



**SISTEM DETEKSI SIMBOL PADA SIBI (SISTEM ISYARAT BAHASA
INDONESIA) MENGGUNAKAN *MEDIAPIPE* DAN *RESNET-50***



TUGAS AKHIR

**Program Studi
S1 TEKNIK KOMPUTER**

Oleh:

Nofal Anam

18410200050

FAKULTAS TEKNOLOGI DAN INFORMATIKA

UNIVERSITAS DINAMIKA

2022

**SISTEM DETEKSI SIMBOL PADA SIBI (SISTEM ISYARAT
BAHASA INDONEISA) MENGGUNAKAN MEDIAPIPE DAN
RESNET-50**

TUGAS AKHIR

Diajukan sebagai salah satu syarat untuk menyelesaikan

Program Sarjana Teknik



UNIVERSITAS
Dinamika

Disusun Oleh :

Nama	: Nofal Anam
NIM	: 18410200050
Program Studi	: S1 Teknik Komputer

**FAKULTAS TEKNOLOGI DAN INFORMATIKA
UNIVERSITAS DINAMIKA**

2022

TUGAS AKHIR

SISTEM DETEKSI SIMBOL PADA SIBI (SISTEM ISYARAT BAHASA INDONESIA) MENGGUNAKAN MEDIAPIPE DAN RESNET-50

Dipersiapkan dan disusun oleh

Nofal Anam

NIM: 18410200050

Telah diperiksa, diuji dan disetujui oleh Dewan Penguji

Pada: Januari 2022

Susunan Dewan Pembahas

Pembimbing:

I. Heri Pratikno, M.T., MTCNA., MTCRE.

NIDN: 0716117302

II. Yosefine Triwidyastuti, M.T.

NIDN 0729038504




Digitally signed by
Universitas Dinamika
Date: 2022.01.13 12:02:56
+07'00'

Digitally signed by
Universitas Dinamika
Date: 2022.01.13
12:24:09 +07'00'

Pembahas:

Pauladie Susanto, S.Kom., M.T.

NIDN: 0729047501



cn=Pauladie Susanto, o=FTI
Undika, ou=Prodi S1 TK,
email=pauladie@dinamika.a
c.id, c=ID
2022.01.13 13:28:56 +07'00'

Tugas Akhir ini telah diterima sebagai salah satu persyaratan
untuk memperoleh gelar sarjana



Digitally signed by
Universitas Dinamika
Date: 2022.01.21 10:13:05
+07'00'

Tri Sagirani, S.Kom., M.MT.

NIDN: 0731017601

Dekan Fakultas Teknologi dan Informatika
UNIVERSITAS DINAMIKA



*"I don't believe that we have to follow a passion, but we have to follow our
obsession" – Sabda PS*

UNIVERSITAS
Dinamika



UNIVERSITAS
Dinamika

Dipersembahkan kepada siapapun yang ingin belajar tentang kecerdasan buatan.

*Tugas akhir yang diharapkan dapat membantu untuk pengembangan teknologi ke
arah yang lebih baik bagi manusia.*

SURAT PERNYATAAN
PERSETUJUAN PUBLIKASI DAN KEASLIAN KARYA ILMIAH

Sebagai mahasiswa Universitas Dinamika, saya:

Nama : Nofal Anam
NIM : 18.41020.0050
Program Studi : S1 Teknik Komputer
Fakultas : Fakultas Teknologi dan Informatika
Jenis Karya : Laporan Tugas Akhir
Judul Karya : SISTEM DETEKSI SIMBOL PADA SIBI (SISTEM
ISYARAT BAHASA INDONESIA) MENGGUNAKAN
MEDIAPIPE DAN RESNET-50

Menyatakan dengan sesungguhnya bahwa:

1. Demi pengembangan Ilmu Pengetahuan, Teknologi dan Seni, saya menyetujui memberikan kepada Universitas Dinamika Hak Bebas Royalti Non-Eksklusif (*Non-Exclusive Royalti Free Right*) atas seluruh isi/ sebagian karya ilmiah saya tersebut di atas untuk disimpan, dialihmediakan dan dikelola dalam bentuk pangkalan data (*database*) untuk selanjutnya didistribusikan atau dipublikasikan demi kepentingan akademis dengan tetap mencantumkan nama saya sebagai penulis atau sebagai pemilik pencipta dan Hak Cipta
2. Karya tersebut di atas adalah karya asli saya, bukan plagiat baik sebagian maupun keseluruhan. Kutipan, karya atau pendapat orang lain yang ada dalam karya ilmiah ini adalah semata hanya rujukan yang dicantumkan dalam Daftar Pustaka saya
3. Apabila dikemudian hari ditemukan dan terbukti terdapat tindakan plagiat pada karya ilmiah ini, maka saya bersedia untuk menerima pencabutan terhadap gelar kesarjanaan yang telah diberikan kepada saya.

Demikian surat pernyataan ini saya buat dengan sebenarnya.

Surabaya, 18 Desember 2021

Yang menyatakan



Nofal Anam

NIM: 18.41020.0050

ABSTRAK

Bahasa isyarat menjadi alternatif yang sering digunakan oleh penyandang tuna rungu dan wicara untuk berkomunikasi. Namun cara tersebut sulit digunakan oleh mereka untuk berkomunikasi dengan orang normal, terutama yang belum belajar Bahasa isyarat. Sementara tetap perlu adanya interaksi dan komunikasi antara penyandang tuna rungu, wicara dengan orang normal. Penggunaan teknologi *deep learning* telah banyak diterapkan dalam berbagai hal, termasuk Bahasa isyarat. Pengembangan *deep learning* dalam ranah Bahasa isyarat masih berlanjut, termasuk dalam penelitian ini. Penulis melakukan penelitian terhadap metode *Pre-trained Convolutional Neural Network* model ResNet-50 untuk klasifikasi simbol Sistem Isyarat Bahasa Indonesia (SIBI). Penulis juga menggunakan teknologi *Mediapipe Holistic* yang dikembangkan oleh Google untuk mendeteksi posisi pose dan tangan dari setiap simbol isyarat, sehingga lebih mudah membedakan gerakan simbol. Telah dibuat 1200 citra gambar yang terdiri dari 6 *class* sebagai *dataset* dengan pembagian 960 untuk *training*, 180 validasi, dan 60 *testing*. Hasil dari proses *training* menggunakan ResNet-50 mendapatkan akurasi *training* 33% dan akurasi *validation* 40%, dengan adanya penambahan *Mediapipe* telah meningkatkan nilai akurasi *training* menjadi 88% dan akurasi *validation* 87%. Sehingga telah terjadi peningkatan nilai akurasi *training* sebesar 55% dan akurasi *validation* sebesar 47%. Perhitungan *performance metrics* menggunakan *Mediapipe* dan ResNet-50 telah menghasilkan nilai performa rata-rata keenam simbol sebagai berikut: *precision* 86%, *recall* 86%, dan *F1-score* 85%. Sedangkan akurasi *testing* menggunakan *Mediapipe* dan ResNet-50 mempunyai nilai persentase setiap simbol sebagai berikut: ‘Aku’ 90%, ‘Cinta’ 100%, ‘Dia’ 100%, ‘Kamu’ 50%, ‘Maaf’ 100%, dan ‘Sedih’ 80%.

Kata Kunci: Bahasa Isyarat, *Deep Learning*, ResNet50, *Mediapipe*, *Pre-trained Convolutional Neural Network*.

KATA PENGANTAR

Puji syukur saya panjatkan kepada Tuhan Yang Maha Esa atas segala rahmat yang telah diberikan - Nya, sehingga penulis dapat menyelesaikan Laporan Tugas Akhir yang berjudul “Sistem Deteksi Simbol Pada SIBI (Sistem Isyarat Bahasa Indonesia) Mengguankan Mediapipe dan ResNet-50”.

Dalam usaha menyelesaikan penulisan Laporan Tugas Akhir ini penulis banyak mendapat bantuan dari berbagai pihak. Oleh karena itu penulis mengucapkan terima kasih kepada:

1. Allah SWT, karena dengan rahmatnya dan hidayahnya penulis dapat menyelesaikan Laporan Tugas Akhir ini.
2. Orang Tua dan Seluruh Keluarga penulis tercinta yang telah memberikan dorongan dan bantuan baik moral maupun materi sehingga penulis dapat menempuh dan menyelesaikan laporan tugas akhir ini.
3. Ibu Tri Sagirani, S.Kom., M.MT. selaku Dekan Fakultas Teknologi dan Informatika (FTI) Universitas Dinamika.
4. Bapak Pauladie Susanto, S.Kom., M.T. selaku Ketua Program Studi S1 Teknik Komputer terima kasih atas bimbingan yang diberikan dan kesempatannya serta tuntunan baik itu materi secara tertulis maupun lisan sehingga penulis dapat menyelesaikan Tugas Akhir.
5. Bapak Pauladie Susanto, S.Kom., M.T. selaku Dosen Pembahas atas saran dan masukannya sehingga penulis dapat menyelesaikan laporan Tugas Akhir ini.
6. Bapak Heri Pratikno, M.T., MTCNA., MTCRE. selaku dosen pembimbing yang telah memberikan masukan dan solusi agar tugas akhir ini dapat selesai dan menjadi lebih baik lagi.
7. Ibu Yosefine Triwidyastuti, M.T. selaku dosen pembimbing yang banyak memberikan dukungan penuh berupa motivasi, saran, dan wawasan bagi penulis selama pelaksanaan tugas akhir dan pembuatan laporan Tugas Akhir.
8. Teman - teman seperjuangan S1 Teknik Komputer angkatan 2018 Universitas Dinamika, yang telah membantu penulis dalam berupa saran selama proses pengerjaan.

9. Sally Febriana Monica, yang selalu memberikan semangat dan dukungan kepada penulis untuk dapat segera menyelesaikan laporan tugas akhir ini serta menemani penulis selama proses pengerjaannya.

Penulis berharap semoga laporan ini dapat berguna dan bermanfaat untuk menambah wawasan bagi pembacanya. Penulis juga menyadari dalam penulisan laporan ini banyak terdapat kekurangan. Oleh karena itu penulis sangat mengharapkan saran dan kritik untuk memperbaiki kekurangan dan berusaha untuk lebih baik lagi.

