

Klasifikasi Isyarat Bahasa Indonesia Menggunakan Metode *Convolutional Neural Network*

<http://dx.doi.org/10.28932/jutisi.v8i2.4863>

Riwayat Artikel

Received: 30 Mei 2022 | Final Revision: 4 Agustus 2022 | Accepted: 5 Agustus 2022

Creative Commons License 4.0 (CC BY – NC)



Muhammad Ezar Al Rivan^{✉#1}, Suryanto Hartoyo^{#2}

[#]*Informatika, Fakultas Ilmu Komputer dan Rekayasa*

[#] *Universitas Multi Data Palembang, Jl. Rajawali No. 14 Palembang 30114 Indonesia*

¹*meedzhar@mdp.ac.id*

²*suryantohartoyo5@gmail.com*

[✉]Corresponding author: *meedzhar@mdp.ac.id*

Abstrak — Bahasa Isyarat Indonesia adalah isyarat kata yang awalnya diambil dari isyarat yang disampaikan oleh anak tunarungu. Bahasa isyarat adalah hal yang biasa bagi orang tuli dan bisu, tetapi tidak asing bagi orang biasa. Untuk itu diperlukan perantara alternatif yang dapat menjadi penerjemah antara penyandang tunarungu dan tuna wicara serta masyarakat biasa. Penelitian ini bertujuan untuk mengklasifikasikan sistem isyarat bahasa Indonesia menggunakan metode Convolutional Neural Network dengan arsitektur VGG (Visual Geometric Group)-16 dan Alexnet. Data terdiri dari huruf A sampai dengan huruf Z yaitu 320 data uji, 1600 data latih, dan 320 data validasi, dan data akan diubah ukurannya menjadi ukuran 224 x 224 piksel, dilanjutkan dengan grayscale dan augmentasi. Hasil pengujian VGG-16 menunjukkan bahwa klasifikasi menggunakan VGG-16 dengan optimasi Adam mendapatkan tingkat akurasi tertinggi, yaitu 99,32% untuk setiap huruf, 91,18% untuk keseluruhan. Sedangkan hasil klasifikasi menggunakan VGG-16 dengan optimasi SGD mendapatkan tingkat akurasi terendah, yaitu 98,85% untuk setiap huruf dan 84,96% untuk keseluruhan. Sedangkan dari hasil pengujian AlexNet terlihat bahwa hasil klasifikasi menggunakan AlexNet dengan Adam optimizer mendapatkan tingkat akurasi tertinggi, yaitu 99,16% untuk setiap huruf dan 89,04% untuk keseluruhan. Sedangkan hasil klasifikasi menggunakan AlexNet dengan pengoptimal SGD mendapatkan tingkat akurasi terendah, yaitu 97,33% untuk setiap huruf dan 68,33% untuk keseluruhan.

Kata kunci— AlexNet; Bahasa Isyarat Indonesia; Convolutional Neural Network; VGG-16

Classification of Indonesian Sign Language Using Convolutional Neural Network

Abstract — Indonesian Sign Language is word signs initially taken from the signs conveyed by deaf children. Sign language is common for the deaf and mute, but it is no stranger to ordinary people. For this reason, alternative intermediaries are needed who can become translators between deaf and speech impaired sufferers and ordinary people. This study aims to classify the Indonesian sign system using the Convolutional Neural Network method with VGG-16 and Alexnet architecture. The data divided by each letter from the letter A to the letter Z is 320 test data, 1600 train data, and 320 validation data, and the data will be resized to a size of 224 x 224 pixels, followed by grayscale and augmentation. The results of the VGG-16 test show that the classification using VGG-16 with the Adam optimizer gets the highest level of accuracy, which is 99.32% for each letter, 91.18% for the whole. While the classification results using VGG-16 with the SGD optimizer get the lowest level of accuracy, which is 98.85% for each letter and 84.96% for the whole. Meanwhile, from the AlexNet test results, it can be seen that the results of the classification using AlexNet with the Adam optimizer get the highest level of accuracy, which is 99.16% for each letter and 89.04% for the whole. While the classification results using AlexNet with the SGD optimizer get the lowest level of accuracy, which is 97.33% for each letter and 68.33% for the whole.

