIVERSITAS SYIAH KUALA**

**DARUSSALAM, BANDA ACEH**

**2022**

# **PERNYATAAN KEASLIAN PROPOSAL**

Saya menyatakan dengan sesungguhnya proposal tugas akhir dengan judul ”**Perancangan dan Implementasi Aplikasi Mobile Identifikasi Autism Spectrum Disorder (ASD) berbasis Convolutional Neural Network (CNN)**” bukan merupakan tiruan atau duplikasi dari tugas akhir atau karya ilmiah yang telah dipublikasi oleh orang lain, kecuali bagian sumber informasinya dicantumkan sebagaimana mestinya. Bila ditemukan hal-hal yang tidak sesuai dengan isi pernyataan ini, saya menerima dan menghormati segala konsekuensi akademis yang diberikan Program Teknik Komputer Universitas Syiah Kuala.

|  |
| --- |
| Banda Aceh, September 2022  Al Yafi  NIM : 1904111010051 |

# **PENGESAHAN PROPOSAL**

**APLIKASI MOBILE IDENTIFIKASI AUTISM SPECTRUM DISORDER (ASD) DENGAN BERBASIS CONVOLUTIONAL NEURAL NETWORK**

**Diajukan Oleh :**

**Al Yafi**

**1904111010051**

|  |  |  |
| --- | --- | --- |
| Telah disetujui oleh  Komisi Pembimbing  Banda Aceh, 2022 | | |
| Pembimbing I  Dr. Ir. Melinda, S.T., M.Sc., IPU., ASEAN Eng.  NIP. 19790610200212201 | Pembimbing II  Prof. Dr. Fitri Arnia, S.T., M.Eng.Sc NIP. 199010252020072101 |
| Mengetahui,  Koordinator Program Studi Teknik Komputer  Dr. Kahlil Muchtar, S.T., M.Eng  NIP. 198512022017091101 | | |

# **ABSTRAK**

*Autism* *Spectrum* *Disorder* (ASD) merupakan sebuah kondisi neurodevelopmental yang dicirikan oleh kesulitan dalam interaksi sosial, komunikasi, dan perilaku yang repetitif. Ini merupakan spektrum kondisi, dengan tingkat keparahan yang bervariasi dari individu ke individu. Beberapa orang dengan ASD mungkin memiliki keterbatasan yang signifikan dalam berkomunikasi dan bersosialisasi, sementara yang lain mungkin memiliki keterbatasan yang lebih ringan dan mampu bekerja dan berfungsi secara independen. Diagnosis ASD dibuat oleh profesional kesehatan mental berdasarkan evaluasi perilaku dan gejala yang dicatat dalam waktu yang cukup lama sehingga dibutuhkan diagnosis terhadap anak penderita ASD dengan perlakuan medis yang cepat dan tepat sasaran. EEG atau Electroencephalography adalah sebuah metode pemeriksaan yang digunakan untuk memantau aktivitas listrik di dalam otak. Dalam konteks *Autism* *Spectrum* *Disorder* (ASD), data EEG dapat digunakan untuk membantu dalam diagnosis atau evaluasi kondisi ini. Saat ini fasilitas diagnosis dini penderita ASD dengan sinyal *Electroencephalogram* (EEG) masih terbatas, sehingga sehingga dibutuhkan biaya yang besar untuk mengakses fasilitas tersebut. Penelitian ini mengusulkan sebuah Aplikasi *Mobile* yang yang diharapkan dapat meningkatkan akses bagi penderita ASD untuk mendiagnosis data EEG secara mandiri. Aplikasi *Mobile* yang dibangun dapat membedakan penderita ASD dan normal berdasarkan data sinyal otak EEG secara *asynchronous*. Selain itu, penelitian ini menghasilkan algoritma *preprocessing* dan data sinyal EEG BCI2000 sehingga dapat diautomasi menggunakan *python*. Penelitian ini juga menghasilkan luaran model terlatih *Deep Learning* *Convolutional Neural Network* yang di-*deploy* menggunakan Python-Flask sehingga diagnosis sinyal EEG penderita ASD dan normal dapat digunakan diberbagai *platform* melalui restAPI.

# **DAFTAR ISI**

[PERNYATAAN KEASLIAN PROPOSAL i](#_Toc127140173)

[PENGESAHAN PROPOSAL ii](#_Toc127140174)

[ABSTRAK iii](#_Toc127140175)

[DAFTAR ISI iv](#_Toc127140176)

[DAFTAR TABEL vii](#_Toc127140177)

[DAFTAR SINGKATAN viii](#_Toc127140178)

[BAB 1 PENDAHULUAN 1](#_Toc127140179)

[**1.1** **LATAR BELAKANG** 1](#_Toc127140180)

[**1.2** **RUMUSAN MASALAH** 3](#_Toc127140181)

[**1.3** **RUANG LINGKUP** 3](#_Toc127140182)

[**1.4** **TUJUAN** 4](#_Toc127140183)

[**1.5** **URGENSI DAN MANFAAT PENELITIAN** 4](#_Toc127140184)

[**1.6** **SISTEMATIKA PENULISAN** 4](#_Toc127140185)

[BAB 2 TINJAUAN PUSTAKA 6](#_Toc127140186)

[**2.1** ***AUTISM SPECTRUM DISORDERS (ASD)*** 6](#_Toc127140187)

[**2.2** ***ELECTROENCEPHALOGRAPHY* (EEG)** 6](#_Toc127140188)

[**2.3** **KARAKTERISTIK GELOMBANG OTAK** 7](#_Toc127140189)

[**2.4** ***BRAIN COMPUTER INTERFACE* (BCI)** 9](#_Toc127140190)

[**2.5** **BCI2KREADER** 10](#_Toc127140191)

[**2.6** ***PRE-PROCESSING* *DATASET*** 11](#_Toc127140192)

[**2.7** ***DEEP LEARNING*** 13](#_Toc127140193)

[**2.7.1 TensorFlow 13**](#_Toc127140194)

[**2.7.2 Convolutional Neural Network (CNN) 14**](#_Toc127140195)

[**2.7.3 Dropout 16**](#_Toc127140196)

[**2.7.4 Loss dan Akurasi 17**](#_Toc127140197)

[**2.8** **ARSITEKTUR SISTEM APLIKASI MOBILE** 18](#_Toc127140198)

[**2.8.1** **Layanan Firebase** 18](#_Toc127140199)

[**2.8.2** **Python Flask** 19](#_Toc127140200)

[**2.8.3** **Flutter** 20](#_Toc127140201)

[**2.9** **UJI PEFORMA SISTEM** 20](#_Toc127140202)

[**2.9.1** **Convusion Matrix** 20](#_Toc127140203)

[**2.9.2** **Uji Perfroma Mobile Apps** 21](#_Toc127140204)

[**2.10** **PENELITIAN TERKAIT** 22](#_Toc127140205)

[BAB 3 METODELOGI DAN JADWAL PENELITIAN 27](#_Toc127140206)

[**3.1** **METODE PENELITIAN** 27](#_Toc127140207)

[**3.1.1** **Bahan Penelitian** 27](#_Toc127140208)

[**3.1.2** **Alat Penelitian** 28](#_Toc127140209)

[**3.1.3** **Alur Penelitian** 29](#_Toc127140210)

[**3.1.4** **Identifikasi Masalah** 30](#_Toc127140211)

[**3.1.3 Studi Literatur** 30](#_Toc127140212)

[**3.1.5** **Sumber Dataset** 30](#_Toc127140213)

[**3.1.6** **Preprocessing menggunakan Python Brain Computer Interface 2000 (BCI2K) dan CV2** 30](#_Toc127140214)

[**3.1.7** **Klasifikasi sinyal EEG menggunakan Convolutional Neural Network (CNN)** 31](#_Toc127140215)

[**3.1.8** **Membangun Cloud Classifier Engine Environtment menggunakan Flask dan Firebase** 33](#_Toc127140216)

[**3.1.9** **Membangun Aplikasi Mobile** 34](#_Toc127140217)

[**3.1.10** **Analisis Peforma Sistem** 35](#_Toc127140218)

[**3.2** **JADWAL PENELITIAN** 36](#_Toc127140219)

[BAB 4 LUARAN 39](#_Toc127140220)

[DAFTAR PUSTAKA 40](#_Toc127140221)

**DAFTAR GAMBAR**

# **DAFTAR TABEL**

# **DAFTAR SINGKATAN**

Autism Spectrum Disorder : (ASD)

Brain Computer Interface : (BCI)

Electroencephalograph : (EEG)

False Positif : (FP)

False Negatif : (FN)

King Abdulaziz University : (KAU)

Machine Learning : (ML)

Support Vector Machine : (SVM)

True Positif : (TP)

Convolutional Neural Network : (CNN)

True Negatif : (TN)

# **BAB 1 PENDAHULUAN**

## **LATAR BELAKANG**

Diagnosis penderita *Autism Spectrum Disorder* (ASD) pada anak umumnya hanya mengandalkan beberapa parameter pemeriksaan fisik, prilaku, dan pemeriksaan tumbuh kembang anak, pemeriksaan ini meliputi dengan memeriksa keluhan utama seperti keterlambatan dalam perkembangan bahasa, regresi kemampuan berbahasa, keterlambatan dalam kemampuan komunikasi, perilaku abnormal, bentuk fisik, prilaku dan sebagainya [24]. Model pemeriksaan ini akan membutuhkan waktu lama karena harus memperhatikan tumbuh kembang anak dan harus memerlukan usaha yang lebih untuk memastikan bahwa seorang anak menderita ASD atau tidak. Interpretasi sinyal EEG secara manual dapat menjadi sulit karena beberapa hal berikut:

1. Keanekaragaman pola aktivitas otak: Sinyal EEG dapat menunjukkan pola aktivitas otak yang sangat beragam, dan ini bisa membuat sulit untuk menentukan apa yang normal dan apa yang anormal [3].
2. Variasi individual: Setiap individu memiliki pola aktivitas otak yang unik dan berbeda, sehingga sulit untuk membandingkan sinyal EEG antar individu dan menentukan apa yang normal dan apa yang anormal [3].
3. Kebisingan: Sinyal EEG dapat terkontaminasi oleh kebisingan dari luar seperti gerakan otot, elektromagnetik, atau sumber-sumber eksternal lainnya. Ini membuat sulit untuk membedakan antara sinyal EEG yang benar-benar mencerminkan aktivitas otak dan sinyal yang terkontaminasi [3].
4. Kemampuan interpretasi yang berbeda: Interpretasi sinyal EEG sangat bergantung pada keahlian dan pengalaman individu yang melakukannya. Beberapa orang mungkin memiliki kemampuan interpretasi yang lebih baik dibandingkan orang lain, dan ini bisa mempengaruhi kualitas interpretasi yang diperoleh [3].

Karena alasan-alasan ini, interpretasi sinyal EEG secara manual membutuhkan keterampilan dan pengalaman yang tinggi, dan seringkali harus dikonfirmasi oleh analisis komputer untuk memastikan akurasi dan validitas hasil.

Beberapa teknologi canggih saat ini digunakan untuk membantu dalam mendiagnosis Autism Spectrum Disorder (ASD) melalui EEG (Electroencephalography), di antaranya:

1. Analisis EEG otomatis: Ini adalah teknologi yang menggunakan algoritma komputer untuk menganalisis sinyal EEG dan mengidentifikasi pola aktivitas otak yang mungkin terkait dengan ASD. Analisis otomatis ini dapat membantu dalam mengurangi variasi individual dan meningkatkan akurasi dalam interpretasi sinyal EEG [4].
2. Brain-Computer Interface (BCI): BCI adalah teknologi yang menghubungkan otak dan komputer, memungkinkan sinyal otak untuk dipantau dan dianalisis secara real-time. Dalam konteks ASD, BCI dapat digunakan untuk memantau dan menganalisis respon otak terhadap rangsangan sosial dan lingkungan, membantu dalam mengidentifikasi pola aktivitas otak yang anormal [4].
3. Artificial Intelligence (AI): AI adalah teknologi yang menggunakan algoritma komputer untuk menganalisis dan mempelajari data secara otomatis. Dalam konteks ASD, AI dapat digunakan untuk membantu dalam analisis EEG, meningkatkan akurasi dalam mengidentifikasi pola aktivitas otak yang mungkin terkait dengan kondisi ini [4].

Teknologi-teknologi ini membantu dalam mempercepat dan mempermudah proses diagnostik dan membantu dalam membuat diagnosis yang lebih akurat dan cepat. Namun, harus diingat bahwa teknologi ini hanya memberikan informasi tambahan dan bahwa diagnosis akhir harus dibuat oleh profesional kesehatan mental berdasarkan evaluasi perilaku dan gejala yang dicatat dalam waktu yang cukup lama. Namun hingga saat ini, masih sedikit teknologi yang dapat menganalisis sinyal EEG dan mendiagnosis penderita ASD secara otomatis sehingga membutuhkan bannyak biaya dan waktu dan membuat banyak anak penderita ASD tidak menyadari penyakit ini secara dini dan menimbulkan dampak buruk bagi perkembangannya [4].

