

Perancangan Sistem Penerjemah Sign Language To Text Berbasis Image Processing

Design A Translation System For Sign Language To Text Using Image Processing

1st Bagas Ruli Pandapotan

Fakultas Ilmu Terapan
Universitas Telkom

Bandung, Indonesia

bagasrulipanda@student.telkomuniversity.ac.id

2nd Suci Aulia

Fakultas Ilmu Terapan
Universitas Telkom

Bandung, Indonesia

suciaulia@telkomuniversity.ac.id

3rd Sugondo Hadiyoso

Fakultas Ilmu Terapan
Universitas Telkom

Bandung, Indonesia

sugondo@telkomuniversity.ac.id

Abstrak—Bahasa isyarat adalah bahasa yang digunakan untuk berkomunikasi dengan menggunakan gerakan tubuh dan/atau ekspresi wajah. Makna atau arti dari gerakan tubuh dalam bahasa isyarat juga telah disepakati agar dapat digunakan untuk bertukar informasi. Bahasa isyarat ini biasa digunakan oleh penyandang tunarungu dan tunawicara untuk saling bertukar informasi dengan masyarakat normal pada umumnya. Hal ini yang membuat masyarakat normal mengalami kesulitan untuk berkomunikasi karena tidak semua memahami arti dari bahasa isyarat tersebut. Sehingga, untuk mengatasi masalah tersebut dilakukan upaya perancangan sistem penerjemah agar dapat dimengerti oleh masyarakat umum. Pada proyek akhir ini, telah dirancang sistem penerjemah bahasa isyarat per karakter menggunakan metode *Convolutional Neural Network* dengan arsitektur VGG-19 berbasis *image processing* dengan menggunakan bahasa pemrograman *python*. Data berupa dataset gambar gerakan tangan dalam bahasa isyarat akan dikumpulkan dan digunakan sebagai acuan objek pengenalan untuk pendeteksian sistem penerjemah ini, yang selanjutnya akan diproses untuk menjalankan program penerjemah bahasa isyarat per karakter ini menjadi sebuah teks. Hasil dari proyek akhir ini menunjukkan bahwa sistem dapat menerjemahkan BISINDO per karakter dengan akurasi pengujian sebesar 100%.

Kata kunci — convolutional neural network, BISINDO, VGG-19.

Abstract—Sign language is the language used to communicate using body movements and/or facial expressions. The meaning of body movements in sign language has also been agreed so that it can be used to exchange information. This sign language is commonly used by deaf and speech impaired people to exchange

information with normal society in general. This makes it difficult for normal people to communicate because not all understand the meaning of sign language. Thus, to overcome this problem, efforts were made to design a translation system so that it could be understood by the general public. In this Final Project, a sign language translator system has been designed percharacter using the *Convolutional Neural Network* method with a VGG-19 architecture based on image processing using the python programming language. Data in the form of a dataset of images of hand movements in sign language will be collected and used as a reference for recognition objects for the detection of this translation system, which will then be processed to run this sign language translator program percharacter into a text. The results of this final project is the system can translate BISINDO per character with a test accuracy of 100%.

Keyword — convolutional neural network, BISINDO, VGG-19.

I. PENDAHULUAN

Berkomunikasi merupakan salah satu hal yang harus dilakukan oleh makhluk hidup pada umumnya, khususnya umat manusia. Bahasa menjadi salah satu media untuk berkomunikasi. Sangat banyak bahasa yang dapat digunakan, salah satunya adalah bahasa isyarat. Bahasa isyarat tentunya lebih mengutamakan berbagai gestur atau gerakan isyarat dibandingkan dengan suara. Gerakan isyarat ini dapat berupa, ekspresi wajah, gerakan tangan, dan jari. Tentunya hal ini akan cukup sulit karena tidak semua masyarakat memahami bahasa isyarat tersebut, karena bahasa isyarat ini lebih sering digunakan untuk komunikasi antara penyandang Tunarungu dan/atau Tunawicara dengan masyarakat umum. Tunawicara merupakan suatu hambatan di

dalam komunikasi verbal, begitu juga dengan Tunarungu yang merupakan hambatan dalam pendengaran verbal[1]. Itulah sebabnya bahasa isyarat dibutuhkan untuk berkomunikasi. Diharapkan melalui perancangan sistem penerjemah ini, segala kesulitan dalam komunikasi bahasa isyarat dapat diatasi.

