

**PENERAPAN *CONVOLUTIONAL NEURAL NETWORK* DALAM  
DETEKSI *AUTISM SPECTRUM DISORDER* PADA ANAK  
MELALUI IDENTIFIKASI TULISAN TANGAN**

**SKRIPSI**

**MEYSHA SABRINA AZ-ZAHRA**

**211401060**



**PROGRAM STUDI S-1 ILMU KOMPUTER  
FAKULTAS ILMU KOMPUTER DAN TEKNOLOGI INFORMASI  
UNIVERSITAS SUMATERA UTARA**

**2025**

**PENERAPAN *CONVOLUTIONAL NEURAL NETWORK* DALAM  
DETEKSI *AUTISM SPECTRUM DISORDER* PADA ANAK  
MELALUI IDENTIFIKASI TULISAN TANGAN**

**SKRIPSI**

**Diajukan untuk melengkapi tugas dan memenuhi syarat memperoleh ijazah  
Sarjana Ilmu Komputer**

**MEYSHA SABRINA AZ-ZAHRA**

**211401060**



**PROGRAM STUDI S-1 ILMU KOMPUTER  
FAKULTAS ILMU KOMPUTER DAN TEKNOLOGI INFORMASI  
UNIVERSITAS SUMATERA UTARA**

**2025**

**PERSETUJUAN**

Judul : PENERAPAN *CONVOLUTIONAL NEURAL NETWORK* DALAM DETEKSI *AUTISM SPECTRUM DISORDER* PADA ANAK MELALUI TULISAN TANGAN

Kategori : SKRIPSI

Nama : MEYSHA SABRINA AZ-ZAHRA

Nomor Induk Mahasiswa : 211401060

Program Studi : SARJANA (S-1) ILMU KOMPUTER

Fakultas : ILMU KOMPUTER DAN TEKNOLOGI INFORMASI UNIVERSITAS SUMATERA UTARA

Komisi Pembimbing :

Pembimbing 2 Pembimbing 1

Herriyance, S.T., M.Kom  
NIP. 198010242010121002

Dewi Sartika Br. Ginting, S.Kom, M.Kom  
NIP. 199005042019032023

Diketahui/Disetujui Oleh:  
Program Studi S-1 Ilmu Komputer  
Ketua

Dr. Amalia S.T., M.T.  
NIP. 197812212014042001

**PERNYATAAN****PENERAPAN *CONVOLUTIONAL NEURAL NETWORK* DALAM  
DETEKSI *AUTISM SPECTRUM DISORDER* PADA ANAK  
MELALUI IDENTIFIKASI TULISAN TANGAN****SKRIPSI**

Saya menyatakan bahwa skripsi ini sepenuhnya adalah hasil penelitian saya sendiri, dengan pengecualian pada beberapa kutipan dan ringkasan yang telah dicantumkan sumbernya.

Medan, 19 Desember 2024



Meysha Sabrina Az-Zahra

211401060

## **PENGHARGAAN**

Segala puji dan syukur dipanjatkan kepada Allah SWT yang telah memberikan limpahan rahmat, nikmat, dan karunia yang begitu luar biasa, sehingga penulis dapat menempuh pendidikan hingga akhirnya menyelesaikan skripsi ini sebagai prasyarat untuk memperoleh gelar Sarjana Komputer di Program Studi S-1 Ilmu Komputer Universitas Sumatera Utara.

Pada kesempatan ini, penulis mengucapkan terima kasih sebesar-besarnya kepada yang teristimewa Ayahanda Syafruddin dan Ibunda Marlina selaku orang tua penulis yang selalu memberikan kasih sayang, doa-doa baik yang dipanjatkan, motivasi, dukungan moral dan materiil, serta segala bentuk perjuangan yang tidak ternilai sehingga penulis dapat berada di titik ini.

Penyusunan skripsi ini tidak terlepas dari bantuan serta dukungan dari berbagai pihak. Untuk itu, penulis mengucapkan terima kasih kepada:

1. Ibu Dr. Maya Silvi Lydia, B.Sc., M.Sc. selaku Dekan Fakultas Ilmu Komputer dan Teknologi Informasi Universitas Sumatera Utara.
2. Ibu Dr. Amalia ST., M.T. selaku Ketua Program Studi S-1 Ilmu Komputer Universitas Sumatera Utara.
3. Ibu Sri Melvani Hardi S.Kom., M.Kom selaku Sekretaris Program Studi S-1 Ilmu Komputer Universitas Sumatera Utara.
4. Bapak Dr. Eng Ade Candra S.T., M.Kom. selaku Dosen Penasihat Akademik yang telah memberi banyak nasihat baik, dukungan, dan bimbingan kepada penulis mulai dari awal perkuliahan sampai dengan penyusunan skripsi ini.
5. Ibu Dewi Sartika Br Ginting, S.Kom, M.Kom selaku Dosen Pembimbing I yang telah banyak membantu penulis dalam penyelesaian skripsi ini melalui masukan, arahan, dan bimbingan berharga yang diberikan.
6. Bapak Herriyance, S.T., M.Kom selaku Dosen Pembimbing II yang telah memberikan banyak arahan dan bimbingan yang berharga selama penyusunan skripsi ini.

7. Seluruh bapak dan ibu dosen Fasilkom-TI USU, khususnya dosen Program Studi S-1 Ilmu Komputer yang telah memberi banyak ilmu dan wawasan yang sangat berarti sehingga penulis bisa sampai di tahap ini.
8. Adik penulis Nazriel Alif Al-Ikram yang telah menjadi teman cerita sekaligus teman adu mulut, namun tetap menjadi pendukung yang baik.
9. Keluarga besar penulis yang telah memberikan dukungan dan doa sehingga penulis dapat menyelesaikan pendidikan ini dengan baik.
10. Para pendidik Pondok Peduli Autis dan SLB Negeri Autis Sumatera Utara atas ilmu dan kesempatan yang telah diberikan kepada penulis.
11. Sahabat baik sejak putih-biru dan putih-abu Dinda Ilyasa, Fattaahunisa Alayya, Hilda Khalilah, Rafida Ilham, dan Roudotul Hasanah yang telah membawa kebahagiaan setiap kali bertemu, berbagi cerita, memberi dukungan, serta saling mendoakan satu sama lain.
12. Sahabat menjalani hari-hari sebagai mahasiswa ilmu komputer Nayata Sandra, Nurhalimah, Dea Oktavia, dan Aurick Muhammad yang telah banyak memberikan semangat, keceriaan, serta tempat berkeluh kesah sepanjang masa perkuliahan.
13. Teman-teman 'Kom C(ool)' dan 'Kom C Girls' yang telah memberikan memori dan cerita yang akan selalu dikenang, serta teman-teman stambuk 2021 yang turut membantu dan memberi pembelajaran baru.

Kepada semua pihak yang telah memberikan dukungan dan doa, yang tidak bisa disebutkan satu per satu. Semoga Allah SWT senantiasa memberikan keberkahan dan kebaikan atas segala bantuan yang telah diberikan dan semoga penelitian ini dapat bermanfaat serta menjadi inspirasi di masa depan.

Medan, 19 Desember 2024



Meysha Sabrina Az-Zahra

211401060

## ABSTRAK

*Autism Spectrum Disorder* (ASD) ditandai dengan gangguan yang mempengaruhi kemampuan komunikasi, interaksi sosial, dan perilaku pada individu. Anak dengan ASD sering mengalami kesulitan dalam aktivitas motorik halus, termasuk menulis tangan, yang sering kali menghasilkan tulisan dengan kualitas lebih rendah dibandingkan anak-anak dengan perkembangan normal. Tantangan ini dapat mencerminkan gangguan pada kemampuan koordinasi motorik dan persepsi sensorik. Deteksi dini sangat penting dilakukan untuk memberikan intervensi yang sesuai sehingga dapat meningkatkan perkembangan anak sehingga dapat meningkatkan kemampuan anak dalam aspek sosial, emosional, dan akademis. Penelitian ini bertujuan untuk mendeteksi ASD melalui pendekatan identifikasi tulisan tangan dengan menggunakan metode *Convolutional Neural Network* (CNN). Arsitektur CNN yang digunakan adalah *Xception*, memanfaatkan *depthwise separable convolution* untuk meningkatkan efisiensi dan akurasi pemrosesan data gambar. Dataset yang digunakan terdiri dari 500 sampel gambar tulisan tangan anak-anak yang dikategorikan ke dalam empat kelas, yaitu normal, ASD ringan, sedang, dan berat. Seluruh data diproses melalui tahapan *preprocessing* yang meliputi normalisasi dan *resizing* gambar untuk memastikan konsistensi data. Metode CNN yang digunakan mencapai tingkat akurasi sebesar 94% pada data pelatihan.

Kata Kunci: *Autism Spectrum Disorder*, CNN, Tulisan Tangan, *Xception*.

**APPLICATION OF CONVOLUTIONAL NEURAL NETWORK IN  
DETECTING AUTISM SPECTRUM DISORDER IN CHILDREN  
THROUGH HANDWRITING IDENTIFICATION**

**ABSTRACT**

*Autism Spectrum Disorder (ASD) is defined by difficulties in communication, social interactions, and behavioral patterns. Children with ASD often face difficulties in fine motor activities, including handwriting, which frequently results in handwriting of lower quality compared to children with typical development. These challenges are often indicative of impairments in motor coordination and sensory perception. Early detection is crucial to providing appropriate interventions, which can significantly enhance children's development in social, emotional, and academic aspects. This study aims to detect ASD through handwriting identification using Convolutional Neural Networks (CNN). The CNN architecture used is Xception, leveraging depthwise separable convolutions to enhance the efficiency and accuracy of image data processing. The dataset consists of 500 handwritten image samples from children, categorized into four classes: normal, mild ASD, moderate ASD, and severe ASD. All data undergoes preprocessing steps, including normalization and resizing, to ensure data consistency. The CNN method achieved an accuracy rate of 94% on the training data.*

*Keywords: Autism Spectrum Disorder, CNN, Handwriting, Xception.*



## DAFTAR ISI

PERSETUJUAN .....	i
PERNYATAAN .....	ii
PENGHARGAAN .....	iii
ABSTRAK.....	v
<i>ABSTRACT</i> .....	vi
DAFTAR ISI.....	vii
DAFTAR GAMBAR .....	ix
DAFTAR TABEL.....	xi
<b>BAB 1 PENDAHULUAN .....</b>	<b>1</b>
1.1. Latar Belakang .....	1
1.2. Rumusan Masalah .....	3
1.3. Batasan Masalah.....	3
1.4. Tujuan Penelitian.....	4
1.5. Manfaat Penelitian.....	4
1.6. Metodologi Penelitian .....	4
1.7. Penelitian Relevan.....	5
1.8. Sistematika Penulisan.....	6
<b>BAB 2 LANDASAN TEORI .....</b>	<b>8</b>
2.1. <i>Deep Learning</i> .....	8
2.1.1. <i>Deep Supervised Learning</i> .....	9
2.1.2. <i>Deep Semi-Supervised Learning</i> .....	9
2.1.3. <i>Deep Unsupervised Learning</i> .....	10
2.2. <i>Convolutional Neural Network (CNN)</i> .....	10
2.2.1. <i>Convolutional Layer</i> .....	12
2.2.2. <i>Pooling Layer</i> .....	14
2.2.3. <i>Fully Connected Layer</i> .....	15
2.3. <i>Xception</i> .....	15
2.4. Motorik Halus .....	17
2.5. Kemampuan Menulis Tangan .....	17
2.6. <i>Autism Spectrum Disorder (ASD)</i> .....	18
<b>BAB 3 ANALISIS DAN PERANCANGAN .....</b>	<b>22</b>

3.1.	Analisis Sistem.....	22
3.1.1.	Analisis Masalah.....	22
3.1.2.	Analisis Data.....	23
3.1.3.	Analisis Kebutuhan.....	29
3.1.4.	Diagram Umum Sistem.....	30
3.2.	Pemodelan Sistem .....	30
3.2.1.	<i>Use Case Diagram</i> .....	30
3.2.2.	<i>Activity Diagram</i> .....	31
3.2.3.	<i>Flowchart</i> .....	33
3.2.4.	<i>Sequence Diagram</i> .....	34
3.3.	Perancangan <i>Interface</i> .....	35
<b>BAB 4 IMPLEMENTASI DAN PENGUJIAN.....</b>		<b>37</b>
4.1.	Implementasi .....	37
4.1.1.	Halaman Utama .....	37
4.1.2.	Halaman Prediksi .....	37
4.1.3.	Halaman Panduan .....	38
4.2.	Pengujian Sistem .....	39
4.2.1.	Tahap Pelatihan Model .....	39
4.2.2.	Tahap Pengujian.....	42
<b>BAB 5 PENUTUP .....</b>		<b>49</b>
5.1.	Kesimpulan.....	49
5.2.	Saran.....	49
<b>DAFTAR PUSTAKA.....</b>		<b>50</b>

## DAFTAR GAMBAR

<b>Gambar 2.1.</b> Perbedaan Cara Kerja <i>Machine Learning</i> dan <i>Deep Learning</i> .....	8
<b>Gambar 2.2.</b> Arsitektur CNN .....	11
<b>Gambar 2.3.</b> Operasi Konvolusi .....	12
<b>Gambar 2.4.</b> Struktur Umum <i>Activation Function</i> .....	13
<b>Gambar 2.5.</b> <i>Max Pooling</i> .....	14
<b>Gambar 2.6.</b> <i>Fully Connected Layer</i> .....	15
<b>Gambar 2.7.</b> Arsitektur <i>Xception</i> .....	16
<b>Gambar 3.1.</b> Diagram Ishikawa.....	22
<b>Gambar 3.2.</b> Contoh Data.....	23
<b>Gambar 3.3.</b> Gambar Input dan Representasi Nilai Piksel dalam RGB.....	24
<b>Gambar 3.4.</b> Proses Konvolusi .....	25
<b>Gambar 3.5.</b> <i>Feature Map</i> .....	26
<b>Gambar 3.6.</b> Penerapan ReLU.....	26
<b>Gambar 3.7.</b> Proses <i>Max Pooling</i> .....	27
<b>Gambar 3.8.</b> <i>Flattening</i> .....	27
<b>Gambar 3.9.</b> Hasil <i>Fully Connected Layer</i> .....	28
<b>Gambar 3.10.</b> Diagram Umum Sistem .....	30
<b>Gambar 3.11.</b> <i>Use Case Diagram</i> .....	31
<b>Gambar 3.12.</b> <i>Activity Diagram</i> .....	32
<b>Gambar 3.13.</b> <i>Flowchart</i> Sistem.....	33
<b>Gambar 3.14.</b> <i>Flowchart</i> CNN dengan Arsitektur <i>Xception</i> .....	34
<b>Gambar 3.15.</b> <i>Sequence Diagram</i> .....	35
<b>Gambar 3.16.</b> Halaman Utama .....	35
<b>Gambar 3.17.</b> Halaman Prediksi.....	36
<b>Gambar 4.1.</b> Tampilan Halaman Utama.....	37
<b>Gambar 4.2.</b> Tampilan Halaman Prediksi .....	37
<b>Gambar 4.3.</b> Tampilan Hasil Prediksi .....	38
<b>Gambar 4.4.</b> Tampilan Halaman Panduan.....	38
<b>Gambar 4.5.</b> Informasi Kelas .....	39
<b>Gambar 4.6.</b> <i>Model Summary</i> .....	40

<b>Gambar 4.7.</b> Proses <i>Training Model</i> .....	41
<b>Gambar 4.8.</b> Grafik <i>Accuracy</i> dan <i>Validation Accuracy</i> .....	41
<b>Gambar 4.9.</b> Grafik <i>Loss</i> dan <i>Validation Loss</i> .....	42
<b>Gambar 4.10.</b> <i>Confusion Matrix</i> .....	47

**DAFTAR TABEL**

<b>Tabel 2.1.</b> Tingkat Keparahan ASD .....	19
<b>Tabel 4.1.</b> Hasil Pengujian Sistem .....	43
<b>Tabel 4.2.</b> Tabel <i>Confusion Matrix</i> .....	47

## **BAB 1**

### **PENDAHULUAN**

#### **1.1. Latar Belakang**

*Autism Spectrum Disorder* (ASD) merupakan gangguan pada perkembangan sistem saraf yang mempengaruhi kemampuan komunikasi, interaksi sosial, serta perilaku individu (American Psychiatric Association, 2013). Sekitar 50-85% anak dengan ASD mengalami masalah motorik. Beberapa masalah motorik yang sering muncul termasuk koordinasi tubuh yang buruk, kekuatan genggam yang lemah, kecepatan motorik yang terganggu, cara berjalan yang tidak normal, dan kontrol motorik halus yang kurang (seperti kesulitan menulis atau mengendalikan objek). Masalah motorik ini dapat sangat mempengaruhi perkembangan anak di bidang akademis, sosial, dan komunikasi (Kangarani-Farahani et al., 2024).