Penelitian ini mengusulkan sebuah aplikasi mobile yang dapat membedakan penderita ASD dan normal berdasarkan data sinyal otak EEG secara *asynchronous* artinya pengguna hanya perlu memiliki data EEG yang sudah direkam kemudian meng-*input* data tersebut menggunakan *Smartphone* dan informasi hasil diagnosis akan ditampilkan di *Smartphone*. Penelitian ini menghasilkan algoritma *preprocessing* dan data sinyal EEG BCI2000 menggunakan *library python* BCI2K yang memungkinkan *preprocessing* data EEG diautomasi menggunakan *python*. Selain itu, penelitian ini menghasilkan luaran model terlatih *Deep Learning* dengan menggunakan metode *Convolutional Neural Network* (CNN) yang di-*deploy* menggunakan Python-Flask sehingga diagnosis penderita ASD dan normal berdasarkan sinyal EEG dapat digunakan diberbagai *platform* melalui restAPI, tetapi penelitian ini hanya difokuskan pada penggunakan *platform Android* menggunakan multi-platform *Framework* yakni Flutter. Penelitian ini juga diharapkan membantu masyarakat sehingga mempercepat diagnosa penderita ASD agar dapat ditangani secara tepat sasaran. Data penderita ASD dan data normal diklasifikasikan dengan benar ke kelas yang sesuai. Penanganan penyakit ini membutuhkan kerja sama yang erat dan terpadu sehingga klasifikasi ASD ini akan sangat membantu penderita dan dapat membuat diagnosis dalam waktu yang lebih cepat memungkinkan penderita memperoleh penanganan dan terapi secara tepat sasaran.

## **RUMUSAN MASALAH**

Rumusan masalah pada penelitian ini adalah:

1. Tenaga kesehatan/dokter kesulitan mengnalisis data EEG secara manual dan analisis dilakukan berdasarkan evaluasi perilaku dan gejala yang dicatat dalam waktu yang cukup lama.
2. Untuk beberapa format data EEG dibutuhkan metode preprocessing khusus dan tidak otomatis.
3. Implementasi model *Deep Learning* terlatih yang telah diautomasi pada Aplikasi *Mobile*.

## **RUANG LINGKUP**

Adapun ruang lingkup pada penelitian ini adalah membuat aplikasi *mobile* yang dapat mengidentifikasi penderita ASD secara *asynchronous*, membangun Deep Learning model untuk klasifikasi penderita ASD dan membuat arsitektur *Cloud* untuk memungkinkan menggunakan model *Deep* *Learning* pada aplikasi *mobile*.

## **TUJUAN**

Tujuan pada Penelitan ini adalah :

1. Merancang dan membangun Aplikasi *Mobile* berbasis android yang dapat membedakan data sinyal EEG normal dan ASD secara *Asynchronous*.
2. Membangun algoritma *preprocessing* sinyal EEG BCI200 menggunakan pemrograman pyhton.
3. Membangun sistem automasi klasifikasi sinyal EEG menggunakan model *Deep Learning* dengan metode *Convolutional* *Neural* *Network* (CNN) berbasis *Framework* *Tensorflow.*

## **URGENSI DAN MANFAAT PENELITIAN**

Adapun manfaat yang diharapkan dari penelitian ini yaitu dapat mendiagnosis sinyal EEG antara ASD dan Normal menggunakan aplikasi mobile secara *asynchronous* yang memungkinkan penggunaan aplikasi dan alat EEG tidak perlu digunakan secara bersamaan sehingga menurunkan biaya operasional untuk melakukan diagnosis. Penelitian ini juga diharapkan membantu masyarakat dan mempercepat diagnosis ASD dan dapat ditangani secara tepat sasaran.

## **SISTEMATIKA PENULISAN**

Penelitian Tugas Akhir ini terdiri dari komponen-komponen beserta tahapan dan penjelasan pada setiap bab dan subbabnya. Adapun sistematika penulisan pada tugas akhir ini adalah sebagai berikut:

BAB 1 PENDAHULUAN

Pada bab ini berisi tentang latar belakang, rumusan masalah, tujuan, urgensi dan manfaat penelitian penelitian dan sistematika penulisan penelitian tugas akhir membahas pengimplementasian model *Deep Learning* dan ektraksi model yang akan digunakan restAPI agar memungkinkan dilakukannya identifikasi ASD dan Normal menggunakan aplikasi *mobile*.

BAB 2 TINJAUAN PUSTAKA

Pada bab ini berisi tentang teori yang berkaitan dengan ASD, arsitektur aplikasi seacra keseluruhan seperti sistem klasifikasi menggunakan *Deep Learning* dengan metode CNN (Convolutional Neural Network), dan Cloud Service untuk penerapan model *Deep Learning* pada aplikasi mobile.

BAB 3 METODOLOGI DAN JADWAL PENELITIAN

Bab ini berisi penjabaran metode penelitian yang digunakan secara lebih rinci. Bab ini juga menjelaskan tentang model yang digunakan untuk klasifikasi citra yang menggunakan *Convolutional Neural Network* (CNN), serta tahapan penelitian, bahan dan alat, dan jadwal penelitian.

BAB 4 LUARAN PENELITIAN

Pada bab ini berisi tentang hasil dan kesimpulan dari penelitian.

DAFTAR PUSTAKA

Bab ini berisi referensi yang digunakan pada penelitian tugas akhir ini. Referensi ini didapat dari berbagai sumber seperti buku, dan jurnal-jurnal penelitian, referensi online.

# **BAB 2 TINJAUAN PUSTAKA**

## ***AUTISM SPECTRUM DISORDERS (ASD)***

Autism Spectrum Disorder (ASD) adalah kondisi neurodiversitas yang dicirikan oleh pola perilaku, komunikasi, dan interaksi sosial yang abnormal. Kondisi ini biasanya ditemukan sejak masa kecil dan dapat mempengaruhi banyak aspek dari kehidupan seseorang, termasuk keterampilan berkomunikasi, kemampuan untuk membangun dan mempertahankan hubungan sosial, dan kemampuan untuk belajar dan beradaptasi dengan lingkungan baru [3].

ASD berada di spektrum, artinya bahwa gejala dan tingkat keparahan dapat bervariasi dari individu ke individu. Beberapa individu dengan ASD mungkin memiliki keterampilan berkomunikasi yang baik dan mampu belajar dan beradaptasi dengan lingkungan dengan baik, sementara yang lain mungkin memiliki keterbatasan yang signifikan dalam hal ini [5].

Faktor yang mempengaruhi pengembangan ASD belum sepenuhnya diketahui, namun penelitian menunjukkan bahwa kombinasi faktor genetik dan lingkungan mungkin memainkan peran. Beberapa tipe terapi dan intervensi, seperti terapi bahasa, terapi perilaku, dan terapi sosial, dapat membantu individu dengan ASD untuk meningkatkan keterampilan dan memfasilitasi interaksi sosial yang lebih baik [5].

Diagnosis ASD dapat dibuat oleh profesional kesehatan mental melalui evaluasi perilaku dan gejala yang dicatat dalam waktu yang cukup lama, dan dapat memerlukan beberapa tes tambahan, seperti tes neuropsikologis dan tes EEG. Diagnosis dini dan intervensi yang tepat dapat membantu individu dengan ASD untuk mencapai potensinya dan meningkatkan kualitas hidup mereka [6].

## ***ELECTROENCEPHALOGRAPHY* (EEG)**

Electroencephalogram (EEG) adalah teknik pengukuran yang digunakan untuk memantau aktivitas listrik dalam otak. Ini dilakukan dengan menempatkan elektroda pada kepala individu dan merekam sinyal elektrik yang dihasilkan oleh sel-sel saraf di dalam otak.

EEG merupakan salah satu alat yang paling penting dalam neurodiagnostik, dan sering digunakan untuk membantu menentukan diagnosa kondisi medis seperti epilepsy, insomnia, dan cedera otak. Dalam konteks autisme, EEG dapat digunakan untuk memantau aktivitas listrik dalam otak individu dengan diagnosa ASD dan membandingkannya dengan individu tanpa diagnosa untuk mencari perbedaan yang signifikan.

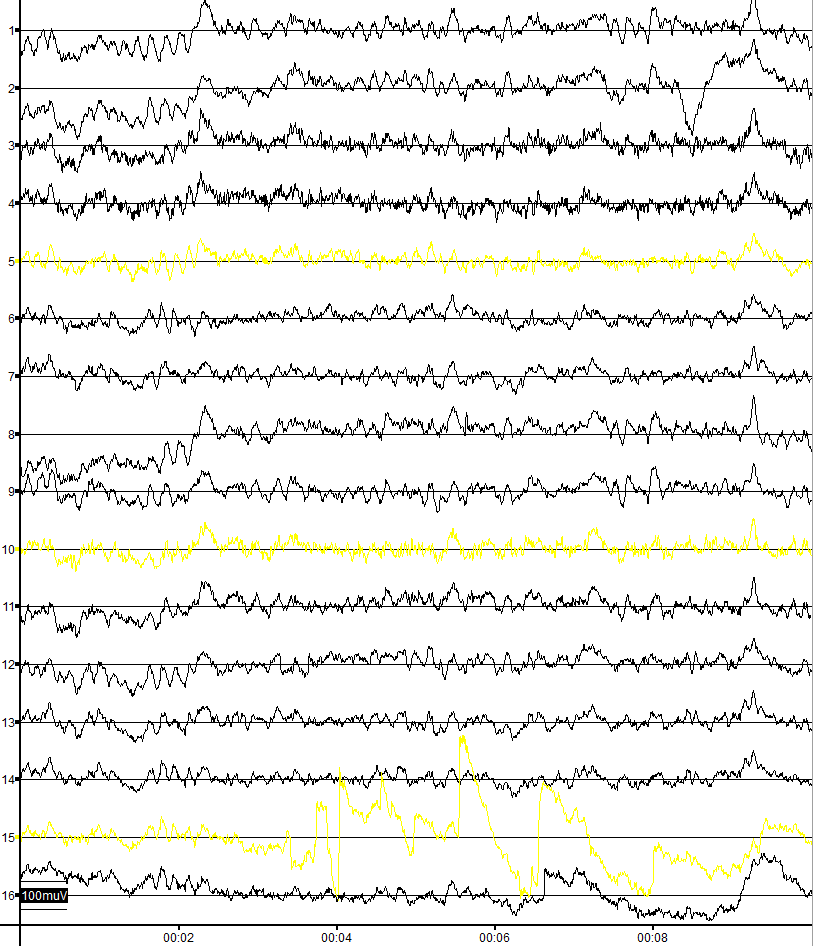
Proses EEG sangat non-invasif dan tidak menyakitkan, dan umumnya memakan waktu sekitar satu jam. Elektroda yang digunakan dalam proses ini biasanya ditempelkan pada kepala dengan menggunakan gel elektrolitik, dan sinyal yang dihasilkan diterjemahkan menjadi pola gelombang yang dapat dianalisis oleh profesional kesehatan untuk menentukan apakah ada perbedaan dalam aktivitas listrik dalam otak.

Meskipun EEG adalah alat yang berguna dan penting dalam proses diagnostik, interpretasi sinyal EEG tidak selalu mudah dan memerlukan interpretasi yang akurat dari profesional kesehatan yang berpengalaman. Oleh karena itu, hasil EEG harus diinterpretasikan dengan hati-hati dan dalam konteks yang lebih luas dari gejala dan riwayat medis individu.

## **KARAKTERISTIK GELOMBANG OTAK**

Gelombang otak adalah sinyal elektrik yang dihasilkan oleh aktivitas sel saraf di dalam otak. Ada beberapa jenis gelombang otak yang dikenali, masing-masing memiliki frekuensi dan amplitudo yang berbeda. Gelombang delta adalah gelombang otak dengan frekuensi terendah dan amplitudo tertinggi. Ini biasanya terjadi pada saat tidur dan menunjukkan aktivitas listrik yang sangat rendah dalam otak. Gelombang theta adalah gelombang otak dengan frekuensi sedang dan amplitudo sedang. Ini biasanya terjadi pada saat tidur lelap atau meditasi dan menunjukkan sedikit aktivitas listrik dalam otak. Gelombang alfa adalah gelombang otak dengan frekuensi sedang dan amplitudo rendah. Ini biasanya terjadi pada saat relaksasi dan menunjukkan sedikit aktivitas listrik dalam otak. Gelombang beta adalah gelombang otak dengan frekuensi tertinggi dan amplitudo rendah. Ini biasanya terjadi pada saat konsentrasi dan menunjukkan aktivitas listrik yang sangat tinggi dalam otak. Gelombang gamma adalah gelombang otak dengan frekuensi yang sangat tinggi dan amplitudo yang sangat rendah. Ini menunjukkan aktivitas listrik yang sangat kompleks dan intens dalam otak. Masing-masing jenis gelombang otak memiliki karakteristik yang berbeda dan dapat memberikan informasi tentang bagaimana otak berfungsi. Oleh karena itu, analisis gelombang otak dapat digunakan untuk memahami kondisi medis seperti epilepsy, insomnia, dan cedera otak [24].