Sampai saat ini, ada banyak cara atau jenis bahasa isyarat yang digunakan untuk berkomunikasi di Indonesia, yaitu SIBI atau Sistem Bahasa Isyarat Indonesia dan BISINDO atau Bahasa Isyarat Indonesia. Kedua jenis bahasa isyarat tersebut lumrah digunakan untuk berkomunikasi dengan bahasa isyarat di Indonesia. Kedua metode ini, menggunakan metode *hand on hand* atau gestur gerakan tangan ke tangan untuk penyampaian informasinya[2].

Pada proyek akhir ini, telah dirancang sistem penerjemah bahasa isyarat per karakter berbasis *Image Processing* dengan menggunakan metoda CNN berarsitektur VGG-19, dan menggunakan bahasa pemrograman *python*. Data berupa dataset gambar gerakan tangan dalam bahasa isyarat akan dikumpulkan dan digunakan sebagai acuan objek pengenalan untuk pendeteksian sistem penerjemah ini, yang selanjutnya akan diproses untuk menjalankan program penerjemah bahasa isyarat per karakter ini.

II. METODE

A. Google Colab

Google Colab merupakan platform yang sering dimanfaatkan untuk *Machine Learning* (ML) maupun *Deep Learning* (DL). Platform ini pun mendukung fitur *sharing* seperti pada Google Drive untuk melakukan pembatasan akses. Aplikasi Google Colab ini menggunakan bahasa pemrograman *python*, dengan penggunaan yang lebih ringan dibanding dengan platform pemrograman *python* lainnya[3].

B. Bahasa Isyarat Indonesia (BISINDO)

Bahasa Isyarat Indonesia atau BISINDO merupakan bahasa isyarat yang diperuntukkan untuk penyandang tunarungu. BISINDO adalah bahasa isyarat yang mengadopsi nilai budaya asli Indonesia dan mudah dapat digunakan untuk berkomunikasi diantara penyandang tunarungu di kehidupan sehari-hari. Kepraktisan ini membuat BISINDO lebih mudah dipahami, berbeda dengan SIBI yang peluncurannya tidak melalui persetujuan dari GERKATIN (Gerakan Kesejahteraan Tunarungu Indonesia) yang pada akhirnya tidak dapat digunakan di kehidupan sehari-hari[2].

Bahasa Isyarat Indonesia ini menggunakan kedua tangan sebagai gerakan bahasa isyaratnya,

dengan menghimpun seluruh huruf A-Z hingga keseluruhan kata dan angka yang ada dalam kamus Bahasa Indonesia. Sehingga, metode BISINDO inilah yang akhirnya digunakan dan diresmikan sebagai bahasa isyarat resmi yang digunakan di Indonesia. Hal ini dapat terjadi karena bentuk gestur tangan untuk alfabet dari BISINDO ini terlihat mirip dengan huruf yang dimaksud. Bentuk gestur tangan pada BISINDO dapat dilihat pada gambar 1.



GAMBAR 1
GESTUR BISINDO

C. Pengolahan Citra (*Image Processing*)

Pengolahan citra merupakan proses memanipulasi dan menganalisis citra dengan bantuan komputer dengan menggunakan deteksi tepi yang menentukan titik-titik tepi dari objek data yang digunakan. Seiring berjalannya waktu, teknik pengolahan citra terus berkembang sehingga sekarang memungkinkan untuk mengambil informasi dari pengolahan citra yang telah dilakukan[4].

D. Convolutional Neural Network (CNN)

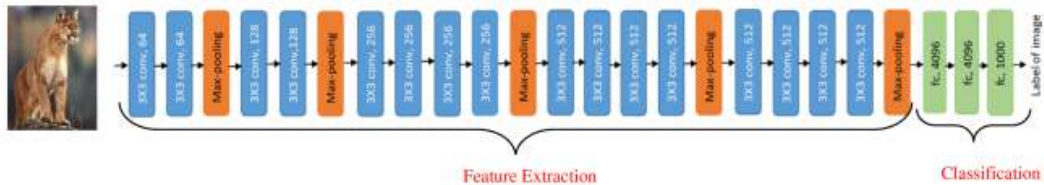
Convolutional Neural Network atau CNN merupakan salah satu metoda dalam pengolahan citra yang didesain untuk mengolah data dua dimensi. CNN pertama kali dikembangkan dengan nama NeoCognitron oleh Kunihiko Fukushima, seorang peneliti dari NHK Broadcasting Science Research Laboratories, Kinuta, Setagaya, Tokyo, Jepang[5]. Pada CNN, data yang akan dipropagasikan pada jaringan citranya adalah citra dua dimensi, yang pada akhirnya akan membuat operasi linear dan parameter bobot pada CNN berbeda[6].