Tulisan tangan yang baik membantu anak dalam belajar, berkomunikasi, dan membangun rasa percaya diri. Anak dengan ASD sering mengalami kesulitan dalam keterampilan motorik halus dan kasar, yang membuat mereka kesulitan dalam menulis dengan rapi dan presisi. Dalam hal fungsi sensorik, terkadang sulit merasakan posisi tubuh mereka dengan baik (*proprioception*), yang membuat gerakan menulis menjadi tidak lancar. Cara anak dengan ASD melihat dan memahami bentuk huruf mungkin berbeda. Anak dengan ASD cenderung fokus pada detail kecil daripada gambaran keseluruhan. Ini bisa mempengaruhi cara mereka melihat dan menulis huruf, sehingga hasil tulisan mereka bisa berbeda dengan anak lain.

Penelitian menunjukkan bahwa anak-anak dengan ASD memiliki kualitas tulisan tangan yang lebih rendah, terutama terkait dengan kesulitan motorik yang mempengaruhi pembentukan huruf. Namun, kemampuan dalam aspek lain seperti menyusun, mengatur ukuran, dan memberikan jarak huruf sama baiknya dengan anak-anak yang berkembang normal (Fuentes et al., 2009). Dengan memahami masalah ini, orang tua dan profesional diharapkan bisa lebih baik dalam memberikan intervensi yang sesuai.

Wakil Menteri Kesehatan, dr. Dante Saksono menyatakan bahwa prevalensi anak dengan autisme di Indonesia terus meningkat setiap tahun. Maka dari itu, deteksi dini penting dilakukan untuk melakukan intervensi lebih awal, yang terbukti dapat meningkatkan perkembangan dan kemampuan beradaptasi pada anak ASD. Anak-anak bisa mendapatkan penanganan khusus sesuai kebutuhan mereka, seperti dalam berkomunikasi, perkembangan bahasa, dan mengatasi masalah perilaku. Penelitian menunjukkan bahwa intervensi dini mampu memberikan perbaikan signifikan pada fungsi kognitif, bahasa, dan sosial-emosional anak-anak dengan autisme (Vivanti et al., 2014).

*Convolutional Neural Network* (CNN) adalah salah satu cabang *artificial intelligence* yang menggunakan *multi-layer neural networks* untuk mengidentifikasi, mengenali, mengklasifikasikan objek, maupun mendeteksi dan segmentasi objek dalam gambar (Taye, 2023). CNN terdiri dari beberapa lapisan, di mana setiap lapisan memiliki fungsi untuk mengekstrak fitur hierarkis dari gambar. Lapisan pertama, yang disebut lapisan konvolusi, berfungsi untuk mendeteksi pola-pola dasar seperti tepi, sudut, dan tekstur. Kemudian, pola-pola dasar ini digabungkan pada lapisan-lapisan berikutnya untuk membentuk representasi yang lebih kompleks, seperti bentuk objek, wajah, atau bahkan adegan dalam gambar. Selain itu, CNN juga mampu melakukan deteksi dan segmentasi objek, yang menjadikannya sangat berguna dalam berbagai aplikasi seperti pengenalan wajah, analisis medis, dan lain-lain.

*Xception* adalah arsitektur CNN yang merupakan pengembangan dari arsitektur *Inception* sebelumnya (Chollet, 2017). Salah satu keunggulan utama *Xception* adalah penggunaan teknik *depthwise separable convolution*, yang mengoptimalkan konvolusi. Teknik ini memungkinkan *Xception* untuk meningkatkan efisiensi komputasi sambil menjaga kinerja yang tinggi. *Xception* memiliki 36 *layer* yang disusun ke dalam 14 modul, dengan setiap modul terdiri dari beberapa lapisan konvolusi dan *batch normalization*. Struktur modular ini membantu arsitektur untuk belajar representasi yang lebih kompleks dari data gambar.

Penelitian ini menerapkan CNN dengan arsitektur *Xception* untuk mengklasifikasikan tulisan tangan anak-anak sebagai pendekatan deteksi ASD. Hasil klasifikasi diharapkan dapat membantu mengidentifikasi anak-anak dengan ASD pada tingkat ringan, sedang, atau berat, serta memberikan informasi yang lebih tepat dalam menentukan intervensi yang sesuai, mendukung penanganan dini yang lebih efektif.

## 1.2. Rumusan Masalah

Deteksi dini *Autism Spectrum Disorder* (ASD) pada anak-anak sangat penting untuk memastikan intervensi yang tepat dan dapat meningkatkan perkembangan sosial dan komunikasi anak. Namun, keterlambatan dalam proses diagnosis sering terjadi karena variasi gejala dan tingkat keparahan ASD. Mengingat anak dengan ASD cenderung mengalami kesulitan motorik halus yang dapat tercermin dalam tulisan tangan mereka, dalam penelitian ini digunakan *Convolutional Neural Network* (CNN) dengan arsitektur *Xception* diharapkan dapat membantu mengotomatisasi proses identifikasi pada tulisan tangan, sehingga deteksi ASD dapat dilakukan dengan lebih cepat dan akurat.

## 1.3. Batasan Masalah

Berikut adalah batasan masalah yang diterapkan dalam penelitian ini yaitu mencakup beberapa hal seperti.

1. Model yang digunakan dalam penelitian ini adalah *Convolutional Neural Network* (CNN).
2. Data yang digunakan berupa gambar tulisan tangan, di mana data *training* dan *validation* bersumber dari situs *Kaggle* (<https://www.kaggle.com/datasets/imranliaqat32/autism-spectrum-disorder-in-childrenhandgestures/data>) dengan total 500 data. Sementara untuk data *testing* diperoleh dari SLB Negeri Autis Sumatera Utara sebanyak 25 data.
3. Jenis tulisan tangan yang digunakan hanya mencakup huruf Latin dan angka, tanpa mencakup simbol atau karakter dari alfabet lain.



4. Klasifikasi yang diterapkan terdiri dari empat kelas, yaitu Normal, Ringan, Sedang, dan Berat.
5. Sistem difokuskan untuk anak-anak dengan gejala *Autism Spectrum Disorder* (ASD).
6. Program dikembangkan menggunakan bahasa pemrograman *Python* dan diimplementasikan ke dalam *website*.

#### **1.4. Tujuan Penelitian**

Penelitian ini memiliki tujuan untuk membangun sistem berbasis *website* yang diharapkan dapat membantu orang tua dalam mendeteksi *Autism Spectrum Disorder* (ASD) pada anak-anak mereka melalui tulisan tangan. Sistem ini diharapkan dapat mengklasifikasikan tingkat keparahan ASD menggunakan model *Convolutional Neural Network* (CNN), yang nantinya dapat memudahkan orang tua untuk melakukan deteksi dini sehingga memungkinkan intervensi yang lebih tepat dan cepat.

#### **1.5. Manfaat Penelitian**

Berikut beberapa manfaat yang diharapkan dari penelitian ini antara lain yaitu.

1. Memberikan wawasan mengenai deteksi *Autism Spectrum Disorder* (ASD) menggunakan teknologi, khususnya melalui *deep learning* seperti *Convolutional Neural Network* (CNN) untuk menganalisis tulisan tangan.
2. Membantu profesional atau ahli terkait saat melakukan diagnosis.
3. Menjadi referensi bagi penelitian selanjutnya dalam penerapan *deep learning* atau CNN untuk mendeteksi gangguan perkembangan lainnya.

#### **1.6. Metodologi Penelitian**

Metode penelitian yang digunakan dalam penelitian ini adalah sebagai berikut.

##### **1. Studi Pustaka**

Pada tahap ini, dilakukannya studi literatur untuk mengumpulkan referensi melalui berbagai bacaan terpercaya, baik melalui jurnal, *e-book*,

artikel ilmiah, maupun situs internet yang relevan. Referensi tersebut berkaitan dengan *Autism Spectrum Disorder* (ASD) serta penerapan metode *Convolutional Neural Network* (CNN) dengan arsitektur *Xception* dalam berbagai kasus deteksi atau klasifikasi.

## 2. Analisis dan Perancangan Sistem

Tahap ini mencakup perancangan sistem menggunakan diagram alir (*flowchart*), *use case diagram*, *activity diagram*, *sequence diagram*, dan diagram Ishikawa untuk menggambarkan alur proses penelitian secara keseluruhan.

## 3. Implementasi Sistem

Pada tahap ini dilakukan pengembangan model CNN dengan arsitektur *Xception* berdasarkan rancangan yang telah dibuat.

## 4. Pengujian Sistem

Sistem yang telah diimplementasikan diuji menggunakan data pengujian untuk mengevaluasi performanya. Pengujian dilakukan dengan menggunakan metrik evaluasi seperti *accuracy*, *precision*, *recall*, dan *f1-score*.

## 5. Dokumentasi Sistem

Tahap akhir melibatkan pencatatan semua proses dan hasil penelitian dalam bentuk dokumentasi yang terstruktur untuk keperluan skripsi.

### 1.7. Penelitian Relevan

Berbagai studi sebelumnya yang berkaitan dengan penelitian ini antara lain yaitu.

1. Penelitian oleh Fatimah et al., 2019 dengan judul “*Personality Features Identification from Handwriting Using Convolutional Neural Networks*” membahas penggunaan grafologi dan CNN untuk mengidentifikasi ciri kepribadian melalui tulisan tangan. Dengan menganalisis fitur struktural seperti margin, spasi, dan kemiringan, serta karakteristik spesifik huruf, penelitian ini mengumpulkan 15.000 sampel tulisan tangan. Analisis

struktural menghasilkan akurasi berkisar antara 82,5% hingga 100%, sementara analisis simbol mencapai 98,03%.

2. Penelitian oleh Wang et al., 2024 yang berjudul “*LSTM-CNN: An Efficient Diagnostic Network for Parkinson’s Disease Utilizing Dynamic Handwriting Analysis*” memperkenalkan metode LSTM-CNN untuk menganalisis tulisan tangan dalam diagnosis penyakit parkinson. Hasil penelitian mencapai akurasi sebesar 90,7%.
3. Penelitian oleh Alsaidi et al., 2024 yang berjudul “*A Convolutional Deep Neural Network Approach to Predict Autism Spectrum Disorder Based on Eye-Tracking Scan Paths*” memperkenalkan model T-CNN-ASD, sebuah jaringan saraf dalam yang dirancang untuk deteksi dini ASD menggunakan data pelacakan mata. Hasil penelitian menunjukkan bahwa model ini mencapai akurasi tinggi sebesar 95,59%.
4. Penelitian oleh penulis Rabbi et al., 2021 yang berjudul “*A Convolutional Neural Network Model for Early-Stage Detection of Autism Spectrum Disorder*” menyajikan studi tentang penggunaan kecerdasan buatan, khususnya CNN, untuk deteksi dini ASD pada anak-anak melalui gambar wajah. Hasil akhir menunjukkan model CNN memperoleh akurasi tertinggi sebesar 92,31%, melampaui algoritma lainnya seperti *Random Forest*, *Gradient Boosting Machine*, dan *Multi-Layer Perceptron*.
5. Penelitian oleh Sherkatghanad et al. (2020) yang berjudul “*Automated Detection of Autism Spectrum Disorder Using a Convolutional Neural Network*” memperkenalkan arsitektur CNN untuk mengklasifikasikan pasien ASD dan kelompok pembanding berdasarkan pola konektivitas fungsional otak menggunakan data pencitraan otak *resting-state* fMRI. Hasilnya didapati bahwa CNN mendapat akurasi rata-rata sebesar 70,22% pada dataset *Autism Brain Imaging Exchange* (ABIDE).

## 1.8. Sistematika Penulisan

Penulisan skripsi ini disusun berdasarkan dengan sistematika seperti berikut.

## **BAB 1 PENDAHULUAN**

Bab ini menguraikan latar belakang penelitian, rumusan masalah, batasan masalah, tujuan dan manfaat penelitian, metodologi yang digunakan, penelitian relevan, serta struktur penulisan yang diterapkan dalam skripsi ini.

## **BAB 2 LANDASAN TEORI**

Membahas teori-teori yang mendasari penelitian, termasuk penjelasan tentang *Autism Spectrum Disorder* (ASD), *Convolutional Neural Network* (CNN), Motorik Halus, Kemampuan Menulis Tangan, serta arsitektur *Xception* yang digunakan dalam penelitian ini.

## **BAB 3 ANALISIS DAN PERANCANGAN**

Memaparkan analisis terhadap kebutuhan sistem, proses perancangan sistem menggunakan *flowchart*, *use case diagram*, *activity diagram*, *sequence diagram*, dan diagram Ishikawa serta penyusunan pelatihan model menggunakan CNN dengan arsitektur *Xception* untuk mendeteksi ASD melalui gambar tulisan tangan.

## **BAB 4 IMPLEMENTASI DAN PENGUJIAN**

Menjelaskan implementasi sistem berdasarkan rancangan yang telah dibuat, hasil pengujian sistem menggunakan data uji, serta evaluasi performa model.

## **BAB 5 PENUTUP**

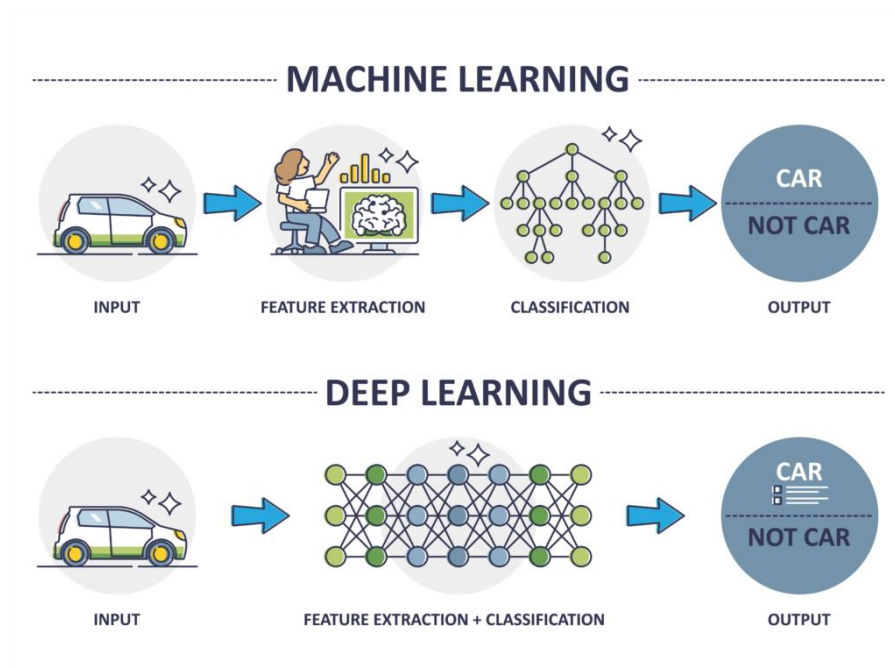
Berisi kesimpulan dari hasil penelitian, serta saran untuk pengembangan penelitian lebih lanjut.

## BAB 2

### LANDASAN TEORI

#### 2.1. *Deep Learning*

*Deep learning* termasuk bagian *machine learning* yang menggunakan *neural networks* dengan banyak lapisan untuk memproses data sehingga dapat mengenali pola yang kompleks. Berbeda dari *machine learning* tradisional, *deep learning* mampu mengekstraksi fitur secara otomatis. Konsep hierarki dalam *deep learning* memungkinkan jaringan untuk mempelajari data secara bertahap, dari representasi sederhana hingga kompleks melalui struktur lapisan yang saling terkait. Konsep ini mencerminkan cara kerja otak manusia dalam mengenali pola dan membangun pemahaman (Alzubaidi et al., 2021).



**Gambar 2.1.** Perbedaan Cara Kerja *Machine Learning* dan *Deep Learning* (Pemrograman Matlab, 2023)

Gambar 2.1. merupakan visualisasi perbedaan cara kerja antara *machine learning* dan *deep learning*. Lapisan input terdiri dari unit-unit yang berisi nilai-nilai yang diteruskan ke setiap neuron di *hidden layer*. Kemudian, hasil prediksi dikeluarkan dari model melalui lapisan output. Jumlah unit di lapisan output ini sesuai dengan jumlah kelas yang diinginkan sebagai hasil prediksi.

*Hidden layer* yang berada di antara lapisan input dan output memiliki tugas untuk menerapkan bobot pada data masukan dan memprosesnya melalui *activation function*. (Ahmed et al., 2023).

*Deep learning* digunakan dalam berbagai aplikasi, termasuk pengenalan gambar, *natural language processing*, dan pengenalan suara, yang sering kali melibatkan data yang tidak terstruktur. Dengan perkembangan dataset yang besar dan sumber daya komputasi yang lebih kuat, teknik *deep learning* telah merevolusi berbagai bidang, seperti kesehatan, keuangan, dan sistem otonom.

### **2.1.1. Deep Supervised Learning**

Teknik ini menggunakan data berlabel, yaitu data yang sudah memiliki input dan output yang diinginkan. Misalnya, sebuah sistem mencoba menebak output dari input  $x_t$ , lalu menghitung kesalahannya (*loss*). Parameter jaringan diperbarui secara bertahap agar hasilnya semakin mendekati output yang diinginkan. Setelah pelatihan selesai, sistem bisa memberikan jawaban yang tepat untuk berbagai pertanyaan berdasarkan data tersebut.