Surabaya, 12 Januari 2022

Penulis



UNIVERSITAS
Dinamika

DAFTAR ISI

	Halaman
ABSTRAK.....	vii
KATA PENGANTAR.....	viii
DAFTAR ISI.....	x
DAFTAR GAMBAR.....	xii
DAFTAR TABEL	xiii
DAFTAR LAMPIRAN.....	xiv
BAB I PENDAHULUAN.....	1
1.1 Latar Belakang.....	1
1.2 Rumusan Masalah.....	2
1.3 Batasan Masalah	3
1.4 Tujuan	3
1.5 Manfaat Penelitian	3
BAB II TINJAUAN PUSTAKA	5
2.1 Tuna Rungu dan Tuna Wicara	5
2.2 SIBI (Sistem Isyarat Bahasa Indonesia)	5
2.3 Deep Learning.....	6
2.4 Convolutional Neural Network (CNN).....	7
2.4.1 Convolution Layer.....	8
2.4.2 Operasi Pooling	9
2.4.3 Flattening.....	10
2.4.4 Fully Connected.....	10
2.5 Mediapipe	10
2.5.1 Mediapipe Holistic	11
2.6 <i>ResNet-50</i>	12
2.7 Menghitung Performa	14
2.7.1 Akurasi.....	15
2.7.2 Presisi.....	15
2.7.3 <i>Recall</i>	15

2.7.4 <i>F1 Score</i>	15
BAB III METODOLOGI PENELITIAN	16
3.1 <i>Dataset</i>	16
3.2 Deteksi <i>Landmark</i> Mediapipe.....	17
3.3 Proses Training ResNet-50	20
3.4 Proses <i>Testing</i>	21
BAB IV HASIL DAN PEMBAHASAN	22
4.1 Hasil Proses <i>Training</i> Model	22
4.2 Perbandingan Performa.....	24
4.3 Prediksi	26
BAB V PENUTUP.....	31
5.1 Kesimpulan	31
5.2 Saran	32
DAFTAR PUSTAKA.....	33
LAMPIRAN.....	35
BIODATA PENULIS.....	47



UNIVERSITAS
Dinamika

DAFTAR GAMBAR

	Halaman
Gambar 2. 1 Sistem Isyarat Bahasa Indonesia	6
Gambar 2. 2 Arsitektur Deep Learning.....	7
Gambar 2. 3 Arsitektur CNN	8
Gambar 2. 4 Proses Konvolusi.....	9
Gambar 2. 5 Proses Flattening	10
Gambar 2. 6 Mediapipe Holistic Pipeline	11
Gambar 2. 7 Blok Residual Learning.....	12
Gambar 2. 8 Blok Residual Learning pada ResNet-50.....	13
Gambar 2. 9 Arsitektur ResNet-50.....	13
Gambar 2. 10 Confusion Matrix.....	16
Gambar 3. 1 Contoh Dataset Bahasa Isyarat.....	16
Gambar 3. 2 Flowchart Deteksi dan Ekstraksi Landmark	18
Gambar 3. 3 Citra Asli (kiri) Citra Dengan Landmark (kanan).....	19
Gambar 3. 4 Proses Mengolah Landmark Pada Background Putih.....	19
Gambar 3. 5 Folder Hasil Ekstraksi Landmark.....	19
Gambar 3. 6 Flowchart Proses Training	20
Gambar 3. 7 Flowchart Proses Testing	21
Gambar 4. 1 Grafik Akurasi Dan Loss Model ResNet-50.....	23
Gambar 4. 2 Grafik Akurasi Dan Loss Mediapipe Dan Model ResNet-50	24
Gambar 4. 3 Confusion Matrix Model ResNet-50.....	25
Gambar 4. 4 Confusion Matrix Mediapipe Dan Model ResNet-50.....	25
Gambar 4. 5 Interface Prediksi SIBI.....	27
Gambar L21.1 Hasil turnitin halaman 1.....	46

DAFTAR TABEL

	Halaman
Tabel 3. 1 Detail Rangkuman Dataset.....	17
Tabel 4. 1 Hasil Training Model ResNet-50.....	25
Tabel 4. 2 Hasil Training Mediapipe dan Model ResNet-50.....	25
Tabel 4. 3 Hasil Performa Model ResNet-50.....	29
Tabel 4. 4 Hasil Performa Mediapipe dan Model ResNet-50.....	29
Tabel 4. 5 Hasil Prediksi Model ResNet-50.....	31
Tabel 4. 6 Hasil Prediksi Mediapipe dan Model ResNet-50.....	32



UNIVERSITAS
Dinamika

DAFTAR LAMPIRAN

Halaman

Lampiran 1 Program Deteksi dan Ekstraksi Landmark	35
Lampiran 2 Training Model ResNet-50 dan ResNet-50 Dengan Mediapipe	38
Lampiran 3 Program Testing SIBI ResNet-50 dan Mediapipe	43
Lampiran 4 Program Testing SIBI ResNet-50	45
Lampiran 5 Hasil Turnitin	46



UNIVERSITAS
Dinamika

BAB I

PENDAHULUAN

1.1 Latar Belakang

Bahasa isyarat menjadi salah satu alternatif sebagai media komunikasi bagi penyandang tuna rungu dan tuna wicara. Bahasa isyarat adalah suatu ungkapan yang dilakukan dengan menggunakan gerakan-gerakan badan dan mimik muka sebagai simbol dari makna Bahasa. Menurut data penyandang disabilitas di Indonesia yang dihimpun oleh kemensos (Kemensos, 2021), jumlah penyandang tuna rungu 13.800 dan tuna wicara 5.580 orang.

Dalam perkembangannya di Indonesia, Bahasa isyarat tergolong menjadi 2 jenis, yakni SIBI (Sistem Isyarat Bahasa Indonesia) dan BISINDO (Bahasa Isyarat Indonesia). Sistem Isyarat Bahasa Indonesia (SIBI) merupakan bentuk Bahasa lisan yang diubah kedalam Bahasa isyarat, dengan beberapa kosakata Bahasa isyarat tersebut diambil dari Bahasa Isyarat Amerika (*America Sign Language*). Hal tersebut bertujuan untuk memudahkan jika harus berkomunikasi dalam Bahasa asing.

Interaksi dan komunikasi tidak hanya terjadi antar penyandang tuna rungu atau tuna wicara, namun juga dengan orang normal. Dengan demikian hubungan sosial diantara mereka akan terjalin dengan baik. Namun terdapat kesenjangan komunikasi bagi penyandang tuna rungu atau tuna wicara dengan orang normal. Pemahaman makna dan persepsi diantara keduanya memang membutuhkan pemahaman yang rumit, terutama bagi orang normal yang belum pernah belajar Bahasa isyarat. Dewasa ini, teknologi *deep learning* telah banyak dikembangkan untuk mengatasi permasalahan ini.

Deep learning secara singkat adalah sebuah model yang bisa mempelajari dengan sendirinya sebuah metode komputasi. Perkembangan *deep learning* dalam permasalahan Bahasa isyarat sedang mengalami perkembangan. Pada penelitian sebelumnya yang telah dilakukan oleh (Putra, 2021), membuat rancangan sistem deteksi simbol SIBI (Sistem Isyarat Bahasa Indonesia) menggunakan

Convolutional Neural Network. Pada penelitian tersebut dibuat *dataset* yang dibatasi sejumlah 660 citra image dan terbagi ke dalam 6 simbol. Hasil dari penelitian tersebut didapatkan akurasi *Training* sebesar 91% dan validasi 90%. *Dataset* juga perlu dilakukan beberapa proses *Image Pre-processing* sebelum dilakukan *Training*.

Sebagai upaya untuk meningkatkan akurasi dari pengujian sebelumnya, pada tugas akhir ini penulis mengusulkan pembelajaran simbol SIBI menggunakan *Pre-trained Convolutional Neural Network model ResNet50*. Model ini dipilih karena mengacu pada hasil penelitian (Grandhi, et al., 2021). Pada penelitian tersebut dilakukan perbandingan hasil training *dataset* huruf ASL (*American Sign Language*) menggunakan beberapa model yaitu *CNN baseline*, *VGG16 Scratch*, *VGG16 Pretrained*, *InceptionNet*, dan *ResNet-50*. Didapatkan bahwa model *ResNet-50* memberikan hasil terbaik dengan akurasi tes mencapai 99.88%.

Selain itu penulis juga mengusulkan menggunakan teknologi Mediapipe yang dikembangkan oleh *Google*. Berdasarkan penelitian (Halder & Tayade, 2021) Mediapipe dapat digunakan secara efisien sebagai alat untuk mendeteksi Gerakan tangan yang rumit secara tepat. Mediapipe digunakan karena teknik pemrosesan gambar yang terlalu kompleks dengan data komputasi dan konsumsi waktu yang tinggi. *Training* dan *testing* dengan beberapa macam Bahasa isyarat menunjukkan bahwa Mediapipe dapat diadaptasi ke Bahasa isyarat lainnya.

1.2 Rumusan Masalah

Berdasarkan latar belakang diatas, maka dapat dirumuskan masalah pada Tugas Akhir ini sebagai berikut:

1. Berapa besar performa sistem deteksi simbol pada Sistem Isyarat Bahasa Indonesia menggunakan *Pre-trained Convolutional Neural Network model ResNet-50*?
2. Berapa besar performa setelah menambahkan teknologi *Mediapipe* pada sistem?

1.3 Batasan Masalah

Dalam pembuatan Tugas Akhir ini, pembahasan masalah dibatasi pada beberapa hal berikut:

1. Pembatasan hanya 6 simbol Bahasa isyarat yang menjadi *dataset*.
2. Pengujian yang dilakukan adalah pengujian akurasi deteksi.
3. Pengujian tidak dilakukan dengan *Real-time* hanya dengan citra yang di *Load*.
4. Mendeteksi *landmark* pada tubuh dan tangan menggunakan *Mediapipe*.

1.4 Tujuan

Berdasarkan latar belakang dan rumusan masalah diatas, mendapatkan tujuan pada tugas akhir ini sebagai berikut:

1. Mengetahui performa yang didapatkan pada sistem deteksi simbol Siste Isyarat Bahasa Indonesia menggunakan *Pre-trained Convolutional Neural Network* model *ResNet-50*.
2. Mengetahui peningkatan performa setelah menambahkan teknologi *Mediapipe* pada sistem.

1.5 Manfaat Penelitian

Adapun dari Tugas Akhir ini dapat diperoleh manfaat sebagai berikut:

1. Bagi penulis yaitu untuk menambah pengetahuan dan penerapan mengenai deteksi simbol pada Sistem Isyarat Bahasa Indoensia menggunakan *Pre-trained Convolutional Neural Network* model *ResNet-50*.
2. Bagi mahasiswa yaitu menjadi referensi bagi mahasiswa yang akan melakukan penelitian menggunakan *Pre-trained Convolutional Neural Network* model *ResNet-50*.
3. Penelitian ini memberikan kontribusi berupa sistem deteksi yang bisa digunakan pada Sistem Isyarat Bahasa Indonesia.

4. Terbentuknya *Dataset* Sistem Isyarat Bahasa Indonesia sebanyak 1200 citra gambar.



UNIVERSITAS
Dinamika

BAB II

TINJAUAN PUSTAKA

2.1 Tuna Rungu dan Tuna Wicara

Tunarungu merupakan istilah yang digunakan untuk menggambarkan kondisi dimana seseorang yang mengalami kekurangan atau kehilangan kemampuan mendengar baik sebagian atau seluruhnya (Setyawan, 2019). Sedangkan tunawicara adalah seseorang yang mengalami kesulitan dalam mengungkapkan pikiran melalui Bahasa verbal, sehingga sulit bahkan tidak dimengerti orang lain. Sehingga dalam berkomunikasi harus menggunakan komunikasi nonverbal (Widiarti & Rahayu, 2020).

Bahasa isyarat menjadi salah satu alternatif sebagai media komunikasi bagi penyandang tunarungu dan tunawicara. Bahasa isyarat adalah Bahasa yang dilakukan dengan menggunakan gerakan-gerakan badan dan mimik muka sebagai simbol dari makna Bahasa. Di Indonesia, Bahasa isyarat yang berkembang dibagi menjadi 2 jenis, yakni SIBI (Sistem Isyarat Bahasa Indonesia) dan BISINDO (Bahasa Isyarat Indonesia). Seperti Bahasa lisan pada umumnya, Bahasa isyarat akan berbeda dengan Bahasa isyarat di daerah lain. Dengan pertimbangan Bahasa isyarat yang banyak dan beragam, akhirnya tercipta sistem hasil rekayasa dari orang normal untuk berkomunikasi dengan para tunarungu yang disebut dengan Sistem Isyarat Bahasa Indonesia.

2.2 SIBI (Sistem Isyarat Bahasa Indonesia)

Sistem Isyarat Bahasa Indonesia (SIBI) merupakan bentuk Bahasa lisan yang diubah kedalam Bahasa isyarat, dengan beberapa kosakata Bahasa isyarat tersebut diambil dari Bahasa Isyarat Amerika (*America Sign Language*). Hal tersebut bertujuan untuk memudahkan jika harus berkomunikasi dalam Bahasa asing.