E. Arsitektur VGG-19

VGG atau *Visual Geometry Group* merupakan salah satu model arsitektur yang digunakan dalam metode pengolahan citra secara *Convolutional Neural Network* atau CNN. Ada dua jenis VGG yaitu, VGG-16 dan VGG-19. Arsitektur VGG ini akan melakukan pemodelan dengan cara memberikan penambahan lapisan pada citra untuk dilakukan pengetesan dan pengujian dan

mengurangi tingkat kesalahan dalam pengolahan citra pada CNN. VGG-19 memiliki 19 lapisan atau layer dengan pembagiannya adalah 16 convolution layers dan 3 fully connected layers yang akan mengklasifikasikan sebuah citra menjadi 1000 kategori objek. VGG-19 dilatih dengan menggunakan database dari ImageNet yang memiliki 1000 kategori objek tersebut[7]. Ukuran citra standar atau

default dari VGG-19 ini adalah 224x224[8] yang nantinya akan dimasukkan dalam proses pemodelan, tetapi tidak menutup kemungkinan untuk dapat dilakukan modifikasi sesuai dengan spesifikasi yang dibutuhkan. Berikut adalah bentuk dari arsitektur pada VGG-19 dapat dilihat pada gambar 1.2 .



GAMBAR 1.2 ARSITEKTUR VGG-19

F. TensorFlow

Sebuah platform *open-source* untuk penggunaan *machine learning*. Tensorflow termasuk menjadi salah satu library dari pemrograman python yang

memiliki ekosistem alat, pustaka yang fleksibel yang memudahkan setiap pengguna dalam menerapkan prosedur *machine learning* [9].



GAMBAR 1.3 TENSORFLOW

G. Keras

Merupakan salah satu interface library yang bertujuan menyederhanakan implementasi algoritma-algoritma deep learning di atas TensorFlow. Keras biasa digunakan untuk tujuan pengembangan dan pengevaluasian model deep learning[10]. Keras dapat bekerja di atas kerangka kerja TensorFlow yang merupakan salah satu library dalam model machine learning. Keras relative

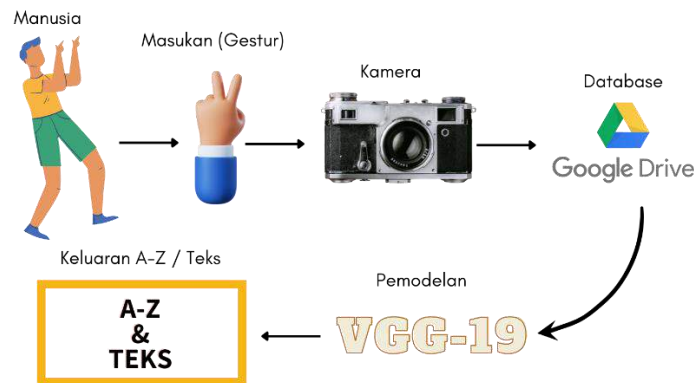
mudah dipelajari dan digunakan karena menyediakan front-end python dengan abstraksi tingkat tinggi dan memiliki beberapa opsi back-end untuk tujuan komputasi. Keras juga menyediakan beberapa fitur terbaik dalam menyusun model, memproses dataset, memvisualisasikan grafik, dan beberapa hal lainnya[11].