Contoh teknik *supervised learning* dalam *deep learning* adalah *Recurrent Neural Network* (RNN), *Convolutional Neural Network* (CNN), dan *Deep Neural Network* (DNN). RNN juga mencakup metode seperti *Gated Recurrent Unit* (GRU) dan *Long Short-Term Memory* (LSTM).

Kelebihan teknik ini adalah kemampuannya memanfaatkan pengetahuan sebelumnya untuk menghasilkan data atau memprediksi output. Namun, kekurangannya adalah jika data pelatihan kurang lengkap, model bisa salah mengklasifikasikan data baru. Meski begitu, teknik ini cukup sederhana dan mampu menghasilkan kinerja yang tinggi.

### **2.1.2. Deep Semi-Supervised Learning**

Teknik ini menggunakan dataset yang sebagian datanya berlabel. RNN, termasuk GRU dan LSTM, juga dapat diterapkan dalam pembelajaran *semi-supervised*. Kelebihan utama teknik ini adalah dapat mengurangi jumlah data berlabel yang diperlukan, sehingga

lebih efisien. Namun, kelemahannya adalah jika terdapat fitur input yang tidak relevan dalam data pelatihan, model bisa menghasilkan keputusan yang salah.

Salah satu contoh penerapan *semi-supervised learning* adalah pengklasifikasi dokumen teks. Karena sulit mendapatkan jumlah besar dokumen teks berlabel, teknik ini sangat cocok digunakan untuk tugas klasifikasi teks.

### 2.1.3. *Deep Unsupervised Learning*

Teknik ini memungkinkan proses pembelajaran dilakukan tanpa membutuhkan data berlabel (tanpa label). Dalam metode ini, model belajar mengenali fitur penting atau pola tersembunyi dalam data untuk menemukan struktur atau hubungan yang belum diketahui. Teknik seperti *generative networks*, pengurangan dimensi, dan *clustering* sering digunakan dalam kategori ini.

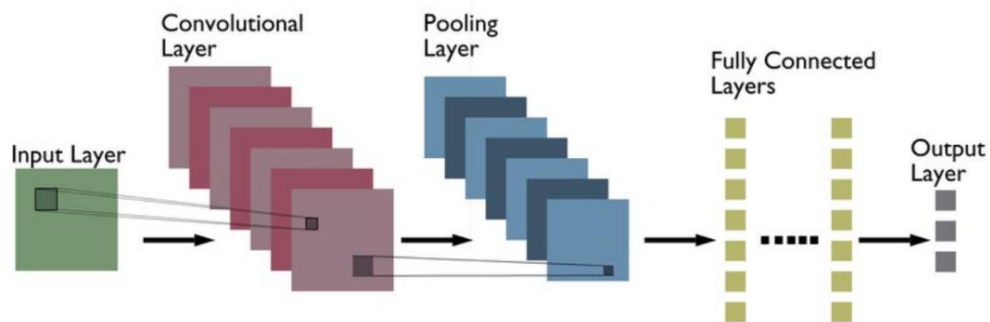
Kelemahan utama dari teknik ini adalah sulit memberikan informasi akurat terkait pengelompokan data dan seringkali membutuhkan komputasi yang kompleks. Salah satu metode yang paling populer dalam *unsupervised learning* adalah *clustering*.

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

*Convolutional Neural Network (CNN)* pertama kali diperkenalkan Yann LeCun bersama dengan timnya pada tahun 1998. LeCun mengenalkan versi awal CNN yang dikenal sebagai *LeNet* untuk mengenali karakter tulisan tangan, seperti angka. Model ini menjadi terobosan dalam pengenalan pola dan menjadi dasar dari berbagai penelitian lanjutan dalam pengembangan teknologi CNN. Seiring waktu, penelitian terus dilakukan untuk meningkatkan performa dan efisiensi CNN, menjadikannya salah satu metode paling populer dalam *deep learning* hingga saat ini.

CNN merupakan cabang *artificial intelligence* yang menggunakan *multi-layer neural networks* untuk mengidentifikasi, mengenali, mengklasifikasikan objek, maupun mendeteksi dan segmentasi objek dalam gambar. Lapisan-lapisan ini dilatih untuk mengenali pola-pola dasar, seperti

garis dan lengkungan, sebelum beralih ke pola yang lebih kompleks, seperti wajah dan objek (Taye, 2023).



**Gambar 2.2.** Arsitektur CNN (Bharatheedasan et al., 2023)

Gambar 2.2. memperlihatkan contoh arsitektur CNN. CNN memiliki 3 lapisan utama, yaitu *convolutional layer*, *pooling layer*, dan *fully connected layer*. Saat membangun CNN, selain memilih *activation function*, *loss function*, dan *optimizer*, juga harus mengatur banyak *hyperparameter* lain yang sangat mempengaruhi kinerja model. Seperti yang diketahui, tidak ada set *hyperparameter* tetap yang dapat menjamin solusi optimal setiap saat. Oleh karena itu, serangkaian percobaan sangat penting saat mengatur *hyperparameter* (Li et al., 2021).

CNN menghadapi beberapa tantangan dalam penggunaannya, salah satunya adalah *overfitting*. *Overfitting* terjadi ketika model terlalu kompleks dan cenderung belajar dari detail spesifik pada data pelatihan, sehingga gagal mengenali pola umum yang dapat diterapkan pada data baru. Untuk mengatasi masalah ini, teknik seperti *dropout* dapat digunakan. *Dropout* bekerja dengan menonaktifkan beberapa neuron secara acak selama proses pelatihan, sehingga model tidak terlalu bergantung pada neuron tertentu dan meningkatkan kestabilan proses pembelajaran.

Selain itu, keterbatasan jumlah data pelatihan juga menjadi kendala yang dapat memicu *overfitting*. Augmentasi data menjadi solusi yang efektif dalam situasi ini, dengan menciptakan variasi artifisial dari data yang ada, seperti mengubah ukuran, orientasi, atau kondisi pencahayaan gambar. Teknik ini tidak hanya meningkatkan jumlah data pelatihan tetapi juga

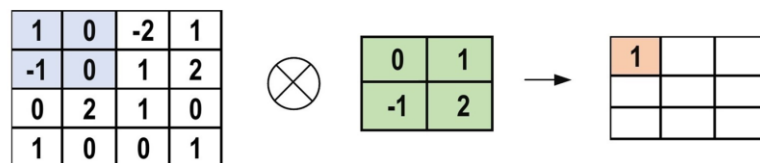


membantu model mengekstraksi fitur yang lebih umum dan aplikatif (Sarigül et al., 2019).

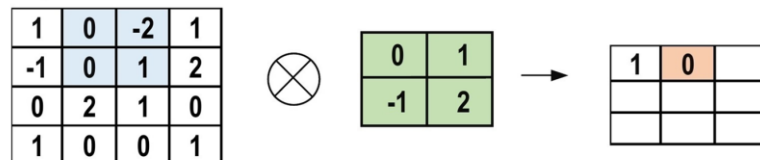
### 2.2.1. Convolutional Layer

*Convolutional layer* adalah komponen penting dalam CNN yang berfungsi mengekstraksi fitur dari data input menggunakan *convolutional filter* (kernel). Kernel berupa *grid* angka yang disebut bobot kernel.

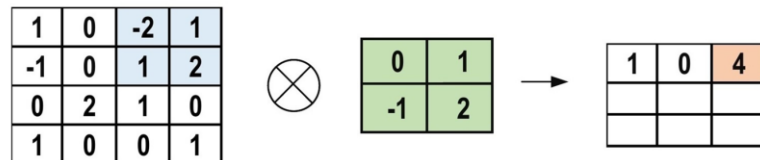
Step-1



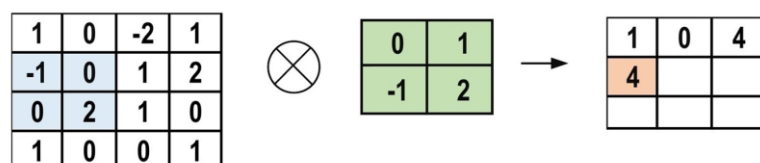
Step-2



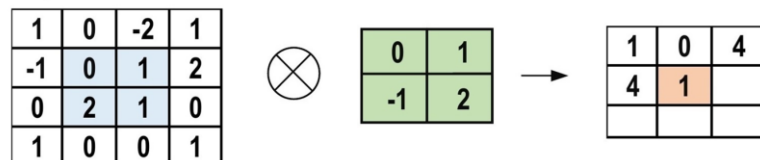
Step-3



Step-4



Step-5



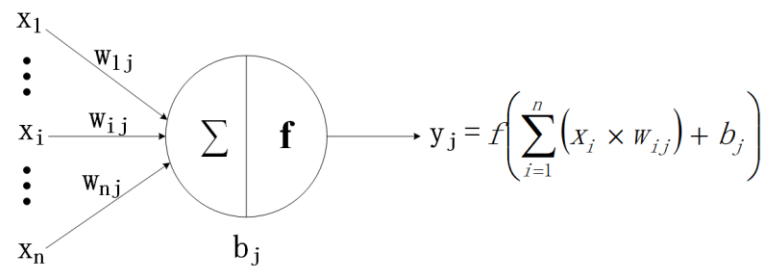
**Gambar 2.3.** Operasi Konvolusi (Alzubaidi, 2021)

Gambar 2.3. menunjukkan bagaimana operasi konvolusi melibatkan penggunaan kernel yang bergerak melintasi seluruh

gambar, di mana proses menghitung *weight dot* antara gambar dan kernel. Proses ini berlanjut hingga kernel tidak dapat digeser, dan hasilnya adalah *feature map* yang menggambarkan karakteristik dari gambar input. Nilai 1 pada *step-1* diperoleh dengan cara seperti berikut.

$$(1 \times 0) + (0 \times 1) + (-1 \times -1) + (0 \times 2) = 0 + 0 + 1 + 0 = 1$$

*Activation function* (fungsi aktivasi) sangat penting dalam *neural networks* karena membantu jaringan mempelajari pola-pola kompleks melalui transformasi non-linear. Beberapa sifat umum dari fungsi aktivasi adalah menambahkan elemen non-linear untuk membantu model belajar lebih baik, tidak membuat model menjadi terlalu lambat atau rumit, memastikan aliran gradien tetap lancar selama pelatihan, serta menjaga data tetap terdistribusi dengan baik agar model bisa belajar lebih optimal (Dubey et al., 2022).



**Gambar 2.4.** Struktur Umum *Activation Function* (Li et al., 2021)

Pada Gambar 2.4.  $x_i$  mewakili fitur input, di mana terdapat  $n$  fitur yang masuk ke neuron  $j$  secara bersamaan.  $w_{ij}$  adalah nilai bobot antara fitur input  $x_i$  dan neuron  $j$ , sedangkan  $b_j$  yaitu nilai bias yang menggambarkan keadaan internal neuron  $j$ . Output dari neuron  $j$  dilambangkan sebagai  $y_j$ , dan  $f(\cdot)$  adalah *activation function* yang digunakan, seperti fungsi *Rectified Linear Unit* (ReLU), *softmax*, *sigmoid*, atau fungsi lainnya.

ReLU adalah adalah fungsi linier tersegmentasi. Ketika input positif, maka outputnya adalah nilai itu sendiri sehingga turunannya satu. Sedangkan jika input bernilai negatif, output akan menjadi nol.

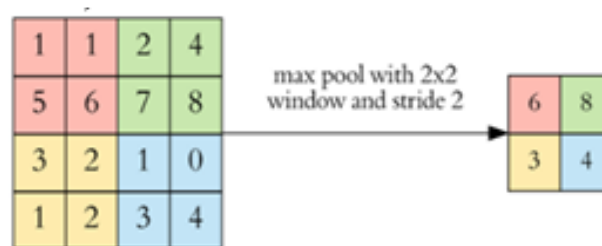
$$\text{ReLU}(x) = \max(0, x) \quad (1)$$

*Softmax* adalah fungsi aktivasi yang digunakan untuk *multi-class classification*, yaitu sampel hanya dapat diklasifikasikan ke dalam satu kelas saja. Nilai-nilai yang dihasilkan fungsi ini merepresentasikan probabilitas prediksi untuk setiap kelas. Semakin besar nilainya, semakin tinggi kemungkinan data termasuk dalam kelas tersebut.

$$\text{Softmax}(x) = \frac{e^{x_i}}{\sum_{j=1}^K e^{x_j}} \quad (2)$$

### 2.2.2. Pooling Layer

*Pooling layer* berfungsi untuk mengecilkan dimensi peta fitur (*feature maps*) tanpa menghilangkan informasi penting. Proses ini membantu mengompresi data dan parameter, mengurangi risiko *overfitting*, serta membuat jaringan lebih tahan terhadap perubahan lokal kecil. Selain itu, *pooling* juga mempercepat waktu pelatihan model, meningkatkan efisiensi ekstraksi fitur, dan mempertahankan sebagian besar informasi dominan meskipun ukuran peta fitur berkurang secara signifikan (Zhao et al., 2024).

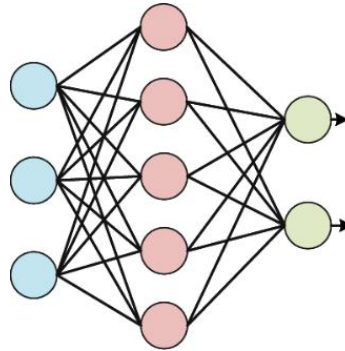


**Gambar 2.5.** *Max Pooling* (Elyasi & Hosseini, 2023)

Pada Gambar 2.5. menunjukkan proses perhitungan *max pooling*. *Max pooling*, yang umum digunakan memilih nilai terbesar dari sekumpulan nilai fitur untuk mewakili area tertentu dalam gambar.

### 2.2.3. *Fully Connected Layer*

*Fully connected layer* terletak di akhir arsitektur CNN dan berfungsi sebagai pengklasifikasi. Lapisan ini mengubah data dari *feature map* menjadi format yang siap untuk klasifikasi atau regresi, di mana neuron pada lapisan output sesuai dengan jumlah kelas yang diinginkan.

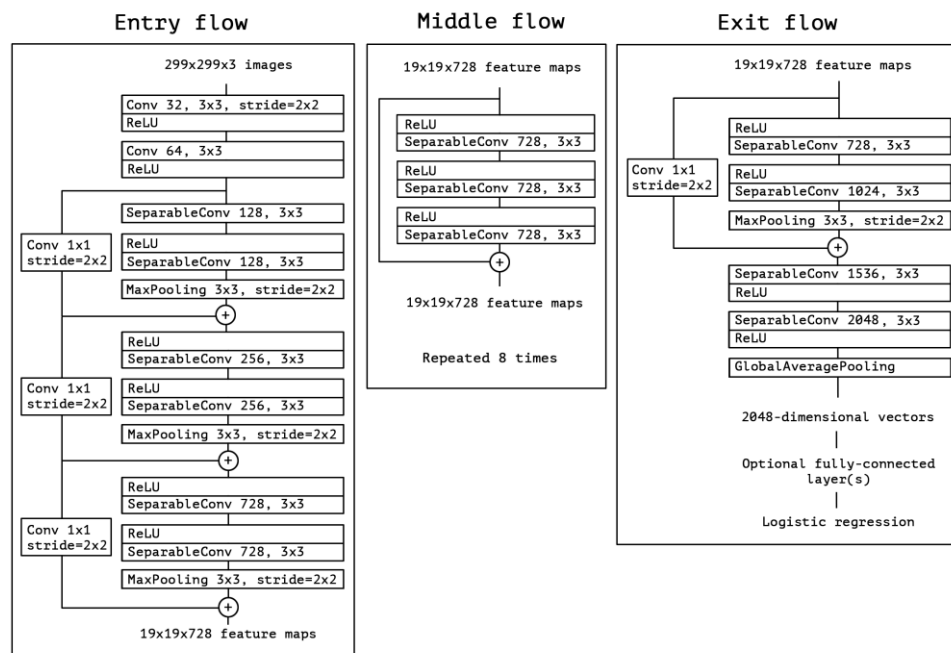


**Gambar 2.6.** *Fully Connected Layer* (Zhao, 2024)

Pada gambar 2.6. memperlihatkan lapisan *fully connected* yang setiap neuron terhubung langsung dengan semua neuron di lapisan sebelumnya. Inputnya berupa vektor satu dimensi yang dihasilkan dari proses *flattening*. *Fully connected layer* mengintegrasikan informasi lokal dari lapisan sebelumnya untuk menghasilkan output akhir dan bekerja sama dengan *activation function* untuk menjalankan tugas seperti klasifikasi atau regresi pada fitur yang telah diekstraksi.

## 2.3. *Xception*

*Extreme Inception* atau yang dikenal dengan *Xception* adalah arsitektur CNN yang diperkenalkan pada tahun 2017 (Chollet, 2017). *Xception* merupakan pengembangan dari arsitektur *Inception* dengan menggunakan *depthwise separable convolution*. *Xception* memiliki 36 layer yang disusun ke dalam 14 modul dan terbagi menjadi 3 tahapan utama, yaitu *entry flow*, *middle flow*, dan *exit flow*.