Keywords— *AlexNet; Convolutional Neural Network; Indonesian Sign Language; VGG-16.*

I. PENDAHULUAN

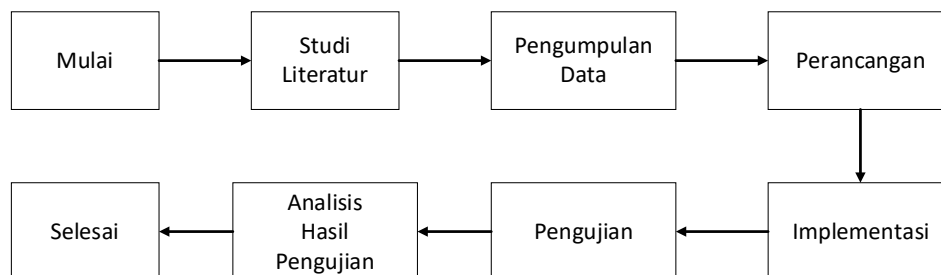
Bahasa isyarat merupakan bahasa komunikasi yang digunakan untuk percakapan sehari-hari oleh penyandang tunarungu dan tuna wicara [1]. Bagi orang yang bukan tuna rungu dan tuna wicara, bahasa isyarat adalah hal yang tidak asing lagi. Bahasa Isyarat Indonesia merupakan simbol kata yang aslinya diambil dari gerak tubuh yang disampaikan oleh anak tuna rungu. Pada tahun 1978, SLB B Zinnia memimpin dalam mengadopsi sistem bahasa isyarat di Indonesia [2]. Penelitian yang menggunakan bahasa isyarat telah dilakukan pada penelitian [3]. Bahasa isyarat yang digunakan pada penelitian tersebut yaitu *American Sign Language (ASL)*. Penelitian lain yang menggunakan ASL juga dilakukan oleh [4] [5] dan [6]. Penelitian bahasa isyarat lain yang juga menggunakan Bahasa Isyarat Indonesia sudah dilakukan [2]. Pada penelitian [7]. DeepCNN dan BiLSTM digunakan untuk mengenali Bahasa Isyarat Indonesia. Bahasa Isyarat juga menjadi objek dalam penelitian [8] [9] dan [10].

Untuk melakukan klasifikasi menggunakan media citra dapat menggunakan *Convolutional Neural Network (CNN)*. Selain sebagai klasifikasi, CNN bisa juga digunakan untuk melakukan ekstraksi fitur. CNN memiliki banyak arsitektur diantaranya yaitu ResNet, AlexNet, VGG, LeNet dan MobileNet. Pada penelitian [7], ResNet digunakan sebagai ekstraksi fitur dengan menggunakan BiLSTM sebagai pengklasifikasi. Pada penelitian [11] arsitektur AlexNet dan LeNet digunakan untuk melakukan klasifikasi ASL dengan hasil yang tidak jauh berbeda yaitu AlexNet 91,618% dan LeNet 92,468%. Penelitian [12] menggunakan AlexNet untuk klasifikasi diabetes retinopati dengan hasil 96%. Penelitian [13] membandingkan AlexNet dan LeNet untuk pengenalan gestur tangan didapat hasil dengan perbedaan kecil yaitu AlexNet 99,45% dan LeNet 99,49%. Arsitektur VGG digunakan pada penelitian [14] untuk melakukan klasifikasi citra dengan hasil 95,4%. VGG juga digunakan dalam penelitian [15] digunakan untuk klasifikasi salak dengan hasil 95,83%.

Penelitian ini mengangkat permasalahan untuk melakukan klasifikasi bahasa isyarat Indonesia menggunakan CNN dengan arsitektur Adam dan VGG-16. Dari klasifikasi menggunakan CNN dapat diketahui arsitektur mana yang memberikan hasil terbaik. Pada penelitian sebelumnya, perbandingan arsitektur AlexNet dan VGG-16 belum dilakukan sehingga penelitian ini dilakukan untuk mengetahui perbandingan arsitektur AlexNet dan VGG-16.

II. METODE PENELITIAN

Pada Gambar 1 merupakan tahapan yang dilakukan untuk klasifikasi Sistem Isyarat Bahasa Indonesia menggunakan CNN dengan arsitektur VGG-16 dan AlexNet.



