

## Sistem Deteksi Dini Autism Spectrum Disorder (ASD) Berbasis Face Recognition Menggunakan Metode Transfer Learning Resnet50

Fadhila Purboningrum<sup>1</sup>, Prof. Dr. Eng. Fitri Utaminingrum, S.T., M.T. <sup>2</sup> Fais Al Huda, S.Kom., M.Kom.<sup>3</sup>

Program Studi Teknik Informatika, Fakultas Ilmu Komputer, Universitas Brawijaya  
Email: <sup>1</sup>fadhilap@student.ub.ac.id, <sup>2</sup>f3\_ningrum@ub.ac.id, <sup>3</sup>faisalhuda@ub.ac.id

### Abstrak

*Autism Spectrum Disorder (ASD)* merupakan suatu kondisi neurologis yang memengaruhi perkembangan anak. ASD memiliki prevalensi global 1 dari 100 anak. Sifat dan gejala ASD yang beragam, seperti kesulitan berkomunikasi, berinteraksi sosial, dan perilaku repetitif menjadi tantangan tersendiri dalam proses diagnosis yang akurat. Karakteristik wajah, diidentifikasi sebagai kunci potensial, menjadi dasar untuk pengembangan sistem deteksi dini menggunakan *face recognition* dan *deep learning*. Penelitian ini mengusulkan sistem deteksi dini ASD dengan memanfaatkan Android dan metode *transfer learning* arsitektur model ResNet50. Sistem ini diimplementasikan dalam bentuk aplikasi Android dengan arsitektur model ResNet50 yang telah dilatih pada *dataset* ImageNet. Penggunaan teknologi Android dipilih untuk aksesibilitas yang lebih luas, dengan 67% penduduk Indonesia sebagai pengguna *smartphone*. Hasil penelitian menunjukkan bahwa nilai *epoch* optimal untuk model adalah 51 dengan nilai *accuracy* sebesar 91%, nilai *precision* 92%, serta nilai *recall* dan *f1-score* masing-masing sebesar 90%. Selain itu, berdasarkan pengujian dan analisis terhadap hasil pengujian didapatkan akurasi luaran aplikasi sebesar 100% pada kelas *autistic* dan 60% pada kelas *non-autistic* dengan rata-rata waktu komputasi 2,827 detik. Saran melibatkan penambahan variasi *dataset*, validasi data dengan pakar, dan optimasi sisi *client* Android untuk meningkatkan kecepatan tampilan hasil. Penelitian ini memberikan landasan untuk mendemonstrasikan potensi *face recognition* dan model *deep learning* dalam membantu proses diagnosis *Autism Spectrum Disorder (ASD)* dengan akurasi tinggi, serta untuk mendukung intervensi dini dan meningkatkan kesadaran terhadap ASD.

**Kata kunci:** Aplikasi Mobile Android, Autism Spectrum Disorder (ASD), Face Recognition, Karakteristik Wajah, ResNet50, Transfer Learning.

### Abstract

*Autism Spectrum Disorder (ASD)* is a neurological condition that affects children's development. ASD has a global prevalence of 1 in 100 children. The diverse nature and symptoms of ASD, such as difficulties in communication, social interaction, and repetitive behaviors, pose a challenge for accurate diagnosis. Facial characteristics, identified as potential keys, form the basis for the development of an early detection system using *face recognition* and *deep learning*. This research proposes an ASD early detection system utilizing Android and the ResNet50 model architecture transfer learning method. The system is implemented in the form of an Android application with ResNet50 model architecture that has been trained on ImageNet dataset. The use of Android technology was chosen for wider accessibility, with 67% of Indonesia's population as smartphone users. The results show that the optimal epoch value for the model is 51 with an accuracy value of 91%, a precision value of 92%, and a recall and f1-score value of 90% each. In addition, based on testing and analysis of the test results, the accuracy of the application output is 100% in the autistic class and 60% in the non-autistic class with an average computing time of 2.827 seconds. Suggestions involve adding dataset variations, validating data with experts, and optimizing the Android client side to increase the speed of displaying results. This research provides a foundation to demonstrate the potential of face recognition and the deep learning model in assisting the diagnosis process of Autism Spectrum Disorder (ASD) with high accuracy, as well as to support early intervention and raise awareness of ASD.