Aktivitas otak manusia dapat diperoleh dari perekam sinyal listrik di otak yang dikenal sebagai electroencephalography (EEG). Ketika sel saraf diaktifkan secara elektrik atau neurologis, EEG mendeteksi potensi dalam tubuh yang dihasilkan oleh neuron di otak melalui jaringan sel saraf [7].



Gambar 2.1 contoh sinyal EEG pada penderita ASD [17]

Gambar 2.1 menunjukkan hasil percobaan sinyal EEG yang pernah dilakukan untuk mengetahui fungsionalitas otak maupun gangguan yang terjadi pada sistem neurologis otak anak autis.

## ***BRAIN COMPUTER INTERFACE* (BCI)**

Brain-Computer Interface (BCI) adalah sistem teknologi yang memungkinkan manusia untuk berinteraksi dengan komputer dengan menggunakan aktivitas otak mereka sebagai input. BCI mengambil sinyal elektroencephalography (EEG) dari otak dan mengonversinya menjadi perintah untuk computer [15].

BCI dapat digunakan untuk berbagai tujuan, seperti membantu penderita kelumpuhan atau gangguan bicara untuk berinteraksi dengan lingkungan mereka, atau membantu atlet profesional meningkatkan kinerja mereka dengan memanfaatkan sinyal otak mereka. BCI juga memiliki potensi untuk digunakan dalam bidang militer, kedokteran, dan hiburan [16].

BCI mengandalkan pada prinsip dasar dari neuroplastisitas, yaitu kemampuan otak untuk memodifikasi dan memperluas jaringan saraf seiring dengan latihan dan pengalaman. Dengan melatih otak untuk mengirimkan sinyal yang spesifik dan terfokus, seseorang dapat membangun kemampuan untuk mengendalikan perangkat BCI dengan aktivitas otak mereka [15].

Meskipun BCI memiliki potensi yang sangat besar, teknologi ini masih dalam tahap awal dan memiliki beberapa kendala, seperti kurangnya akurasi dan keandalan, dan masalah teknis yang masih perlu diselesaikan [16]. Namun, dengan terus berkembangnya penelitian dan pengembangan, diharapkan BCI akan menjadi lebih efektif dan dapat digunakan secara luas dalam berbagai aplikasi.

Data EEG yang digunakan pada penelitian ini merupakan data yang berformat BCI2000 dengan extensi “.dat” sehingga membutuhkan modul khusus untuk membaca data tersebut menggunakan bahasa pemrograman tertentu [8]. Pada matlab data ini dapat dibaca menggunakan modul EEG Lab namun sulit diimplementasi kedalam system android. Namun pada pemrograman python, sejauh ini hanya ada satu modul yang dapat membaca data ini yakni BCI2kReader yang masih dalam tahap pengembangan lebih lanjut sehingga pemakaian modul ini diperlukan kehati-hatian dalam penggunaanya [14]. Maka dari itu penelitian ini menggunakan pemrograman python yang lebih fleksibel dan dapat dimplementasikan di berbagai platform dan mempermudah automasi seluruh proses seperti *preprocessing* hingga klasifikasi data EEG.

## **BCI2KREADER**

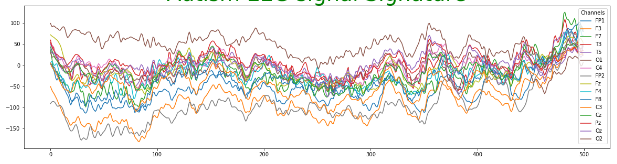
BCI2000Reader adalah sebuah paket Python yang digunakan untuk membaca dan menganalisis data BCI2000. BCI2000 adalah sebuah sistem open-source untuk pengambilan dan pemrosesan data BCI yang digunakan oleh banyak peneliti dalam bidang neuroteknologi. Dengan menggunakan BCI2000Reader, peneliti dapat dengan mudah mengimpor data BCI2000 ke dalam lingkungan Python mereka dan menganalisisnya menggunakan berbagai library analisis data populer, seperti NumPy, Pandas, dan Matplotlib. BCI2000Reader memiliki antarmuka yang mudah digunakan dan memungkinkan peneliti untuk membaca data EEG, stimulus, dan respon dari file BCI2000 [14].

BCI2000Reader juga memiliki beberapa fitur canggih, seperti pemrosesan data online dan offline, dukungan untuk format data yang berbeda, dan dukungan untuk beberapa jenis peralatan BCI. Hal ini membuat BCI2000Reader menjadi pilihan yang baik bagi peneliti yang ingin memproses dan menganalisis data BCI2000 mereka dengan efisien [14].

BCI2000Reader sangat berguna bagi peneliti yang menggunakan data BCI2000 dalam penelitian mereka, karena memungkinkan mereka untuk memproses dan menganalisis data secara cepat dan efisien. Ini juga membantu memastikan konsistensi dan akurasi data, sehingga membuat hasil penelitian lebih valid dan dapat dipercaya. Oleh karena itu, BCI2000Reader merupakan paket yang sangat berguna bagi para peneliti dalam bidang neuroteknologi [15].

Untuk membaca data BCI2000 yang berformat “.dat”. dalam bahasa pemrograman Python, digunakan modul BCI2KReader. Karenamodul ini masih dalam pengembangan, fungsi-fungsi yang tersedia pada modul ini masih sedikit yakni hanya sebatas membaca, dan memotong sinyal BCI2000 [27].

Pada gambar 2.3, pembacaan sinyal EEG BCI2000 menggunakan modul Python Yakni BCI2kReader mengembalikan data dalam bentuk *Channel* dan *Data* *Point* dalam format Numpy matrix [27, 28].



Gambar 2.3 contoh hasil pembacaan sinyal EEG BCI2000 dengan modul Python yakni BCI2kReader [27]

## ***PRE-PROCESSING* *DATASET***

digunakan pada penelitian ini merupakan data yang berformat BCI2000 dengan extensi “.dat” sehingga membutuhkan modul khusus yakni BCI2kReader untuk membaca data tersebut menggunakan bahasa pemrograman python. Bentuk hasil pembacaan data EEG menggunakan modul BCI2kReader ini adalah data dengan tipe Numpy *Matrix* dimana data berbentuk matriks dua dimensi dengan jumlah *Collums* sebanyak channel yang terdapat pada alat EEG serta jumlah *Rows* sebanyak *time-steps* atau lama waktu perekaman data EEG tersebut dalam satuan detik [14].

Agar data dapat diklasifikasi menggunakan Convolutional Neural Network (CNN), maka data matrix EEG ini akan dirubah kedalam bentuk citra RGB. Untuk mengubah matrix menjadi citra RGB, hal yang pertama dilakukan adalah dengan memahami dasar-dasar citra digital. Citra digital adalah representasi visual dari data numerik yang disimpan sebagai matrix (juga dikenal sebagai grid) dari nilai-nilai piksel. Piksel adalah singkatan dari "picture element " dan merepresentasikan unit terkecil dari sebuah citra [2].

Dalam citra RGB, setiap piksel memiliki tiga nilai intensitas warna: merah (R), hijau (G), dan biru (B). Setiap nilai intensitas warna tersebut dapat berkisar antara 0 hingga 255, dengan 0 menunjukkan tidak adanya warna dan 255 menunjukkan warna maksimal [2].

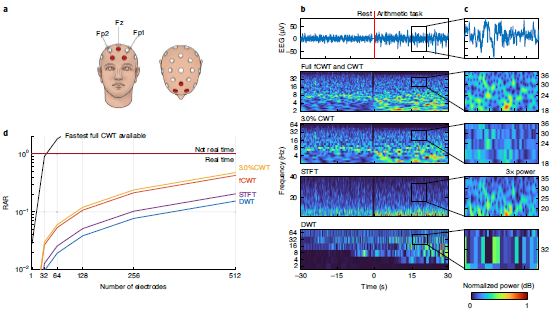
Untuk mengubah matrix menjadi citra RGB, hal yang perlu dilakaukan adalah dengan mengaitkan setiap elemen dari matrix dengan nilai intensitas warna R, G, atau B untuk setiap piksel. Ini bisa dilakukan dengan memasukkan nilai-nilai dari matrix ke dalam format data yang sesuai untuk citra RGB, seperti file gambar dengan ekstensi .jpeg atau .png [2].

Beberapa bahasa pemrograman seperti Python memiliki library seperti OpenCV dan Pillow yang dapat membantu Anda dalam memproses dan menyimpan citra digital. Anda dapat menggunakan fungsi seperti cv2.imwrite() atau Image.fromarray() untuk mengubah matrix menjadi citra RGB dan menyimpannya sebagai file gambar [2].

Merubah matrix menjadi bentuk citra RGB memiliki beberapa manfaat, antara lain:

1. Visualisasi data: Merubah matrix menjadi citra RGB memungkinkan untuk melihat data dalam bentuk visual yang lebih mudah dipahami oleh manusia. Ini memudahkan Anda untuk melihat pola dan tren dalam data yang mungkin tidak terlihat dalam bentuk angka-angka saja [9].
2. Analisis citra: Setelah matrix diubah menjadi citra RGB, dapat menggunakan berbagai teknik analisis citra untuk mengekstrak informasi dan mengambil keputusan berdasarkan data yang ada [9].
3. Penyimpanan dan pembagian data: Merubah matrix menjadi citra RGB membuat data lebih mudah disimpan dan dibagikan. Citra digital bisa disimpan sebagai file gambar dan dibagikan dengan mudah melalui internet atau media elektronik lainnya [9].
4. Kompresi data: Merubah matrix menjadi citra RGB juga memungkinkan untuk mengompresi data sehingga lebih mudah disimpan dan dibagikan. Ada berbagai metode kompresi citra yang dapat digunakan, seperti kompresi lossless dan lossy [9].

Dengan demikian, merubah matrix menjadi citra RGB membantu dalam membuat data lebih mudah dipahami, dianalisis, disimpan, dan dibagikan. Ini bisa sangat berguna dalam berbagai aplikasi, seperti pemrosesan citra digital, visualisasi data, dan banyak lagi [9].

 Gambar 2.2 Merubah sinyal ke bentuk Time Frequency Spectrum Image [1]

## ***DEEP LEARNING***

*Deep* *learning* adalah subbidang pembelajaran mesin yang algoritmanya terinspirasi oleh struktur otak manusia. Struktur ini disebut jaringan saraf tiruan atau *Artificial* *Neural* *Networks* (ANN). Pada dasarnya *Deep Learning* adalah jaringan saraf dengan tiga atau lebih lapisan ANN. Itu dapat belajar dan beradaptasi dengan sejumlah besar data dan memecahkan masalah yang sulit dipecahkan dengan algoritme pembelajaran mesin lainnya [5].

Penelitian ini menggunakan metode Deep learning untuk mengklasifikasikan data sinyal EEG dari subjek dan memberikan diagnosis apakah subjek memiliki gejala ASD atau tidak. Penelitian ini juga menggunakan modul *TensorFlow* [35] dalam menggunakan deep learning agar mudah diautomasi dan diterapkan ke berbagai platform.

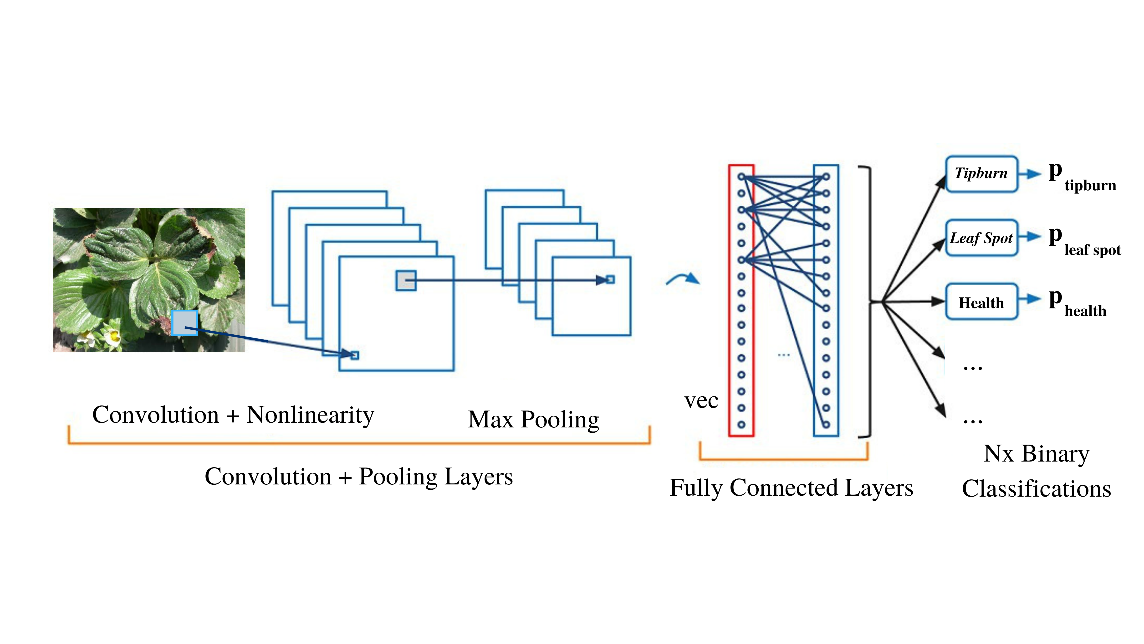
### **TensorFlow**

TensorFlow adalah platform open source end-to-end untuk machine learning. Ini memiliki ekosistem alat, pustaka, dan sumber daya komunitas yang komprehensif dan fleksibel yang memungkinkan peneliti mendorong Machine Learning mutakhir dan pengembang dengan mudah membangun dan menerapkan aplikasi yang didukung *Machine* *Learning* [35]. TensorFlow menawarkan beberapa tingkat abstraksi sehingga tool ini merupakan tool yang tepat untuk kebutuhan penelitian ini. Membangun dan melatih model dengan menggunakan *Keras* API [36] tingkat tinggi yang memudahkan penggunaan machine learning untuk penelitian agar dapat diautomasi, dan diterapkan diberbagai platform.