Sistem Isyarat Bahasa Indonesia (SIBI) terdiri dari tatanan sistematis tentang seperangkat isyarat gerak tubuh, gerak tangan, dan gerak bibir yang melambangkan kosakata Bahasa Indonesia. Dalam perkembangannya Sistem Isyarat Bahasa

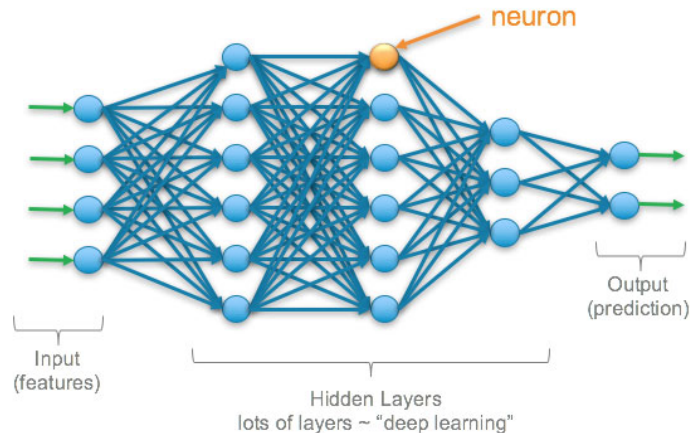
Indonesia (SIBI) masih menggunakan dari Bahasa isyarat luar negeri yang lalu ditambah dengan isyarat-isyarat lokal Indonesia dan buatan (Nasir, et al., 2021)



Gambar 2.1 Sistem isyarat bahasa Indonesia
(Sumber: Maharsi, 2018)

2.3 Deep Learning

Sebelum membahas mengenai *deep learning* terlebih dahulu kita perlu ketahui definisi dari *machine learning* dan *deep learning* menurut (Cholissodin, et al., 2019). *Machine learning* adalah kecerdasan buatan yang fokus belajar dari data dan pengembangan sistem yang mampu belajar secara “mandiri” tanpa harus berulang kali diprogram manusia. Sedangkan *deep learning* adalah salah satu cabang dari *machine learning* yang terdiri dari algoritma pemodelan abstraksi tingkat tinggi pada data menggunakan sekumpulan fungsi transformasi *non-linear* yang ditata berlapis-lapis dan mendalam.



Gambar 2. 2 *Arsitektur Deep Learning*
(Sumber: Rinaghan, 2018)

Deep learning merupakan pembelajaran yang berbasis pada fitur berbentuk hirarki yang dapat diskalakan dan disesuaikan dengan kasus yang diproses (Cholissodin, et al., 2019). Berdasarkan hasil penelitian (Yifei, 2017), metode *deep learning* memiliki beberapa parameter yaitu:

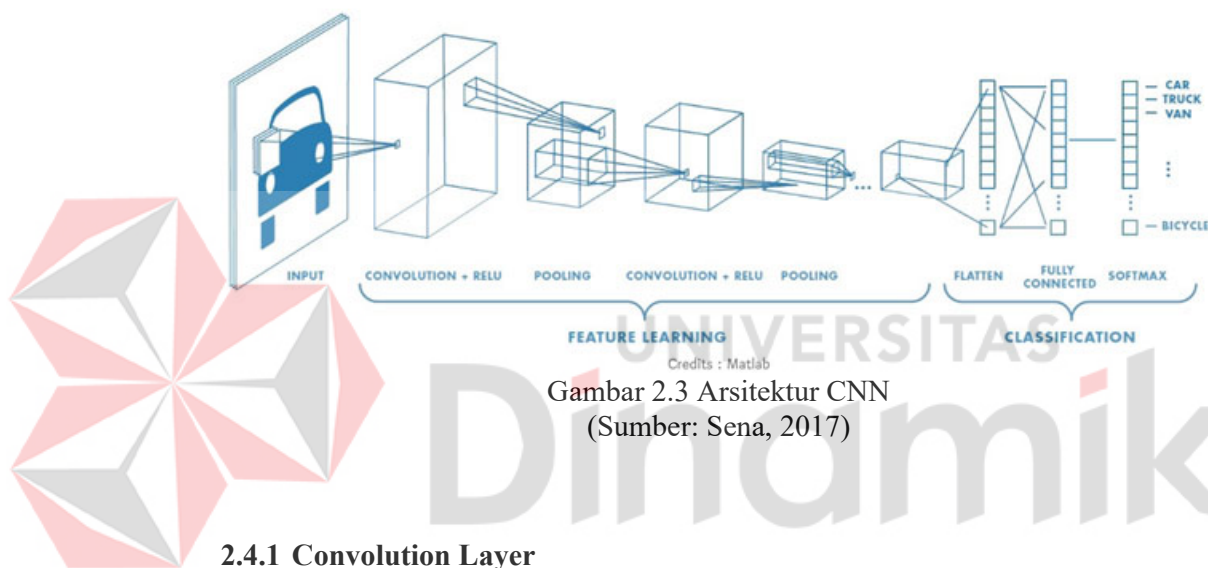
1. *Hidden layer*, adalah jumlah lapisan tersembunyi dan jumlah *neuron* di setiap lapisan dalam arsitektur *deep learning*.
2. *Epochs*, adalah jumlah iterasi atau pengulangan yang harus pada set data Ketika proses *training*. *Epochs* menandakan satu siklus algoritma *deep leaning* belajar dari keseluruhan *training dataset*.
3. *Learning rate*, adalah salah satu parameter *training* untuk menghitung nilai koreksi bobot pada waktu proses *training*. Nilai *learning rate* adalah antara nol (0) sampai satu (1). Semakin besar nilai *learning rate* yang telah disetting sampai dengan mendapatkan hasil dari model dengan kinerja yang paling ideal berdasarkan tingkat akurasi.

2.4 Convolutional Neural Network (CNN)

Convolutional Neural Network (CNN) adalah pengembangan dari *neural network* yang umum digunakan untuk klasifikasi citra. CNN digunakan untuk menganalisis citra yang merupakan vektor berdimensi tinggi sehingga melibatkan

banyak parameter untuk mencirikan jaringan. CNN dapat mengenali objek tertentu bahkan jika tampil melalui cara yang berbeda-beda, ini disebut sebagai terjemahan invarian (Nahata & P. Singh, 2020). Arsitektur CNN dapat mengenali terjemahan invarian melalui perhitungan matematis sebagai berikut:

1. *Convolution*
2. *Pooling*
3. *Flattening*
4. *Full Connection*



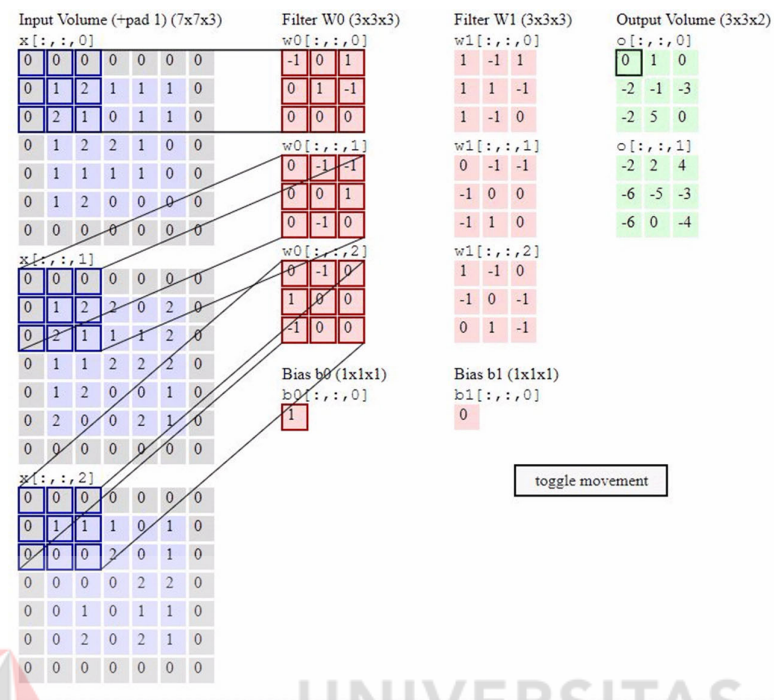
Gambar 2.3 Arsitektur CNN
(Sumber: Sena, 2017)

2.4.1 Convolution Layer

Convolution layer adalah salah satu tahap pada arsitektur CNN yang berfungsi melakukan operasi konvolusi. *Convolution layer* merupakan proses utama yang mendasari jaringan arsitektur CNN. Operasi konvolusi merupakan operasi pada dua fungsi yang menerapkan fungsi *output* sebagai *Feature Map* dari *input* citra. *Input* dan *Output* ini dapat dilihat sebagai dua *argument* yang bernilai nyata.

Convolution layer tersusun dari beberapa neuron yang membentuk sebuah filter dengan Panjang dan tinggi (pixels). Sebagai contoh, *layer* pertama pada *feature extraction layer* biasanya adalah *conv. Layer* dengan ukuran 5x5x3. Panjang 5 *pixels*, tinggi 5 *pixels*, dan tebal/jumlah 3 buah sesuai dengan *channel* dari *image* tersebut. Ketiga *filter* ini akan digeser keseluruh bagian dari gambar. Setiap pergeseran akan dilakukan operasi “dot” antara input dan nilai dari *filter* tersebut

sehingga menghasilkan sebuah output atau biasa disebut sebagai *activation map* atau *feature map*.



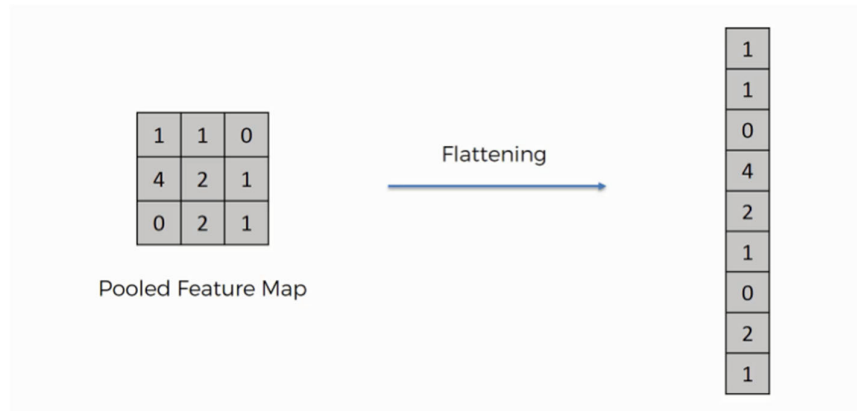
Gambar 2.4 Proses konvolusi
(Sumber: Sena, 2017)

2.4.2 Operasi Pooling

Operasi pooling merupakan operasi untuk mengurangi ukuran *matriks* atau jumlah parameter jika ukuran gambar terlalu besar. Pada dasarnya *pooling layer* terdiri dari sebuah *filter* dengan ukuran dan *stride* tertentu yang akan secara bergantian bergeser pada seluruh area *feature map*. Pooling yang biasa digunakan yaitu *max-pooling*, *sum-pooling*, dan *average pooling*. Proses operasi matematisnya sesuai dengan namanya masing-masing. Lapisan *pooling* yang dimasukkan diantara lapisan konvolusi secara berturut-turut dalam arsitektur model CNN dapat secara progresif mengurangi ukuran *volume output* pada *Feature Map*, sehingga mengurangi jumlah parameter dan perhitungan di jaringan, untuk mengendalikan *Overfitting*.