Gambar 1. Metode Penelitian

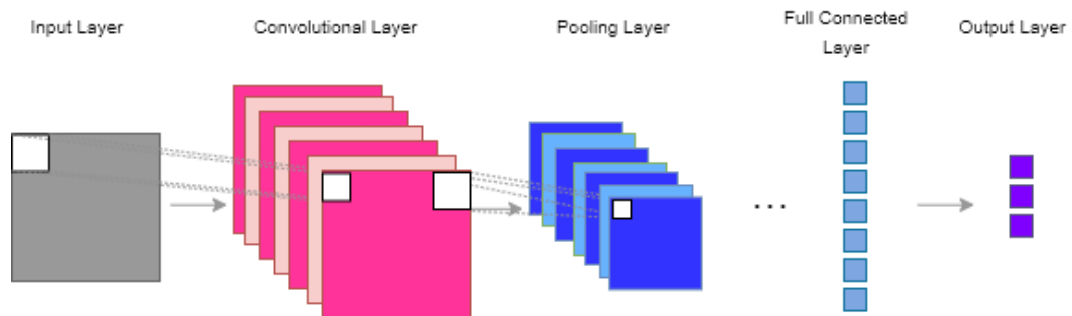
A. Studi Literatur

Tahap ini diawali dengan melakukan pencarian literatur berupa buku dan jurnal yang berkaitan dengan klasifikasi SIBI dengan menggunakan metode CNN

1) *Sistem Isyarat Bahasa Indonesia (SIBI)*: Bahasa Isyarat Indonesia adalah simbol kata yang aslinya diambil dari gerakan tubuh yang disampaikan oleh anak tunarungu. Sistem isyarat ketika SLB B Zinnia merintis pendekatan bahasa isyarat pertama kali pada tahun 1978 di Indonesia. Kemudian, pada tahun 1981 SLB B Karya Mulya yang berada di Surabaya menggunakan SIBI. Ada berbagai bahasa isyarat di dunia, rata-rata setiap negara memiliki bahasa isyarat tersendiri [2].

2) *Convolutional Neural Network (CNN)*: *Convolutional Neural Network* merupakan salah satu dari *Deep Learning* yang digunakan untuk melakukan klasifikasi. Klasifikasi ini menggunakan *convolutional layer* yang bertujuan untuk menghitung konvolusi antara input dengan filter. Metode CNN dapat mengambil dan mengekstrak fitur dengan cara *unsupervised*. Hal

ini yang menjadikan metode CNN berbeda dengan metode *machine learning* yang memerlukan fitur yang ditentukan sebelumnya [16]. Pada Gambar 2 berikut merupakan model *Convolutional Neural Network*.



Gambar 2. Convolutional Neural Network [17]

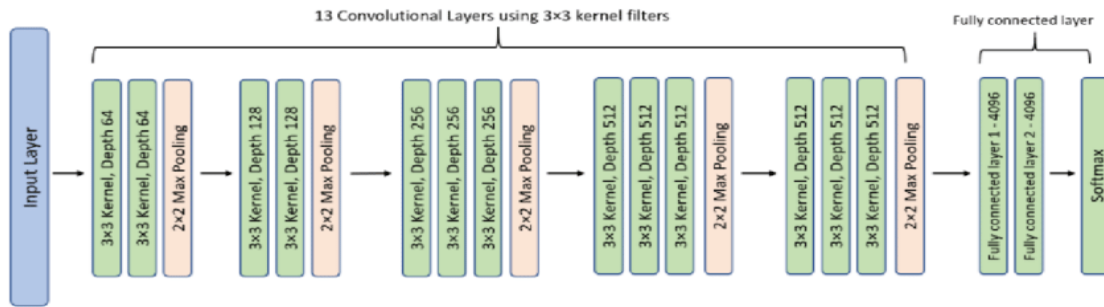
Convolutional Layer merupakan *layer* yang digunakan untuk melakukan perhitungan keluaran dari neuron yang terhubung ke *local region* dari inputan citra. Setiap neuron menggunakan filter secara bergeser untuk perkalian antara *region* kecil yang terhubung ke citra input dan melakukan operasi konvolusi pada output *layer* [16]. *Convolutional Layer* termasuk blok utama dalam *Convolutional Neural Network* yang terdiri dari *filter* yang dipelajari secara acak untuk melakukan operasi konvolusi untuk tujuan ekstraksi fitur untuk mempelajari representasi fitur dari input *layer*. Konvolusi pada data citra bertujuan untuk melakukan ekstraksi fitur dari input citra. Hasil dari konvolusi yaitu berupa transformasi linear dari data yang bersesuaian dengan informasi spasial yang tersedia pada data input.