Penelitian ini menggunakan modul tensorflow untuk mendesain, membangun, menjalankan, dan pengimplementasikan model *Deep Learning* untuk mencapai tujuan utama pada penelitian ini yakni mengklasifikasikan sinyal EEG dan memberikan diagnosis dini untuk mengetahui subjek menderita ASD atau tidak secara otomatis.

### **Convolutional Neural Network (CNN)**

*Convolutional Neural Network* (CNN) adalah bagian dari jaringan saraf dalam, yang merupakan jaringan saraf tiruan yang banyak digunakan dalam pengenalan dan pemrosesan gambar. Algoritma ini dibuat terutama untuk menangani data piksel dan gambar visual. Neuron dari algoritma *Convolutional Neural Network* dimaksudkan untuk berfungsi mirip dengan *lobus frontal*, terutama daerah korteks visual, pada otak manusia dan hewan. Korteks visual adalah area otak yang bertanggung jawab untuk memproses informasi dalam bentuk *input* visual. Inilah yang membedakan CNN dengan teknik *neural network* lainnya dalam pengolahan citra [26]. Contohnya termasuk pengenalan wajah, kategorisasi gambar, dan aplikasi CNN lainnya dalam visi komputer. Seperti jaringan saraf sederhana, CNN berisi parameter yang dapat dipelajari, seperti jaringan saraf, yaitu bobot, *bias*, dan lain-lain [27].



Gambar 2.7 proses klasifikasi menggunakan *Convolutional Neural Network* (CNN) [28].

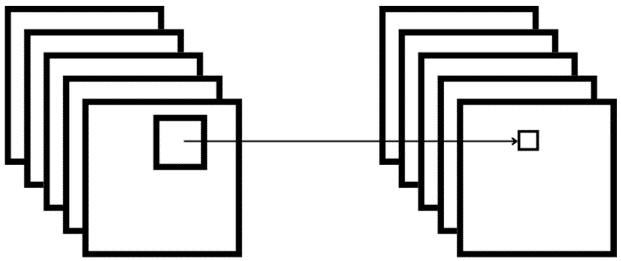
Ada enam lapisan CNN, yaitu:

1. Lapisan masukan

Lapisan input CNN harus berisi data gambar. Data gambar, seperti yang ditetapkan sebelumnya, diwakili oleh matriks tiga dimensi. Itu harus dibentuk kembali menjadi satu kolom [29].

1. Lapisan Konvolusi

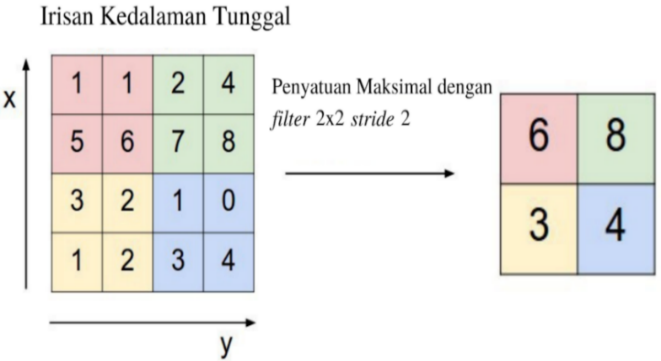
Karena mengekstraksi properti gambar, lapisan konvo juga disebut lapisan ekstraktor fitur. Untuk memulai, bagian dari gambar ditautkan ke lapisan konvoi, yang melakukan operasi konvolusi dan menghitung perkalian titik antara bidang reseptif dan filter. Metode ini menghasilkan bilangan bulat tunggal dari volume keluaran. Filter selanjutnya diteruskan ke bidang reseptif berikutnya dari gambar masukan yang identik dan prosesnya diulang. Pendekatan yang sama diulang sampai seluruh gambar telah diproses. Luaran akan dikirim ke tingkat berikutnya [30].



Gambar 2.9 lapisan konvolusi [31].

1. Lapisan Penyatuan

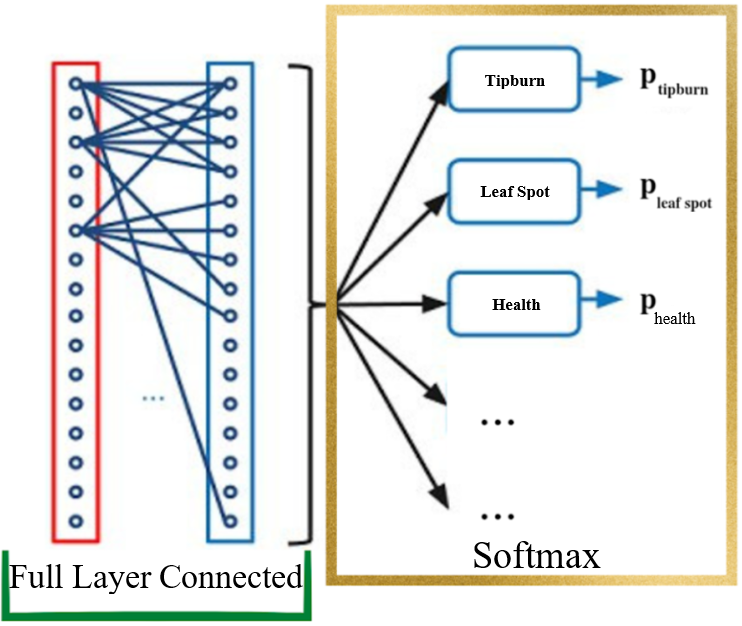
Selama konvolusi, lapisan penyatuan digunakan untuk membantu mengurangi *volume* spasial pada gambar masukan [32].



Gambar 2.10 lapisan penyatuan [31].

1. *Softmax*

 Lapisan terakhir CNN adalah *softmax*, umumnya dikenal sebagai lapisan logistik. Itu ada di bagian paling bawah dari lapisan yang terhubung sepenuhnya. *Softmax* digunakan untuk multi-klasifikasi, sedangkan logistik digunakan untuk klasifikasi biner [33].



Gambar. 2.12 lapisan *softmax* [28].

1. Lapisan Keluaran

Lapisan keluaran adalah semacam regularisasi yang berupaya meminimalkan *overfitting* dengan meningkatkan akurasi pengujian, terkadang dengan mengorbankan akurasi pelatihan. Lapisan *output* dalam arsitektur jaringan kami secara acak mengirimkan *input* dari lapisan sebelumnya ke lapisan berikutnya dengan probabilitas “p” untuk setiap *mini-batch* di *set* pelatihan kami [30]. Lapisan *output* memanfaatkan fungsi kerugian seperti entropi silang kategori *untuk* menghitung kesalahan prediksi. Setelah *forward pass*, *backpropagation* memperbarui bobot dan bias untuk mengurangi kesalahan dan kerugian [34].

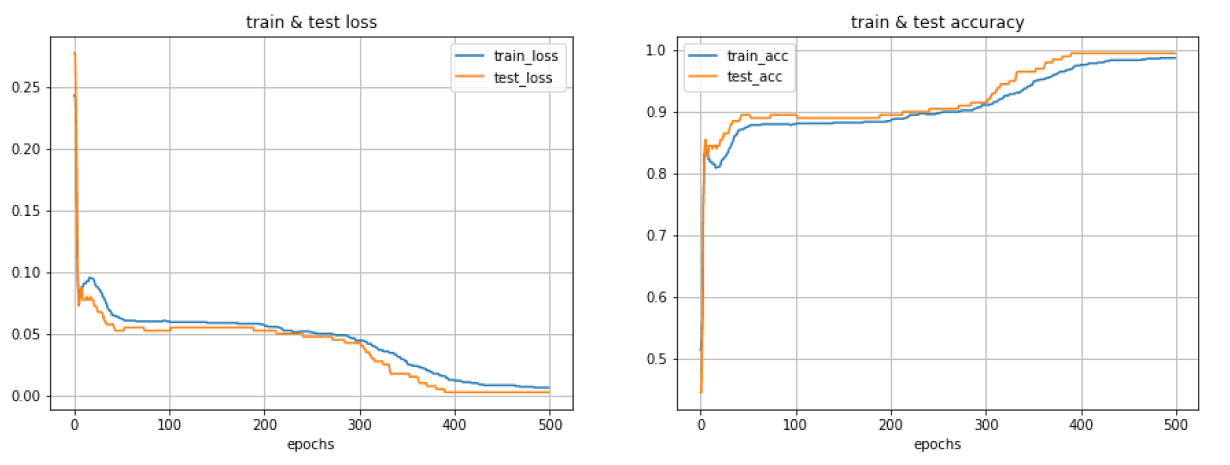
### **Dropout**

*Dropout* adalah pendekatan regularisasi jaringan saraf yang melibatkan pemilihan neuron secara acak. Putus sekolah mengacu pada penghapusan neuron dari lapisan tersembunyi dan terlihat jaringan. Menghapus neuron melibatkan penghapusan sementara dari jaringan yang ada. Neuron yang akan dihapus akan dipilih secara acak. Setiap neuron akan diberi probabilitas antara 0 dan 1. Ini mencegah *overfitting* dan mempercepat proses pelatihan [36].

### **Loss dan Akurasi**

Loss atau kerugian menunjukkan tingkat kerugian pada klasifikasi citra, dimana semakin rendah loss maka model yang dibangun semakin baik. Namun hal tersebut tidak berlaku jika model telah mengalami overfitting. Nilai loss didapatkan dari perhitungan pada training dan validation dan hasilnya menginterpretasikan seberapa bagus model ketika melakukan training dan validation.

Untuk nilai loss tidak dinyatakan dalam persen seperti accuracy, hal tersebut karena berasal dari penjumlahan error dari setiap contoh epoch atau validation sets. Pada dasarnya tujuan model pembelajaran digunakan untuk mengurangi nilai loss sesuai dengan parameter model dengan mengganti nilai weight vector melalui metode optimasi yang berbeda seperti backpropagation pada Neural Network. Jika disimpulkan maka nilai loss menggambarkan seberapa bagus atau buruknya sutau model setelah dilakukan iterasi dari optimasi.



Gambar 2.6 Tampilan loss dan accuracy

Pada Gambar 2.6 menunjukkan grafik hubungan antara loss, accuracy dan epoch. Nilai akurasi dapat diperoleh setelah mempelajari parameter model dan pelatihan lebih lanjut (tidak ada pelatihan yang dilakukan). Sampel uji kemudian dimasukkan ke dalam model, jumlah kesalahan dalam model (kehilangan lingkaran nol) dicatat, dan kemudian dibandingkan dengan target yang sebenarnya. Kemudian tingkat kesalahan klasifikasi dihitung. Misalnya, jika jumlah sampel uji 1000 dan model yang dihasilkan memberikan nilai benar 952, maka nilai akurasi yang diperoleh adalah 95,2% [16].

## **ARSITEKTUR SISTEM APLIKASI MOBILE**

Flutter adalah toolkit UI yang dibuat oleh Google. Ini digunakan untuk membangun aplikasi seluler lintas platform untuk Android dan iOS dan aplikasi desktop untuk Windows, Mac, dan Linux [40]. UI yang dibuat dengan Flutter selalu bergantung pada tumpukan teknologi backend untuk fungsionalitas inti seperti proses klasifikasi sinyal menggunakan *deep learning*. Salah satu teknologi backend tersebut adalah framework Flask dan firebase [15]. Flask adalah kerangka kerja web Python untuk membangun aplikasi web [40].