**Keywords:** Android Mobile Application, Autism Spectrum Disorder (ASD), Face Recognition, Face Characteristics, ResNet50, Transfer Learning.

## 1. PENDAHULUAN

Autisme, atau ASD, adalah kondisi neurologis pada anak yang berdampak seumur hidup. Gejalanya, seperti hambatan komunikasi dan sosialisasi, muncul sebelum usia 3 tahun dengan tingkat keparahan yang beragam. Anak-anak ASD cenderung acuh terhadap suara dan menghindari kontak sosial. (Rahayu, 2015).

ASD dapat memengaruhi fungsi kognitif, emosional, dan psikomotorik, serta menimbulkan masalah perkembangan sosial dan komunikasi dewasa. Prevalensi global ASD mencapai 1 dari 100 anak, dengan variasi di berbagai kelompok sosiodemografis (Zeidan et al., 2022). Diagnosis ASD menantang, sementara proses diagnosis yang kompleks dan mahal membatasi akses bagi banyak keluarga. Persepsi orangtua dan dokter spesialis anak sangat penting dalam mendeteksi ASD dini, mengingat dampaknya pada tantangan akademis, hiperaktivitas, agresi, kecemasan, dan depresi jika tidak ditangani dengan baik (Arumugam et al., 2021).

Wajah manusia memiliki karakteristik unik yang dapat digunakan untuk mengidentifikasi identitas, perilaku, emosi, usia, dan jenis kelamin (Ahmed et al., 2022). Beberapa penelitian menunjukkan bahwa karakteristik wajah pada penderita ASD dapat membantu dalam diagnosis, bahkan ada korelasi antara karakteristik wajah dan tingkat keparahan ASD (Musser, 2020). Dengan memanfaatkan karakteristik wajah ini, dapat dikembangkan sistem deteksi dini ASD melalui teknologi computer vision dengan face recognition dan pendekatan deep learning. Sistem ini dapat menyediakan alat diagnostik cepat, murah, dan mudah untuk membantu dalam keputusan pengujian ASD lebih lanjut pada anak, memastikan pertemuan dengan spesialis yang tepat dan mengurangi jumlah anak yang tidak terdiagnosis ASD.

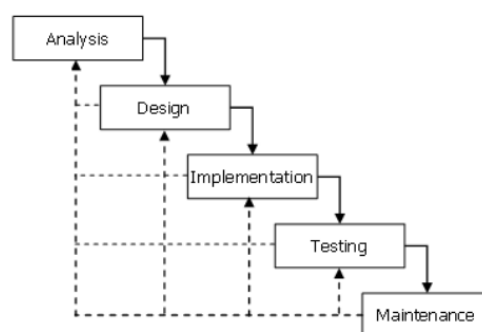
Computer vision, terutama face recognition, telah mengubah cara diagnosis medis dilakukan, termasuk pengenalan patah tulang, gumpalan darah, dan tumor ganas. Pendekatan deep learning, khususnya menggunakan Convolutional Neural Network (CNN), dapat diterapkan melalui pembangunan model atau transfer learning, terutama efektif saat

menghadapi dataset terbatas atau masalah klasifikasi serupa (Nandre et al., 2022).

Penelitian ini mengembangkan aplikasi Android menggunakan deep learning (CNN dan ResNet50) untuk mengklasifikasikan anak-anak sebagai autistic atau non-autistic berdasarkan karakteristik wajah. Pemilihan teknologi didasarkan pada fakta bahwa 67% penduduk Indonesia menggunakan smartphone pada tahun 2022 (Ahdiat, n.d.). ResNet50 dipilih karena kemampuan optimal dalam face recognition (Nandre et al., 2022). Diharapkan, sistem ini, dengan transfer learning ResNet50, dapat memberikan akurasi tinggi dalam mengklasifikasikan anak-anak, mendukung upaya mengurangi jumlah anak tidak terdiagnosis ASD dan meningkatkan kesadaran serta intervensi dini terhadap Autism Spectrum Disorder (ASD).