Pooling Layer adalah operasi *down-sampling* dalam dimensi spasial (lebar, tinggi). *Pooling Layer* adalah *layer* yang memperkecil dimensi *layer* sebelumnya dari $w * w * f$ menjadi $x * x * f$ dimana nilai $x < w$. Dimana w adalah ukuran *width* dari input sebelumnya, sedangkan x adalah ukuran filter. Notasi f adalah operasi filter. Operasi *pooling layer* bisa dikatakan hampir mirip dengan operasi konvolusi, di tahapan ini tidak ada proses *learning* dari *pooling layer*, kemudian tidak ada *filter* yang digunakan untuk melakukan proses pada *pooling layer*, proses konvolusi melakukan transfer *window* ke input. Pada *pooling layer*, berbagai cara dapat dilakukan seperti *max pooling*, *average pooling* dan *sum pooling*. Proses *Max pooling* bekerja dengan cara menentukan nilai terbesar pada *windows* [16].

Fully Connected Layer yang terhubung pada *layer* yang terdiri dari beberapa *node* [16]. Pada *Fully Connected Layer* terjadi tahapan menggabungkan fitur-fitur yang telah diekstraksi ke dalam *class*. Cara kerjanya yaitu dengan mengacak nilai bobot dan bias kemudian dilakukan perkalian. Ketika nilainya masih jauh, diperbarui dan diulangi sampai memperoleh nilai bobot dan bias yang terbaik untuk *class*. Proses ini bisa membutuhkan waktu yang lama tergantung pada banyaknya fitur yang dihasilkan [18].

3) *Preprocessing*: *Preprocessing* adalah sebuah proses yang dilakukan untuk membantu meningkatkan dan memperbaiki kualitas gambar. *Preprocessing* bertujuan untuk mengubah *pixel* gambar sesuai dengan tujuan *preprocessing* itu sendiri sehingga hasil *preprocessing* nanti dapat digunakan pada proses selanjutnya [19]. *Grayscale* adalah citra dengan memiliki warna skala abu-abu yang digunakan untuk membuat pengambilan informasi lebih mudah daripada gambar berwarna [20]. *Augmentasi* adalah proses perubahan atau mengubah suatu gambar. Komputer dapat mendeteksi gambar yang diubah adalah gambar yang berbeda, namun sebenarnya manusia tetap dapat mengetahui gambar tersebut adalah gambar yang sama. *Augmentasi* dapat berupa *rotate*, *flip*, *crop*, *blur* serta memberikan *noise* pada data gambar [21]. *Resize* adalah proses menyesuaikan ukuran gambar yang tadinya kecil menjadi lebih besar atau sebaliknya dari besar menjadi lebih kecil dari ukuran gambar sebelumnya yang bertujuan untuk menyesuaikan dengan ukuran yang seharusnya digunakan [22].

4) *VGG-16*: *VGG-16* adalah arsitektur CNN yang dikembangkan oleh Simonyan dan Zisserman pada tahun 2014. Dari penelitian tersebut didapatkan pengaruh kedalaman *layer* terhadap nilai error pada CNN. Penelitian tersebut kemudian diikutsertakan pada perlombaan *ImageNet Large Scale Visual Recognition Competition (ILSVRC)* tahun 2014. Mereka berhasil menjadi pemenang kedua [16]. Pada Gambar 3 terdapat arsitektur *VGG-16*.



Gambar 3. Arsitektur VGG-16 [15]

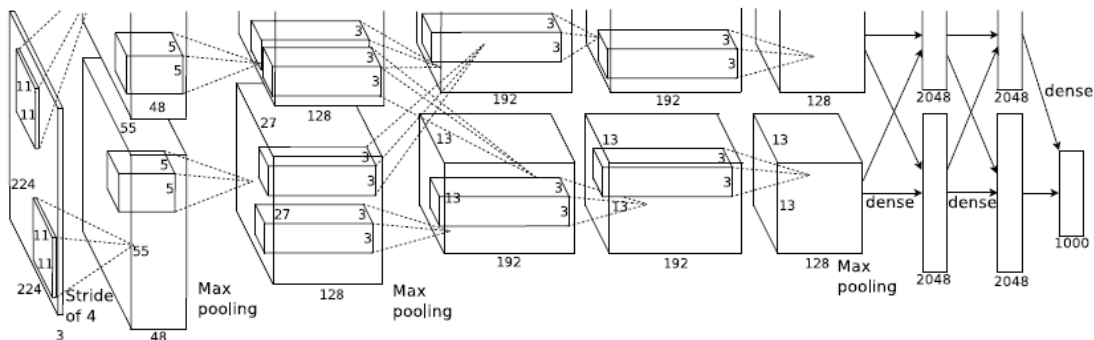
Lapisan konvolusi pertama dan kedua terdiri dari 64 *filter kernel* fitur dan ukuran *filter*-nya adalah 3 x 3. Ketika gambar input (gambar RGB dengan kedalaman 3) dilewatkan ke lapisan konvolusi pertama dan kedua, dimensi berubah menjadi 224 x 224 x 64. Kemudian *output* yang dihasilkan dilewatkan ke lapisan *max pooling*.