2.4.3 Flattening

Flattening adalah proses mengubah *matriks* menjadi bentuk *vector* panjang. *Vektor* tersebut kemudian digunakan sebagai *input* data yang melewati *artificial neural network* untuk melanjutkan proses.



Gambar 2.5 Proses flattening
(Sumber: SuperDataScience, 2018)

2.4.4 Fully Connected

Fully connected layer adalah sebuah lapisan dimana semua *neuron* aktivasi dari lapisan sebelumnya terhubung semua dengan neuron di lapisan selanjutnya sama seperti halnya dengan *neural network* biasa. Lapisan ini biasanya digunakan untuk tujuan melakukan transformasi pada dimensi data agar dapat diklasifikasikan secara *linear*.

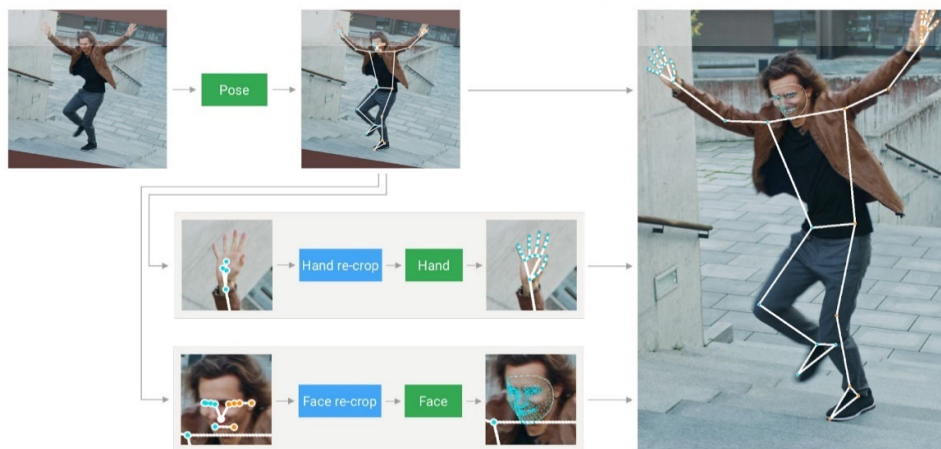
2.5 Mediapipe

Mediapipe adalah sebuah *framework* yang di kembangkan oleh Google yang dapat membangun *pipelines* untuk memproses data persepsi dari berbagai format audio dan video. Diluncurkan pada 2019, *framework* ini menyediakan sekumpulan *machine learning solutions* seperti *Face Detection*, *Iris*, *Hair Segmentation*, *Holistic*, dan lainnya yang dapat dilihat pada web (MediapipeDev, 2019). *Solutions* yang disediakan kompatibel dengan sistem operasi android dan ios, serta dengan Bahasa C++, Python, JS, dan Coral.

2.5.1 Mediapipe Holistic

Mediapipe holistic pipeline mengintegrasikan model yang berbeda untuk komponen *pose*, wajah dan tangan, dimana masing-masing dioptimalkan untuk domain khususnya. Namun karena spesialisasinya berbeda, *input* ke satu komponen tidak cocok untuk yang lain. Oleh karena itu, *Mediapipe holistic* dirancang sebagai *pipeline* yang dapat menangani berbagai wilayah menggunakan resolusi gambar yang sesuai.

Pertama, *holistic* memperkirakan *pose* manusia dengan *detector pose* *BlazePose* dan model *landmark* berikutnya. Kemudian, dengan menggunakan *landmark pose inferred* diperoleh tiga *crop regions of interest* (ROI) untuk masing-masing tangan (2x) dan wajah, dan menggunakan model *re-crop* untuk meningkatkan ROI. Bingkai *input* resolusi penuh kemudian dipotong ke ROI dan menerapkan model wajah dan tangan khusus tugas untuk memperkirakan *landmark* yang sesuai. Terakhir, menggabungkan semua *landmark* dengan model *pose*. *Mediapipe Holistic* menggunakan model *landmark pose*, wajah, dan tangan di *Mediapipe Pose*, *Mediapipe Face Mesh*, dan *Mediapipe Hands* masing-masing untuk menghasilkan total 543 *landmark* (33 *landmark pose*, 468 *landmark wajah*, dan 21 *landmark tangan* setiap tangan).

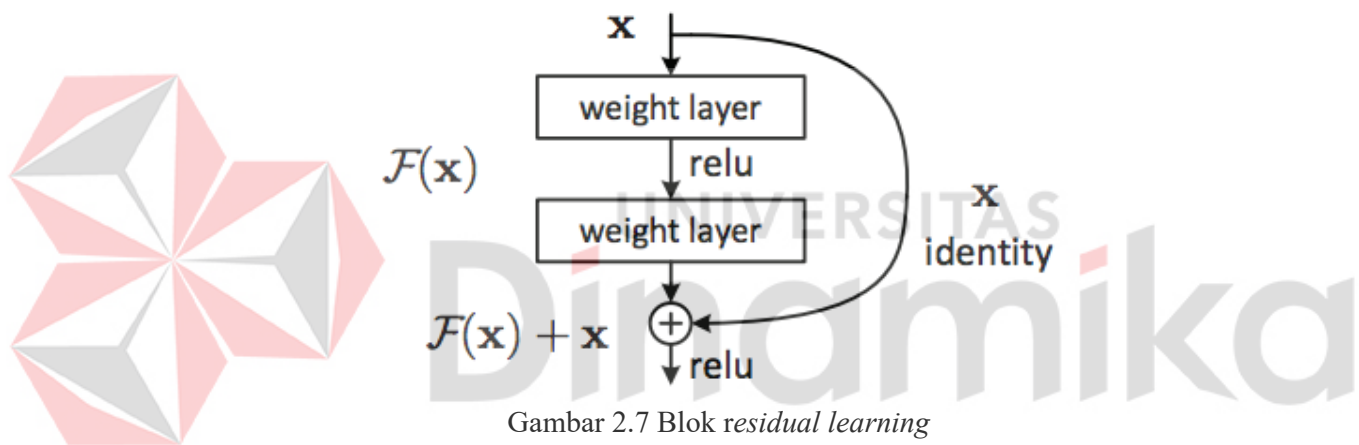


Gambar 2.6 Mediapipe Holistic pipeline
(Sumber: MediapipeDev, 2019)

2.6 ResNet-50

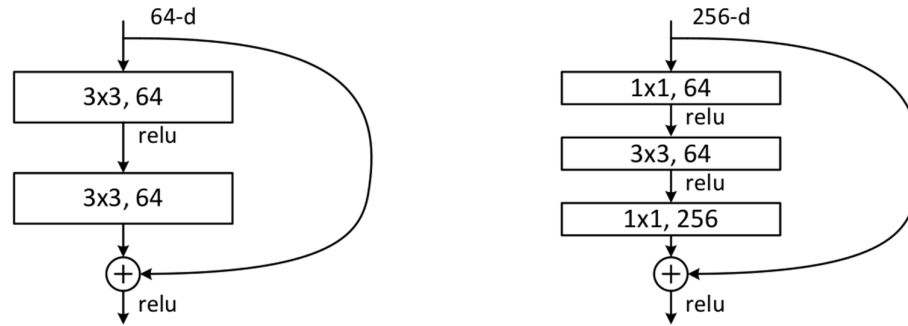
Residual Network (ResNet) merupakan salah satu *deep neural networks* yang memiliki performa generalisasi paling baik pada pengenalan objek. Hasil tersebut didasarkan pada dasar deteksi menggunakan PASCAL VOC 2007 dan 2012, dan COCO (He, et al., 2016).

Pada *deep*-CNN, semakin besar *layer* pada arsitektur maka proses pelatihan menjadi lebih sulit dan akurasi mulai stagnan dan menurun. Pembelajaran residual didesain untuk mengatasi masalah penurunan akurasi (He, et al., 2016). Pembelajaran residual menerapkan metode *shortcut* atau *skip connection* untuk secara langsung menghubungkan *input* pada beberapa *layer* berikutnya.

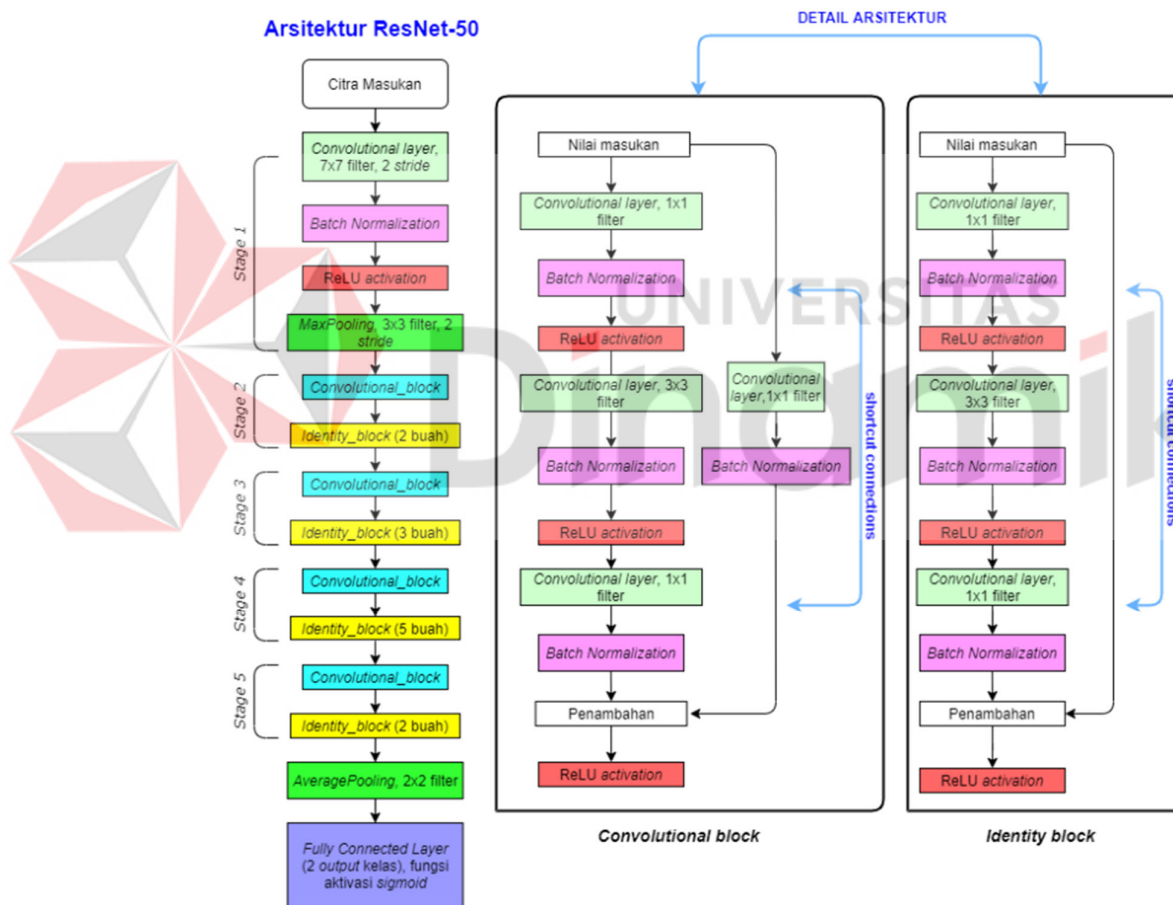


Gambar 2.7 Blok *residual learning*
(Sumber: Kaushik, n d.)

ResNet-50 berarti arsitektur ResNet dengan 50 *layer deep*-CNN, merupakan arsitektur *deep*-CNN yang menggunakan pembelajaran residual pada 2015. Gambar 7 merupakan contoh *residual learning* yang digunakan pada *architecture ResNet 18* dan 34 *layer*. Ada sedikit perubahan yang dibuat untuk *ResNet-50* dan yang lebih tinggi. Jika sebelumnya *shortcut connections* melewati 2 *layer* pada *ResNet-50* menjadi 3 *layer* dan juga terdapat 1x1 *convolution layer* yang ditambahkan.