## 2. METODOLOGI

Penelitian ini merupakan penelitian implementatif pengembangan lanjut, fokus pada penerapan langsung pada perangkat lunak yang terintegrasi dengan model machine learning. Dengan menggunakan metode SDLC jenis Waterfall, penelitian ini mengikuti tahapan perancangan yang dibagi menjadi beberapa langkah untuk menghasilkan sistem aplikasi berbasis Android sebagai solusi. Tahapan metode SDLC ini disesuaikan dengan kebutuhan penelitian dan diilustrasikan dalam diagram alir penelitian. Diagram alir metodologi penelitian diilustrasikan pada Gambar



Gambar 1. Waterfall SDLC Model

### 2.1 Teknik Pengumpulan Data

Teknik pengumpulan data yang digunakan dalam penelitian ini didapat dari Google Drive milik James Paladin (<https://bit.ly/autism-jamespaladin>). Dataset yang didapatkan akan

dilakukan pembagian data pada penelitian ini adalah 80:10:10 untuk *Training:Validation:Test* pada masing-masing kelas *autistic* dan *non-autistic*. Data untuk masing-masing kelas dapat dilihat pada Gambar 2.



Gambar 2. Contoh Dataset

### 3. REKAYASA KEBUTUHAN

#### 3.1 Gambaran Umum Sistem

Dalam penelitian ini, dikembangkan aplikasi menggunakan deep learning, khususnya CNN dengan transfer learning ResNet50, untuk mengklasifikasikan anak-anak sebagai autistic atau non-autistic berdasarkan karakteristik wajah. Aplikasi ini berjalan pada perangkat Android, memanfaatkan model ResNet50 untuk klasifikasi citra wajah anak melalui kamera ponsel. Setelah tahapan preprocessing, model ResNet50 diintegrasikan pada aplikasi Android melalui API Predict untuk pengenalan fitur dan proses klasifikasi. Output sistem mencakup hasil klasifikasi ke dalam dua kelas, yaitu autistic dan non-autistic, beserta nilai confidence prediksi pada citra wajah anak.

#### 3.2 Kebutuhan Fungsional

Kebutuhan fungsional untuk sistem Deteksi Dini Autism Spectrum Disorder (ASD) Berbasis Face Recognition Menggunakan Metode Transfer Learning Resnet50 melibatkan:

1. Sistem dapat menangkap dan mengunggah citra wajah anak melalui kamera *handphone*.
2. Sistem dapat mengunggah citra wajah anak melalui galeri *handphone*.
3. Sistem dapat melakukan klasifikasi citra wajah anak menggunakan model ResNet50.
4. Sistem dapat menampilkan hasil klasifikasi dan nilai confidence kepada pengguna melalui layar aplikasi.

### 3.3 Kebutuhan Non Fungsional

Kebutuhan non fungsional untuk sistem Deteksi Dini Autism Spectrum Disorder (ASD) Berbasis Face Recognition Menggunakan Metode Transfer Learning Resnet50 melibatkan: Berikut adalah kebutuhan non fungsional pada penelitian ini:

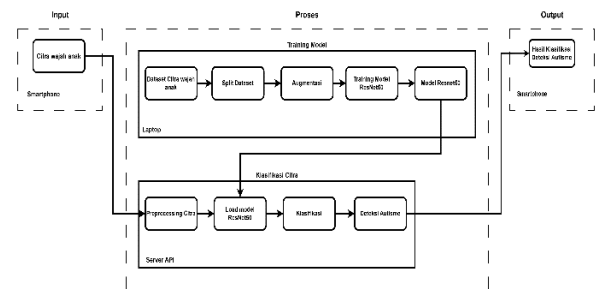
#### 1. Usability

Sistem memiliki tampilan antarmuka aplikasi mudah dipelajari dan digunakan pengguna.

## 4. PERANCANGAN DAN IMPLEMENTASI

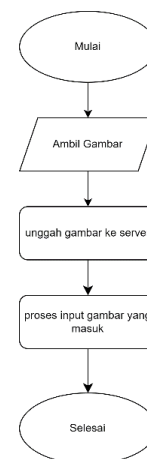
### 4.1 Perancangan Sistem

Perancangan sistem terdiri dari dua bagian: perancangan perangkat keras dan perangkat lunak. Berikut pada Gambar 3. adalah keseluruhan blok diagram sistem.