Lapisan konvolusi ketiga dan keempat adalah 124 fitur *filter kernel* dan ukuran filter adalah 3 x 3. Setelah kedua *layer* maksimal dengan langkah 2 maka *output* yang dihasilkan akan dikurangi menjadi 56 x 56 x 128.

Lapisan kelima, keenam, dan ketujuh merupakan lapisan konvolusi yang memiliki ukuran *kernel* 3 x 3. Ketiganya menggunakan 256 *fitur kernel*. Lapisan ini diikuti oleh penyatuan maksimal lapisan.

Lapisan kedelapan hingga tiga belas adalah dua *set* lapisan konvolusi dengan ukuran *kernel* 3 x 3. Semua set lapisan konvolusi ini memiliki 512 filter kernel. Setelah lapisan ini adalah lapisan penyatuan maksimal.

Lapisan Empat belas dan lima belas lapisan yang terhubung sepenuhnya lapisan tersembunyi 4096 unit diikuti oleh *softmax* lapisan keluaran (lapisan 16) sebanyak 1000 unit.



Gambar 4. Arsitektur AlexNet [23]

5) *Alexnet*: Alexnet adalah arsitektur CNN yang menggunakan model *convolution* untuk *transfer learning*. Ada lima *layer Convolution matrix* dan tiga *layer fully connected* yang terhubung pada arsitektur AlexNet. *Convolution* pertama memiliki 11 x 11 x 3 ukuran filter [18]. Arsitektur AlexNet dapat dilihat pada Gambar 4.

Convolution pertama memiliki 11 x 11 x 3 ukuran filter. Ukuran filter diasumsikan dalam volume balok, maka $x = 11, y = 11, z = 3$. Volume tersebut merupakan hasil dari ukuran gambar 227 x 227 x 3. Ukuran 96 filter pada *layer* pertama dengan ukurannya adalah 55 x 55 x 96. Ukuran ini diperoleh dari hasil dari *convolution* dan *max pooling* pada tahapan sebelumnya. Pada *layer* kedua ukuran yang dimiliki 55 x 55 x 96 dan akan menggunakan ukuran kernel 3 x 3 x 27 untuk dieliminasi (*Max pooling*). Pada Gambar 2.3 merupakan bentuk arsitektur AlexNet. Hasil dari tahapan ini adalah terbentuknya *layer* baru dengan 96 filter. Pada *layer* kedua ukuran baru yang dimiliki, yaitu 27 x 27 x 256 dengan *max pooling* matriks 3 x 3 x 26 dan 384 filter. *Layer* ketiga memiliki ukuran 13 x 13 x 384 dengan jumlah 384 filter dan sama seperti *layer* keempat. Sedangkan *layer* kelima adalah 6 x 6 x 256 untuk diekstraksi agar bisa memulai penggabungan berbagai fungsi

6) *Optimizer*: *Optimizer* adalah fungsi yang berguna untuk mempercepat proses pembelajaran dalam menemukan bobot optimal, meminimalkan loss, dan memaksimalkan akurasi. Masing-masing algoritma *optimizer* menggunakan nilai *learning rate* beragam untuk menentukan performa belajar secara cepat dan lambat [24]. *Optimizer* yang digunakan pada tahapan ini yaitu : *Optimizer* Adam dan SGD.