GAMBAR 1.4 KERAS

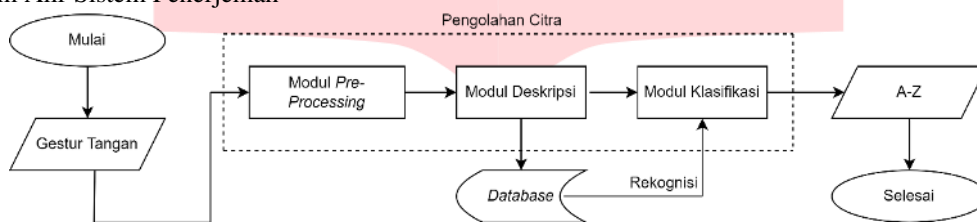
III. HASIL DAN PEMBAHASAN

A. Blok Diagram Sistem Penerjemah



GAMBAR 3.1 BLOK DIAGRAM SISTEM PENERJEMAH SIGN LANGUAGE TO TEXT

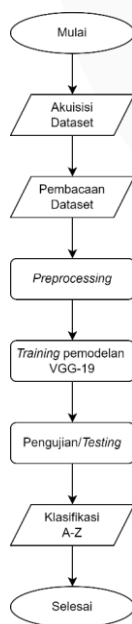
B. Diagram Alir Sistem Penerjemah



GAMBAR 3.2 DIAGRAM ALIR SISTEM PENERJEMAH SIGN LANGUAGE TO TEXT

C. Diagram Alir Pendeteksian Citra

Pada proses penerjemahan dari bahasa isyarat ini akan melalui tahap pendeteksian dari citra gestur yang diambil menggunakan tangan dari penulis beserta partisipan lainnya. Sehingga diperlukan sistem dari pendeteksian ini. Diagram Alir dari pendeteksian citra ini ditunjukkan pada gambar 3.3 .



GAMBAR 3.3 DIAGRAM ALIR SISTEM PENDETEKSIAN CITRA SAMPAI PENERJEMAHAN

Proses dari pendeteksian ini adalah sebagai berikut.

1. Akuisisi Dataset
 Dataset baru yang berisi kumpulan citra gestur tangan yang diambil langsung oleh penulis dengan melakukan pengambilan citra gerakan tangan BISINDO menggunakan kamera DSLR Canon EOS1300D 55mm dengan citra yang berformat *Joint Photographic Experts Group* (JPEG atau JPG) beresolusi 3.456 x 2.304. Dataset yang diambil berjumlah 1.040 citra yang terdiri dari 520 citra dataset awal, dan 520 citra dataset baru untuk validasi. Keseluruhan dataset ini terdiri dari huruf A-Z dengan masing-masing huruf berjumlah 20 citra dengan 4 kondisi. Citra yang telah diambil untuk dijadikan dataset ini akan diupload ke Google Drive sesuai dengan kelas yang digunakan.
2. Pembacaan Dataset
 Akan dilakukan pembacaan dataset berupa gestur tangan BISINDO yang akan diterjemahkan yaitu huruf A-Z, dan 10 kata.

3. Preprocessing

Pada tahap ini diberikan pelabelan untuk huruf A-Z pada program Google Colab, yang dijadikan sebagai acuan pembacaan dataset. Kemudian, dilakukan pembagian dataset untuk training dan testing. Pembagian ini ditetapkan jumlah citra untuk dilakukan training sebesar 80% dari keseluruhan dataset, sedangkan untuk dilakukan testing sebesar 20% dari keseluruhan dataset. Setelah dilakukan pembagian tersebut, maka dilakukanlah perubahan ukuran atau resize citra menjadi 64 x 64, dikarenakan resolusi tersebut yang terbaik dan dapat dilakukan pemodelan arsitektur VGG-19 untuk dilakukan pelatihan dan pengujian.

4. Pelatihan Pemodelan Arsitektur VGG-19

Pada tahap ini dilakukan pemodelan arsitektur VGG-19 pada citra setelah dilakukan tahap resize dan penentuan jumlah data citra untuk training dan testing pada tahap preprocessing. Pemodelan ini dilakukan dengan cara penentuan model VGG pada program, penentuan jumlah kelas yang akan diberikan pemodelan sesuai dengan jumlah yang ada pada database yakni huruf A-Z yaitu 26 kelas, lalu melakukan percobaan epochs dan batch untuk mendapatkan akurasi tertinggi atau maksimal yang selanjutnya akan dilakukan pemodelan dan disimpan dengan file .h5 di Google Drive.