Gambar 2.8 Blok *residual learning* pada ResNet-50
(Sumber: Kaushik, n.d.)



Gambar 2.9 Arsitektur ResNet-50
(Sumber: Nashrullah, et al., 2020)

2.7 Menghitung Performa

Dalam menentukan atau menghitung performa deep learning, tidak cukup hanya dengan satu angka untuk mengindikasikan probabilitas mendapatkan prediksi yang benar. User mungkin hanya butuh hasil prediksi dari sampel data yang menurut mereka menarik atau memprediksi data dengan beberapa kondisi saja. Sehingga hasil ini tidak bisa digunakan untuk menentukan atau membandingkan performa. Untuk itu, telah dikembangkan berbagai tipe untuk mengevaluasi performa deep learning salah satunya *confusion matrix*.

Confusion matrix berisi informasi perbandingan antara nilai aktual dan prediksi dari klasifikasi sistem (Santra & Christy, 2012). Terdapat empat data yang umumnya digunakan untuk menghitung performa yaitu *true positives (TP)*, *true negatives (TN)*, *false positives (FP)*, dan *false negatives (FN)* (Jiao & Du, 2016). Keempat data tersebut memiliki arti masing-masing. *True positive* memiliki hasil prediksi dan nilai aktual positif. *False positives* memiliki nilai aktual positif dan hasil prediksi negatif. *False negative* berarti nilai aktual adalah negatif dan hasil prediksi adalah positif. *True negative* berarti memiliki nilai aktual dan prediksi negatif. Terdapat empat teknik untuk menghitung performa setelah diketahui confusion matrixnya yaitu akurasi, presisi, *recall*, dan *F1 score*.

		Prediction			
		Positives	Negatives		
Real	Positives	TP	FN	RP = TP + FN	
	Negatives	FP	TN		
		PP = TP + FP PN = TN + FN		RN = FP + TN	

Gambar 2.10 *Confusion matrix*
(Sumber: Jiao & Du, 2016)

2.7.1 Akurasi

Akurasi adalah ukuran perhitungan performa yang paling intuitif dan hanya rasio pengamatan yang diprediksi dengan benar terhadap total pengamatan. Akurasi merupakan ukuran yang bagus tetapi hanya jika model menggunakan data simetris dimana jumlah data pada setiap *class* adalah hampir sama. Untuk menghitung akurasi menggunakan persamaan (2.1)

$$Akurasi = \frac{TP+TN}{TP+FP+FN+TN} \quad (2.1)$$

2.7.2 Presisi

Presisi adalah rasio prediksi positif yang benar terhadap keseluruhan prediksi positif. Presisi menunjukkan seberapa tepat model yang telah di-*training* dalam memprediksi kejadian positif. Untuk menghitung presisi menggunakan persamaan (2.2).

$$Presisi = \frac{TP}{TP+FP} \quad (2.2)$$

2.7.3 Recall

Recall adalah rasio prediksi positif yang benar terhadap seluruh data aktual yang positif. Untuk menghitung recall menggunakan persamaan (2.3).

$$Recall = \frac{TP}{TP+FN} \quad (2.3)$$

2.7.4 F1 Score

F1 Score adalah rata-rata bobot antara presisi dan *recall*. Itulah kenapa nilai ini menghitung *false positive* dan *false ngative*. *F1 Score* lebih berguna dibandingkan akurasi terutama jika dataset yang digunakan jumlahnya tidak seimbang untuk setiap *class*-nya. Untuk menghitung *F1 Score* menggunakan persamaan (2.4).

$$F1\ Score = 2 \times \frac{(recall \times presisi)}{(recall + presisi)} \quad (2.4)$$

BAB III

METODOLOGI PENELITIAN

3.1 Dataset

Dataset yang digunakan merupakan citra gambar manusia yang memeragakan simbol SIBI (Sistem Isyarat Bahasa Indonesia). Terdapat 1200 citra yang terdiri dari 6 simbol (*classes*) yang berbeda, dimana 3 merupakan kata ganti orang yaitu ‘Saya’, ‘Dia, dan ‘Kamu’ serta 3 kata ganti perasaan ‘Cinta, ‘Maaf’, dan ‘Sedih’. Contoh *dataset* dapat dilihat pada Gambar 3.1.



Gambar 3.1 Contoh *dataset* bahasa isyarat

Dataset diambil menggunakan kamera *smartphone*. Untuk mendapatkan *dataset* yang beragam dilakukan beberapa hal yaitu ada 9 subjek berbeda yang menjadi model *dataset*, pemotretan citra menggunakan kamera dari 6 tipe *smartphone* berbeda, posisi kamera *smartphone* adalah *portrait* dan *landscape*, latar belakang citra bebas, citra diambil dari sudut yang berbeda, dan ukuran (*dimensions*) *pixel* citra ada 9 macam. 1 subjek dipotret secara *portrait* dan *landscape* sehingga diasumsikan sebagai 2 subjek yang berbeda.

Dataset dibagi menjadi 3 bagian yaitu *training*, *validation*, dan *testing* dengan perbandingan 80%, 15%, dan 5%. Dari total 1200 citra gambar didapatkan 960

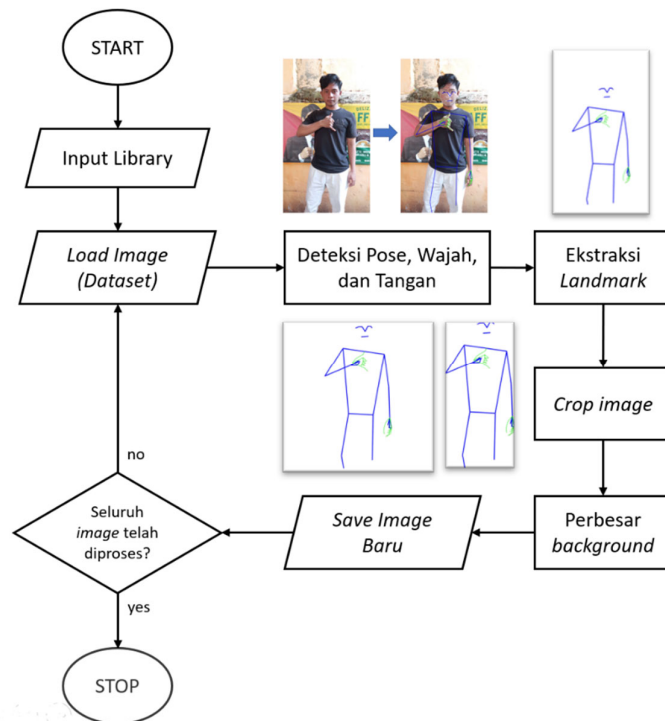
sebagai data *training*, 180 sebagai data *validation*, dan 60 sebagai data *testing*. Secara detail informasi *dataset* dapat dilihat pada Tabel 3.1.

Tabel 3.1 Detail rangkuman *dataset*

No	Subjek	Usia	Gender	Jenis Smartphone	Position	Jumlah Data (Citra)
1	Subjek 1	23 Tahun	Laki-laki	Samsung A32	Portrait	Training: 96 Validation: 18 Testing: 6
2	Subjek 2	22 Tahun	Perempuan	Samsung A32	Portrait	Training: 96 Validation: 18 Testing: 6
3	Subjek 3	21 Tahun	Laki-laki	Realme 3 Pro	Portrait	Training: 96 Validation: 18 Testing: 6
4	Subjek 4	21 Tahun	Perempuan	Redmi note 8 Pro	Portrait	Training: 96 Validation: 18 Testing: 6
5	Subjek 5	22 Tahun	Laki-laki	Samsung S8	Portrait	Training: 96 Validation: 18 Testing: 6
6	Subjek 6	23 Tahun	Laki-laki	Samsung A32	Landscape	Training: 96 Validation: 18 Testing: 6
7	Subjek 7	23 Tahun	Perempuan	Realme 3 Pro	Landscape	Training: 96 Validation: 18 Testing: 6
8	Subjek 8	21 Tahun	Laki-laki	Sony XZ3	Landscape	Training: 96 Validation: 18 Testing: 6
9	Subjek 9	22 Tahun	Perempuan	Infinite Hot 10 Play	Landscape	Training: 96 Validation: 18 Testing: 6
10	Subjek 10	26 Tahun	Laki-laki	Realme 3 Pro	Landscape	Training: 96 Validation: 18 Testing: 6

3.2 Deteksi *Landmark* Mediapipe

Proses *training* menggunakan arsitektur ResNet-50 dilakukan 2 kali dengan *dataset* yang telah diproses deteksi *landmark* menggunakan Mediapipe dan tidak. Proses deteksi *landmark* dilakukan sebelum *training*, hal ini bertujuan untuk melacak posisi wajah, *pose*, dan tangan. Hasilnya kemudian diekstrak sehingga didapatkan *background* putih dengan gambar *landmark*. Pada Gambar 3.2 menunjukkan *flowchart* proses deteksi dan ekstraksi *landmark* pada citra.



Gambar 3.2 Flowchart deteksi dan ekstraksi *landmark*

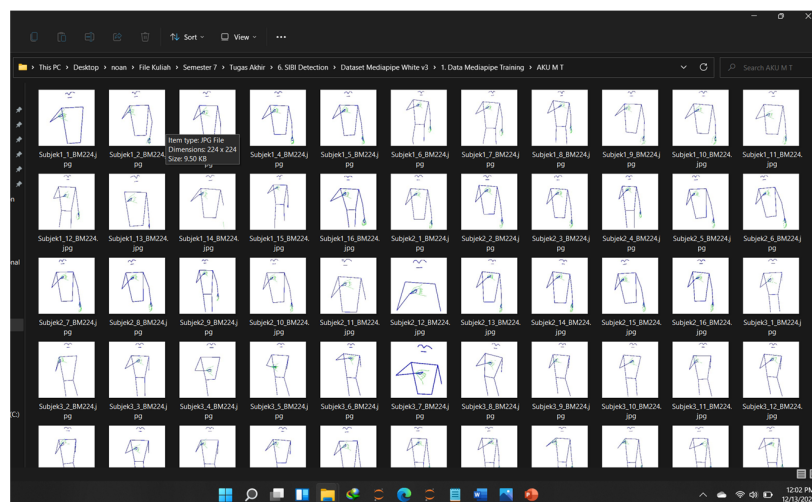
Terdapat beberapa proses yang harus dilakukan. Pertama *dataset* asli dideteksi *landmark*-nya menggunakan Mediapipe Holistic seperti Gambar 3.3. Untuk mempermudah proses klasifikasi pada saat *training*, *landmark* diekstrak ke *background* putih. Untuk tetap menjaga bentuk *landmark* saat citra di-*resize* maka citra hasil ekstraksi di-*crop* hanya pada bagian *landmark*-nya, kemudian *background* putih diperbesar hingga membentuk ukuran persegi. Proses mengolah *landmark* pada *background* putih tersebut seperti ditunjukkan pada Gambar 3.4. Terakhir citra di-*resize* menjadi ukuran 224x224 dan disimpan. Seluruh citra asli diproses dan disimpan ke sebuah folder baru seperti Gambar 3.5.