Gambar 3. Diagram Sistem

### 4.2 Perancangan Perangkat Keras dan Input Sistem

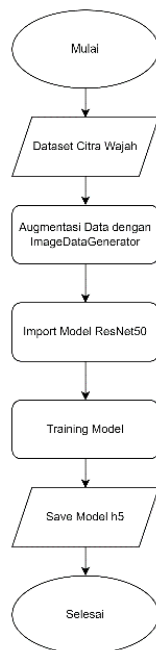


Gambar 4. Flowchart Proses Input Sistem

Bagian input menggunakan kamera atau galeri Android untuk mengakuisisi citra wajah anak dalam format RGB. Proses melibatkan

pelatihan model ResNet50 dengan Tensorflow dan Keras pada laptop, hasilnya disimpan sebagai model ".h5". Model ini digunakan dalam proses klasifikasi dan deteksi autisme pada aplikasi Android. Proses klasifikasi melibatkan pembacaan data citra wajah anak, resize menjadi 224 x 224 piksel, dan klasifikasi menggunakan model ResNet50 melalui API Predict. Hasil klasifikasi menentukan apakah citra masuk dalam kategori autistic atau non-autistic, yang ditampilkan pada layar perangkat Android.

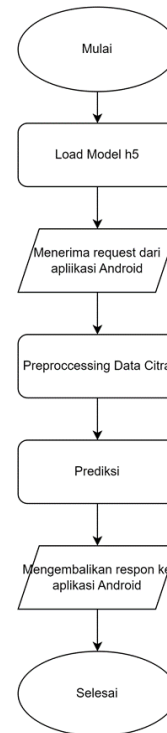
### 4.3 Perancangan Model Klasifikasi Dengan ResNet50



Gambar 5. Flowchart Proses Perancangan Model

Perancangan model klasifikasi dengan menggunakan ResNet50 melibatkan beberapa langkah yang teratur, seperti yang ditunjukkan dalam diagram alir yang disajikan pada Gambar 5.2. Proses ini dimulai dengan memuat dataset, langkah penting dalam pengumpulan data untuk melatih dan menguji model. Setelah itu, dataset ini diaugmentasidiimport ke model ResNet50, yang dirancang dengan arsitektur yang dapat menghasilkan tingkat klasifikasi terbaik.

### 4.4 Perancangan API Predict



Gambar 6. Flowchart Proses Perancangan API Predict

Tahapan perancangan *Application Programming Interface (API)* untuk melakukan proses *predict*, sejalan dengan diagram alir yang telah diilustrasikan pada Gambar 6. proses ini terfokus pada pengembangan sisi *back-end* aplikasi Android untuk menghubungkan komunikasi dan pertukaran data antar program. Dalam penelitian ini, fokus utama API yang dibangun adalah untuk menghubungkan aplikasi Android dengan model pembelajaran mesin yang sudah dirancang sebelumnya. Pada penelitian ini penulis menggunakan bahasa pemrograman Python dengan *framework* Flask. Setelah menentukan bahasa pemrograman dan *framework* yang digunakan, selanjutnya adalah menentukan *endpoint* API antara Android dan *server* yang akan mengatur *request* dan *response* antara aplikasi dan *server*. Setelah *endpoint* pada *server* telah ditentukan, dilakukan proses integrasi model pembelajaran mesin yang telah dalam format file h5 ke dalam *server* API.

### 4.5 Perancangan Output Sistem

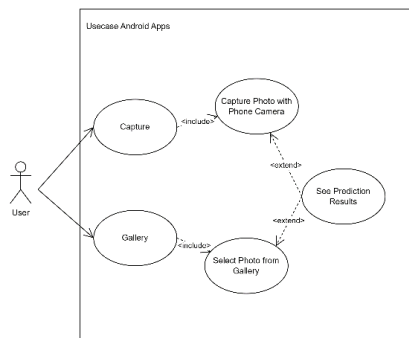


Gambar 7. Flowchart Proses Perancangan Output Sistem

Output yang dihasilkan dari sistem akan ditampilkan melalui tampilan dari *smartphone* pengguna. Pengguna akan melakukan pengambilan atau unggah gambar melalui kamera/galeri, lalu pengguna akan menunggu beberapa saat untuk sistem selesai melakukan proses prediksi berdasarkan dari gambar yang diunggah tersebut.