5. Pengujian

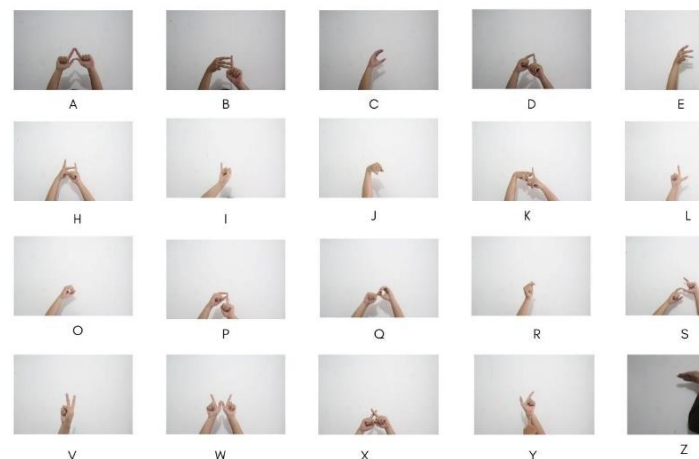
Pada tahap ini, dilakukan pengujian atau testing pada dataset berjumlah 20% dari keseluruhan dataset. Pengujian ini dilakukan dengan menggunakan file penyimpanan pemodelan VGG-19 yang diberikan pada 20% dataset untuk diuji apakah penerjemahan berjalan dengan baik atau tidak, dan seluruh kelas A-Z dan 10 kata dapat terbaca dengan baik oleh sistem. Apabila terbaca semua, dan tidak ada error atau kesalahan, itu berarti hasil pemodelan arsitektur VGG-19 yang sudah disimpan tadi dapat dikatakan baik atau berhasil, sehingga dapat berjalan ke proses selanjutnya.

6. Klasifikasi Teks & A-Z

Setelah dilakukan proses training dan testing maka selanjutnya masuk ke tahap akhir yaitu pengklasifikasian teks dan/atau huruf A-Z beserta penampilan citra gestur tangan dari teks atau huruf yang diklasifikasikan.

D. Dataset

Pada proyek akhir ini, dataset diambil dengan melibatkan 2 partisipan, yaitu penulis beserta seorang teman dari penulis. Dataset yang pertama diambil dengan menggunakan gestur tangan dari penulis mulai dari A-Z secara keseluruhan, kemudian dataset ini yang dijadikan data training dan testing untuk dataset awal. Lalu, dataset yang baru akan diambil dengan menggunakan gestur tangan dari dua orang partisipan lainnya, yang mana dataset ini diambil secara acak dan/atau random mulai dari huruf A-Z, yang kemudian dataset ini akan di training dan testing untuk dijadikan validasi apakah sistem penerjemah ini dapat berjalan atau tidak. Jika nilai dataset dapat diterjemahkan setelah melalui proses pelatihan dan pengujian, maka berkas dari data uji ini akan disimpan dan digunakan dalam proses penerjemahan bahasa isyarat per karakter ini terhadap huruf A-Z dan 10 kata. Dataset yang digunakan pada proyek akhir ini dapat dilihat di gambar 3.4.



GAMBAR 3.4
CITRA DATASET

IV. HASIL DAN PEMBAHASAN

Analisa dari hasil yang telah dilakukan dibutuhkan sebagai penilaian atas program yang telah dirancang dan dibuat ini. Simulasi yang dilakukan diantaranya adalah dengan melakukan pengujian besaran *epoch* dan *batch size*.

A. Simulasi Epoch pada Pemodelan Arsitektur VGG-19

Pada proses pengujian proyek akhir ini telah disimulasikan pengaruh besaran nilai epochs dan batch size pada arsitektur VGG-19 terhadap tingkat akurasi pada dataset yang dilakukan pelatihan. Pada proses pelatihan ini dilakukan proses percobaan sebanyak 8 kali percobaan dengan jumlah epochs yang berbeda. Pelatihan ini dilakukan sampai menemukan nilai akurasi paling tinggi atau maksimal. Disaat sudah menemukan nilai akurasi