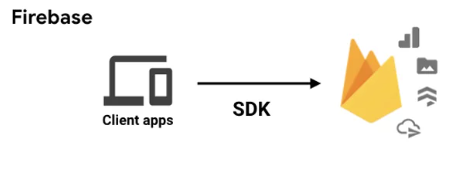
Gambar 2.8 Arsitektur aplikasi mobile menggunakan

Flutter, firebase, Flask [43].

Pada gambar 2.6 diperlihatkan arsitektur aplikasi *mobile* yang menggunakan *framework* Flutter dengan fungsi inti Flask dan Firebase [16,22]. Flask digunakan untuk menjalankan fungsi-fungsi inti dari aplikasi mobile tersebut dan digunakan *Cloud* *service* Firebase untuk menyimpan database.

### **Layanan Firebase**

Firebase adalah layanan milik google yang memiliki beberapa fitur yang dapat membantu developer untuk pengembangan fungsi inti dari sebuah aplikasi. Dengan adanya layanan ini, pengembangan aplikasi dapat lebih difokuskan pada bagian tampilan[42].



Gambar 2.9 Arsitektur hubungan aplikasi klien

dengan firebase [41].

Firebase memiliki beberapa layanan yang sangat berguna bagi pengembangan aplikasi *mobile*. Layanan pada firebase seperti *authentication* [16] membuat aplikasi menjadi lebih aman bagi pengguna. Tidak hanya itu, firebase juga menyediakan *Cloud* *Firestore* dan *Cloud* *Storage* untuk menyimpan *database* dan juga memungkinkan pengguna untuk meng-*upload* semua jenis *file* kedalam penyimpanan berbasis cloud tersebut [16]. Dengan ini, aplikasi mobile yang diintegrasikan dengan firebase akan tetap berjalan dengan kinerja tinggi tanpa harus memakan penyimpanan local tambahan.

### **Python Flask**

Flask merupakan web framework dari bahasa pemrograman python. Flask digunakan sebagai inti dan kerangka kerja aplikasi. Dengan menggunakan Flask dan bahasa python, pengembangan dapat membuat API yang terstruktur dengan mudah dan tidak memerlukan suatu alat atau Pustaka tertentu didalam penggunaannya[21].

Pada penelitian ini, python flask digunakan untuk preprocessing data EEG dan menjalankan model deep learning terlatih untuk klasifikasi sinyal EEG penderita ASD dan normal. Flask juga dapat mengembalikan data berupa restAPI yang dapat diakses diberbagai platform. Pada penelitian ini [22], platform yang digunakan adalah android yang dibangun dengan framework flutter [16].

### **Flutter**

****

Gambar 2.10 Flutter Mobile Apps[14].

Flutter merupakan *framework* open source yang diciptakan oleh google untuk membangun aplikasi yang indah, berjalan dengan *native* dan *multi*-*platform* seperti Android, iOS, web, maupun desktop hanya dengan satu *codebase* [14]. Flutter memiliki dua komponen yakni Software development kit (SDK) yang merupakan tools yang berguna untuk menjalankan aplikasi diberbagai platform, dan Framework UI yang meruapakan komponen tampilan seperti tombol, text, dan lainnya sesuai kustomisasi kebutuhan. Kelebihan Flutter lainnya adalah memiliki kustomisasi yang tidak terlalu rumit sehingga dapat menghemat waktu pengembangan[15].

Pada penelitian ini, flutter akan diintegrasikan dengan *Cloud Service Firebase* [16]dan digunakan untuk membangun sebuah aplikasi mobile dengan kinerja tinggi yang memungkinkan pengguna untuk mengimput data file EEG dengan mudah dan langsung mendapat hasil diagnosis secara cepat.

## **UJI PEFORMA SISTEM**

### **Convusion Matrix**

Confusion Matrix adalah matriks yang menampilkan data Klasifikasi aktual dan prediksi. Matriks konfusi berukuran n x n. dimana n adalah jumlah kelas yang berbeda [38]. Untuk mengevaluasi kinerja metode yang diusulkan, Validasi silang digunakan [35]. Kinerja klasifikasi telah dievaluasi sebagai Parameter seperti karakteristik operasi penerima (ROC) [1] benar positif (TP), benar Negatif (TN), positif palsu (FP), negatif palsu (FN). Sensitivitas, spesifisitas, skor F1, dan akurasi keseluruhan menggunakan rumus :

 (2.7)

 (2.8)

 (2.9)

 (2.10)

Keterangan :

1. TP merupakan jumlah data dari kelas aktual True dan diklasifikasikan dengan benar sebagai kelas prediksi Positif.
2. TN merupakan jumlah data dari kelas aktual True namun diklasifikasikan sebagai kelas prediksi Negatif.
3. FP merupakan jumlah data dari kelas False namun diklasifikasikan sebagai kelas prediksi Positif.
4. FP merupakan jumlah data dari kelas False dan diklasifikasikan dengan benar sebagai kelas prediksi Negatif.

### **Uji Perfroma Mobile Apps**

Tahap-tahap pengujian performa aplikasi mobile yang memiliki sistem Deep Learning meliputi beberapa langkah berikut:

1. Verifikasi Data Latih: Langkah ini melibatkan memastikan bahwa data latih yang digunakan untuk membentuk model Deep Learning adalah benar dan valid [20].
2. Uji Validitas Model: Langkah ini melibatkan menguji model Deep Learning untuk memastikan bahwa model tersebut memiliki akurasi dan kinerja yang baik. Ini bisa dilakukan dengan membandingkan hasil prediksi model dengan data latih dan menghitung metrik seperti presisi, recall, dan F1 score [20].
3. Uji Performa: Langkah ini melibatkan menguji performa aplikasi mobile yang menggunakan model Deep Learning dalam kondisi normal dan ekstrem. Ini termasuk melakukan pengujian pada berbagai ukuran layar, resolusi, dan jaringan seluler [20].
4. Uji Keamanan: Langkah ini melibatkan menguji keamanan aplikasi mobile yang menggunakan model Deep Learning, termasuk memastikan bahwa data pribadi pengguna tidak dapat dicuri atau dicuri oleh pihak ketiga [20].
5. Uji Scalability: Langkah ini melibatkan menguji skalabilitas aplikasi mobile yang menggunakan model Deep Learning, termasuk memastikan bahwa aplikasi tersebut dapat berfungsi dengan baik pada skala besar dan dapat diperluas untuk mengatasi peningkatan beban [20].
6. Uji Kompatibilitas: Langkah ini melibatkan menguji kompatibilitas aplikasi mobile yang menggunakan model Deep Learning dengan perangkat dan sistem operasi yang berbeda [20].
7. Uji Usabilitas: Langkah ini melibatkan menguji usabilitas aplikasi mobile yang menggunakan model Deep Learning, termasuk memastikan bahwa aplikasi mudah digunakan dan memiliki antarmuka yang intuitif [20].

Semua langkah-langkah ini penting untuk dilakukan untuk memastikan bahwa aplikasi mobile yang menggunakan model Deep Learning bekerja dengan baik dan memenuhi standar kualitas yang diharapkan.

## **PENELITIAN TERKAIT**

A. Studi literatur terkait IoT klasifikasi sinyal EEG

Tabel 2.1 Studi literatur terkait IoT klasifikasi sinyal EEG

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| No | Referensi | Latar Belakang | Metode | Hasil |
| 1 | Ref [3] | Tujuan :   * mengembangkan aplikasi smartphone berbasis Android untuk monitoring deteksi epilepsi berdasarkan hasil klasifikasi sinyal EEG. | * SVM, * ANN, * KNN | * Aplikasi berbasis android yang dapat mengidentifikasi epilepsi melalui sinyal EEG yang datanya diambil langsung dari alat penangkap sinyal EEG |
| 2 | Ref [12] | * Prototype untuk melakukan Mind Wave EEG dan aplikasi Android yang berkomunikasi melalui Bluetooth. | * Cloud Computing | * Aplikasi yang dapat memproses sisnyal EEG untuk berbagai aplikasi seperti BCI, Neurofeedback, prediksi serangan epilepsi, dll. |
| 3 | Ref [4] | Tujuan :   * mengeksplorasi kelayakan mendeteksi keadaan emosional, dan beban kerja mental selama mengemudi berbasis VR menggunakan EEG dan Brain Computer Interface (BCI) berbasis EEG. | * Bayes network * naïve Bayes * SVM * Multilayer * Perceptron * KNN * random forest, and J48. | Hasil klasifikasi menjanjikan, yakni :   * lebih dari 80% akurasi dalam mengklasifikasikan keterlibatan dan beban kerja mental adaptif untuk digunakan dalam intervensi autisme. |

B. Studi literatur terkait klasifikasi sinyal EEG

Tabel 2.2 Studi literatur terkait klasifikasi sinyal EEG

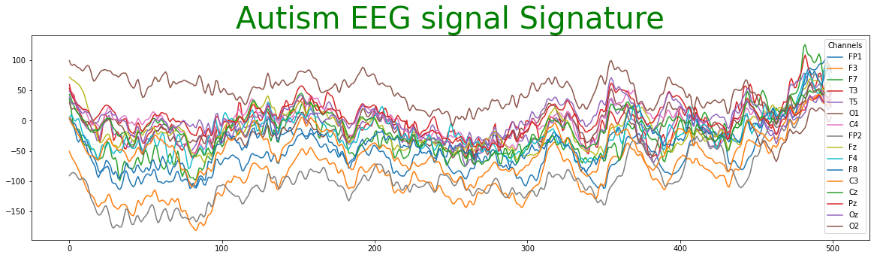
|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 1 | Ref [9] | Tujuan :   * menyajikan perbandingan klasifikasi sinyal Electroencephal EEG dan BCI * mengekstrak pola sinyal EEG berkualitas buruk dan secara bersamaan dengan adanya suara artefak | * SVM * KNN * MLP-ANN * LR | * LR mendapat akurasi klasifikasi EEG sebesar 73,03% * SVM mendapat akurasi sebesar 68,97%. |
| 2 | Ref [10] | * Mengatasi masalah implementasi protokol elisitasi emosi, ekstraksi fitur, * klasifikasi. Rekaman EEG dari database eENTERFACE tujuan simulasi. | * Nearest Neighbor classifier (kNN) | * tingkat akurasi klasifikasi 90,77% |
| 3 | Ref [1] | * menggunakan sinyal saraf untuk mengontrol aplikasi ponsel | * IoT | * headset elektroensefalografi nirkabel (EEG) untuk penggerak aplikasi ponsel |
| 4 | Ref [2] | Tujuan :   * memperoleh scalogram sinyal EEG * mengekstrak fitur dari skalogram dari sinyal EEG subjek normal dan ASD. | * CNN | * CNN GoogLeNet, AlexNet, MobileNet dan SqueezeNet mencapai akurasi validasi 75%, 75,84%, 79,45%, dan 82,98% |
| 7 | Ref [1] | Tujuan :   * Klasifikasi Emosi menggunakan signal EEG. * Menambah fitur citra wajah untuk memprediksi ciri-ciri anak autisme | * Preprocessing: ICA * Proses ekstraksi: Wavelet Denoising * Klasifikasi: MDA | Mendapat perbandingan akurasi dari dua metode :   * Akurasi klasifikasi sebesar 70% dengan metode ICA * Dan nilai akurasi 65% tanpa ICA |
| 8 | Ref [5] | Tujuan :   * Diagnosis kejang epilepsi sinyal Electroencephalgraphy (EEG). | * Deep Learning Convolutional Neural Network (CNN) | * Akurasi klasifikasi bernilai 72.49%, * Nilai loss bernilai 0.576. |
| 9 | Ref [6] | Tujuan :   * mengklasifikasikan berbagai pola sampel rekaman EEG yaitu autis, epilepsi dan normal menggunakan algoritma deep learning. * Untuk menghilangkan artifak dari dataset EEG | * Preprocessing: Independent Component Analysis (ICA) * Klasifikasi: Deep Convolutional Neural Network (DCNN) | * Akurasi klasifikasi yang dibuat oleh model DCNN yang didapatkan mencapai 80% untuk validasi kumpulan data pelatihan. |
| 10 | Ref [7] | Tujuan :   * Deteksi Autisme Berbasis EEG Menggunakan CNN Melalui Transformasi Data Saluran Berbasis Korelasi. | * CNN | * akurasi klasifikasi 100%. |
| 11 | Ref [8] | Tujuan:   * mengembangkan kerangka kerja diagnostik yang efisien sinyal EEG untuk mengidentifikasi ASD secara otomatis. | * Deep Learning Convolutional Neural Network (CNN) * Machine Learning (ML) Classifier Model | * model berbasis Deep Learning CNN mencapai akurasi yang lebih tinggi (99,15%) * dengan model berbasis ML (95,25%) |