### 4.5 Perancangan Aplikasi Android

Aplikasi Android mendeteksi autisme melalui analisis citra wajah pengguna yang diambil dengan kamera *smartphone*. Hasil klasifikasi, apakah pengguna autistic atau non-autistic, ditampilkan pada layar dengan tingkat *confidence*.



Gambar 8. Use case Diagram Aplikasi Android

Berikut penjelasan dari *use case diagram* pada Gambar 8.:

- a. *User*: Pengguna aplikasi dengan akses penuh.

- b. *Capture*: Pengambilan gambar melalui kamera *smartphone*.
- c. *Gallery*: Unggah gambar dari galeri *smartphone*.
- d. *Capture photo with phone camera*: Ambil foto langsung dari kamera ponsel.
- e. *Select photo from gallery*: Pilih foto dari galeri untuk diunggah.
- f. *See prediction results*: Lihat tampilan hasil prediksi.

### 4.6 Implementasi Perangkat Keras dan Input Sistem

Implementasi perangkat keras dan input sistem dirancang berdasarkan perancangan yang sudah dibuat sebelumnya. Keseluruhan komponen perangkat keras yang dibutuhkan pada penelitian ini telah terhubung pada *smartphone* yang akan digunakan. Hal ini termasuk dengan kamera yang akan digunakan untuk mengambil gambar wajah pengguna, modul WiFi dan baterai juga sudah tersedia pada perangkat *smartphone*.

### 4.7 Implementasi Model Klasifikasi ResNet50

Implementasi ResNet50 untuk klasifikasi dua kelas (autistic dan non-autistic) melibatkan langkah-langkah awal seperti loading dataset dari Google Drive, dilanjutkan dengan augmentasi data menggunakan ImageDataGenerator. Model ResNet50 diimpor dengan metode transfer learning, lapisan teratasnya disesuaikan, dan lapisan Global Average Pooling ditambahkan. Untuk menangani klasifikasi dua kelas, lapisan Dense, Dropout, dan softmax ditambahkan. Proses pelatihan model melibatkan penggunaan callbacks seperti EarlyStopping, ModelCheckpoint, dan LearningRateScheduler untuk mencegah overfitting dan mengatur learning rate. Setelah pelatihan, model disimpan dalam format H5. Tujuan utama implementasi ini adalah mengadaptasi ResNet50 untuk tugas klasifikasi dua kelas dengan efektif.

### 4.8 Implementasi Server API Predict

Proses implementasi *server API predict* menggunakan *framework* Flask. Model yang telah dilatih sebelumnya diimpor ke dalam Flask. *Application Programming Interface* atau



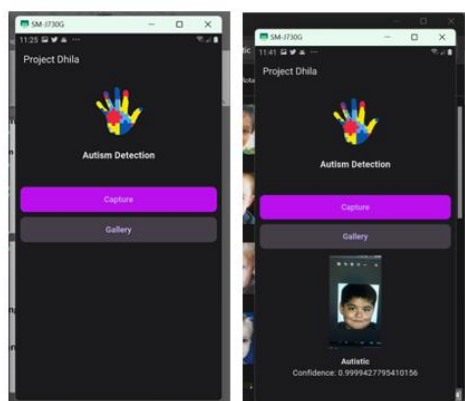
API adalah kumpulan protokol dan alat untuk membangun aplikasi perangkat lunak. Dalam konteks web, API biasanya merujuk pada serangkaian definisi yang memungkinkan perangkat lunak untuk berinteraksi satu sama lain. API dapat digunakan untuk memfasilitasi berbagai jenis interaksi, seperti meminta data, mengirim data, atau melakukan aksi pada perangkat lunak lain. Implementasi API di sini untuk menerima POST *request* dengan gambar sebagai input, menyimpannya secara sementara, dan memprosesnya untuk prediksi menggunakan model. Hasil prediksi, termasuk label dan tingkat kepercayaan (*confidence*), disusun dalam *response* berupa JSON. Setelah selesai, file sementara tersebut dihapus. Flask dijalankan pada alamat dan port tertentu untuk melayani proses *request*. Proses ini memungkinkan pengintegrasian model deteksi autisme ke dalam aplikasi secara efisien melalui API Predict.

#### 4.9 Implementasi Output Sistem

Proses implementasi output sistem merupakan hasil dari klasifikasi citra wajah yang dilakukan oleh pengguna. Proses klasifikasi dilakukan di *server* melalui API *Predict*, hasil dari proses klasifikasi atau *predict* ini akan dikirimkan kembali ke tampilan aplikasi Android pada *smartphone* pengguna.