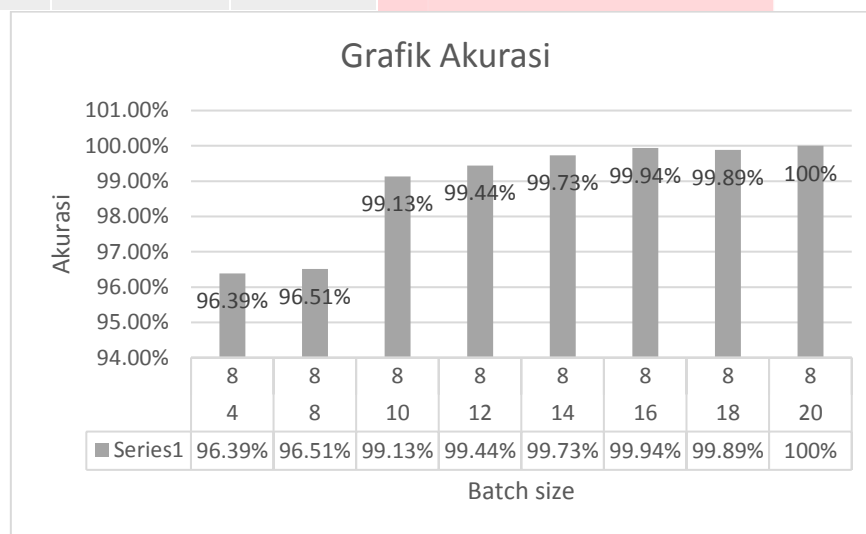
yang maksimal, maka file penyimpanan dari pengujian epochs dan batch size ini akan digunakan untuk proses selanjutnya yaitu validasi pada citra dari huruf A-Z dan citra dari 10 kata. Hasil dari simulasi ini dapat dilihat pada tabel 4.1 .

TABEL 4. 1
HASIL PERCOBAAN EPOCH DAN BATCH SIZE

Data Akurasi			
No.	Epochs	Batch Size	Akurasi
1	4	8	96,39%
2	8	8	96,51%
3	10	8	99,13%

4	12	8	99,44%
5	14	8	99,73%
6	16	8	99,94%
7	18	8	99,89%
8	20	8	100%

Berdasarkan tabel 4.1 yang berisi hasil dari percobaan terhadap pelatihan pemodelan arsitektur VGG-19 ini, maka akan dibuat grafik yang memperlihatkan peningkatan akurasi setelah dilakukan penambahan epoch dan batch size. Grafik ini dapat dilihat pada gambar 4.2 .



GAMBAR 4.2
GRAFIK PERCOBAAN EPOCH DAN BATCH SIZE

B. Rata-rata Waktu Proses

Dalam proses perancangan sistem ini, akan ada tahap training atau pelatihan dan tahap testing atau pengujian. Setelah tahap pelatihan, maka akan masuk ke tahap selanjutnya yaitu tahap pengujian dimana pada tahap ini akan dilakukan pengujian validasi hasil penyimpanan pemodelan arsitektur

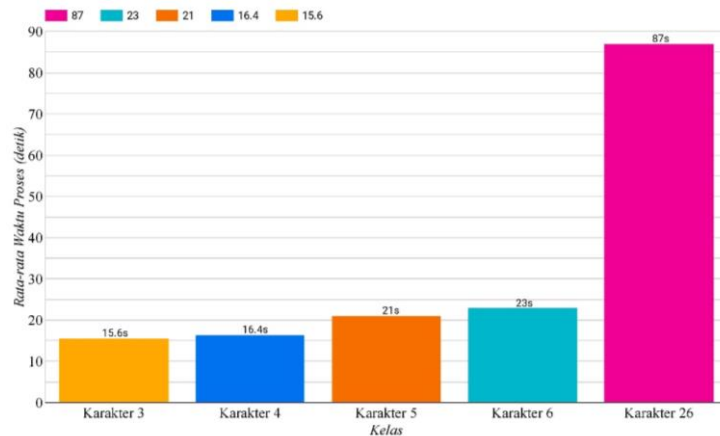
VGG-19 pada data citra apakah dapat berjalan dengan baik atau tidak. Dalam proses pengujian ini, akan dilakukan penghitungan rata-rata waktu proses yang akan dijadikan acuan berapa lama sistem ini akan bekerja. Tabel rata-rata ini dapat dilihat pada tabel 4.2 di bawah ini.

TABEL 4. 2
RATA-RATA WAKTU PROSES

No.	Kelas	Pengujian Ke-... (detik)										Rata-rata Waktu Proses (detik)
		1	2	3	4	5	6	7	8	9	10	
1.	3 Karakter	22	17	17	17	23	11	15	14	10	10	15,6
2.	4 Karakter	17	17	15	20	15	14	20	17	15	14	16,4
3.	5 Karakter	22	22	20	20	30	19	22	17	21	17	21
4.	6 Karakter	27	27	20	21	19	24	19	19	19	35	23
5.	26 Karakter	97	85	93	84	87	85	84	84	88	83	87

Berdasarkan tabel rata-rata waktu proses ini, dibuat grafik yang menunjukkan rata-rata dari waktu proses

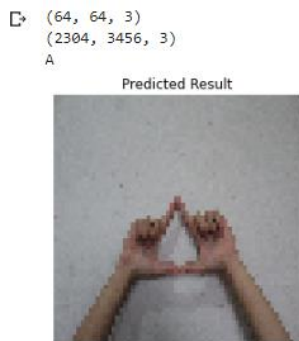
ini secara lengkap yang dapat dilihat pada gambar 4.3 .