**Gambar 2.7.** Arsitektur Xception (Chollet, 2017)

Gambar 2.7. menunjukkan bahwa ukuran input arsitektur *Xception* adalah  $299 \times 299 \times 3$ . *Entry flow* terdiri dari konvolusi normal yang diikuti oleh *depthwise separable convolutions*. *Middle flow* terdiri dari 8 *miniblock*, yang masing-masing berisi 3 *depthwise separable convolutions*. Terakhir, *exit flow* terdiri dari *depthwise separable convolutions*, *flattening*, dan output.

Konsep dari *depthwise separable convolutions* adalah untuk mengurangi jumlah perkalian matriks dalam jaringan, sehingga mengurangi ukuran model. Alih-alih melakukan konvolusi normal dalam satu langkah dengan menggunakan filter yang banyak, *depthwise separable convolutions* melakukan konvolusi dalam dua langkah. Langkah pertama disebut *depthwise convolution*, yang menggunakan satu filter untuk setiap lapisan input. Output dari langkah ini akan memiliki ukuran tinggi $\times$ lebar $\times$ kedalaman dari gambar asli, dengan syarat *padding* digunakan selama konvolusi. Output dari *depthwise convolution* akan menjadi input bagi *pointwise convolution*. Dalam *pointwise convolution*, ukuran filter adalah  $1 \times 1 \times \text{depth}$ , di mana *depth* adalah kedalaman dari input sebelumnya.

## 2.4. Motorik Halus

Motorik halus merujuk pada kemampuan untuk melakukan gerakan kecil yang memerlukan koordinasi otot-otot kecil, seperti jari tangan, dengan penglihatan. Kemampuan ini tentunya penting dalam menjalankan berbagai aktivitas sehari-hari, seperti menulis, menggambar, mengancingkan pakaian, atau menggunakan alat makan. Pengembangan motorik halus pada anak merupakan aspek penting dalam tahap tumbuh kembang mereka, yang tidak hanya mendukung keterampilan praktis, tetapi juga berdampak pada aspek kognitif, emosional, dan sosial.

Kemampuan motorik halus dipengaruhi oleh stimulasi lingkungan, latihan, dan dukungan perkembangan neurologis. Aktivitas yang melibatkan koordinasi mata dan tangan, seperti meronce, bermain *puzzle*, atau menulis, dapat membantu memperkuat kemampuan ini. Selain itu, keterampilan motorik halus juga berperan dalam mendukung kemampuan akademis, terutama dalam menulis tangan, yang melibatkan gerakan presisi tinggi dan koordinasi yang baik.

Keterampilan ini juga erat kaitannya dengan perkembangan sistem saraf. Anak-anak dengan gangguan perkembangan, seperti *Autism Spectrum Disorder* (ASD), sebagian besar mengalami kesulitan dalam motorik halus, yang dapat mempengaruhi kemampuan mereka dalam aktivitas sehari-hari dan interaksi sosial. Sehingga penting untuk orang tua dan pendidik memberikan rangsangan yang sesuai melalui berbagai aktivitas yang menarik dan disesuaikan dengan usia anak. Penguatan motorik halus tidak hanya mendukung aktivitas fisik, tetapi juga mempengaruhi keterampilan kognitif seperti konsentrasi, daya ingat, dan penyelesaian masalah.

## 2.5. Kemampuan Menulis Tangan

Tulisan tangan berfungsi sebagai media komunikasi yang memungkinkan seseorang untuk mengekspresikan pemikiran dan idenya. Kemampuan menulis tangan juga berperan penting dalam pengembangan keterampilan lain seperti membaca, pengenalan huruf, dan pemahaman visual-spasial (Achymy et al., 2022). Selain itu, menulis tangan melibatkan

koordinasi motorik halus yang dapat meningkatkan perkembangan saraf, terutama pada anak-anak.

Keterampilan menulis tangan memiliki dampak besar terhadap perkembangan kognitif dan sosial. Kegiatan ini melibatkan interaksi antara otak dan tangan, yang merangsang area-area otak yang terkait dengan pembelajaran dan pengenalan. Ini juga meningkatkan fokus, daya ingat, dan pemahaman terhadap materi yang ditulis. Dari sisi sosial, kemampuan menulis dapat meningkatkan rasa percaya diri anak dalam berkomunikasi dan berpartisipasi di lingkungan akademis (Fuentes et al., 2009).

Kemampuan menulis tangan memiliki dampak yang sangat signifikan pada perkembangan kognitif dan sosial anak. Aktivitas menulis tangan melibatkan hubungan kompleks antara otak dan otot, yang tidak hanya mendukung pengembangan kemampuan membaca dan menulis, tetapi juga meningkatkan keterampilan visual-spasial, yang penting dalam pembelajaran dan pemecahan masalah. Menulis tangan memfasilitasi pembelajaran yang lebih mendalam karena melibatkan keterlibatan fisik dan mental yang lebih tinggi dibandingkan dengan mengetik. Selain itu, keterampilan menulis tangan juga berfungsi untuk melatih koordinasi motorik halus, yang berperan penting dalam perkembangan neurologis anak, dan dapat memberikan pengaruh positif dalam meningkatkan keterampilan sosial mereka, seperti berinteraksi dengan teman sebaya dan berpartisipasi di lingkungan akademik. Dengan demikian, kemampuan menulis tangan menjadi salah satu elemen penting dalam mendukung perkembangan holistik anak, baik secara kognitif, sosial, maupun emosional.

## **2.6. *Autism Spectrum Disorder (ASD)***

*Autism Spectrum Disorder (ASD)* yaitu adanya gangguan perkembangan sistem saraf yang mempengaruhi kemampuan komunikasi, interaksi sosial, serta perilaku individu (American Psychiatric Association, 2013). Penggunaan istilah “*spectrum*” mengacu pada perbedaan yang luas dalam gejala, keterampilan, dan tingkat keparahan yang dialami oleh penderita ASD. Hal ini menunjukkan bahwa setiap individu dapat memiliki

kombinasi tantangan yang unik, dengan beberapa mungkin memiliki kemampuan komunikasi yang sangat terbatas, sementara yang lain mungkin mampu berkomunikasi dengan baik tetapi memiliki masalah intelektual. Beberapa orang memerlukan dukungan yang besar dalam kehidupan sehari-hari mereka, sementara lainnya dapat hidup secara mandiri.

Menurut *American Psychiatric Association* yang tercantum dalam *Diagnostic and Statistical Manual of Mental Disorders* (DSM-5), membagi ASD ke dalam tiga tingkat keparahan, yaitu ringan, sedang, dan berat. Klasifikasi ini didasarkan pada sejauh mana individu dengan ASD membutuhkan dukungan dalam aspek komunikasi, interaksi sosial, serta perilaku repetitif atau restriktif. Tingkat ringan membutuhkan sedikit dukungan, tingkat sedang memerlukan dukungan substansial, dan tingkat berat memerlukan dukungan yang sangat substansial. Penjelasan dapat dilihat pada Tabel 2.1. di bawah ini.

**Tabel 2.1.** Tingkat Keparahannya ASD

<b>Tingkat Keparahan ASD</b>	<b>Komunikasi Sosial</b>	<b>Ketertarikan Spesifik &amp; Perilaku Berulang-ulang</b>
Tingkat 3 Butuh dukungan yang sangat substansial	Keterbatasan parah dalam keterampilan komunikasi sosial verbal dan nonverbal, menyebabkan kesulitan besar dalam berinteraksi sosial. Hal ini membuat seseorang jarang memulai percakapan dan hanya merespons interaksi sosial dengan cara yang sangat terbatas, seperti menggunakan sedikit kata-kata yang bisa dimengerti dan hanya bertindak untuk memenuhi kebutuhan dasar.	Perilaku kaku, sangat kesulitan menghadapi perubahan, atau pola perilaku yang terbatas dan berulang bisa sangat mengganggu aktivitas di berbagai aspek kehidupan. Hal ini juga menyebabkan stres atau kesulitan besar dalam mengalihkan perhatian atau mengubah tindakan.



Tingkat 2 Butuh dukungan yang substansial	Kesulitan yang jelas dalam komunikasi sosial, baik secara verbal maupun nonverbal, bahkan ketika sudah ada dukungan yang diberikan. Kesulitan ini terlihat dari jaranganya memulai interaksi sosial dan respon yang terbatas atau tidak biasa terhadap ajakan sosial dari orang lain.	Perilaku kaku, kesulitan menghadapi perubahan, kebiasaan yang terbatas dan berulang sering terlihat jelas bahkan oleh orang awam. Selain itu, orang dengan kondisi ini sering merasa stres atau kesulitan untuk mengalihkan perhatian mereka.
Tingkat 2 Butuh dukungan	Kesulitan dalam komunikasi sosial bisa terlihat jelas. Orang dengan kondisi ini mungkin kesulitan memulai interaksi sosial, saat berkomunikasi percakapan dua arah sering tidak berjalan lancar, dan tampak kurang tertarik dengan interaksi sosial.	Kekakuan perilaku mengganggu fungsi seseorang di satu atau lebih situasi, sulit berpindah dari satu aktivitas ke aktivitas lainnya, dan kesulitan dalam mengatur hal-hal yang menghambat kemandirian.

Meskipun hingga saat ini belum ditemukan satu penyebab yang jelas, perkembangan otak yang dipengaruhi oleh faktor genetik dan lingkungan menjadi penyebab potensial dari ASD. Beberapa faktor yang berkontribusi terhadap risiko ASD meliputi adanya riwayat keluarga dengan gangguan serupa, paparan terhadap zat berbahaya selama kehamilan, serta kondisi kesehatan ibu yang dapat mempengaruhi perkembangan janin (Hodges et al., 2020).

Gejala ASD bervariasi pada setiap anak, dan pengamatan terhadap gejala-gejala ini sangat penting untuk diagnosis yang akurat. Beberapa anak tidak merespon ketika dipanggil namanya atau tidak menunjuk objek untuk menunjukkan minat. Selain itu, beberapa individu dengan ASD mungkin menghindari kontak mata dan lebih suka menyendiri. Mereka mungkin kesulitan dalam memahami emosi orang lain maupun mengungkapkan perasaan mereka sendiri. Gejala lainnya termasuk keterlambatan dalam

perkembangan bicara dan bahasa, mengulang kata atau frasa (*echolalia*), memberikan jawaban yang tidak relevan pada pertanyaan, serta mudah terganggu oleh perubahan kecil. Anak memiliki minat yang obsesif dan membuat gerakan berulang seperti mengepakkan tangan, bergoyang, atau berputar-putar. Respon yang berbeda terhadap bunyi, bau, rasa, atau tekstur juga sering ditemukan (Hyman et al., 2020).

## BAB 3

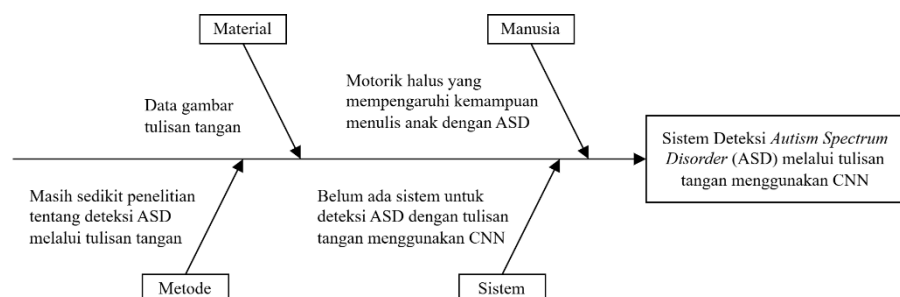
### ANALISIS DAN PERANCANGAN

#### 3.1. Analisis Sistem

Analisis sistem sebagai proses untuk memahami, mengidentifikasi, dan mendokumentasikan komponen-komponen serta hubungan yang ada dalam suatu sistem. Pada prosesnya, sistem dibagi menjadi bagian-bagian yang lebih rinci untuk mempermudah identifikasi dan meminimalkan kemungkinan kesalahan.

##### 3.1.1. Analisis Masalah

Analisis masalah merupakan langkah untuk memahami dan menguraikan masalah untuk mengembangkan solusi yang tepat. Pada penelitian ini, permasalahan utama adalah bagaimana membangun sistem yang mampu mengidentifikasi tingkat keparahan *Autism Spectrum Disorder* (ASD) pada anak berdasarkan analisis tulisan tangan. Permasalahan muncul dari kompleksitas dalam mendeteksi ASD secara dini, mengingat karakteristik gejala ASD yang beragam dan kemampuan tulis-menulis yang berbeda pada tiap anak. Keterbatasan dalam metode deteksi saat ini juga menyebabkan keterlambatan dalam diagnosis dan intervensi yang tepat. Pada tahap analisis masalah, penelitian ini menggunakan diagram Ishikawa (*fishbone*) sebagai berikut.



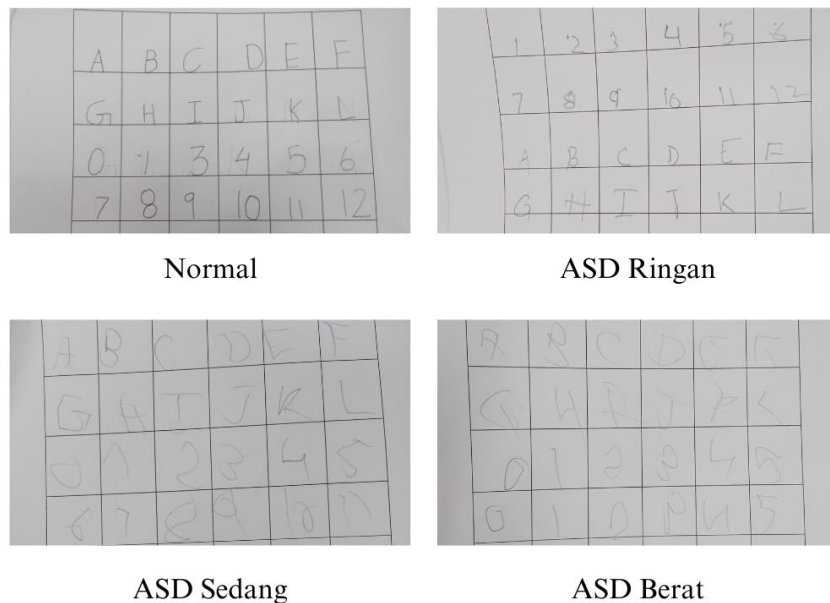
**Gambar 3.1.** Diagram Ishikawa

Gambar 3.1. menunjukkan diagram Ishikawa yang menggambarkan faktor-faktor utama yang berkontribusi pada

pengembangan sistem deteksi ASD melalui identifikasi tulisan tangan menggunakan *Convolutional Neural Network* (CNN). Diagram ini dibagi menjadi empat kategori utama: material, manusia, metode, dan sistem yang masing-masing memiliki elemen-elemen terkait.

### 3.1.2. Analisis Data

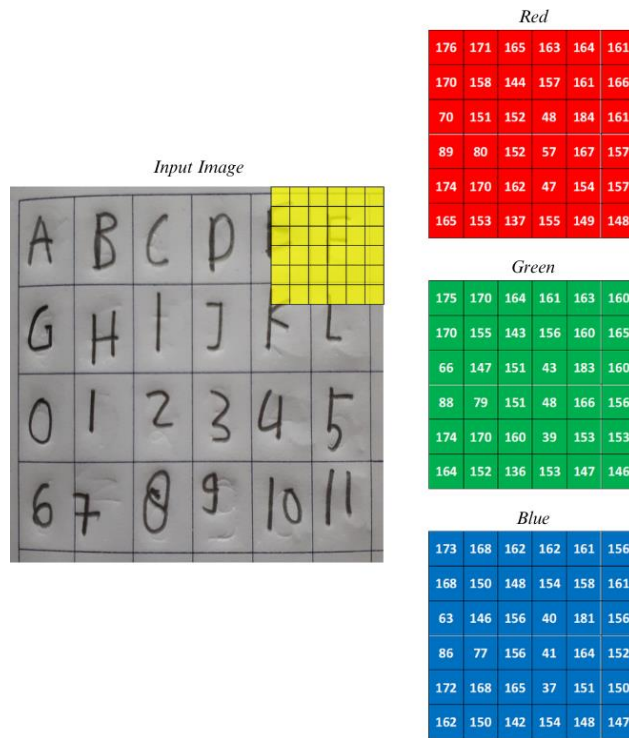
Data yang digunakan untuk penelitian ini bersumber dari situs *Kaggle* (<https://www.kaggle.com/datasets/imranliaqat32/autism-spectrum-disorder-in-childrenhandgestures/data>) dengan jumlah 500 data. Data ini digunakan untuk keperluan pelatihan (*training*) dan validasi (*validation*) dengan rasio 7:3. Sementara itu, untuk data pengujian (*testing*) berasal dari SLB Negeri Autis Sumatera Utara.



**Gambar 3.2.** Contoh Data

Gambar 3.2. memperlihatkan contoh gambar tulisan tangan dari masing-masing kelas, dengan karakteristik berbeda pada tingkat kerumitan tulisan tangan yang mencerminkan kategori normal, ASD ringan, ASD sedang, dan ASD berat.