#### 4.10 Implementasi Aplikasi Android

Proses implementasi aplikasi Android dilakukan dengan menggunakan bahasa pemrograman Dart dengan *framework* pengembangan aplikasi Flutter, menggunakan IDE Visual Studio Code. Aplikasi *mobile* Android ini nantinya akan menjadi sebuah sistem yang dapat digunakan oleh pengguna dalam melakukan deteksi dan klasifikasi autisme pada anak melalui masukkan citra wajah. Tampilan aplikasi dapat dilihat pada Gambar 12.

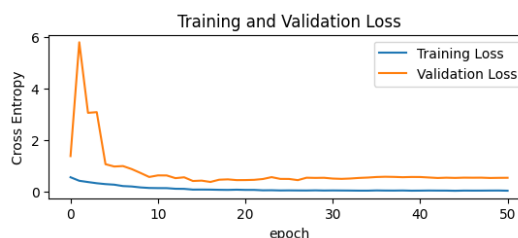


Gambar 12. Tampilan Antarmuka Aplikasi

### 5. PENGUJIAN DAN ANALISIS

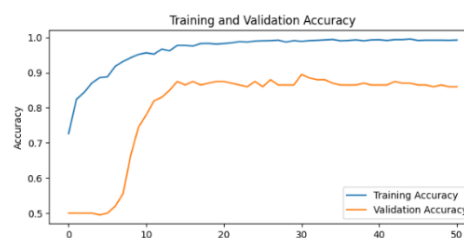
#### 5.1 Pengujian Pengaruh Nilai Epoch Paling Optimal Terhadap Performa Model

Pada pengujian pertama penelitian ini akan diperlihatkan bagaimana nilai *epoch* berpengaruh terhadap nilai *accuracy* dan *loss* dalam proses *training* model ResNet50 menggunakan dataset citra wajah anak yang dibagi menjadi 3 macam data yaitu, data latih (*train*), data validasi (*validation*), dan data uji (*test*) dengan perbandingan 80 % data *train*, 10% data *validation*, dan 10% data *test*. Proses *training* model ini dilakukan dengan 100 *epochs* serta menggunakan mekanisme *callbacks*.



Gambar 12. Grafik Loss

Pada awalnya, nilai *loss* pada data *validation* mencapai 1.3, menunjukkan kesulitan model dalam melakukan generalisasi. Pada *epoch* ke-2, terjadi peningkatan signifikan yang mengindikasikan potensi *overfitting*. Namun, nilai *loss* kemudian menurun dan hampir mencapai *convergence* dengan nilai *loss* pada data *training* pada *epoch* selanjutnya.



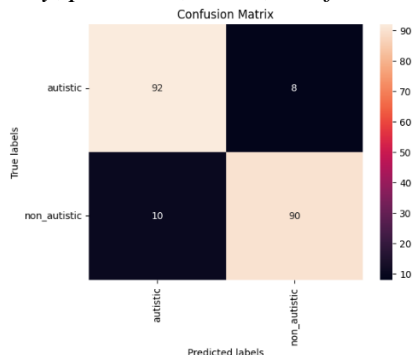
Gambar 13. Grafik Accuracy

Selanjutnya, nilai *accuracy* pada data training awalnya 72%, meningkat seiring berjalannya *epoch*. Di sisi lain, nilai *accuracy* pada data *validation* awalnya 50%, menunjukkan kurangnya generalisasi model. Namun, terjadi peningkatan pada *epoch* ke-10 dan seterusnya.

#### 5.2 Pengujian Confusion Matrix Terhadap Nilai Accuracy, Precision, Recall, dan F1-

**Score**

Pada pengujian kedua ini akan diperlihatkan hasil pengujian *confusion matrix* terhadap nilai *accuracy*, *precision*, *recall*, dan *f1-score*.