Adam merupakan bentuk kombinasi dari RMSprop dan momentum yang merupakan hasil dari metode SGD berdasarkan estimasi adaptif momen orde pertama dan kedua. Adam mencatat nilai rata-rata dari gradien proses sebelumnya secara eksponensial seperti RMSprop. Formula perhitungan *optimizer* Adam dapat dilihat pada persamaan (1).

$$\theta_{t+1} = \theta_t - \frac{\eta}{\sqrt{\hat{v}_t + \epsilon}} \cdot \hat{m}_t \quad (1)$$

dengan $\theta_{(t+1)}$ adalah parameter untuk hasil baru, θ_t adalah parameter untuk hasil lama, η adalah *learning rate*, m_t adalah gradien kuadrat momen orde pertama, v_t adalah gradien kuadrat momen orde kedua, dan ϵ merupakan skalar kecil yang bertujuan untuk mencegah pembagian dengan nilai nol [24].

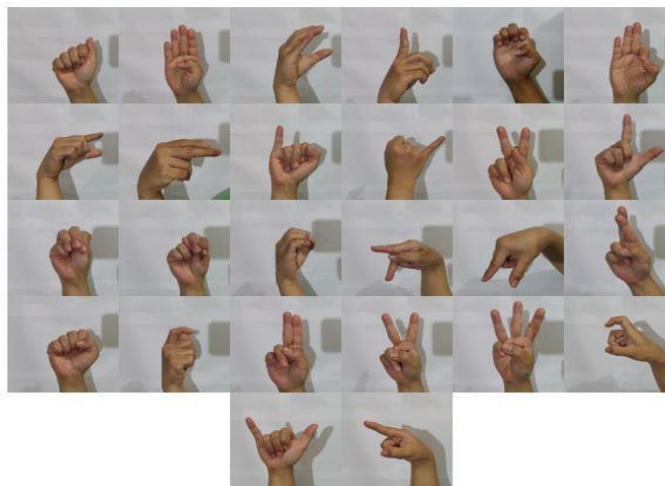
SGD merupakan variasi dari optimasi *gradient descent* yang memperbarui parameter dari data yang telah dilatih. Saat memperbarui parameter, SGD tidak melakukan perulangan sehingga kinerja lebih cepat untuk dataset berjumlah besar. Proses pembaruan parameter pada SGD dapat didefinisikan pada persamaan (2).

$$\theta = \theta - \eta \cdot \nabla_{\theta} J(\theta; \chi^{(i)}; \gamma^{(i)}) \quad (2)$$

dengan θ adalah parameter hasil baru, η adalah *learning rate*, serta $x^{(i)}$ dan $y^{(i)}$ adalah data yang sedang dalam proses pelatihan [19].

B. Pengumpulan Data

Di tahap ini dilakukan pengumpulan data yang berupa dataset *Datasets SIBI Sign Language Alphabets* yang berasal dari [25] yang merupakan dataset bersifat publik. Dataset dengan pembagian data untuk setiap huruf dari huruf A sampai dengan huruf Z adalah 320 data test, 1600 data train dan 320 data validasi. Contoh Gambar huruf SIBI dapat dilihat pada Gambar 5.



Gambar 5. Dataset SIBI [25]

C. Perancangan

Pada tahap ini diperlukan perancangan dan sistem penelitian untuk melakukan penelitian, yaitu penggunaan metode CNN untuk mengklasifikasikan SIBI berdasarkan citra gambar. Dimulai dari *preprocessing* citra dengan menggunakan *grayscale*, *resize* dan *augmentasi* dengan fitur yang diekstraksi kemudian dilakukan klasifikasi menggunakan CNN dan arsitektur yang dipakai adalah arsitektur *VGG-16* dan *AlexNet* dengan skenario pengujian menggunakan *optimizer* Adam dan SGD. Pada Gambar 6 dapat dilihat skema perancangan

D. Implementasi

Tahapan ini merupakan implementasi dari sistem yang telah dirancang menggunakan data yang terdapat pada sistem untuk mengenali dan melakukan klasifikasi terhadap data *train* yang telah dikumpulkan sebelumnya. Tahap ini merupakan sistem yang di test agar dapat mengklasifikasikan SIBI menggunakan metode *CNN* dengan arsitektur *VGG-16* dan *AlexNet* yang dilakukan melalui *optimizer* Adam dan SGD.