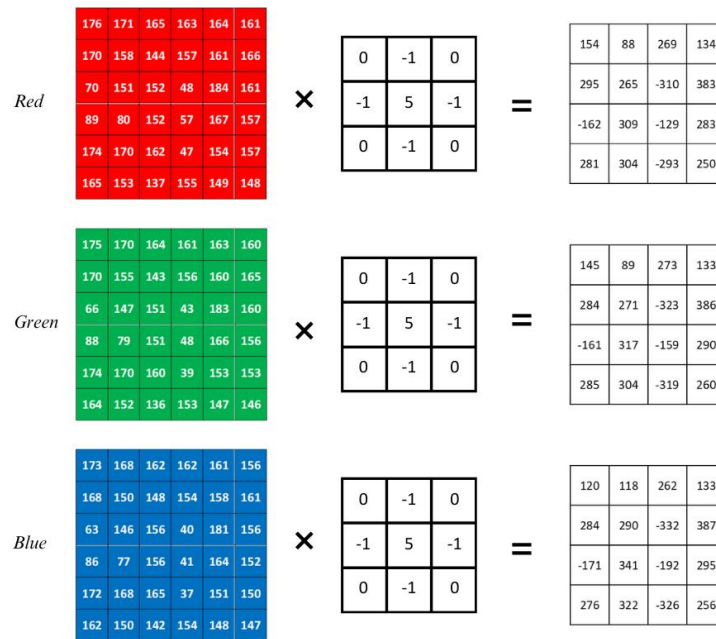
Pada bagian ini akan dibahas contoh perhitungan konvolusi pada data uji untuk menjelaskan konsep CNN yang digunakan. Perhitungan ini bertujuan untuk memberikan pemahaman mendalam tentang bagaimana CNN mengolah data input berupa gambar secara bertahap.



**Gambar 3.3.** Gambar Input dan Representasi Nilai Pixel dalam RGB

Gambar 3.3. menunjukkan gambar input dan representasi nilai pikselnya dalam *channel* RGB. Gambar input berupa citra tulisan tangan, sedangkan representasi nilai piksel menampilkan data numerik dalam bentuk matriks saluran yang merepresentasikan intensitas warna pada setiap piksel. Informasi nilai ini digunakan sebagai input untuk model CNN dalam proses ekstraksi fitur.

Setelah representasi nilai piksel diperoleh, langkah berikutnya adalah menerapkan operasi konvolusi untuk mengekstraksi fitur penting dari gambar. Operasi konvolusi memungkinkan model untuk mengenali pola visual spesifik, seperti garis, tepi, atau sudut, yang tersebar di seluruh gambar. Proses ini dilakukan dengan menggunakan kernel atau filter yang bergerak melintasi gambar untuk menghitung nilai keluaran pada setiap posisi. Hasil dari operasi ini berupa *feature map* yang merepresentasikan informasi penting dari gambar input.



**Gambar 3.4.** Proses Konvolusi

Gambar 3.4. menunjukkan proses konvolusi dengan kernel  $3 \times 3$ . Kernel ini bergerak melintasi gambar input secara bertahap, dengan menghitung nilai piksel yang bersesuaian pada area gambar dengan matriks kernel. Perhitungan konvolusi dengan *channel* RGB dilakukan dengan cara mengaplikasikan kernel secara terpisah pada setiap saluran warna (*red, green, blue*). Caranya dapat dilihat sebagai berikut.

- *Red*

$$(176 \times 0) + (171 \times -1) + (165 \times 0) + (170 \times -1) + (158 \times 5) + (144 \times -1) + (70 \times 0) + (151 \times -1) + (152 \times 0) = 0 - 171 + 0 - 170 + 790 - 144 + 0 - 151 + 0 = 154$$

- *Green*

$$(175 \times 0) + (170 \times -1) + (164 \times 0) + (170 \times -1) + (155 \times 5) + (143 \times -1) + (66 \times 0) + (147 \times -1) + (151 \times 0) = 0 - 170 + 0 - 170 + 775 - 143 + 0 - 147 + 0 = 145$$

- *Blue*

$$(173 \times 0) + (168 \times -1) + (162 \times 0) + (168 \times -1) + (150 \times 5) + (148 \times -1) + (63 \times 0) + (146 \times -1) + (156 \times 0) = 0 - 168 + 0 - 168 + 750 - 148 + 0 - 146 + 0 = 120$$

Hasil dari ketiga saluran ini kemudian dijumlahkan untuk menghasilkan satu nilai pada *feature map* di posisi tertentu. Proses ini dilakukan berulang hingga seluruh area gambar terproses.

419	295	804	400
863	826	-965	1156
-494	967	-480	868
842	930	-938	766

**Gambar 3.5.** *Feature Map*

Gambar 3.5. merupakan feature map yang dihasilkan dari proses konvolusi sebelumnya. Nilai 419 diperoleh dari penjumlahan ketiga saluran (*channel*).

$$154 + 145 + 120 = 419$$

Selanjutnya ReLU digunakan sebagai fungsi aktivasi untuk membantu memperkenalkan non-linearitas dalam model, yang memungkinkan jaringan untuk belajar pola-pola kompleks dari data.

419	295	804	400
863	826	-965	1156
-494	967	-480	868
842	930	-938	766

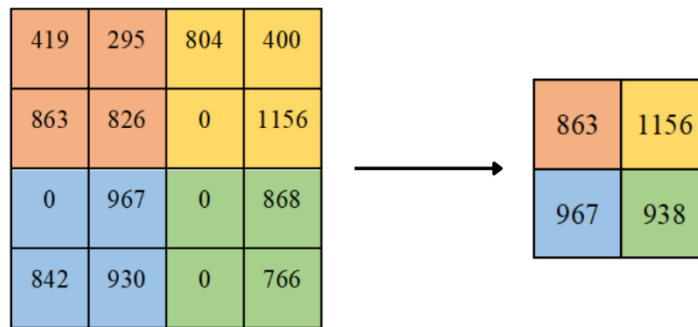
→

419	295	804	400
863	826	0	1156
0	967	0	868
842	930	0	766

**Gambar 3.6.** Penerapan ReLU

Pada Gambar 3.6. menunjukkan bahwa fungsi ini bekerja dengan mengganti semua nilai negatif pada *feature map* menjadi nol, sementara nilai positif dibiarkan tanpa perubahan.

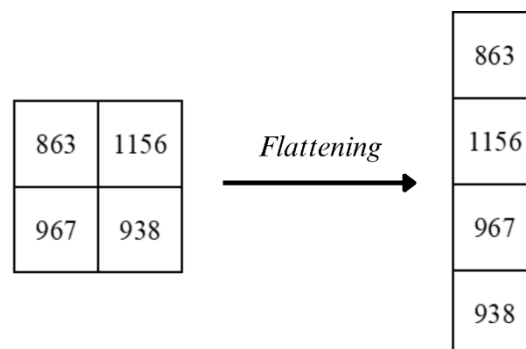
- Jika  $x > 0$ , maka  $x$  tetap.
- Jika  $x \leq 0$ , maka  $x$  diubah menjadi 0.



**Gambar 3.7.** Proses *Max Pooling*

Gambar 3.7. menunjukkan proses *max pooling* sebagai teknik pengurangan dimensi yang digunakan pada penelitian ini. *Max pooling* bekerja dengan memilih nilai maksimum dari sekumpulan piksel dalam area tertentu, dalam hal ini adalah  $2 \times 2$ . Warna yang digunakan menandakan bagian yang diproses pada setiap langkah. Proses ini menghasilkan keluaran berupa *feature map* dengan dimensi yang lebih kecil namun tetap mempertahankan fitur utama.

- Bagian merah =  $\max(419, 295, 863, 826) = 863$
- Bagian kuning =  $\max(804, 400, 965, 1156) = 1156$
- Bagian biru =  $\max(494, 967, 842, 930) = 967$
- Bagian hijau =  $\max(480, 868, 938, 766) = 938$



**Gambar 3.8.** *Flattening*

Pada Gambar 3.8. vektor yang sebelumnya multi-dimensi diubah menjadi vektor satu dimensi melalui proses *flattening*. Vektor hasil *flattening* ini kemudian digunakan sebagai input untuk lapisan *fully connected*.



Selanjutnya, untuk analisis ini digunakan 3 neuron pada lapisan *fully connected* dengan bobot dan bias yang ditentukan sebagai berikut.

$$\text{Bobot } (W) = \begin{bmatrix} 0.1 & 0.2 & 0.3 \\ 0.4 & 0.5 & 0.6 \\ 0.7 & 0.8 & 0.8 \\ 0.2 & 0.3 & 0.4 \end{bmatrix}$$

$$\text{Bias } (b) = [1, 2, 3]$$

Perhitungan dilakukan dengan menghitung hasil kali antara bobot ( $W$ ) dan input ( $x$ ), lalu menambahkan bias ( $b$ ) untuk setiap neuron. Formula yang digunakan adalah.

$$y = W \cdot x + b$$

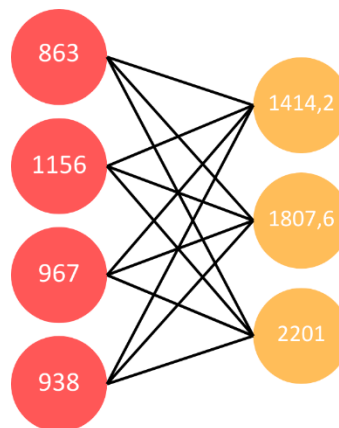
Keterangan:

$b$  = bias

$W$  = bobot

$x$  = input vektor dari proses *flatten*

$y$  = output *fully connected layer*



**Gambar 3.9.** Hasil *Fully Connected Layer*

Gambar 3.9. menunjukkan setiap neuron terhubung ke neuron selanjutnya. Perhitungan pada neuron dapat dipahami dengan menggunakan pendekatan seperti yang ditunjukkan pada contoh berikut.

- Neuron 1

$$(863 \times 0.1) + (1156 \times 0.4) + (967 \times 0.7) + (938 \times 0.2) + 1 = 1414.2$$

- Neuron 2  

$$(863 \times 0.2) + (1156 \times 0.5) + (967 \times 0.8) + (938 \times 0.3) + 2 = 1807.6$$
- Neuron 3  

$$(863 \times 0.3) + (1156 \times 0.6) + (967 \times 0.9) + (938 \times 0.4) + 3 = 2201$$

Sehingga, didapatkan output dari lapisan *fully connected* adalah [(1414.2), (1807.6), (2201)].

### 3.1.3. Analisis Kebutuhan

Analisis kebutuhan yaitu langkah untuk mengidentifikasi dan merumuskan kebutuhan sistem yang nantinya dibangun agar dapat berfungsi sesuai dengan tujuan penelitian.

#### 3.1.3.1. Kebutuhan Fungsional

Kebutuhan fungsional adalah spesifikasi atau tugas utama yang harus dijalankan oleh suatu sistem untuk memenuhi kebutuhan pengguna. Adapun aspek yang dibutuhkan secara fungsional pada sistem ini yaitu.

1. Sistem dapat mengenali gambar tulisan tangan.
2. Sistem mampu mengklasifikasikan tingkat keparahan *Autism Spectrum Disorder* (ASD) berdasarkan tulisan tangan menggunakan *Convolutional Neural Network* (CNN).
3. Sistem dapat memberikan hasil klasifikasi dan pemahaman lebih lanjut kepada pengguna.

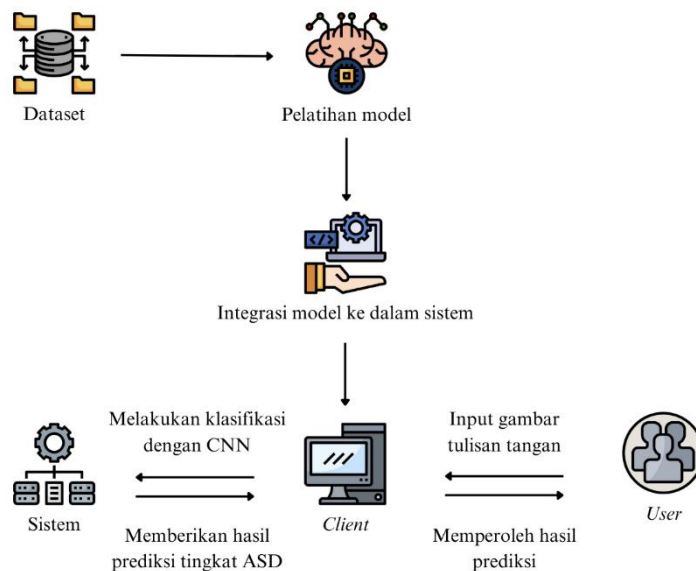
#### 3.1.3.2. Kebutuhan Non-Fungsional

Kebutuhan non-fungsional merujuk pada atribut kualitas dari sistem. Adapun aspek yang dibutuhkan secara non-fungsional untuk sistem ini yaitu sebagai berikut.

1. Sistem memiliki tampilan yang mudah dipahami oleh pengguna (*user-friendly*).
2. Sistem hanya dapat diakses melalui koneksi internet.
3. Sistem bisa diakses tanpa biaya yang besar dan tidak memerlukan perangkat tambahan.

### 3.1.4. Diagram Umum Sistem

Diagram umum sistem adalah representasi visual yang menunjukkan keseluruhan proses, alur kerja, dan interaksi antara komponen-komponen dalam suatu sistem.



**Gambar 3.10.** Diagram Umum Sistem

Pada Gambar 3.10. dapat dilihat bahwa *user* harus melakukan input gambar tulisan tangan terlebih dahulu, kemudian gambar tersebut akan diklasifikasi dengan menggunakan CNN. Hasil prediksi tingkat ASD kemudian akan ditampilkan.

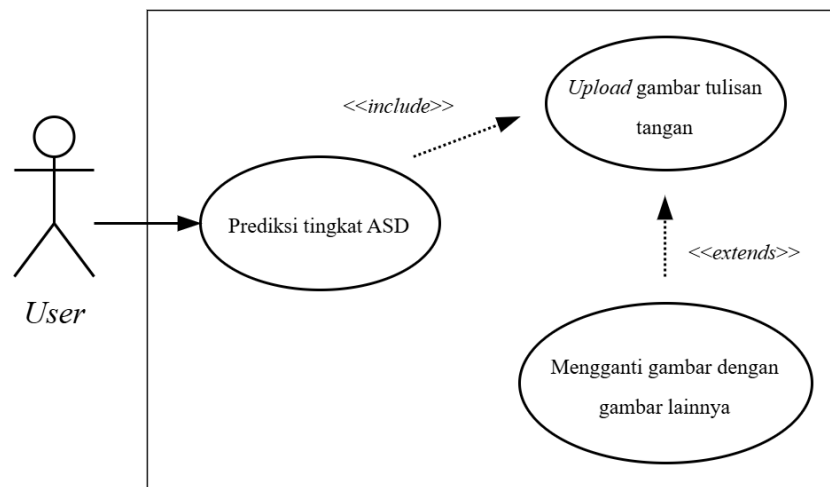
## 3.2. Pemodelan Sistem

Pemodelan sistem berfungsi sebagai penggambaran langkah-langkah interaksi antara pengguna dan sistem secara terstruktur, sehingga sistem dapat beroperasi sesuai dengan rancangan yang dibuat. Penelitian ini menggunakan *use case diagram*, *activity diagram*, *flowchart*, dan *sequence diagram* untuk pemodelan sistem.

### 3.2.1. Use Case Diagram

*Use case diagram* merepresentasikan interaksi antara aktor (pengguna atau sistem eksternal) dengan fungsi atau fitur utama dalam

suatu sistem, menggambarkan keterkaitan antara pengguna dan kebutuhan sistem.