Gambar 14. Hasil *Confusion Matrix*

Berdasarkan *confusion matrix* yang diilustrasikan pada Gambar 14. Model dievaluasi menggunakan data *test* dan didapatkan beberapa kesalahan klasifikasi. Pada kelas *autistic*, 92 dari 100 citra wajah berhasil diklasifikasikan dengan benar, sedangkan 8 citra gagal diklasifikasikan. Pada kelas *non-autistic*, 90 dari 100 citra wajah berhasil diklasifikasikan dengan benar, sementara 10 citra gagal diklasifikasikan. Berdasarkan nilai diatas dapat dilihat hasil perhitungan yang dijabarkan pada Tabel 1.

Tabel 1. Hasil Nilai Accuracy, Precision, Recall, F1-Score

Confusion Matrix	Komponen
Accuracy	0,91
Precision	0,92
Recall	0,90
F1-Score	0,90

**5.3 Pengujian Rata-rata Waktu Komputasi Sistem**

Pengujian ini dilakukan untuk mengetahui seberapa cepat sistem aplikasi dapat melakukan deteksi dan klasifikasi citra wajah *autistic* dan *non-autistic*. Hasil perhitungan rata-rata waktu komputasi sistem dijabarkan pada Tabel 2

Tabel 2. Hasil Waktu Komputasi Sistem

Pengujian ke-n	Waktu Komputasi
1	3,22 detik
2	3,08 detik
3	3,93 detik

4	3,23 detik
5	3,10 detik
6	2,50 detik
7	3,48 detik
8	2,98 detik
9	1,45 detik
10	1,30 detik
<b>Rata-rata</b>	<b>2,827 detik</b>

Berdasarkan Tabel 2. Sistem dapat melakukan klasifikasi *autistic* dan *non-autistic* selama rata-rata 7,332 detik.

**5.4 Pengujian Tingkat Akurasi Sistem**

.Pengujian ini bertujuan untuk mengetahui seberapa baik dan akurat sistem aplikasi melakukan klasifikasi *autistic* dan *non-autistic*.

Tabel 3. Hasil Akurasi Sistem

Kelas	Akurasi
Autistic	100%
Non-Autistic	60%

Berdasarkan Tabel 3. dapat dilihat bahwa sistem dapat mampu melakukan proses klasifikasi dengan akurasi 100% pada proses klasifikasi ke dalam kelas *autistic*. Sedangkan, pada proses klasifikasi ke dalam kelas *non-autistic* sistem menghasilkan akurasi yang rendah pada 60%. Sehingga rata-rata nilai akurasi sistem adalah sebesar 80%.

**5.5 Pengujian Usability Sistem Aplikasi**

Pengujian ini bertujuan untuk mengetahui seberapa baik nilai *usability* dari antarmuka sistem aplikasi menurut pengguna. Pengujian ini dilakukan dengan bantuan dari 6 responden dan menggunakan metode SUS (*System Usability Scale*) berdasarkan penelitian dari (Brooke, 1995). Hasil dari pengujian dapat dilihat secara keseluruhan pada Tabel 4.

Tabel 4. Hasil System Usability Scale

Responden ke-n	Skor SUS
1	90
2	90
3	57,5
4	87,5
5	97,5
6	90
<b>Total Skor SUS</b>	<b>180</b>
<b>Rata-rata Skor SUS</b>	<b>85,41</b>

Dapat dilihat pada salah satu responden yaitu responden ke-3, nilai yang diberikan sebesar 57,9 yang cukup rendah dibandingkan dengan responden lainnya. Hal ini disebabkan, pengguna tersebut awalnya ragu menggunakan aplikasi dan memiliki keterbatasan paket data. Namun setelah menggunakan aplikasi, pengguna tidak merasa kebingungan untuk mendeteksi autisme pada anak melalui foto. Pengguna sangat terbantu dan puas dengan aplikasi ini, meskipun membutuhkan sedikit bimbingan untuk memahami fiturnya.

Selanjutnya, berdasarkan riset yang telah dilakukan sebelumnya, skor SUS dapat dikategorikan ke dalam beberapa kategori: *best imaginable* (91 – 100), *excellent* (71 – 90), *good* (51 – 70), *ok* (40 – 50), *poor* (21 – 30), *worst imaginable* (0 – 20). Sehingga, berdasarkan nilai rata-rata skor SUS yang didapatkan sebelumnya yaitu 85,41 masuk ke dalam kategori *excellent*.