Gambar 3.3 Citra asli (kiri) citra dengan *landmark* (kanan)



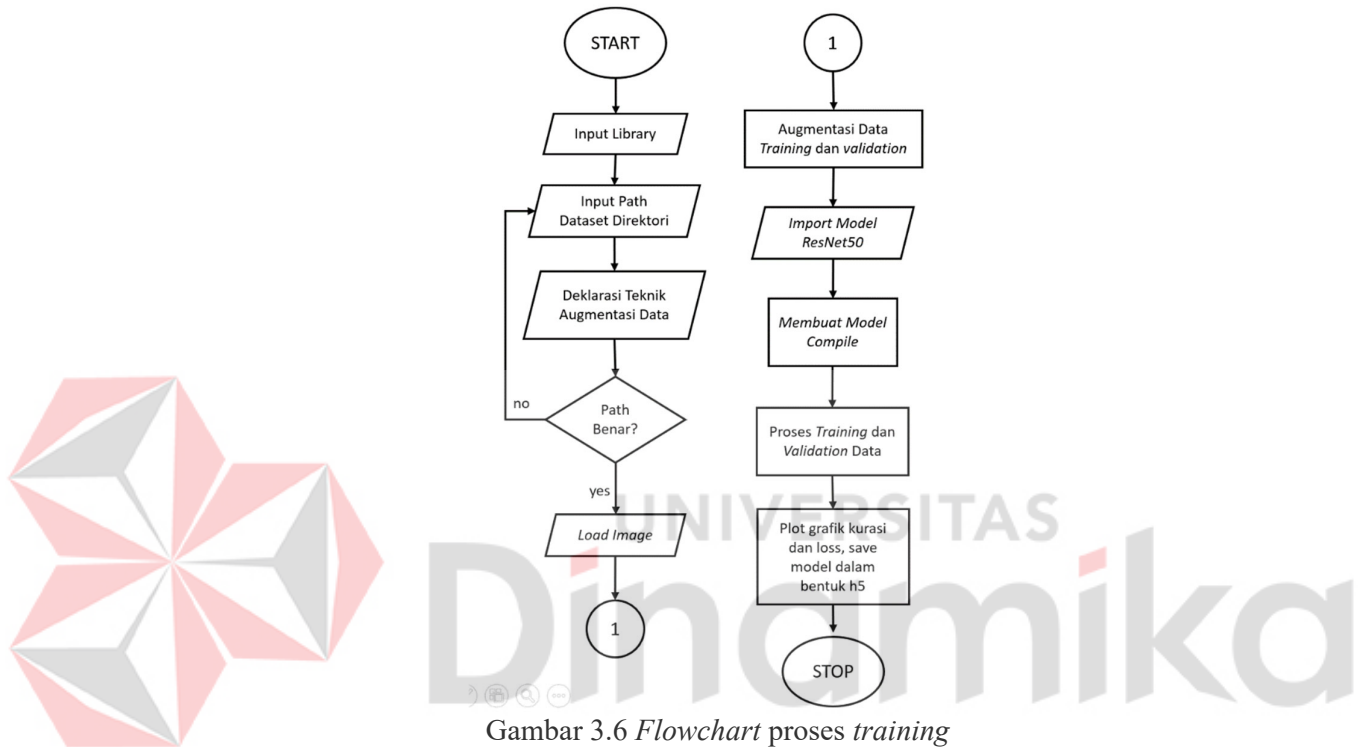
Gambar 3.4 Proses mengolah *landmark* pada *background* putih



Gambar 3.5 Folder hasil ekstraksi *landmark*

3.3 Proses Training ResNet-50

Proses training dilakukan 2 kali dengan *dataset* yang berbeda hasil *dataset* asli dan *dataset* hasil ekstraksi *landmark*. Proses *training* untuk kedua *dataset* sama, hanya berbeda pada menginputkan *path* direktorinya. Proses training seperti ditunjukkan pada flowchart Gambar 3.6.



Gambar 3.6 Flowchart proses training

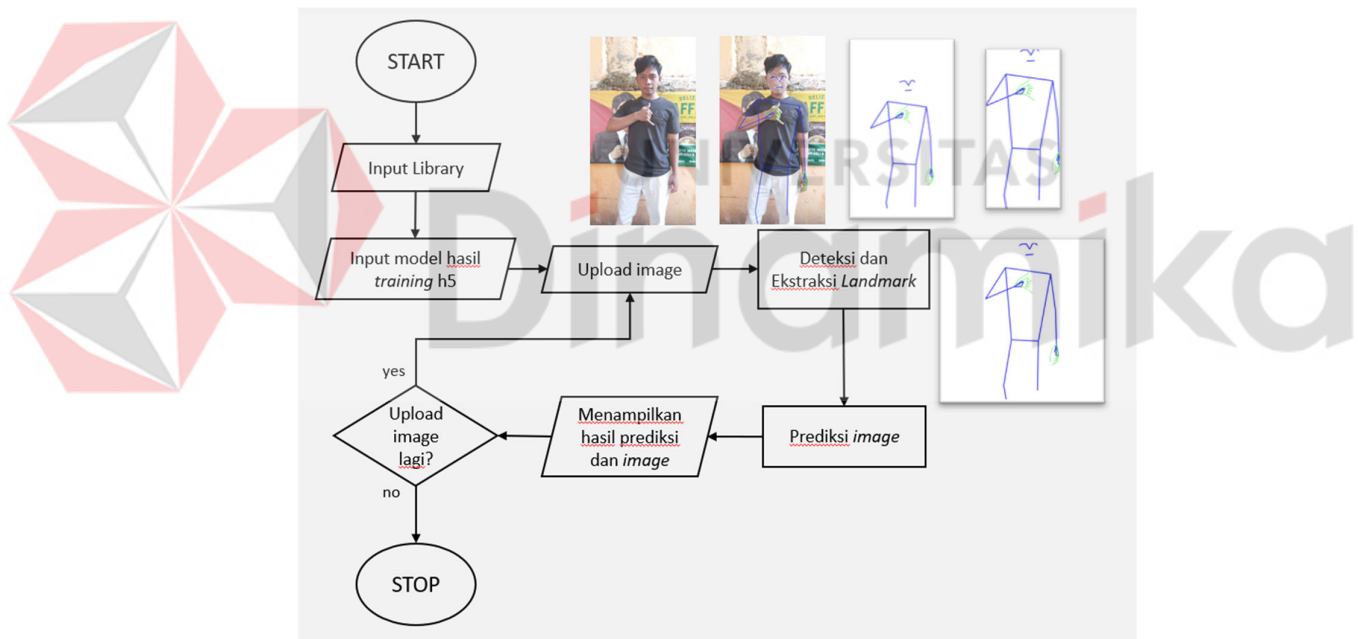
Pertama program menginputkan *library* yang dibutuhkan dan *path* direktori *dataset*. Untuk melakukan proses augmentasi data, perlu dilakukan deklarasi teknik terlebih dahulu. Terdapat beberapa Teknik yang digunakan yaitu *rescale*, *zoom range*, *rotation range*, dan *horizontal flip*. Keempat teknik tersebut dipilih karena dapat memberikan variasi *dataset* yang lebih banyak. Teknik lainnya dianggap penulis tidak dibutuhkan karena data tidak lagi berupa citra asli melainkan telah diproses hingga menjadi ekstraksi *landmark* pada *background* putih, sehingga teknik lainnya tidak memberikan pengaruh atau bahkan merusak *dataset*. Jika *path* benar, maka seluruh *image* akan di-load dan diaugmentasi.

Model arsitektur ResNet-50 yang digunakan di-import dari tensorflow Keras dengan *weights* Imagenet. Selanjutnya dibuatkan model *compile* dengan parameter

input-nya adalah *optimizer* 'adam' dan *loss* *categorical_crossentropy*'. Selanjutnya proses *training* dan *validation* dilakukan untuk melatih arsitektur dengan *dataset* yang disediakan. Hasil dari *training* dan *validation* kemudian diplotting agar dapat dilihat secara visual. Agar hasil *training* dapat digunakan kembali tanpa perlu melakukan hasilnya kemudian disimpan dalam bentuk file h5.

3.4 Proses Testing

Proses *testing* pada *dataset* dilakukan menggunakan fitur *upload files* dari google colab. Sehingga proses testing dapat dilakukan secara umum oleh siapapun. Data *testing* merupakan 5% dari keseluruhan *dataset* dan tidak termasuk dari data *training* atau *validation*. Proses yang dilakukan ketika *testing* ditunjukkan pada *flowchart* Gambar 3.7.



Gambar 3.7 Flowchart proses testing

Pada program *testing*, pertama *input library* yang dibutuhkan dan *load* model hasil *training* h5. Selanjutnya program akan meminta *user* untuk menginputkan *image* untuk diprediksi. Setelah *user* memilih citra, program akan mendeteksi dan ekstraksi *landmark* terlebih dahulu. Hasil ekstraksi kemudian akan di-*resize* menjadi ukuran 224x224 *pixel* lalu diprediksi oleh model. Hasil prediksi kemudian ditampilkan dalam bentuk *text* agar dapat dilihat oleh *user* bersamaan dengan *image* yang telah di-*upload*.

BAB IV

HASIL DAN PEMBAHASAN

4.1 Hasil Proses *Training* Model

Proses *training* dilakukan sebanyak 2 kali dengan dataset yang berbeda yaitu citra asli dan hasil ekstraksi *landmark*. Masing-masing dilakukan *training* sebanyak 250 *epochs* dengan *batch size* 32, yang artinya membutuhkan 30 *iterasi* untuk 1 kali *epoch*. Optimizer yang digunakan pada *training* model adalah ‘adam’ dan *loss* ‘categorical_crossentropy’. Hasilnya didapatkan pada *epochs* ke 250 untuk *training* dengan *dataset* asli mencapai akurasi *training* 33% dan akurasi *validation* 40% sedangkan *training* dengan *dataset* hasil ekstraksi *landmark* mencapai akurasi *training* 88% dan akurasi *validation* 87%. Dengan adanya penambahan Mediapipe maka terjadi peningkatan *akurasi training* sebesar 55% dan peningkatan *akurasi validasi* sebesar 47%. Lebih detail hasil *training* dapat dilihat pada Tabel 4.1 dan Tabel 4.2.