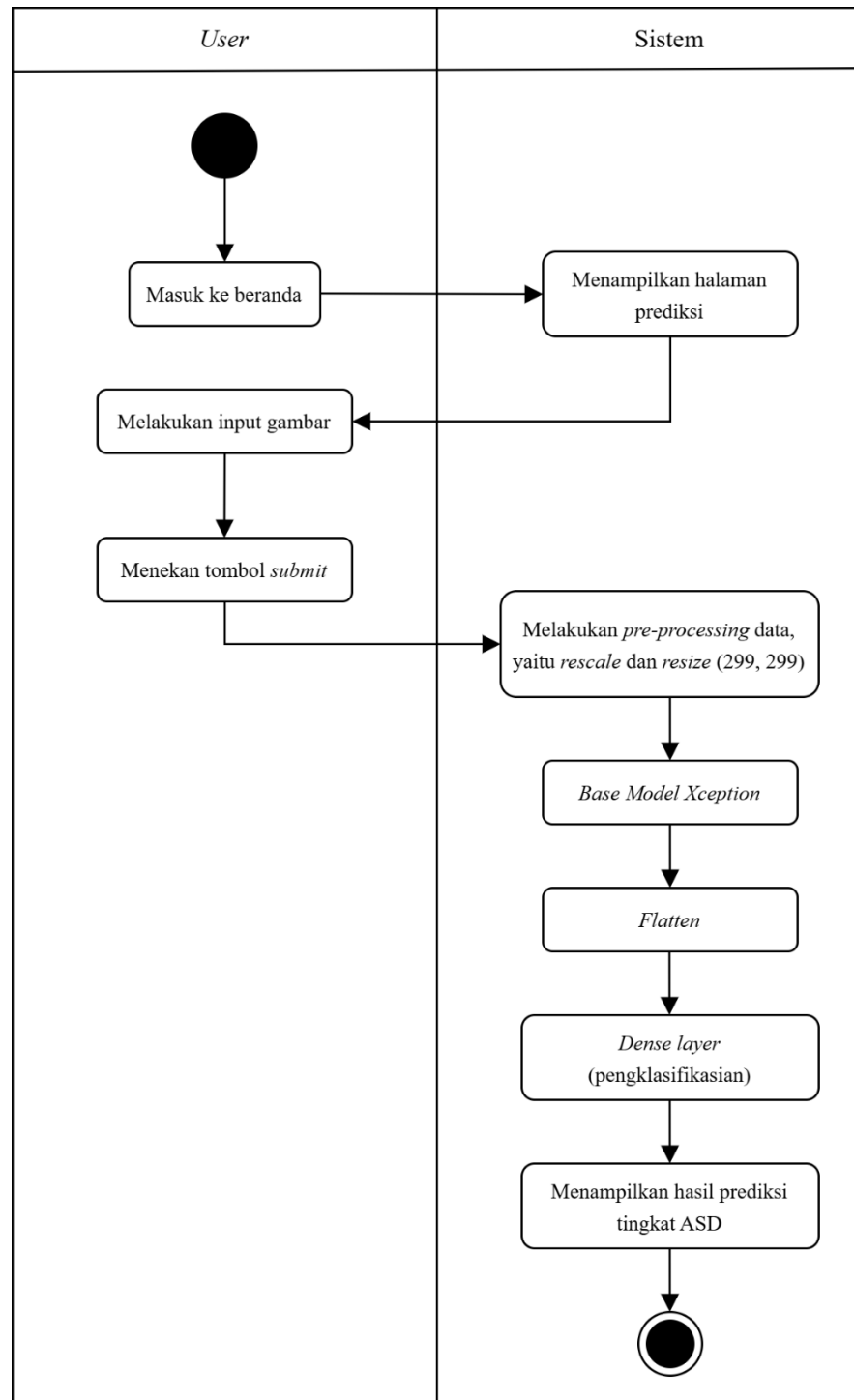


**Gambar 3.11.** *Use Case Diagram*

Gambar 3.11. menunjukkan interaksi antara *user* dan sistem prediksi tingkat *Autism Spectrum Disorder* (ASD) berbasis tulisan tangan. *User* berperan sebagai aktor yang menggunakan sistem untuk menjalankan fungsi utama, yaitu prediksi tingkat ASD. Dalam proses ini, *user* harus mengunggah gambar tulisan tangan melalui fungsi *upload* gambar tulisan tangan, yang merupakan langkah wajib (<<include>>) untuk memulai analisis dan prediksi. Selain itu, sistem juga menyediakan fungsi opsional berupa Mengganti Gambar dengan Gambar Lainnya, yang memungkinkan *user* untuk mengganti gambar jika data yang diunggah dirasa tidak sesuai (<<extends>>).

### 3.2.2. *Activity Diagram*

*Activity diagram* berguna untuk memvisualisasikan alur aktivitas dalam suatu proses, termasuk kondisi awal, transisi antar aktivitas, dan kondisi akhir, serta bagaimana aktivitas tersebut saling berhubungan.



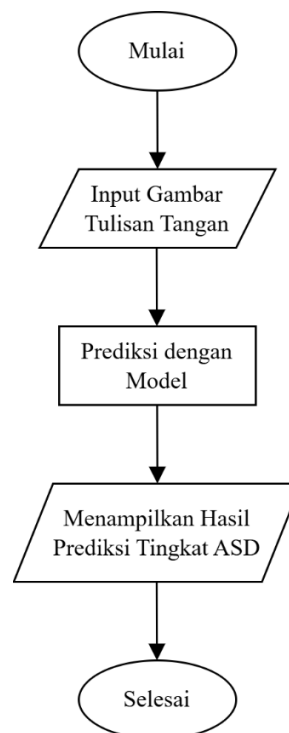
**Gambar 3.12.** *Activity Diagram*

Gambar 3.12. memperlihatkan alur interaksi antara sistem dan pengguna dalam proses prediksi tingkat *Autism Spectrum Disorder* (ASD) berdasarkan input gambar. Pengguna memulai dengan membuka halaman utama aplikasi, lalu mengunggah gambar yang

ingin diprediksi. Setelah itu, pengguna menekan tombol *submit* untuk mengirimkan gambar ke sistem. Di sisi sistem, aplikasi akan menampilkan halaman prediksi yang memungkinkan pengguna untuk mengunggah gambar. Setelah menerima input, sistem kemudian menjalankan proses prediksi menggunakan model yang telah dilatih sebelumnya. Akhirnya, sistem akan menampilkan hasil prediksi yang menunjukkan tingkat ASD berdasarkan gambar yang diunggah oleh pengguna, seperti normal, ringan, sedang, atau berat.

### 3.2.3. Flowchart

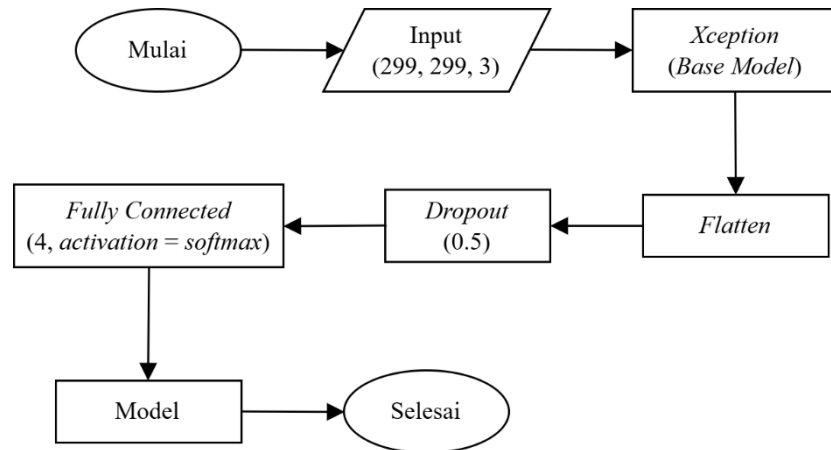
*Flowchart* (diagram alir) menggambarkan alur kerja atau proses dalam suatu sistem, dengan memanfaatkan simbol-simbol tertentu untuk mewakili langkah-langkah dalam proses tersebut.



**Gambar 3.13.** *Flowchart* Sistem

Gambar 3.13. menjelaskan alur sistem dimulai ketika pengguna melakukan prediksi tingkat ASD. Selanjutnya, pengguna diminta untuk mengunggah gambar tulisan tangan, yang akan digunakan sebagai data input. Setelah itu, sistem akan menjalankan langkah

Prediksi dengan Model, di mana model CNN menganalisis gambar untuk memprediksi tingkat ASD. Hasil analisis ini kemudian ditampilkan kepada pengguna yang mencakup informasi tingkat ASD (seperti normal, ringan, sedang, atau berat).

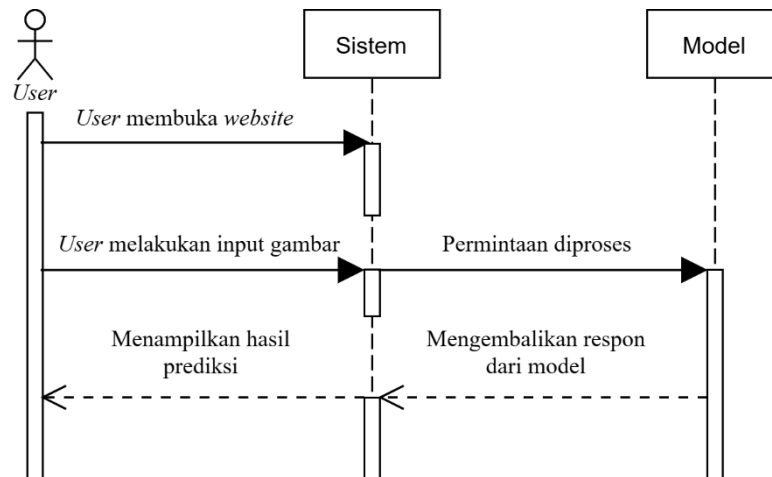


**Gambar 3.14.** Flowchart CNN dengan Arsitektur *Xception*

Berdasarkan pada Gambar 3.14. model menerima input untuk selanjutnya fitur diekstraksi menggunakan model *Xception*, kemudian melalui lapisan *flatten*. Selanjutnya, lapisan *dropout* sebesar 0.5 diterapkan untuk mencegah terjadinya *overfitting*, diikuti oleh lapisan *fully connected* dengan 4 neuron dan fungsi aktivasi *softmax* untuk menghasilkan klasifikasi ke dalam 4 kelas.

### 3.2.4. Sequence Diagram

*Sequence diagram* merujuk pada urutan proses antara entitas dalam suatu sistem berdasarkan waktu, menunjukkan bagaimana pesan atau data ditransmisikan dari satu objek ke objek lainnya secara berurutan.

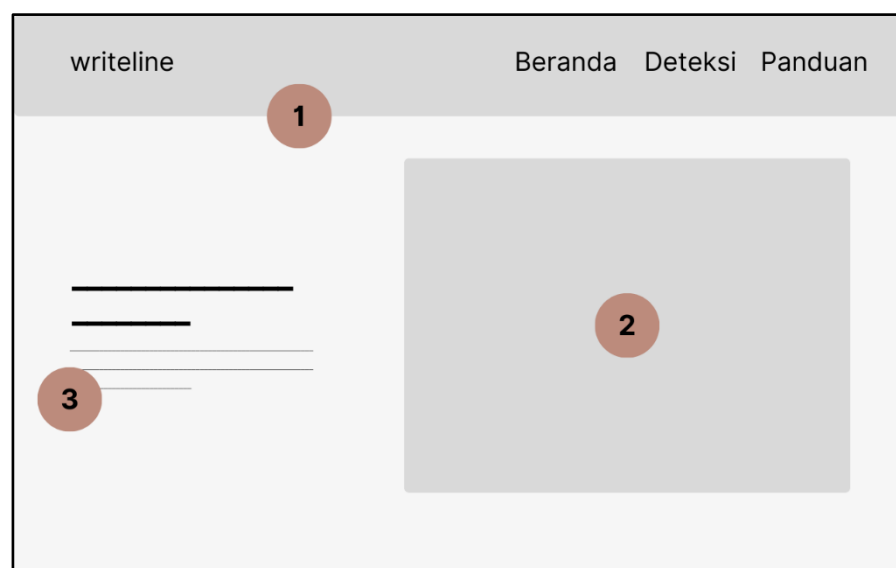


**Gambar 3.15.** *Sequence Diagram*

Gambar 3.15. menjelaskan bahwa 3 komponen yang saling berinteraksi, proses dari awal *user* membuka halaman *website* hingga permintaan *user* selesai dilaksanakan.

### 3.3. Perancangan *Interface*

Perancangan *interface* sebagai salah satu tahapan dalam pengembangan sistem yang berfokus pada pembuatan tampilan antarmuka yang nantinya akan digunakan pengguna dalam berinteraksi dengan sistem. Proses ini bertujuan untuk memastikan pengguna dapat dengan mudah mengunggah gambar, melihat hasil prediksi, dan melakukan interaksi lainnya secara intuitif.

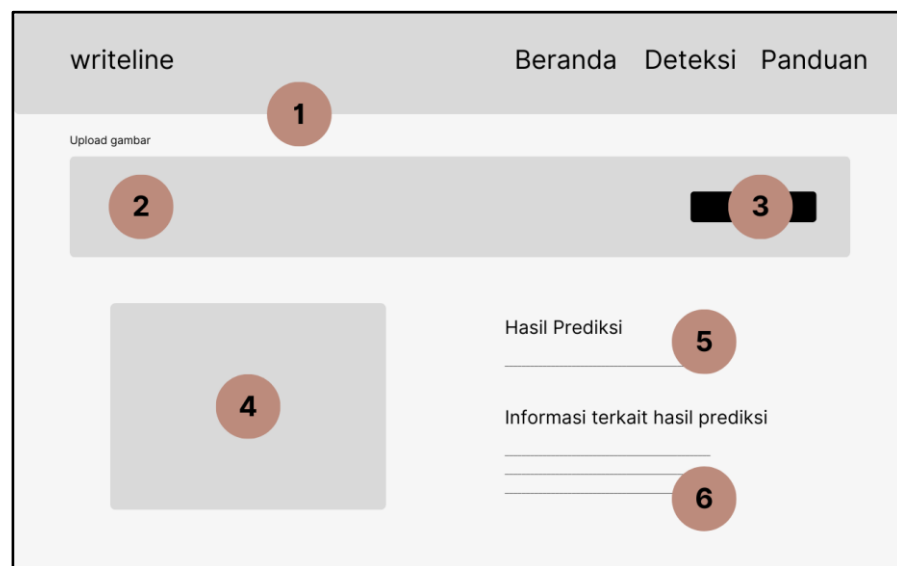


**Gambar 3.16.** Halaman Utama



Gambar 3.16. menunjukkan halaman utama yang dirancang untuk memberikan penjelasan singkat mengenai tujuan sistem, sehingga memudahkan pengguna memahami cara kerja sistem sebelum mulai menggunakannya. Adapun keterangan dari Gambar 3.16. adalah:

1. *Navbar* untuk menampilkan logo serta menu yang tersedia pada sistem.
2. Menampilkan gambar sebagai tambahan estetika sistem.
3. Deskripsi singkat mengenai tujuan sistem.



**Gambar 3.17.** Halaman Prediksi

Gambar 3.17. menunjukkan tampilan mengunggah gambar, kemudian pengguna menekan tombol prediksi, lalu sistem memproses gambar menggunakan model CNN. Hasil prediksi ditampilkan di layar beserta informasi tambahan. Untuk keterangan Gambar 3.17. yaitu:

1. *Navbar* untuk menampilkan logo serta menu yang tersedia pada sistem.
2. Perintah untuk mengunggah gambar.
3. *Button* untuk mengunggah gambar.
4. Gambar yang diunggah akan ditampilkan.
5. Kelas hasil prediksi.
6. Informasi singkat terkait hasil prediksi.

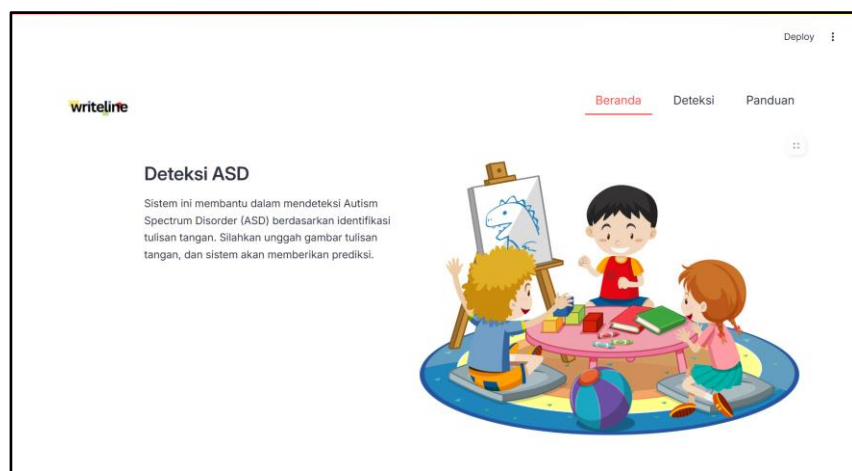
## BAB 4

### IMPLEMENTASI DAN PENGUJIAN

#### 4.1. Implementasi

Implementasi penelitian ini melibatkan pembangunan sistem dalam penerapan *Convolutional Neural Network* (CNN) dengan arsitektur *Xception*. Sistem berbasis *website* dibangun dengan menggunakan bahasa pemrograman *Python* dan *framework Streamlit*.

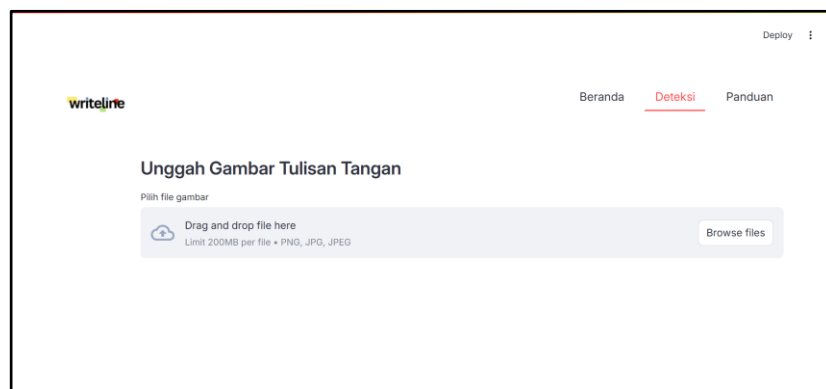
##### 4.1.1. Halaman Utama



**Gambar 4.1.** Tampilan Halaman Utama

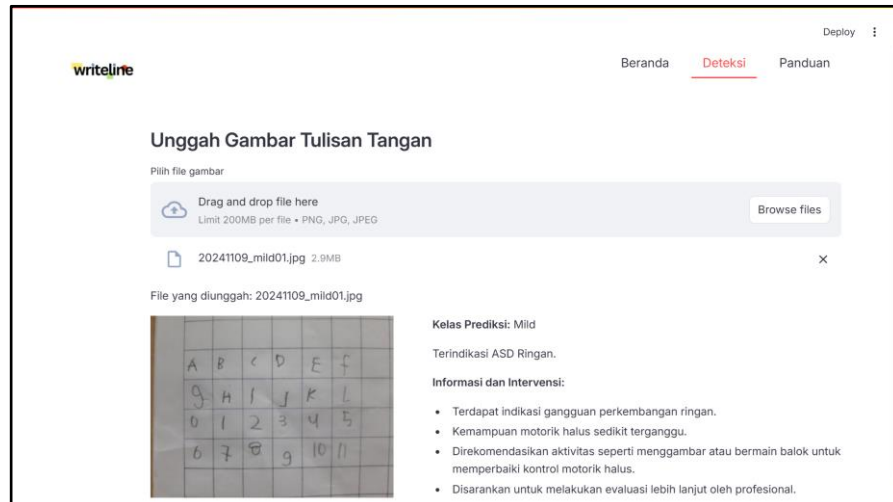
Gambar 4.1. menunjukkan halaman utama atau beranda. Ketika pengguna membuka *website*, halaman ini akan menjadi tampilan pertama yang ditampilkan.