## 5. Kesimpulan

Berdasarkan analisis hasil pengujian penelitian, kesimpulan dibuat mengenai masalah yang dirumuskan sebelumnya. Kesimpulan yang dapat disimpulkan dari penelitian ini secara keseluruhan adalah sebagai berikut.

1. Dari hasil pengujian penelitian pertama, ditemukan bahwa nilai epoch memengaruhi akurasi dan loss. Model menunjukkan kinerja baik pada epoch ke-1, dan melalui epoch berikutnya, model semakin memperbaiki diri dengan nilai loss yang semakin rendah. Penggunaan callbacks melalui early stopping menunjukkan bahwa pelatihan setelah epoch ke-51 tidak signifikan memengaruhi akurasi dan loss terhadap data baru, menjadikan epoch ke-51 sebagai epoch optimal.
2. Dari hasil pengujian penelitian kedua, terlihat bahwa confusion matrix mencapai performa terbaik pada epoch ke-51. Nilai *accuracy* 0,91, *precision* 0,92, *recall* 0,90, dan *f1-score* 0,90, menunjukkan kinerja yang baik pada klasifikasi anak-anak autistic dan non-autistic.
3. Dari hasil pengujian penelitian ketiga, sistem memerlukan rata-rata waktu komputasi sekitar 2,827 detik untuk melakukan klasifikasi autistic dan non-

autistic melalui citra wajah anak menggunakan model ResNet50. Waktu komputasi tercepat adalah 1,3 detik, sedangkan waktu terlamban mencapai 2,98 detik.

4. Dari hasil pengujian penelitian keempat, sistem aplikasi mampu mendeteksi dan mengklasifikasikan dengan akurasi 100% pada kelas autistic dan 60% pada kelas non-autistic. Rata-rata akurasi sistem adalah 80% dari total 10 kali pengujian, dengan hasil sesuai pada 8 data citra wajah.

## 6. DAFTAR PUSTAKA

- Ahdiat, A. (n.d.). *67% Penduduk Indonesia Punya Handphone pada 2022, Ini Sebarannya*. Retrieved November 28, 2023, from <https://databoks.katadata.co.id/datapublish/2023/03/08/67-penduduk-indonesia-punya-handphone-pada-2022-ini-sebarannya>
- Ahmed, Z. A. T., Aldhyani, T. H. H., Jadhav, M. E., Alzahrani, M. Y., Alzahrani, M. E., Althobaiti, M. M., Alassery, F., Alshafut, A., Alzahrani, N. M., & Al-madani, A. M. (2022). Facial Features Detection System To Identify Children With Autism Spectrum Disorder: Deep Learning Models. *Computational and Mathematical Methods in Medicine*, 2022, 1–9. <https://doi.org/10.1155/2022/3941049>
- Arumugam, S. R., Balakrishna, R., Khilar, R., Manoj, O., & Shylaja, C. S. (2021). Prediction of Autism Spectrum Disorder in Children using Face Recognition. *2021 2nd International Conference on Smart Electronics and Communication (ICOSEC)*, 1246–1250. <https://doi.org/10.1109/ICOSEC51865.2021.9591679>
- Brooke, J. (1995). SUS: A quick and dirty usability scale. *Usability Eval. Ind.*, 189.
- Musser, M. (2020, April 13). *Detecting Autism Spectrum Disorder in Children With Computer Vision*. Towards Data Science. <https://towardsdatascience.com/detecting-autism-spectrum-disorder-in-children-with-computer-vision-8abd7fc9b40a>
- Nandre, J., Rai, S., & Kanawade, B. R. (2022). Comparative Analysis of Transfer Learning CNN for Face Recognition. *2022 2nd International Conference on*



*Intelligent Technologies (CONIT)*, 1–6.  
<https://doi.org/10.1109/CONIT55038.2022.9847946>

Rahayu, S. M. (2015). Deteksi dan Intervensi Dini Pada Anak Autis. *Jurnal Pendidikan Anak*, 3(1).

<https://doi.org/10.21831/jpa.v3i1.2900>

Zeidan, J., Fombonne, E., Scolah, J., Ibrahim, A., Durkin, M. S., Saxena, S., Yusuf, A., Shih, A., & Elsabbagh, M. (2022). Global prevalence of autism: A systematic review update. *Autism Research*, 15(5), 778–790.  
<https://doi.org/10.1002/aur.2696>