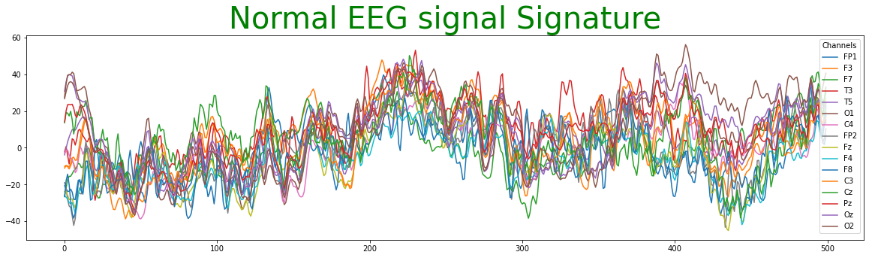
# **BAB 3 METODELOGI DAN JADWAL PENELITIAN**

## **METODE PENELITIAN**

### **Bahan Penelitian**

Penelitian ini menggunakan data sinyal EEG yang didapatkan dari penelitian yang dilakukan pada Universitas King Abdulaziz (KAU), Jeddah Arab Saudi. KAU), Jeddah, Arab Saudi. Dataset yang didapat sebanyak 17 subjek yang terdiri dari 4 sinyal EEG normal dan 13 sinyal EEG untuk penderita ASD dengan extensi *dat*. Data ini diperoleh dari website malhaddad <https://malhaddad.kau.edu.sa/Pages-BCI-Datasets-En.aspx>. Pada penelitian ini data dibagi dan dialokasikan ke dua folder yang masing-masing berisi sinyal EEG normal dan penderita ASD serta menghapus data-data yang tidak layak untuk diproses. Untuk memvisualikan data EEG pada penelitian ini dapat digunakan dua metode yakni menggunakan python dan menggunakan BCI2000 *viewer*, berikut adalah visualisasi data EEG:





Gambar 3.1 Grafik sinyal EEG menggunakan python

Gambar 3.1 merupakan gambar visualisasi sinyal EEG menggunakan *library* python BCI2k *reader*. Pada gambar 3.1 diperlihatkan dua sinyal yakni pada gambar atas merupakan sinyal EEG penderita ASD dan gambar kedua yang merupakan sinyal EEG normal. Pada sinyal ini terdapat 16 kanal (Fp1, Fp2, F7, F3, Fz, F4, F8, T3, C3, Cz, C4, T5, Pz, O1, Oz dan O2) yang masing-masing kanal ditandai dengan warnal yang berbeda (legenda pada grafik). Sinyal ini nantinya akan diproses dalam beberapa tahap seperti membaca sinyal kedalam program, mengolah sinyal, dan pengklasifikasian sinyal menggunakan. *Convolutional Neural Network*

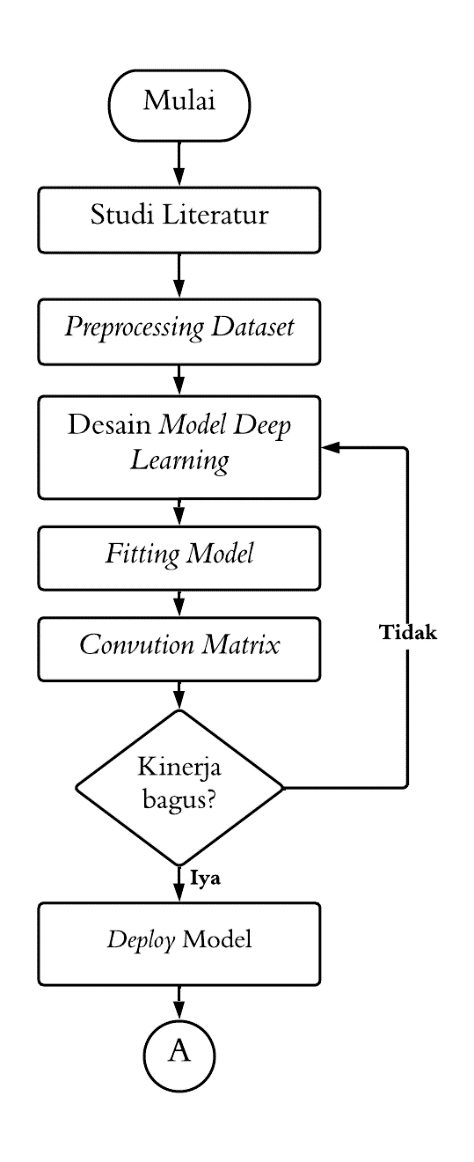
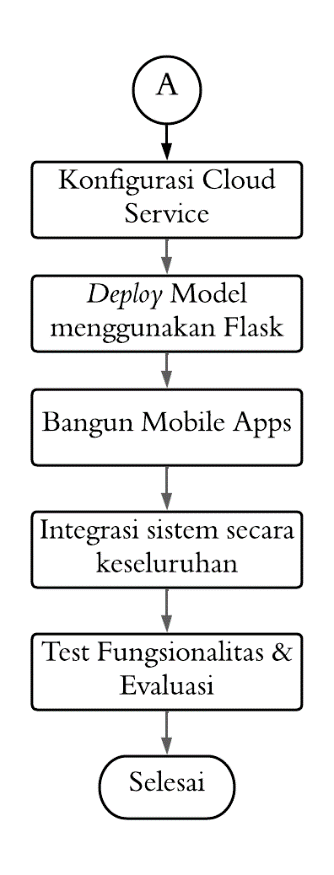
### **Alat Penelitian**

Tabel 3.1 Alat Penelitian

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| No. | Alat | Spesifikasi | Jumlah | Justifikasi |
| 1 | Laptop | * RAM 12 GB * Intel(R) Core(TM) i5-8265U CPU @ 1.60GHz (8 CPUs), ~1.8GHz * NVIDIA GeForce MX230 | 1 | Perangkat  Eksekusi program |
| 2 | Operating System | Windows 11 Home Single Language 64-bit (10.0, Build 22000) (22000.co\_release.210604-1628) | 1 | Sistem operasi untuk pengembangan penelitian |
| 3 | Deep Learning IDE | Jupyter Notebook | 1 | Platform lingkungan pengembangan *deep learning* |
| 4 | Library Deep Learning | TensorFlow | 1 | Framework pengembangan deep learning |
| 5 | Python Preprocessing Library | * BCI2K * Numpy * Pandas * Matplotlib * Flask | 1 | Modul pembaca data EEG dan *Preprocessing* |
| 6 | Could Service | * Fire Base * Flask | 1 | Database API dll |
| 7 | Android SDK | Flutter | 1 | Pengembangan aplikasi *mobile* |
| 8 | Smartphone | Samsung Galaxy M22 | 1 | Emulator output pengembangan aplikasi *mobile* |

### **Alur Penelitian**

Bagian ini menjelaskan tahapan-tahapan dalam pembuatan aplikasi diagnosis ASD menggunakan sinyal EEG. Berikut adalah alur tahapan pada penelitian:

Gambar 3.2 Alur Penelitian

### **Identifikasi Masalah**

Tahap pertama pada penelitian ini adalah mendefenisikan masalah yang berkaitan dengan estimasi sifat tumor tulang jinak atau ganas dan memberikan solusi yang dapat mengatasi masalah tersebut.

### **3.1.3 Studi Literatur**

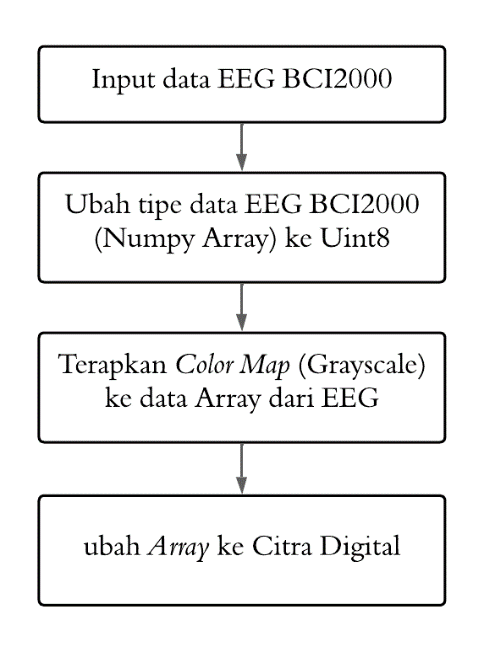
Tahap selanjutnya adalah studi literatur. Pada tahapan ini dilakukan pencarian referensi yang dibutuhkan untuk pada penelitian ini. Tahap studi literatur ini bersifat teori yang mencakup terori mengenai sifat Autism Spectrum Disorder (ASD), sifat gelombang otak, struktur data sinyal EEG, hingga mempelajari teori-teori yang dibutuhkan untuk membangun model yang dapat mengestimasi objek melalui sinyal EEG.

### **Sumber Dataset**

Dataset sinyal EEG bersal dari penelitian pada universitas King Abdulaziz (KAU), Jeddah, Arab Saudi. Dataset yang didapat sebanyak 17 subjek yang terdiri dari 4 sinyal EEG normal dan 13 sinyal EEG untuk penderita ASD dengan extensi *dat*. Data ini diperoleh dari website malhaddad <https://malhaddad.kau.edu.sa/Pages-BCI-Datasets-En.aspx>. Pada penelitian ini data dibagi dan dialokasikan ke dua folder yang masing-masing berisi sinyal EEG normal dan penderita ASD serta menghapus data-data yang tidak layak untuk diproses.

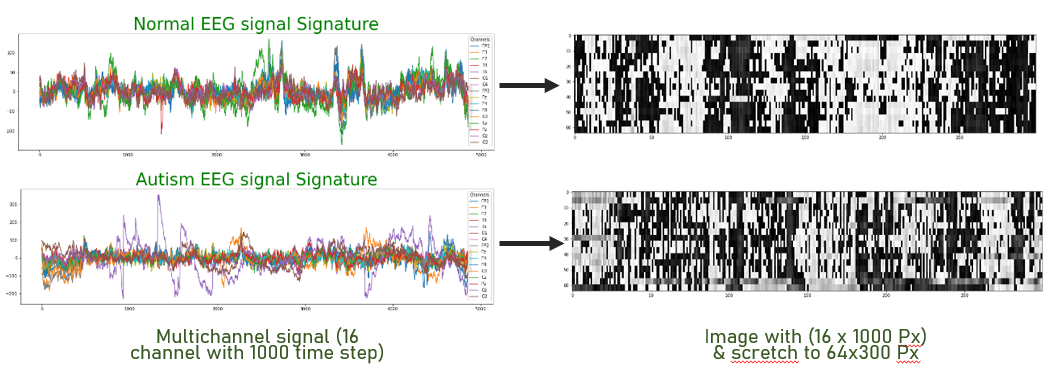
Pada pemrograman Python, sinyal ini hanya bisa diinput menggunakan package BCI2kReader yang nantinya datanya akan terdiri dari channel dan datapoint. Nilai channel dan datapoint yang akan diproses sebelum melakukan training data pada model deep learning.

### **Preprocessing menggunakan Python Brain Computer Interface 2000 (BCI2K) dan CV2**



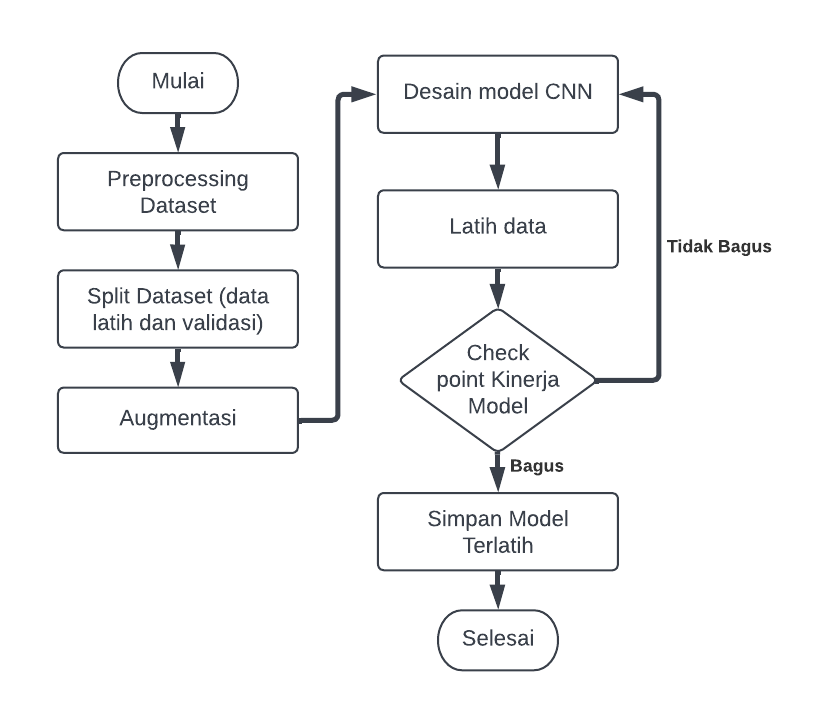
Gambar 3.3 Langkah mengubah data EEG ke bentuk Citra Digital

Pada tahapan ini dilakukan dengan membaca dataset satu persatu yang nantinya datanya akan terdiri dari channel dan datapoint. Proses akan mengambil nilai channel dan datapoint dari sinyal EEG dilakukan dengan menggunakan library Numpy dan membentuk matriks dua dimensi dan data sinyal EEG diubah kedalam bentuk citra digital. Citra digital EEG akan dipisahkan menjadi beberapa bagian yakni data latih yang terdiri dari 80% dari keseluruhan dataset dan data validasi yang berisi 20% data dari keseluruhan dataset. Data latih dan validasi ini dibagi lagi sehingga terdapat empat bagian yakni data latih, label latih, data validasi, serta label validasi dan data siap untuk dilatih oleh model Neural Network.