Tabel 4.1 Hasil *training* model ResNet-50

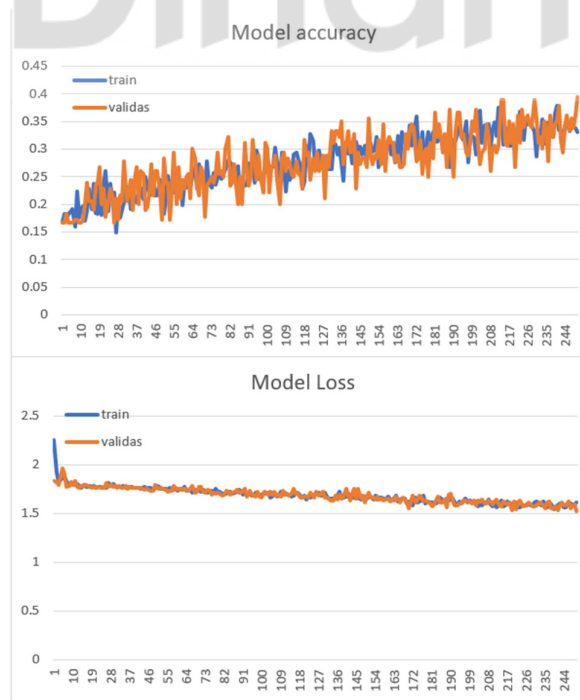
Epochs	Akurasi <i>Training</i> (%)	Akurasi <i>Validation</i> (%)	<i>Training Loss</i>	<i>Validation Loss</i>
25	21	22	1,77	1,76
50	18	28	1,78	1,76
75	24	26	1,73	1,73
100	29	30	1,69	1,66
125	30	26	1,67	1,67
150	31	29	1,65	1,67
175	33	26	1,62	1,68
200	32	33	1,61	1,60
225	35	36	1,58	1,58
250	33	40	1,61	1,52

Tabel 4.2 Hasil *training* Mediapipe dan Model ResNet-50

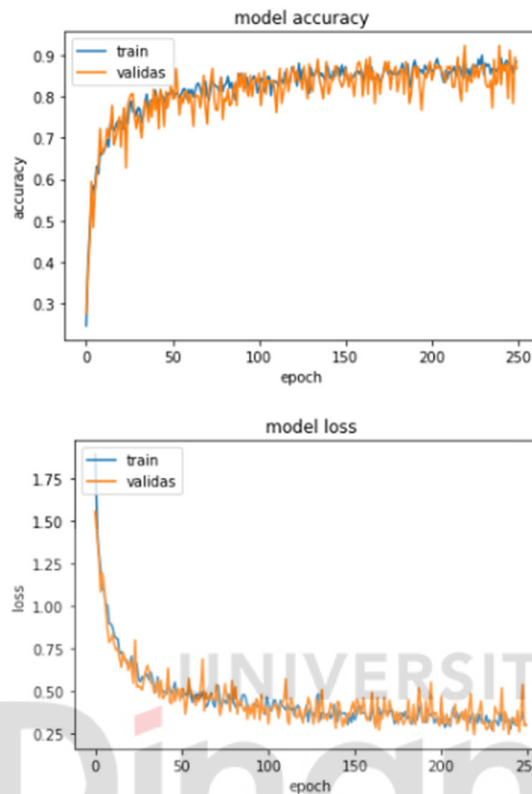
Epochs	Akurasi <i>Training</i> (%)	Akurasi <i>Validation</i> (%)	<i>Training Loss</i>	<i>Validation Loss</i>
25	71	78	0,70	0,53
50	80	83	0,48	0,44
75	81	80	0,48	0,47
100	83	83	0,43	0,40
125	84	84	0,39	0,43
150	85	83	0,39	0,44
175	89	86	0,27	0,38
200	88	88	0,29	0,30
225	88	88	0,30	0,31
250	88	87	0,31	0,30

Pada Tabel 4.1 dan Tabel 4.2 ditunjukkan 2 hasil parameter untuk setiap *training* dan *validation* yaitu akurasi dan *loss*. Akurasi merupakan salah satu parameter yang digunakan untuk menghitung performa model dalam persentase. Akurasi menjadi ukuran seberapa akurat prediksi model dibandingkan data sebenarnya. *Loss* digunakan untuk mengoptimalkan algoritma pembelajaran model. *Loss* adalah jumlah *error* yang dibuat untuk setiap contoh dalam set *training* dan *validation*. Nilai *loss* menunjukkan seberapa buruk atau baik model untuk setiap iterasinya.

Hasil *training* untuk kedua *dataset* menunjukkan perbedaan yang sangat tinggi dengan *loss* pada *training* citra asli yang mencapai 1,61. Hasil tersebut menunjukkan bahwa penggunaan citra asli pada pembelajaran ResNet-50 tidak efektif, sehingga perlu dilakukan *image pre-processing* sebelumnya. Sementara hasil *training* pada *dataset* yang telah diproses menggunakan Mediapipe menunjukkan hasil yang lebih baik. Sehingga penggunaan Mediapipe untuk deteksi *landmark* pada tubuh efektif untuk pembelajaran model ResNet-50 dalam mengklasifikasikan simbol SIBI.



Gambar 4.1 Grafik akurasi dan *loss* model ResNet-50



Gambar 4.2 Grafik akurasi dan *loss* Mediapipe dan Model ResNet-50

Pada Gambar 4.1 menunjukkan grafik proses *training* dengan citra asli terlihat fluktuatif dan *underfitting*. *Underfitting* merupakan kondisi dimana proses *training* menghasilkan *loss* tinggi dan akurasi rendah. Hal ini disebabkan karena model tidak dapat memahami dan mengklasifikasikan data dengan tepat. Sementara pada Gambar 4.2 menunjukkan grafik proses training dengan *images* hasil ekstraksi *landmark* yang terlihat fluktuatif dan optimal.

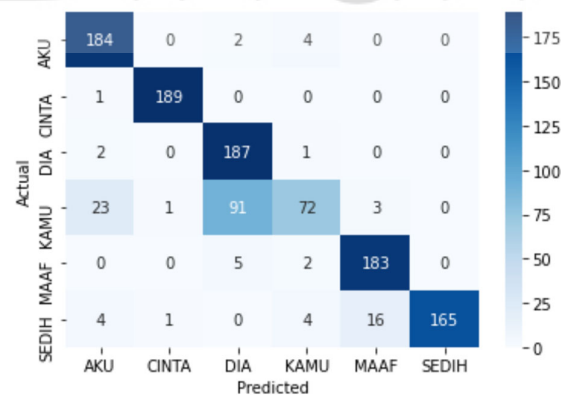
4.2 Perbandingan Performa

Untuk mengukur kinerja atau performa dari model dengan lebih mudah digunakan tabel *confusion matrix*, karena dapat menggambarkan dengan lebih detail jumlah data yang diklasifikasikan dengan benar. Pada Gambar 4.3

ditampilkan *confusion matrix* dari hasil *training* dengan *dataset* citra asli. Didapatkan bahwa hasil klasifikasi model terhadap *dataset* tidak baik, karena setiap simbol dapat terklasifikasi ke 5 simbol lainnya. Pada Gambar 4.4 ditampilkan *confusion matrix* dari hasil *training* dengan *dataset* ekstraksi *landmark* dari citra asli. Didapatkan Sebagian besar dataset telah terklasifikasikan dengan tepat. Pada simbol ‘Aku’, ‘Cinta’, ‘Dia’, dan ‘Maaf’ simbol terklasifikasi dengan tepat. Pada simbol ‘Sedih’ sebagian besar terklasifikasi dengan benar sementara pada simbol ‘Kamu’ yang sebagian besar terklasifikasi pada simbol lain.



Gambar 4.3 *Confusion matrix* model ResNet-50



Gambar 4.4 *Confusion matrix* Mediapipe Dan model ResNet-50

Dari tabel *confusion matrix* dapat dihitung nilai *performance metrics* yang umum dan sering digunakan yaitu *accuracy*, *precision*, *recall*, dan *F1-score*. Pada Tabel 4.3 ditunjukkan nilai performa dari hasil training model ResNet-50 terhadap

dataset citra asli. Didapatkan nilai setiap *performance metrics* dari nilai rata-rata keenam simbol (*class*) adalah *precision* 33%, *recall* 33%, *F1-score* 32%. Sementara pada Tabel 4.4 ditunjukkan nilai performa dari hasil *training* model ResNet-50 terhadap *dataset* yang telah *dideteksi landmark*-nya menggunakan Mediapipe. Didapatkan nilai setiap *performance metrics* dari nilai rata-rata keenam simbol (*class*) adalah *precision* 86%, *recall* 86%, *F1-score* 85%. Dari hasil ini menunjukkan bahwa performa dari model yang di-*training* menggunakan *dataset* ekstraksi *landmark* menggunakan Mediapipe lebih baik dibandingkan dengan citra asli. Dengan adanya penambahan Mediapipe telah meningkatkan nilai *performance matrix*, adapun peningkatan nilai persentase dari masing-masing performa tersebut adalah sebagai berikut: *precision* 53%, *recall* 53% dan *F1-score* 53%.

Tabel 4.3 Hasil performa model ResNet-50

	Precision	Recall	F1-Score	Support
“Aku”	0.17	0.13	0.15	30
“Cinta”	0.62	0.43	0.51	30
“Dia”	0.27	0.50	0.35	30
“Kamu”	0.31	0.13	0.19	30
“Maaf”	0.21	0.20	0.21	39
“Sedih”	0.42	0.57	0.49	30
Accuracy			0.33	180
Macro Avg	0.34	0.33	0.32	180
Weighted Avg	0.34	0.33	0.32	180

Tabel 4.4 Hasil performa Mediapipe dan model ResNet-50

	Precision	Recall	F1-Score	Support
“Aku”	0.86	0.97	0.91	190
“Cinta”	0.99	0.99	0.99	190
“Dia”	0.66	0.98	0.79	190
“Kamu”	0.87	0.38	0.53	190
“Maaf”	0.91	0.96	0.93	190
“Sedih”	1.00	0.87	0.93	190
Accuracy			0.86	1140
Macro Avg	0.88	0.86	0.85	1140
Weighted Avg	0.88	0.86	0.85	1140

4.3 Prediksi

Model yang telah di-*training* disimpan dalam bentuk file h5 digunakan untuk melakukan *testing* atau prediksi. Disediakan 60 citra asli atau 5% dari jumlah

keseluruhan *dataset*. Setiap subjek memiliki 1 citra gambar dari setiap simbol SIBI untuk di-*testing*. Terdapat 2 program yang berbeda yaitu menggunakan proses deteksi dan ekstraksi *landmark* dengan Mediapipe dan yang tidak. Pada program yang menggunakan model ResNet-50 citra hanya akan di-*resize* ke ukuran 224x224 *pixel* kemudian diprediksi. Sementara untuk program yang menggunakan Mediapipe dan model ResNet-50, citra akan dideteksi *landmark*-nya menggunakan Mediapipe. Selanjutnya hasil dari *landmark* akan diestraksi ke *background* putih. *Images* kemudian di-*resize* ke ukuran 224x224 *pixel* untuk selanjutnya diprediksi.

Proses *testing* dibuat agar dapat digunakan secara umum. Terdapat fitur uploading *image* yang dapat digunakan oleh user untuk mengload citra, kemudian *user* memilih citra pada direktori di komputer untuk diupload dan diprediksi, seperti ditunjukkan pada Gambar 4.5.



Gambar 4.5 Interface prediksi SIBI

Hasil dari prediksi untuk model ResNet-50 dapat dilihat pada Tabel 4.5 dan untuk model Mediapipe dan ResNet-50 pada Tabel 4.6. Dari hasil percobaan didapatkan bahwa akurasi prediksi model ResNet-50 adalah 47%, dengan rincian simbol ‘Aku’ 20%, ‘Cinta’ 70%, ‘Dia’ 60%, ‘Kamu’ 30%, ‘Maaf’ 60%, dan ‘Sedih’ 40%. Sedangkan akurasi prediksi Mediapipe dan model ResNet-50 adalah 87%, dengan rincian simbol ‘Aku’ 90%, ‘Cinta’ 100%, ‘Dia’ 100%, ‘Kamu’ 50%, ‘Maaf’

100%, dan 'Sedih' 80%. Dari hasil ini didapatkan bahwa model dengan Mediapipe dan ResNet-50 lebih baik dibandingkan model ResNet-50 saja.