##### 4.1.2. Halaman Prediksi



**Gambar 4.2.** Tampilan Halaman Prediksi

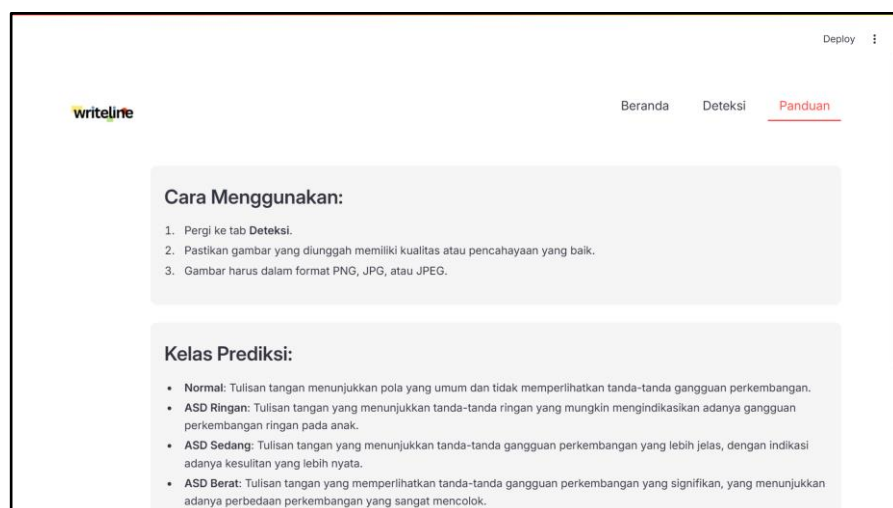
Gambar 4.2. merupakan halaman prediksi, di mana pengguna dapat mengunggah gambar tulisan tangan dan sistem akan memprosesnya.



**Gambar 4.3.** Tampilan Hasil Prediksi

Gambar 4.3. menampilkan halaman ketika pengguna sudah mengunggah gambar serta menyajikan hasil prediksi, dilengkapi dengan informasi tambahan dan rekomendasi intervensi.

#### 4.1.3. Halaman Panduan



**Gambar 4.4.** Tampilan Halaman Panduan

Gambar 4.4. memperlihatkan halaman panduan yang berisi penjelasan singkat mengenai cara menggunakan sistem, informasi

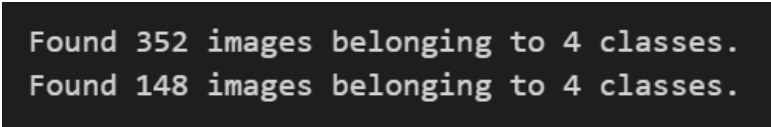
tentang setiap kelas, serta catatan penting terkait batasan dan tujuan sistem.

## 4.2. Pengujian Sistem

Tahapan ini merupakan langkah untuk memastikan sistem yang sudah dikembangkan mampu berfungsi memberikan hasil dengan baik dalam memprediksi tingkat keparahan *Autism Spectrum Disorder* (ASD) berdasarkan tulisan tangan.

### 4.2.1. Tahap Pelatihan Model

Pelatihan model dimulai dengan pembagian dataset menjadi data *training* dan *validation* dengan rasio 7:3. Perbandingan ini dipilih berdasarkan percobaan yang dilakukan, di mana rasio tersebut memberikan hasil terbaik.



```
Found 352 images belonging to 4 classes.
Found 148 images belonging to 4 classes.
```

**Gambar 4.5.** Informasi Kelas

Gambar 4.5. menunjukkan setiap folder dalam dataset sudah terstruktur dengan baik, di mana setiap folder mewakili 4 kelas yang berbeda, yaitu *mild*, *moderate*, *normal*, dan *severe*. Data gambar dinormalisasi dengan membagi nilai pikselnya dengan 255, sehingga semua nilai berada dalam rentang [0,1]. Ini membantu mempercepat pelatihan model.

Data *training* kemudian dilakukan proses augmentasi, seperti *rotasi*, *shear*, *zoom*, dan lainnya. Augmentasi ini dilakukan untuk meningkatkan variasi data sehingga model dapat lebih *robust* terhadap perbedaan pola dalam gambar. Sementara itu, data *validation* hanya dilakukan normalisasi tanpa augmentasi, agar evaluasi model lebih konsisten.

Proses pelatihan model dimulai dengan membuat generator data untuk dataset *training* dan *validation* menggunakan *ImageDataGenerator*. Generator ini mengambil data dari folder

masing-masing dan mengubah gambar menjadi ukuran target (299, 299) yang sesuai dengan arsitektur *Xception*. Ukuran ini dipilih karena *Xception* memerlukan input gambar sebesar (299, 299, 3). Nilai 3 pada dimensi terakhir menunjukkan jumlah saluran warna dalam gambar, yaitu RGB, yang mengindikasikan bahwa gambar yang digunakan berwarna dan memiliki tiga saluran untuk merepresentasikan intensitas warna merah, hijau, dan biru. *Batch size* ditetapkan sebanyak 32 gambar, dan kelas diatur menggunakan mode *categorical* untuk klasifikasi multikelas.

Layer (type)	Output Shape	Param #
xception (Functional)	(None, 10, 10, 2048)	20,861,480
flatten_4 (Flatten)	(None, 204800)	0
dropout_4 (Dropout)	(None, 204800)	0
dense_4 (Dense)	(None, 4)	819,204

**Gambar 4.6.** Model Summary

Pada Gambar 4.6. arsitektur *Xception* digunakan sebagai *base* model tanpa lapisan *fully connected* pada bagian atas. Setelah itu, model dilengkapi dengan lapisan *flatten* untuk mengubah output tensor multidimensi menjadi vektor satu dimensi, kemudian diikuti dengan lapisan *dropout* sebesar 0.5 untuk mencegah *overfitting* dengan menghapus secara acak 50% neuron selama pelatihan. Lapisan terakhir adalah *dense* yang menghasilkan 4 unit output dan menggunakan fungsi aktivasi *softmax* untuk klasifikasi multikelas.

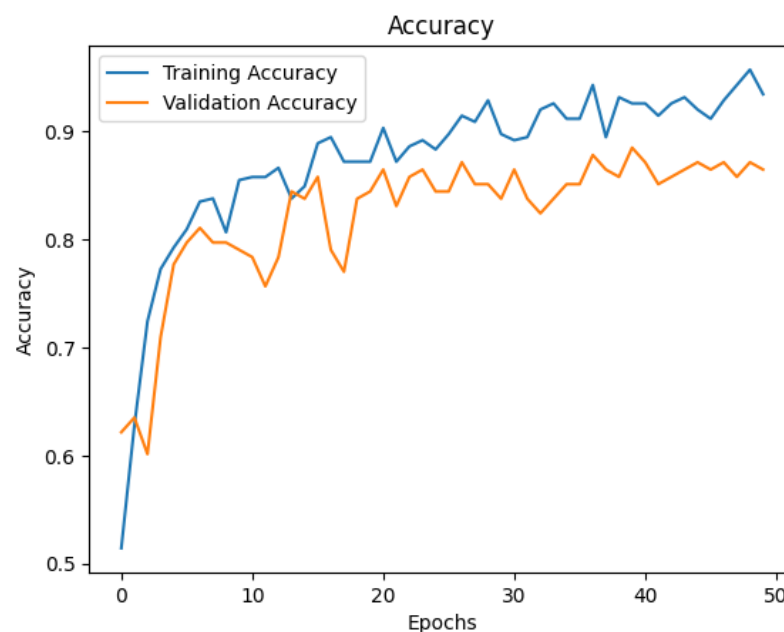
Selanjutnya model dikompilasi menggunakan *optimizer* adam dengan *learning rate* yang kecil, yaitu 0.0001. Penggunaan *learning rate* yang rendah ini bertujuan untuk memastikan bahwa model dapat belajar secara bertahap dan menghindari lompatan besar yang bisa mengganggu proses pelatihan. Fungsi *loss* yang digunakan adalah *categorical\_crossentropy*, yang cocok untuk masalah klasifikasi multikelas. Sebagai metrik evaluasi, model menggunakan *accuracy* untuk memantau kinerja model sepanjang proses pelatihan dan

validasi. Kombinasi ini memungkinkan model untuk memperbarui bobotnya secara efektif selama proses pelatihan guna meminimalkan kehilangan dan meningkatkan akurasi prediksi.

```
Epoch 45/50
11/11 ----- 164s 15s/step - accuracy: 0.9155 - loss: 0.2205 - val_accuracy: 0.8716 - val_loss: 0.3462
Epoch 46/50
11/11 ----- 197s 18s/step - accuracy: 0.9134 - loss: 0.2152 - val_accuracy: 0.8649 - val_loss: 0.3831
Epoch 47/50
11/11 ----- 145s 14s/step - accuracy: 0.9273 - loss: 0.1907 - val_accuracy: 0.8716 - val_loss: 0.3686
Epoch 48/50
11/11 ----- 163s 15s/step - accuracy: 0.9274 - loss: 0.1992 - val_accuracy: 0.8581 - val_loss: 0.3964
Epoch 49/50
11/11 ----- 164s 15s/step - accuracy: 0.9435 - loss: 0.1687 - val_accuracy: 0.8716 - val_loss: 0.3964
Epoch 50/50
11/11 ----- 162s 15s/step - accuracy: 0.9437 - loss: 0.1868 - val_accuracy: 0.8649 - val_loss: 0.4059
```

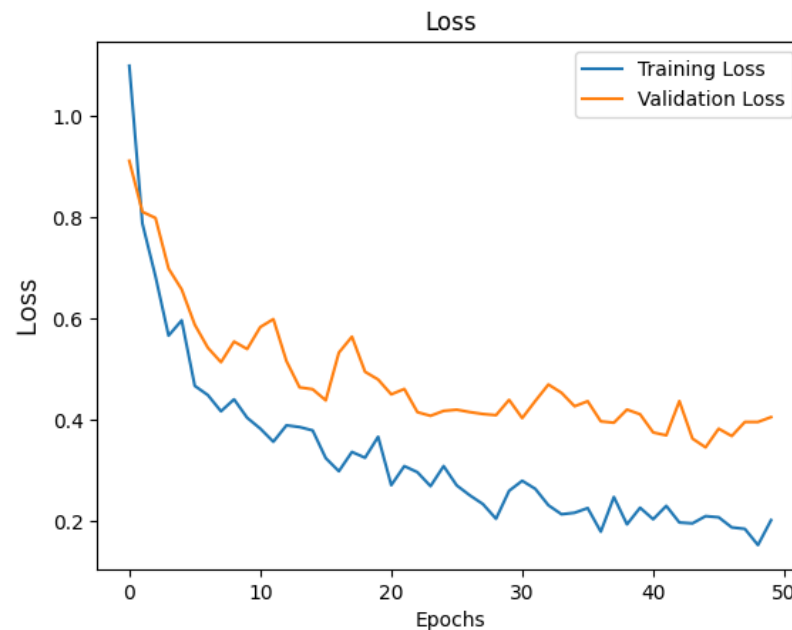
**Gambar 4.7.** Proses *Training Model*

Gambar 4.7. memperlihatkan bahwa setelah pelatihan selama 50 *epoch*, model berhasil mencapai tingkat akurasi sebesar 94% pada data *training* dan 86% pada data *validation*, memperlihatkan bahwa model dapat mengenali pola dengan baik dalam dataset tersebut. Akurasi yang baik ini mencerminkan efektivitas arsitektur model dan parameter yang digunakan selama proses pelatihan. Namun, penting untuk mengevaluasi performa model pada data uji untuk memastikan kemampuan generalisasi model terhadap data baru.



**Gambar 4.8.** Grafik *Accuracy* dan *Validation Accuracy*

Grafik pada Gambar 4.8. menunjukkan peningkatan *accuracy* selama 50 *epochs* untuk *training* dan *validation*. *Training accuracy* terus meningkat hingga mendekati 1, menunjukkan model belajar baik dari data pelatihan. *Validation accuracy* stabil di sekitar 0.85 setelah beberapa *epochs*, mengindikasikan performa yang cukup baik tanpa *overfitting* yang signifikan.



**Gambar 4.9.** Grafik *Loss* dan *Validation Loss*

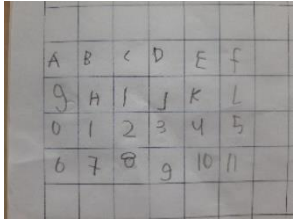
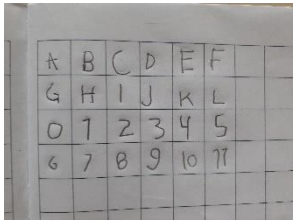
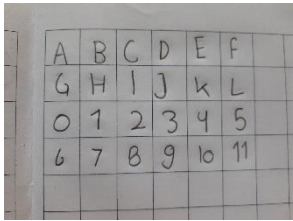
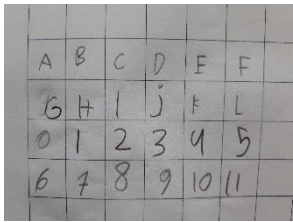
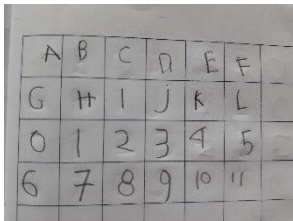
Berdasarkan Gambar 4.9. grafik menunjukkan perubahan nilai *loss* pada data *training* dan *validation* selama 50 *epochs*. Nilai *loss* pada data *training* terus menurun, menunjukkan bahwa model semakin baik dalam memprediksi data pelatihan. Sementara itu, *loss* pada data *training* juga mengalami penurunan meskipun lebih tidak stabil. Hal ini mengindikasikan bahwa model mampu belajar dengan baik tanpa menunjukkan gejala *overfitting* yang signifikan.

#### 4.2.2. Tahap Pengujian

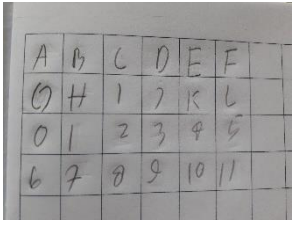
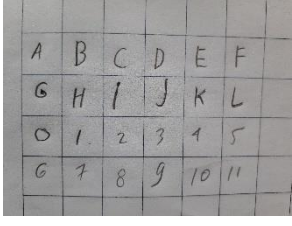
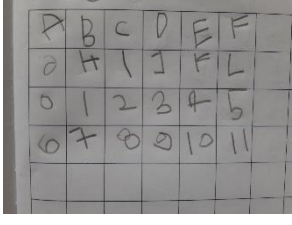
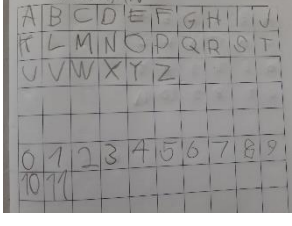
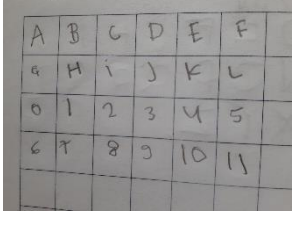
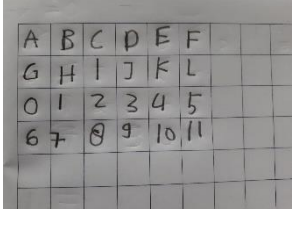
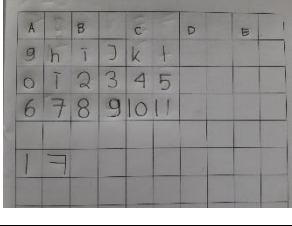
Pengujian dilakukan dengan mengevaluasi kinerja model menggunakan 25 sampel data baru yang belum pernah dilihat sebelumnya, kemudian membandingkan hasil diagnosis dari ahli

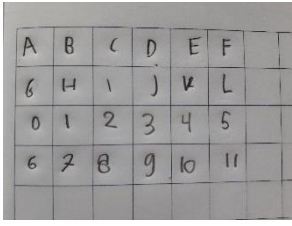
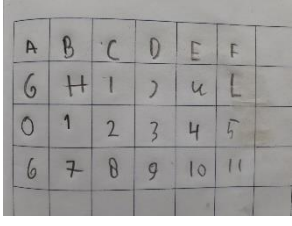
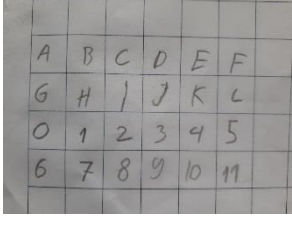
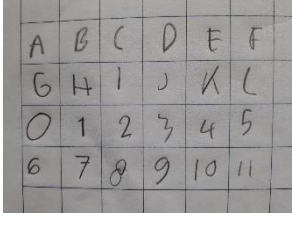
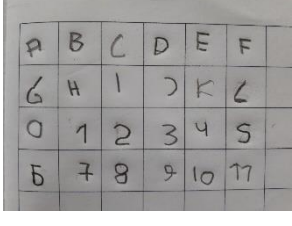
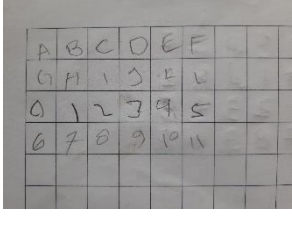
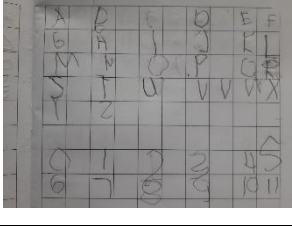
dengan diagnosis yang diberikan oleh sistem. Hasil pengujian dapat dilihat di bawah ini.