****

Gambar 3.4 Contoh meubah sinyal ke bentuk Citra Digital [1]

### **Klasifikasi sinyal EEG menggunakan Convolutional Neural Network (CNN)**

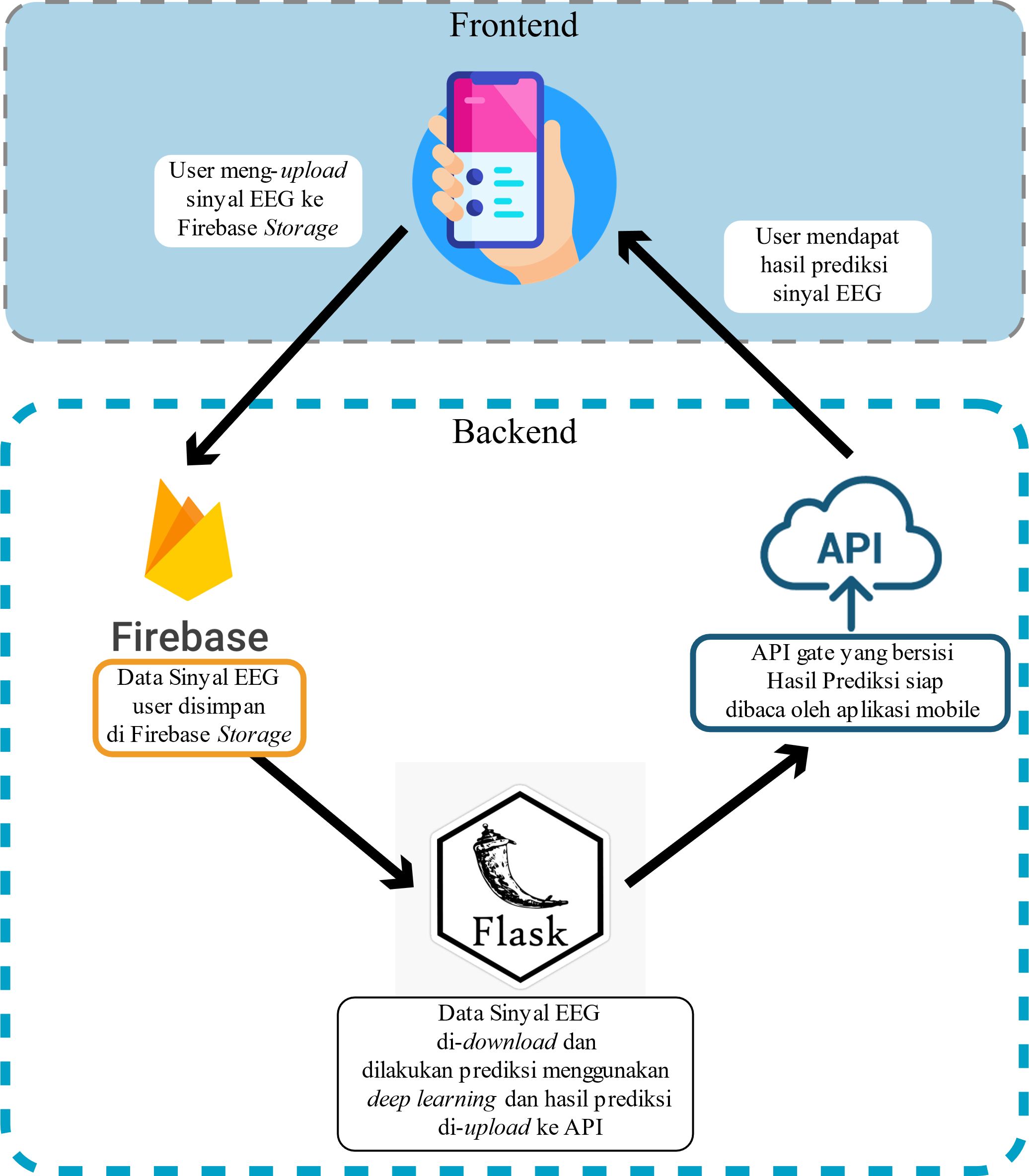


Gambar 3.5 Klasifikasi Data EEG menggunakan CNN

Pengklasifikasian sinyal EEG menggunakan *Convolutional Nerural Network* (CNN) lakukan dengan beberapa tahap yakni:

1. Membangun model *Sequential* deep learning yang terdiri dari tujuh *layer* dengan *layer* terakhir adalah *dense layer* dengan unit sebanyak 1
2. Pada layer terakhis dikonfigurasi dengan mode aktivasi ‘sigmoid’ karena hanya mengklasifikasikan antara dua kelas yakni sinyal EEG penderita autism dan normal.
3. Model akan di *Compile* dengan beberapa parameter untuk klasifikasi dua kelas seperti menggunakan optimasi *‘adam’* untuk meminimalkan nilai *loss* pada pelatihan, dan parameter *'binary\_crossentropy'* untuk memaksimalka klasifikasi antara dua kelas yakni sinyal EEG penderita autism dan normal, dan menggunakan metrics ‘accuracy’ untuk menampilkan dan mengoptimalkan akurasi pada pelatihan data.
4. Model dijalankan dengan menggunakan fungsi *fit* dengan beberapa parameter yang bersisi data dan label pelatihan, data dan label validasi, jumplah *epocs­*, jumlah verbose, suffle, dan fungsi callbacks untuk menghentikan pelatihan jika nilai validasi dan akurasi sudah maksimal dan berguna untuk megoptimalkan waktu pelatihan.
5. Setelah pelatihan dilakukan, model akan disimpan dalam bentuk file dengan extensi h5. File ini berisi *trained* model yang nantinya akan digunakan untuk mengkasifikasi sinyal EEG secara cepat dan ringan untuk dijalankan. File model ini juga akan dideploy menggunakan Python FLASK yang nantinya akan dihubungkan ke firebase dan aplikasi mobile.

### **Membangun Cloud Classifier Engine Environtment menggunakan Flask dan Firebase**



Gambar 3.6 Skema arsitektur IoT yang ditawarkan untuk klasifikasi

penderita ASD menggunakan sinyal EEG

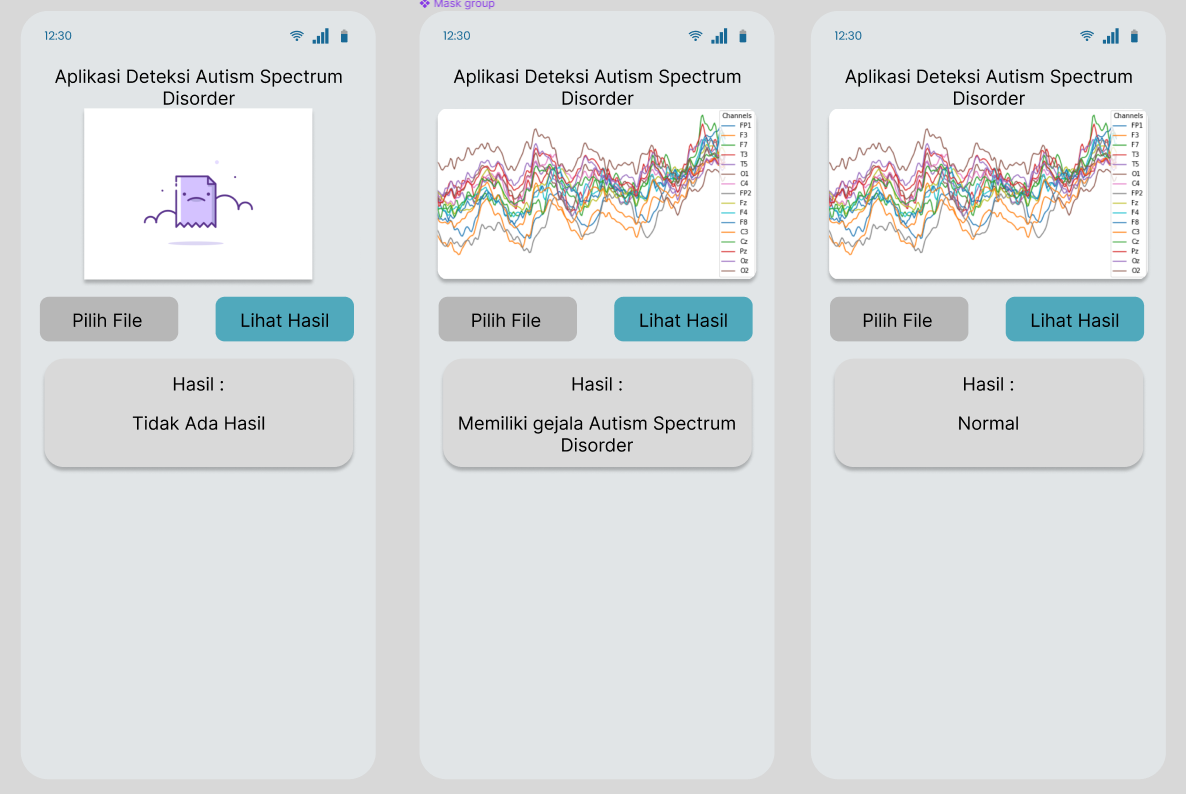
Tahap ini dilakukan dengan *Cloud Environtment* dan *python flask* yang nantinya akan digunakan untuk menjalankan dan menghubungkan model *deep learning* yang sudah dibuat ke aplikasi *mobile*. Tahapan ini dimulai dengan membuat *project* pada *Firebase* dengan menggunakan bebepara fitur pada *Firebase Cloud Service* yakni :

1. *Authentication* untuk penyimpan data akun *user*
2. *CloudFire* *Store* untuk menyimpan data EEG dari setiap *user*

Setelah konfigurasi pada *Firebse* dilakukan, dibangun *Flask app* menggunakan python dengan membuat beberapa fitur seperti :

1. Fitur untuk men-*download* data user dan data sinyal EEG dari firebase dari setiap user.
2. Fitur *preprocessing* sinyal EEG yang nantinya akan mengubah sinyal EEG multichannel ke dalam bentuk citra digital,
3. selanjutnya membuat fungsi prediksi dengan meng-*import* model *deep learning* yang terlah dilatih sebelumnya, dan memprediksi data sinyal EEG dari setiap user dan mengembalikan hasil prediksi dalam bentuk restAPI dengan format json dan hasil prediksi siap untuk ditampilkan pada aplikasi *mobile*.

### **Membangun Aplikasi Mobile**



Gambar 3.7 Mockup aplikasi mobile yang akan dibangun

Pada tahap ini diawali dengan mendesain tampilan dan sistem navigasi pada aplikasi dan dilakukan pembuatan flutter apps dengan memasang package serta flutter app akan dihubungkan dengan firebase dengan sehingga dapat megimput data EEG serta menampilkan hasil klasifikasi dari sinyal EEG BCI2000. berikut adalah fitur-fitur yang akan diterapkan pada aplikasi klasifikasi ASD basis *Deep Learning* :

1. fitur widget untuk tampilan, halaman, dan navigasi.
2. fitur untuk mengupload file sinyal EEG ke firebase.
3. fitur membaca hasil prediksi klasifikasi sinyal EEG dari restAPI flask yang telah dibuat sebelumnya dan menampilkan hasilnya di Aplikasi android.

### **Analisis Peforma Sistem**

Analisis dilakukan dengan beberapa tahap yakni menganalisis hasil training pada model deep learning, dan kinerja keseluruhan aristektur pada aplikasi IoT pada penelitian ini.

1. Analisis pada pelatihan *Deep Learning*

Analisis pada pelatihan *Deep Learning* dilakukan dengan melihat grafik yang dikasilkan dari proses *training* data. Grafik ini menampilkan informasi akurasi dari setiap *epocs* (hyperparameter yang menentukan berapa kali algoritma pembelajaran akan bekerja mengolah seluruh dataset training) yang dilakukan pada *training*. Analisis ini dilakukan dengan melihat apakah grafik *training* dan *validation* terjadi *fit* atau *overfitting.* Jika terjadi overfitting, maka akan dilakukan evaluasi pada *preprocessing* data sinyal EEG dan perbaikan model. Hasil percobaan pengolahan sinyal EEG tersebut juga dianalisis menggunakan confussion matrix berdasarkan parameter-parameter performa metode dalam analisa sinyal yaitu Sensitivitas (Pers 2.7), Spesitivitas (Pers 2.8), Akurasi (Pers 2.9), dan F1 Score (Pers 2.10). Dari analisis pada confussion matrix yang dihasilkan tersebut diharapkan memperoleh hasil yang sesuai dengan tujuan penelitian.