UNIVERSITAS
Dinamika

Tabel 4.5 Hasil prediksi model ResNet-50

No	Subjek	Simbol SIBI (Sistem Isyarat Bahasa Indonesia)					
		Aku	Cinta	Dia	Kamu	Maaf	Sedih
1	Subjek 1	Prediksi: Aku	Prediksi: Aku	Prediksi: Aku	Prediksi: Sedih	Prediksi: Maaf	Prediksi: Sedih
2	Subjek 2	Prediksi: Aku	Prediksi: Cinta	Prediksi: Dia	Prediksi: Aku	Prediksi: Sedih	Prediksi: Dia
3	Subjek 3	Prediksi: Maaf	Prediksi: Sedih	Prediksi: Maaf	Prediksi: Kamu	Prediksi: Maaf	Prediksi: Maaf
4	Subjek 4	Prediksi: Maaf	Prediksi: Cinta	Prediksi: Dia	Prediksi: Kamu	Prediksi: Sedih	Prediksi: Cinta
5	Subjek 5	Prediksi: Cinta	Prediksi: Cinta	Prediksi: Aku	Prediksi: Kamu	Prediksi: Maaf	Prediksi: Sedih
6	Subjek 6	Prediksi: Sedih	Prediksi: Dia	Prediksi: Dia	Prediksi: Maaf	Prediksi: Dia	Prediksi: Dia
7	Subjek 7	Prediksi: Maaf	Prediksi: Cinta	Prediksi: Dia	Prediksi: Dia	Prediksi: Maaf	Prediksi: Maaf
8	Subjek 8	Prediksi: Dia	Prediksi: Cinta	Prediksi: Kamu	Prediksi: Sedih	Prediksi: Maaf	Prediksi: Dia
9	Subjek 9	Prediksi: Maaf	Prediksi: Cinta	Prediksi: Sedih	Prediksi: Maaf	Prediksi: Maaf	Prediksi: Sedih
10	Subjek 10	Prediksi: Maaf	Prediksi: Cinta	Prediksi: Cinta	Prediksi: Dia	Prediksi: Cinta	Prediksi: Sedih
Prosentase		Benar: 2 Salah: 8 Prosentase: 20%	Benar: 7 Salah: 3 Prosentase: 70%	Benar: 6 Salah: 4 Prosentase: 60%	Benar: 3 Salah: 7 Prosentase: 30%	Benar: 6 Salah: 4 Prosentase: 60%	Benar: 4 Salah: 6 Prosentase: 40%

Tabel 4.6 Hasil prediksi Mediappe dan Model ResNet-50

No	Subjek	Simbol SIBI (Sistem Isyarat Bahasa Indonesia)					
		Aku	Cinta	Dia	Kamu	Maaf	Sedih
1	Subjek 1	Prediksi: Aku	Prediksi: Cinta	Prediksi: Dia	Prediksi: Dia	Prediksi: Maaf	Prediksi: Kamu
2	Subjek 2	Prediksi: Aku	Prediksi: Cinta	Prediksi: Dia	Prediksi: Dia	Prediksi: Maaf	Prediksi: Sedih
3	Subjek 3	Prediksi: Kamu	Prediksi: Cinta	Prediksi: Dia	Prediksi: Aku	Prediksi: Maaf	Prediksi: Sedih
4	Subjek 4	Prediksi: Aku	Prediksi: Cinta	Prediksi: Dia	Prediksi: Kamu	Prediksi: Maaf	Prediksi: Sedih
5	Subjek 5	Prediksi: Aku	Prediksi: Cinta	Prediksi: Dia	Prediksi: Kamu	Prediksi: Maaf	Prediksi: Sedih
6	Subjek 6	Prediksi: Aku	Prediksi: Cinta	Prediksi: Dia	Prediksi: Kamu	Prediksi: Maaf	Prediksi: Sedih
7	Subjek 7	Prediksi: Aku	Prediksi: Cinta	Prediksi: Dia	Prediksi: Dia	Prediksi: Maaf	Prediksi: Sedih
8	Subjek 8	Prediksi: Aku	Prediksi: Cinta	Prediksi: Dia	Prediksi: Dia	Prediksi: Maaf	Prediksi: Maaf
9	Subjek 9	Prediksi: Aku	Prediksi: Cinta	Prediksi: Dia	Prediksi: Kamu	Prediksi: Maaf	Prediksi: Sedih
10	Subjek 10	Prediksi: Aku	Prediksi: Cinta	Prediksi: Dia	Prediksi: Kamu	Prediksi: Maaf	Prediksi: Sedih
Prosentase		Benar: 9 Salah: 1 Prosentase: 90%	Benar: 10 Salah: 0 Prosentase: 100%	Benar: 10 Salah: 0 Prosentase: 100%	Benar: 5 Salah: 5 Prosentase: 50%	Benar: 10 Salah: 0 Prosentase: 100%	Benar: 8 Salah: 2 Prosentase: 80%

BAB V

PENUTUP

5.1 Kesimpulan

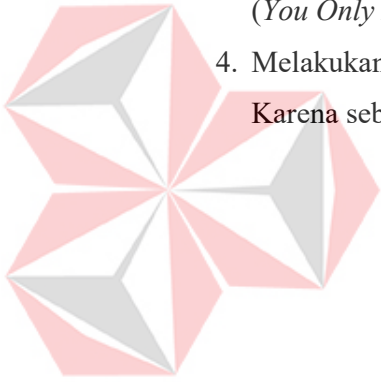
Dari hasil pengujian dan perbandingan data, didapatkan beberapa kesimpulan yaitu:

1. Hasil dari proses *training* menggunakan *Convolutional Neural Network Pre-trained* model ResNet-50 mempunyai nilai akurasi *training* 33% dan akurasi *validation* 40%. Setelah adanya penambahan proses deteksi dan ekstraksi *landmark* menggunakan Mediapipe terjadi peningkatan akurasi *training* sebesar 88% dan akurasi *validation* menjadi 87%. Sehingga dengan adanya proses penambahan Mediapipe maka terjadi peningkatan nilai akurasi *training* sebesar 55% dan akurasi *validation* sebesar 47%.
2. Dari hasil *training* dilakukan proses perhitungan *performance metrics* dengan hasil *training* citra asli menggunakan ResNet-50 tanpa Mediapipe didapatkan nilai performa rata-rata keenam simbol (*class*) adalah *precision* 33%, *recall* 33%, dan *F1-score* 32%. Sementara untuk hasil *training* dengan menggunakan Mediapipe dan model ResNet-50 didapatkan nilai performa rata-rata keenam simbol (*class*) adalah *precision* 86%, *recall* 86%, dan *F1-score* 85%. Sehingga dengan adanya proses penambahan Mediapipe maka terjadi peningkatan nilai performa rata-rata keenam simbol (*class*), yaitu: *precision* 53%, *recall* 53%, dan *F1-score* 53%.
3. Setelah dilakukan proses *training* di kedua sistem tersebut diatas pada proses *testing* menggunakan ResNet-50 tanpa Mediapipe didapatkan hasil sebesar 47%, dengan rincian sebagai berikut: simbol 'Aku' 20%, 'Cinta' 70%, 'Dia' 60%, 'Kamu' 30%, 'Maaf' 60%, dan 'Sedih' 40%. Sedangkan hasil akurasi *testing* menggunakan Mediapipe dan model ResNet-50 mempunyai nilai persentase rata-rata dari keenam simbol sebesar 87%, dengan rincian sebagai berikut: simbol 'Aku' 90%, 'Cinta' 100%, 'Dia' 100%, 'Kamu' 50%, 'Maaf' 100%, dan 'Sedih' 80%.

5.2 Saran

Adapun saran untuk pengembangan pada penelitian ini agar lebih baik, terdapat beberapa saran yaitu:

1. Untuk mendapatkan nilai akurasi yang lebih tinggi, dapat menambahkan jumlah *dataset* dengan beberapa aspek yaitu jumlah simbol (*class*), variasi usia dari subjek yang menjadi model *dataset*, dan variasi tipe kamera dengan resolusi yang berbeda-beda.
2. Menemukan metode *image pre-processing* lain yang lebih memudahkan model *Convolutional Neural Network pre-trained* model ResNet-50 untuk melakukan klasifikasi terhadap *dataset* citra asli.
3. Menggunakan proses *deep learning* selain *Convolutional Neural Network* seperti LSTM (*Long Short Term Memory*), SSD (*Single Short Detector*), YOLO (*You Only Look Once*), atau Faster-RCNN.
4. Melakukan pengujian terhadap *dataset* yang berupa video, bukan citra gambar. Karena sebagian dari bahasa isyarat merupakan gerakan dinamis.



UNIVERSITAS
Dinamika

DAFTAR PUSTAKA

- Cholissodin, I. et al., 2019. *AI, Machine Learning & Deep Learning (Teori & Implementasi)*. Malang: Fakultas Ilmu Komputer Universitas Brawijaya.
- Grandhi, C., Liu, S. & Roharia, D., 2021. American Sign Language Recognition using Deep Learning. *International Conference 2021*.
- Halder, A. & Tayade, A., 2021. Real-time Vernacular Sign Language Recognition using Mediapipe and Machine Learning. *Real-time Vernacular Sign Language Recognition using Mediapipe and* , pp. 9-17.
- He, K., Zhang, X., Ren, S. & Sun, J., 2015. Deep Residual Learning for Image Recognition. *IEEE*, pp. 770-778.
- Jiao, Y. & Du, P., 2016. Performance measures in evaluating machine learning based bioinformatics predictors for classifications. *School of Computer Science and Technology*, 4(4), pp. 320-330.
- Kaushik, A., t.thn. *Understanding ResNet50 architecture*. [Online]
Available at: <https://iq.opengenus.org/resnet50-architecture/>
[Diakses 16 9 2021].
- Kemensos, 2021. *SISTEM INFORMASI MANAGEMENT PENYANDANG DISABILITAS*. [Online]
Available at: <https://simpd.kemensos.go.id/>
[Diakses 12 9 2021].
- Maharsi, J., 2018. *Mengenal Bahasa Isyarat*. [Online]
Available at: <https://komunita.id/2018/11/30/mengenal-bahasa-isyarat/>
- MediapipeDev, 2019. *Live ML anywhere*. [Online]
Available at: <https://google.github.io/Mediapipe/>
[Diakses 16 9 2021].
- Nahata, H. & P. Singh, S., 2020. *Deep Learning Solutions for Skin Cancer Detection and Diagnosis*, Singapore: Nanyang Technological University.

- Nashrullah, F., Wibowo, S. A. & Budiman, G., 2020. Investigasi Parameter Epoch Pada Arsitektur ResNet-50 Untuk Klasifikasi Pornografi. *Journal of Computer, Electronic, and Telecommunication*, 1(1), pp. 1-8.
- Nasir, M. C., Sudaryanto, E. & Kusumaningrum, H., 2021. Penggunaan Sistem Isyarat Bahasa Indonesia (SIBI) Sebagai Media Komunikasi. *Konferensi Nasional*.
- Putra, I. R. W., 2021. *Sistem Deteksi Simbol Pada SIBI (Sistem Isyarat Bahasa Indonesia) Menggunakan Convolutional Neural Network*, Surabaya: Universitas Dinamika.
- Ronaghan, S., 2018. *Deep Learning: Common Architectures*. [Online] Available at: <https://srnghn.medium.com/deep-learning-common-architectures-6071d47cb383> [Diakses 13 9 2021].
- Santra, A. K. & Christy, C. J., 2012. Genetic Algorithm and Confusion Matrix for Document Clustering. *JCSI International Journal of Computer Science Issues*, 9(1), pp. 322-328.
- Setyawan, A., 2019. Komunikasi Antar Pribadi Non Verbal Penyandang Komunikasi Antar Pribadi Non Verbal Penyandang. *Jurnal Kajian Ilmiah Universitas Bhayangkara Jakarta Raya*, 19(2), pp. 165-174.
- SuperDataScience, 2018. *Convolutional Neural Networks (CNN): Step 3 - Flattening*. [Online] Available at: <https://www.superdatascience.com/blogs/convolutional-neural-networks-cnn-step-3-flattening> [Diakses 14 9 2021].
- Widiarti, A. & Rahayu, H., 2020. Pemberdayaan Penyandang Disabilitas Dalam Proses Pembangunan Di Indonesia Ditinjau Dari Perspektif Perundang-undangan. *Jurnal Ilmu Hukum*, 3(2), pp. 274-283.
- Yifei, L., 2017. *Deep Neural Networks and Fraud Detection*, Uppsala: UPPSALA UNIVERSITET.