GAMBAR 4.3
GRAFIK RATA-RATA WAKTU PROSES

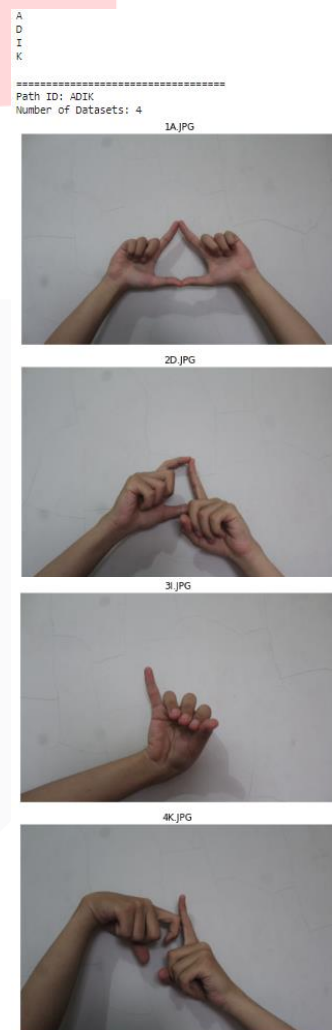
E. Hasil Pengujian

Pada bagian ini, akan diperlihatkan bagaimana hasil dari proses pengujian dari pemodelan arsitektur VGG-19 ini pada kelas yang digunakan, yaitu data citra berupa 26 huruf dan 10 kata yang telah ditentukan. Pada hal ini, akan diperlihatkan beberapa hasil dari pengujian tersebut salah satunya pada huruf A dan kata ADIK. Pada huruf A, akan ditampilkan kelas "A" beserta citra A, begitu pula dengan kata ADIK akan ditampilkan kelas A, D, I, dan K serta citra A, D, I, dan K. Untuk hasil pada huruf A dapat dilihat pada gambar 4.4 .



GAMBAR 4.4
HASIL PENGUJIAN PADA HURUF A

Untuk pengujian pada kata ADIK dapat dilihat pada gambar 4.5 .



GAMBAR 4.5
HASIL PENGUJIAN PADA KATA ADIK

V. KESIMPULAN

Pada proyek akhir ini telah dirancang sebuah sistem untuk menerjemahkan BISINDO per karakter menggunakan metode Convolutional Neural Network dengan arsitektur VGG-19. Penerjemahan BISINDO per karakter ini akan menerjemahkan

bahasa isyarat untuk huruf A-Z, dan 10 kata yaitu “adik, ayah, halo, ibu, kakak, kakek, kamu, mereka, saya, siapa”. Adapun hasil yang diperoleh adalah sebagai berikut.