1. Analisis kinerja keseluruhan aristektur pada aplikasi

Tahap analisis ini dilakukan dengan metode analisis *Black Box* denganmemperhatikan kinerja pada sistem arsitektur secara keseluruhan baik itu pada sistem Cloud, dan Aplikasi Mobile itu sendiri. Pada sistem cloud dilakukan beberapa analisis seperti koneksi untuk upload dan download data pada flask dan memastikan seluruh sistem bekerja dengan baik tanpa error. Pada Aplikasi mobile dilakukan analisis kinerja sistem seperti berjalan dengan baik atau terdapat bug atau error. Jika terdapat kesalahan akan dilakukan evaluasi dan perbaikan kinerja.

Pengujian ini bertujuan untuk menganalisa kinerja aplikasi android yang telah dibangun. Pengujian akan dilakukan sebanyak 30 kali dengan memeriksa bebrapa kriteria-kriteria pengujian. Kriteria pengujian kinerja adalah sebagai berikut:

* + - 1. Pengguna *Register* akun dan *Login*.
      2. Aplikasi Android dapat terhubung dengan Firebase
      3. Aplikasi Android dapat terhubung dengan Cloud Classifier Engine
      4. Pengguna dapat mengupload file Raw data EEG ke cloud storage
      5. Aplikasi Android dapat menampilkan hasil klasifikasi.
      6. Aplikasi Android dapat menampilkan rekomendasi terapi sesuai dengan hasil yang ditampilkan

## **JADWAL PENELITIAN**

Penelitian ini akan dilaksanakan sesuai dengan jadwal yang ditampilkan pada tabel berikut:

Tabel 3.2 Jadwal Penelitian

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| No | Kegiatan | Bulan | | | | | | | | | | | | | | | | | | | | | | | |
| 1 | | | | 2 | | | | 3 | | | | 4 | | | | 5 | | | | 6 | | | |
| 1 | 2 | 3 | 4 | 1 | 2 | 3 | 4 | 1 | 2 | 3 | 4 | 1 | 2 | 3 | 4 | 1 | 2 | 3 | 4 | 1 | 2 | 3 | 4 |
| 1 | Persiapan Penelitian |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
|  | Identifikasi Masalah |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| Studi Literatur |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| Pengummpulan dataset |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| 2 | Pelaksanaan |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
|  | Klasifikasi sinyal EEG menggunakan CNN |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| Evaluasi dan Analisa Hasil |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| merancang dan membuat Aplikasi Mobile |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| Menerapkan Model Deep Learning kedalam Aplikasi Mobile |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| Ujicoba aplikasi dan Evaluasi secara keseluruhan |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| 3 | penyusunan Laporan |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
|  | Penulisan Bab 1 |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| Penulisan Bab 2 |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| Penulisan Bab 3 |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| Penulisan Bab 4 |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| Penulisan Bab 5 |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| Kesimpulan |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| Laporan Final |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| 4 | Presentasi Penelitian |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
|  | Seminar Proposal |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| Seminar Hasil |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| Sidang |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| Publikasi Jurnal |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |

# **BAB 4 LUARAN**

Luaran utama pada penelitian ini adalah sebuah aplikasi mobile yang dapat membedakan penderita ASD dan normal berdasarkan data sinyal otak EEG secara *asynchronous*. Penelitian ini juga menghasilkan algoritma *preprocessing* dan data *pipeline* sinyal EEG BCI2000 sehingga dapat diautomasi menggunakan pemrograman python. Penelitian ini juga menghasilkan luaran model terlatih *Deep Learning* *Convolutional Neural Network* yang di-*deploy* menggunakan Python-Flask sehingga diagnosis sinyal EEG penderita ASD dan normal dapat digunakan diberbagai *platform* melalui restAPI. Penelitian ini juga diharapkan membantu masyarakat dan mempercepat diagnosa penderita ASD agar dapat ditangani secara tepat sasaran.

# **DAFTAR PUSTAKA**

1. Arts, L.P.A., van den Broek, E.L. The fast continuous wavelet transformation (fCWT) for real-time, high-quality, noise-resistant time–frequency analysis. Nat Comput Sci 2, 47–58 (2022). https://doi.org/10.1038/s43588-021-00183-z
2. S. Calcagno, F. La Foresta, and M. Versaci, “Independent component analysis and discrete wavelet transform for artifact removal in biomedical signal processing,” Am. J, doi:10.3844/ajassp.2014.57.68
3. Q. Mohi ud Din and A. K. Jayanthy, “AUTOMATED CLASSIFICATION OF AUTISM SPECTRUM DISORDER USING EEG SIGNALS AND CONVOLUTIONAL NEURAL NETWORKS,” Biomed. Eng. Appl. Basis Commun., vol. 34, no. 02, p. 2250020, Apr. 2022, doi: 10.4015/S101623722250020X.
4. Appl. Sci., vol. 11, no. 1, pp. 57–68, 2014, doi: 10.3844/ajassp.2014.57.68. G. Muhammad, M. Masud, S. U. Amin, R. Alrobaea and M. F. Alhamid, "Automatic Seizure Detection in a Mobile Multimedia Framework," in IEEE Access, vol. 6, pp. 45372-45383, 2018, doi: 10.1109/ACCESS.2018.2859267.
5. J. Fan et al., "A Step towards EEG-based brain computer interface for autism intervention," 2015 37th Annual International Conference of the IEEE Engineering in Medicine and Biology Society (EMBC), 2015, pp. 3767-3770, doi: 10.1109/EMBC.2015.7319213.
6. W.-L. Mao, H. I. K. Fathurrahman, Y. Lee, and T. W. Chang, “EEG dataset classification using CNN method,” J. Phys. Conf. Ser., vol. 1456, no. 1, p. 012017, Jan. 2020, doi: 10.1088/1742-6596/1456/1/012017.
7. M. Ranjani and P. Supraja, "Classifying the Autism and Epilepsy Disorder Based on EEG Signal Using Deep Convolutional Neural Network (DCNN)," 2021 International Conference on Advance Computing and Innovative Technologies in Engineering (ICACITE), 2021, pp. 880-886, doi: 10.1109/ICACITE51222.2021.9404634.
8. Z. J. Peya, M. A. H. Akhand, J. Ferdous Srabonee and N. Siddique, "EEG Based Autism Detection Using CNN Through Correlation Based Transformation of Channels' Data," 2020 IEEE Region 10 Symposium (TENSYMP), 2020, pp. 1278-1281, doi: 10.1109/TENSYMP50017.2020.9230928.
9. Tawhid MNA, Siuly S, Wang H, Whittaker F, Wang K, Zhang Y (2021) A spectrogram image based intelligent technique for automatic detection of autism spectrum disorder from EEG. PLoS ONE 16(6): e0253094. https://doi.org/10.1371/journal.pone.0253094
10. M. Z. Ilyas, P. Saad, M. I. Ahmad and A. R. I. Ghani, "Classification of EEG signals for brain-computer interface applications: Performance comparison," 2016 International Conference on Robotics, Automation and Sciences (ICORAS), 2016, pp. 1-4, doi: 10.1109/ICORAS.2016.7872610.
11. H. Xu and K. N. Plataniotis, "Affect recognition using EEG signal," 2012 IEEE 14th International Workshop on Multimedia Signal Processing (MMSP), 2012, pp. 299-304, doi: 10.1109/MMSP.2012.6343458.
12. A. Campbell et al., “NeuroPhone: brain-mobile phone interface using a wireless EEG headset,” in Proceedings of the second ACM SIGCOMM workshop on Networking, systems, and applications on mobile handhelds - MobiHeld ’10, New Delhi, India, 2010, p. 3. doi: 10.1145/1851322.1851326.
13. E. Džaferović, S. Vrtagić, L. Bandić, J. Kevric, A. Subasi and S. M. Qaisar, "Cloud-based mobile platform for EEG signal analysis," 2016 5th International Conference on Electronic Devices, Systems and Applications (ICEDSA), 2016, pp. 1-4, doi: 10.1109/ICEDSA.2016.7818497.
14. https://firebase.google.com/docs/storage/, “Cloud Storage for Firebase[online].” Diakses pada tanggal 1 juli 2022.
15. <https://firebase.google.com/docs/auth/>, “Firebase Authentication[online].” Diakses pada tanggal 1 juli 2022.
16. https://docs.flutter.dev/resources/architectural-overview/, “Flutter architectural overview[online].” Diakses pada tanggal 20 juni 2022.
17. Z. Khakim and S. Kusrohmaniah, “Dasar - Dasar Electroencephalography ( EEG ) bagi Riset Psikologi The Basics of Electroencephalography ( EEG ) for Psychological Research,” vol. 29, no. 1, pp. 92–115, 2021, doi: 10.22146/buletinpsikologi.52328.
18. Subramanyam AA, Mukherjee A, Dave M, Chavda K. Clinical Practice Guidelines for Autism Spectrum Disorders. Indian J Psychiatry2019;61:254–69. https://www.ncbi.nlm.nih.gov/pmc/articles/PMC6345133/
19. https://www.bci2000.org/mediawiki/index.php/User\_Tutorial:Data\_Analysis, “User\_Tutorial:Data\_Analysis[online].” Diakses pada tanggal 13 juni 2022.
20. https://pypi.org/project/BCI2kReader/, “BCI2kReader 0.32.dev0[online].” Diakses pada tanggal 6 juli 2022.
21. [28] Z. Khakim and S. Kusrohmaniah, “Dasar – Dasar Electroencephalography ( EEG ) bagi Riset Psikologi The Basics of Electroencephalography ( EEG ) for Psychological Research,” vol. 29, no. 1, pp. 92–115, 2021, doi: 10.22146/buletinpsikologi.52328.
22. [31] L. C. Cheong, R. Sudirman, and S. S. Hussin, “Feature Extraction OfEeg Signal Using Wavelet Transform For Autism Classification,” vol. 10, no.19, pp. 8533–8540, 2015.
23. https://numpy.org/doc/stable/reference/generated/numpy.mean.html, “numpy.mean[online].” Diakses pada tanggal 6 juli 2022.
24. https://www.tensorflow.org/?hl=en, “End-to-end Open Source Machine Learning Platform[online].” Diakses pada tanggal 20 juli 2022.
25. https://www.tensorflow.org/about, “Why TensorFlow[online].” Diakses pada tanggal 20 juli 2022.
26. C. B. Ahn, Y. C. Song and D. J. Park, "Adaptive template filtering for signal-to-noise ratio enhancement in magnetic resonance imaging," in IEEE Transactions on Medical Imaging, vol. 18, no. 6, pp. 549-556, June 1999, doi: 10.1109/42.781019.
27. P. Gong, P. Song and S. Chen, "Improved Contrast-Enhanced Ultrasound Imaging With Multiplane-Wave Imaging," in IEEE Transactions on Ultrasonics, Ferroelectrics, and Frequency Control, vol. 65, no. 2, pp. 178-187, Feb. 2018, doi: 10.1109/TUFFC.2017.2781190.
28. Y. Gao, Y. Liu, Y. Wang, Z. Shi and J. Yu, "A Universal Intensity Standardization Method Based on a Many-to-One Weak-Paired Cycle Generative Adversarial Network for Magnetic Resonance Images," in IEEE Transactions on Medical Imaging, vol. 38, no. 9, pp. 2059-2069, Sept. 2019, doi: 10.1109/TMI.2019.2894692.
29. A. Okumura, J. Suzuki, I. Furukawa, S. Ono and T. Ashihara, "Signal analysis and compression performance evaluation of pathological microscopic images," in IEEE Transactions on Medical Imaging, vol. 16, no. 6, pp. 701-710, Dec. 1997, doi: 10.1109/42.650867.
30. Z. Yang, J. Jin and M. Wang, "A signal processing method using pulse-based intermediate values for delta-sigma analog-to-digital conversion," 2015 IEEE International Conference on Digital Signal Processing (DSP), 2015, pp. 173-176, doi: 10.1109/ICDSP.2015.7251853.
31. Ma Debao, Li Wugao, Le Zhongxin and Wang Jiefeng, "A matrix algorithm for SAR quick look imaging on digital signal processing," IGARSS 2000. IEEE 2000 International Geoscience and Remote Sensing Symposium. Taking the Pulse of the Planet: The Role of Remote Sensing in Managing the Environment. Proceedings (Cat. No.00CH37120), 2000, pp. 2257-2259 vol.5, doi: 10.1109/IGARSS.2000.858374.
32. J. C. Cabanillas, G. Kemper and C. del Carpio, "A Conversion Algorithm for ECG signals on a 2D array based on Digital Signal Processing," 2022 11th International Conference on Communications, Circuits and Systems (ICCCAS), 2022, pp. 105-109, doi: 10.1109/ICCCAS55266.2022.98242