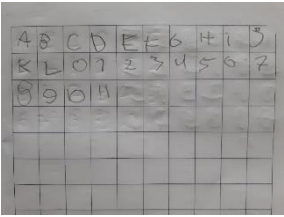
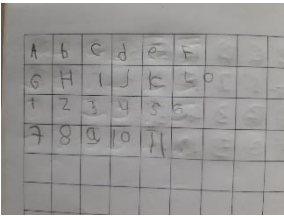
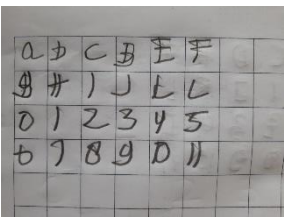
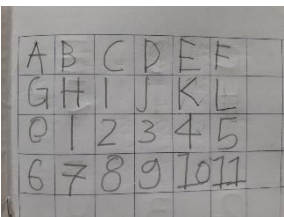
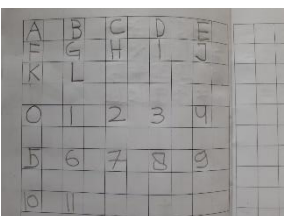
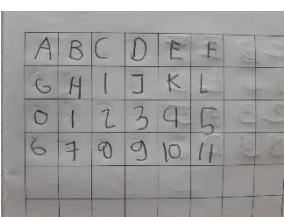
**Tabel 4.1.** Hasil Pengujian Sistem

Gambar	Diagnosis Pakar	Prediksi Sistem	Status Hasil
	ASD Ringan	ASD Ringan	✓
	ASD Ringan	ASD Ringan	✓
	Tidak Terindikasi Gangguan	Tidak Terindikasi Gangguan	✓
	Tidak Terindikasi Gangguan	Tidak Terindikasi Gangguan	✓
	Tidak Terindikasi Gangguan	Tidak Terindikasi Gangguan	✓



	Tidak Terindikasi Gangguan	Tidak Terindikasi Gangguan	✓
	Tidak Terindikasi Gangguan	Tidak Terindikasi Gangguan	✓
	ASD Sedang	ASD Sedang	✓
	ASD Sedang	ASD Ringan	✗
	ASD Ringan	Tidak Terindikasi Gangguan	✗
	ASD Ringan	ASD Ringan	✓
	ASD Ringan	ASD Ringan	✓

	Tidak Terindikasi Gangguan	Tidak Terindikasi Gangguan	✓
	Tidak Terindikasi Gangguan	Tidak Terindikasi Gangguan	✓
	Tidak Terindikasi Gangguan	ASD Ringan	✗
	Tidak Terindikasi Gangguan	Tidak Terindikasi Gangguan	✓
	Tidak Terindikasi Gangguan	Tidak Terindikasi Gangguan	✓
	ASD Berat	ASD Sedang	✗
	ASD Berat	ASD Berat	✓

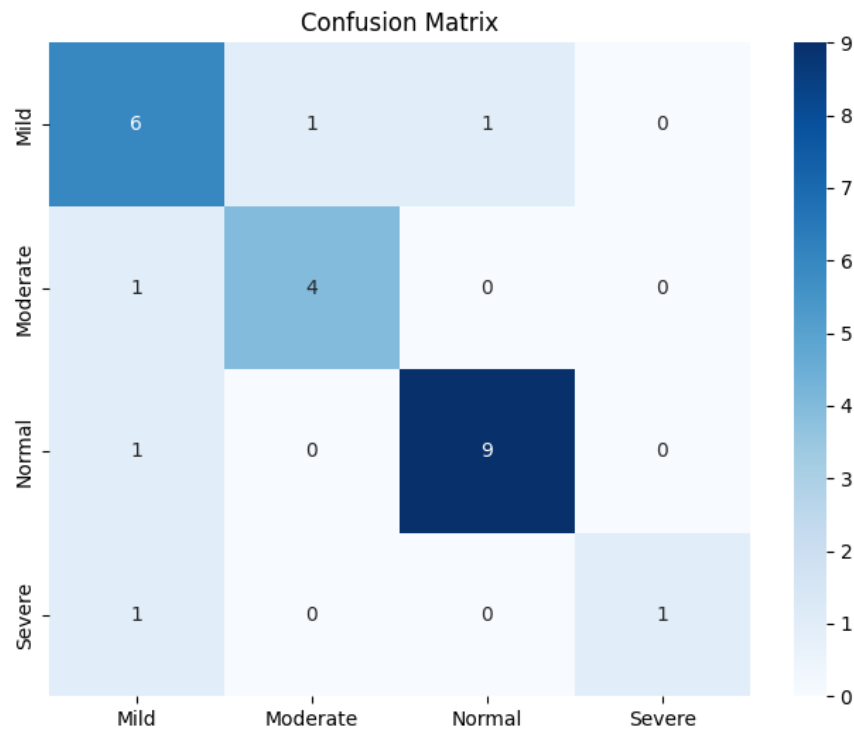
	ASD Sedang	ASD Sedang	✓
	ASD Sedang	ASD Sedang	✓
	ASD Sedang	ASD Sedang	✓
	ASD Ringan	ASD Ringan	✓
	ASD Ringan	ASD Sedang	✗
	ASD Ringan	ASD Ringan	✓

Keterangan:

✓ : Hasil prediksi benar

✗ : Hasil prediksi salah

Penelitian ini menggunakan *confusion matrix* untuk memberikan detail mengenai kinerja model pada tiap kelas. Dari hasil pengujian terhadap 25 data uji pada Tabel 4.1 diperoleh *confusion matrix* sebagai berikut.



**Gambar 4.10.** *Confusion Matrix*

Gambar 4.10. merupakan *confusion matrix* yang selanjutnya akan digunakan untuk menghitung *metric evaluation*. Hasil *confusion matrix* juga dapat dilihat pada Tabel 4.2. berikut.

**Tabel 4.2.** Tabel *Confusion Matrix*

Kelas	Hasil Prediksi			
	Normal	Ringan	Sedang	Berat
Normal	9	1	0	0
Ringan	1	6	1	0
Sedang	0	1	4	0
Berat	0	1	0	1

Berdasarkan *confusion matrix* yang sudah didapatkan, maka *metric evaluation* yang terdiri dari *accuracy*, *precision*, *recall*, dan *f1-score* dapat dihitung sebagai berikut:

1. *Accuracy* (akurasi) menunjukkan seberapa baik model dalam membuat prediksi yang tepat, baik kelas positif maupun negatif.

$$\begin{aligned} \text{Accuracy} &= \frac{\text{True Positive}}{\text{Total Dataset}} \times 100\% \\ &= \frac{20}{25} \times 100\% \\ &= 80\% \end{aligned}$$

2. *Precision* (presisi) mengukur ketepatan model dalam memprediksi kelas positif.

$$\begin{aligned} \text{Precision} &= \frac{\text{True Positive}}{\text{True Positive} + \text{False Positive}} \times 100\% \\ &= \frac{20}{20 + 5} \times 100\% \\ &= 80\% \end{aligned}$$

3. *Recall* (sensitivitas) mengukur kemampuan model untuk mendeteksi semua data positif.

$$\begin{aligned} \text{Recall} &= \frac{\text{True Positive}}{\text{True Positive} + \text{False Negative}} \times 100\% \\ &= \frac{20}{20 + 3} \times 100\% \\ &= 86\% \end{aligned}$$

4. *F1-score* dapat dihitung dengan menggabungkan antara *precision* dan *recall* sehingga menghasilkan nilai rata-rata.

$$\begin{aligned} \text{F1-score} &= 2 \times \frac{\text{Recall} \times \text{Precision}}{\text{Recall} + \text{Precision}} \times 100\% \\ &= 2 \times \frac{0.8 \times 0.86}{0.8 + 0.86} \times 100\% \\ &= 83\% \end{aligned}$$

## BAB 5

### PENUTUP

#### 5.1. Kesimpulan

Berdasarkan temuan dari penelitian yang telah dilaksanakan untuk prediksi tingkat keparahan anak *Autism Spectrum Disorder* (ASD) berdasarkan tulisan tangan, diperoleh kesimpulan sebagai berikut.

1. Penggunaan *Convolutional Neural Network* (CNN) dengan arsitektur *Xception* pada 352 data *training* dan 148 data *validation* menghasilkan akurasi sebesar 94% setelah 50 *epoch* pelatihan.
2. Pada 25 data uji, model memperoleh akurasi sebesar 80%, *precision* 80%, *recall* 86%, dan *f1-score* 83%
3. Pengaturan *dropout* dan *learning rate optimizer* mempengaruhi nilai *accuracy* dan *loss* selama pelatihan model.
4. Beberapa faktor yang dapat mempengaruhi hasil prediksi antara lain adanya kemiripan karakteristik antarkelas, serta jumlah dataset yang terbatas.

#### 5.2. Saran

Beberapa saran yang dapat diajukan untuk penelitian selanjutnya antara lain yaitu.

1. Menambah jumlah dataset guna meningkatkan kemampuan model dalam mengadaptasi variasi data dan memastikan hasil prediksi yang lebih optimal.
2. Mengintegrasikan fitur *real-time* dan dapat diakses melalui ponsel untuk meningkatkan kemudahan bagi pengguna.

## DAFTAR PUSTAKA

- Achmyy, Z., Kadar, M., Razaob, N. A., & Wan Yunus, F. (2022). Factors influencing handwriting development among preschool children: A systematic review. *Kesmas: Jurnal Kesehatan Masyarakat Nasional*, 17(4), 235. <https://doi.org/10.21109/kesmas.v17i4.6209>
- Ahmed, S. F., Alam, M. S. B., Hassan, M., Rozbu, M. R., Ishtiaq, T., Rafa, N., ... & Gandomi, A. H. (2023). Deep learning modelling techniques: current progress, applications, advantages, and challenges. *Artificial Intelligence Review*, 56(11), 13521-13617. <https://doi.org/10.1007/s10462-023-10466-8>
- Alsaidi, M., Obeid, N., Al-Madi, N., Hiary, H., & Aljarah, I. (2024). A convolutional deep neural network approach to predict autism spectrum disorder based on eye-tracking scan paths. *Information*, 15(3), 133. <https://doi.org/10.3390/info15030133>
- Alzubaidi, L., Zhang, J., Humaidi, A. J., Al-Dujaili, A., Duan, Y., Al-Shamma, O., ... & Farhan, L. (2021). Review of deep learning: concepts, CNN architectures, challenges, applications, future directions. *Journal of big Data*, 8, 1-74. <https://doi.org/10.1186/s40537-021-00444-8>
- American Psychiatric Association. (2013). Diagnostic and Statistical Manual of Mental Disorders, Fifth Edition. American Psychiatric Association
- Bharatheedasan, K., Maity, T., Kumaraswamidhas, L. A., & Durairaj, M. (2023). An intelligent of fault diagnosis and predicting remaining useful life of rolling bearings based on convolutional neural network with bidirectional LSTM. *Sādhana*, 48(3), 131. <https://doi.org/10.1007/s12046-023-02169-1>
- Chollet, F. (2017). Xception: Deep learning with depthwise separable convolutions. In *Proceedings of the IEEE conference on computer vision and pattern recognition* (pp. 1251-1258). <https://doi.org/10.1109/CVPR.2017.195>

- Dubey, S. R., Singh, S. K., & Chaudhuri, B. B. (2022). Activation functions in deep learning: A comprehensive survey and benchmark. *Neurocomputing*, 503, 92-108. <https://doi.org/10.1016/j.neucom.2022.06.111>
- Elyasi, N., & Hosseini Moghadam, M. (2022). Classification of skin lesions by Tda alongside Xception neural network. *Journal of AI and Data Mining*, 10(3), 333-344. <https://doi.org/10.22044/jadm.2022.10948.2239>
- Fatimah, S. H., Djamal, E. C., Ilyas, R., & Renaldi, F. (2019). Personality Features Identification from Handwriting Using Convolutional Neural Networks. *2019 4th International Conference on Information Technology, Information Systems and Electrical Engineering (ICITISEE)*, 119-124. <https://doi.org/10.1109/ICITISEE48480.2019.9003855>
- Fuentes, C. T., Mostofsky, S. H., & Bastian, A. J. (2009). Children with autism show specific handwriting impairments. *Neurology*, 73, 1532–1537. <https://doi.org/10.1212/WNL.0b013e3181c0d48c>
- Hodges, H., Fealko, C., & Soares, N. (2020). Autism spectrum disorder: Definition, epidemiology, causes, and clinical evaluation. *Translational Pediatrics*, 9(8), S55–S65. <https://doi.org/10.21037/tp.2019.09.09>
- Hyman, S. L., Levy, S. E., Myers, S. M., AAP Council on Children with Disabilities, Section on Developmental and Behavioral Pediatrics. (2020). Identification, evaluation, and management of children with autism spectrum disorder. *Pediatrics*, 145(1), e20193447. <https://doi.org/10.1542/peds.2019-3447>
- Kangarani-Farahani, M., Malik, M. A., & Zwicker, J. G. (2024). Motor impairments in children with autism spectrum disorder: A systematic review and meta-analysis. *Journal of Autism and Developmental Disorders*, 54, 1977–1997. <https://doi.org/10.1007/s10803-023-05948-1>
- Li, Z., Liu, F., Yang, W., Peng, S., & Zhou, J. (2021). A survey of convolutional neural networks: analysis, applications, and prospects. *IEEE transactions on neural networks and learning systems*, 33(12), 6999-7019.



<https://doi.org/10.1109/TNNLS.2021.3084827>

- Rabbi, M. F., Hasan, S. M. M., Champa, A. I., dan Zaman, M. A. (2021). A Convolutional Neural Network Model for Early-Stage Detection of Autism Spectrum Disorder. *2021 International Conference on Information and Communication Technology for Sustainable Development (ICICT4SD)*, 110-114. <https://doi.org/10.1109/ICICT4SD50815.2021.9397020>
- Sarıgül, M., Ozyildirim, B. M., & Avci, M. (2019). Differential convolutional neural network. *Neural Networks*, 116, 279-287. <https://doi.org/10.1016/j.neunet.2019.04.025>
- Sherkatghanad, Z., Akhondzadeh, M., Salari, S., Zomorodi-Moghadam, M., Abdar, M., Acharya, U. R., Khosrowabadi, R., dan Salari, V. (2020). Automated Detection of Autism Spectrum Disorder Using a Convolutional Neural Network. *Frontiers in Neuroscience*, 13, 1325. <https://doi.org/10.3389/fnins.2019.01325>
- Taye, M. M. (2023). Theoretical understanding of convolutional neural network: Concepts, architectures, applications, future directions. *Computation*, 11(3), 52. <https://doi.org/10.3390/computation11030052>
- Vivanti, G., Prior, M., Williams, K., & Dissanayake, C. (2014). Predictors of outcomes in autism early intervention: Why don't we know more? *Frontiers in Pediatrics*, 2, 58. <https://doi.org/10.3389/fped.2014.00058>
- Wang, X., Huang, J., Chatzakou, M., Medijainen, K., Toomela, A., Nõmm, S., & Ruzhansky, M. (2024). LSTM-CNN: An Efficient Diagnostic Network for Parkinson's Disease Utilizing Dynamic Handwriting Analysis. *Computer Methods and Programs in Biomedicine*, 247. <https://doi.org/10.1016/j.cmpb.2024.10806>
- Zhao, X., Wang, L., Zhang, Y., et al. (2024). A review of convolutional neural networks in computer vision. *Artificial Intelligence Review*, 57, 99. <https://doi.org/10.1007/s10462-024-10721-6>