1. Pada proyek akhir perancangan penerjemah BISINDO ini menggunakan total 1.040 dataset yang terdiri atas 520 data citra untuk huruf A-Z, dan 520 data citra baru huruf A-Z yang digunakan untuk validasi.
2. Pengambilan citra ini ditentukan berbagai kondisi, yaitu kondisi terang dan gelap, serta background polos dan heterogen. Tinggi dan jarak kamera pun ditentukan, yaitu masing-masing berjarak 1 meter.
3. Kamera yang digunakan untuk pengambilan citra ini adalah Canon EOS1300D yang menghasilkan citra beresolusi 3.456 x 2.304 dengan format JPEG atau JPG.
4. Data citra yang telah diambil ini akan disimpan di Google Drive sebagai pusat data dari proyek akhir ini yang kemudian dihubungkan dengan Google Colab.
5. Pada arsitektur VGG-19, nilai ukuran citra standar adalah sebesar 224x224 tetapi pada proyek akhir ini dirubah atau dilakukan resize menjadi 64x64 dikarenakan citra pada ukuran 224x224 menunjukkan error.
6. Pada tahap pengujian dari proyek akhir ini, dilakukan 2 skenario dengan pembedanya adalah keperluan dari data citra yang diambil pada skenario pertama dan skenario kedua. Dari kedua skenario ini, dilakukan pembagian jumlah data latih dan data uji sebesar 80% untuk data latih, dan 20% untuk data uji. Pengujian pun dilakukan dengan melakukan pengujian epochs dan batch size pada data citra yang kemudian disimpan sebagai data uji yang akan digunakan pada tahap validasi dari proses pengujian pada data citra di proses penerjemahan.
7. Pemberian epochs dan batch size dilakukan pada saat melakukan pemodelan arsitektur VGG-19 dan dilakukan beberapa kali dengan besaran yang berbeda sampai menemukan tingkat akurasi yang tertinggi, yaitu pada percobaan ke-8 epoch sebesar 20 dan batch size sebesar 8 dengan tingkat akurasi sebesar 100%. Hasil dari pengujian ini akan digunakan pada tahap validasi di proses penerjemahan.
8. Proses validasi ini menghasilkan bahwa proses pengujian data citra berhasil dengan dapat menerjemahkan data citra menjadi huruf A-Z dan teks sesuai dengan data citra yang dimasukkan.

REFERENSI

- [1] M. T. Informatika and S. Handayani,

“MEDIA PEMBELAJARAN FINGERSPELLING ALPHABET UNTUK PENDERITA TUNARUNGU DAN TUNAWICARA BERBASIS ANDROID.”

- [2] A. Brevia Yunanda, F. Mandita, and A. Primasetya Armin, “Pengenalan Bahasa Isyarat Indonesia (BISINDO) Untuk Karakter Huruf Dengan Menggunakan Microsoft Kinect,” *Fountain Informatics J.*, vol. 3, no. 2, p. 41, Nov. 2018, doi: 10.21111/fij.v3i2.2469.
- [3] R. Maya and F. Lubis, “PENGEMBANGAN ANALISA ALGORITMA AUTOREGRESSIVE INTEGRATED MOVING AVERAGE (ARIMA-BOX JENKINS) PEMODELAN MENGGUNAKAN GOOGLE COLAB (PHYTON).”
- [4] O. Vedak, P. Zavre, A. Todkar, and M. Patil, “Sign Language Interpreter using Image Processing and Machine Learning,” *Int. Res. J. Eng. Technol.*, 2008, [Online]. Available: www.irjet.net.
- [5] T. Varma, R. Baptista, D. C. Pandi, and R. Coutinho, “Sign Language Detection using Image Processing and Deep Learning Index Terms-Deep Learning- Convolution Neural Network (CNN) - American Sign Language (ASL) - Canny Edge Detection,” *Int. Res. J. Eng. Technol.*, 2020, [Online]. Available: www.irjet.net.
- [6] “15696-34686-1-PB.”
- [7] D. Hindarto and H. Santoso, “Plat Nomor Kendaraan dengan Convolution Neural Network.”
- [8] M. Bansal, M. Kumar, M. Sachdeva, and A. Mittal, “Transfer learning for image classification using VGG19: Caltech-101 image data set,” *J. Ambient Intell. Humaniz. Comput.*, 2021, doi: 10.1007/s12652-021-03488-z.
- [9] N. Wiranda, H. S. Purba, and R. A. Sukmawati, “Survei Penggunaan Tensorflow pada Machine Learning untuk Identifikasi Ikan Kawasan Lahan Basah,” *IJEIS (Indonesian J. Electron. Instrum. Syst.)*, vol. 10, no. 2, p. 179, Oct. 2020, doi: 10.22146/ijeis.58315.
- [10] F. Rahutomo, D. Novita Sari, P. Studi Teknik Informatika, J. Teknologi Informasi, and P. Negeri Malang, “JIP (Jurnal Informatika Polinema) IMPLEMENTASI LIBRARY DEEP LEARNING KERAS PADA SISTEM

UJIAN ESSAY ONLINE.”

- [11] A. Santoso and G. Ariyanto,
“IMPLEMENTASI DEEP LEARNING
BERBASIS KERAS UNTUK
PENGENALAN WAJAH,” *J. Tek. Elektro*,
vol. 18, no. 01, [Online]. Available:
<https://www.mathworks.com/discovery/con>
vol.