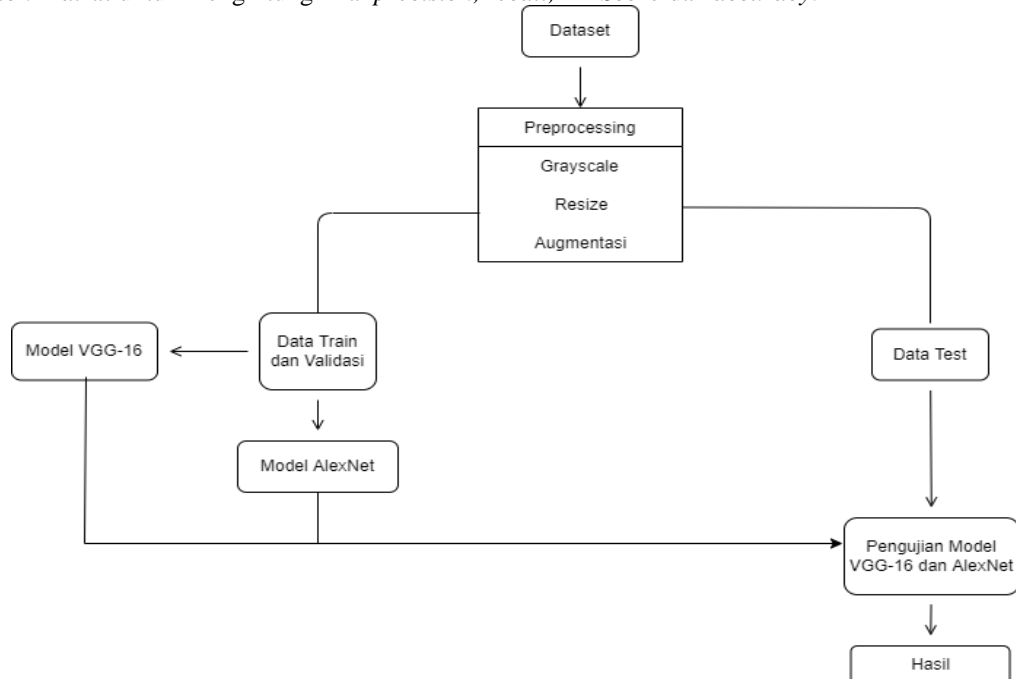
E. Pengujian

Pada tahap ini, sistem akan menguji data test yang diperoleh dan diimplementasikan pada sistem yang terdapat 4 skenario pada arsitektur *VGG-16* dengan menggunakan *optimizer* Adam dan SGD serta pada arsitektur *AlexNet* juga

menggunakan *optimizer* Adam dan SGD, hasil dari klasifikasi akan dicatat untuk mendapatkan *accuracy*, *precision*, *recall* dan *F1-Score* pada tahap selanjutnya.

F. Analisis Hasil Pengujian

Setelah dilakukan pengujian, langkah selanjutnya yaitu melakukan analisis hasil pengujian. Dari hasil pengujian kemudian dihitung untuk mendapatkan tingkat keberhasilan dari metode yang digunakan. Cara yang dilakukan yaitu dengan *Confusion Matrix* untuk menghitung nilai *precision*, *recall*, *F1-Score* dan *accuracy*.

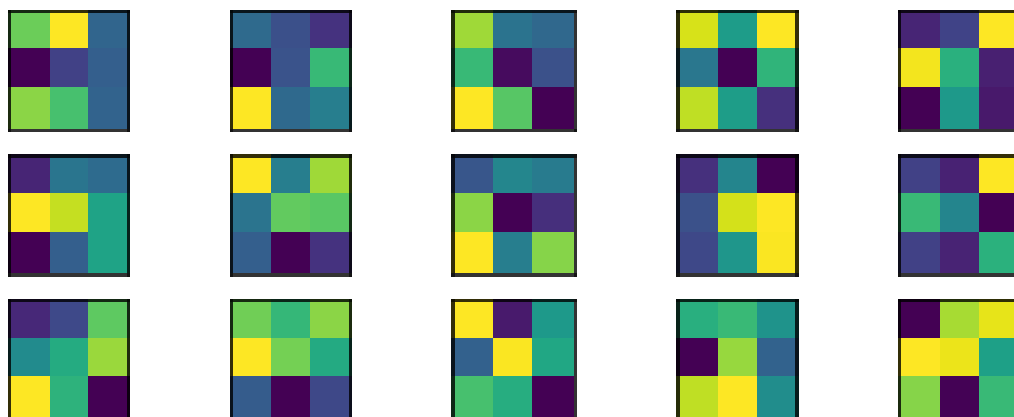


Gambar 6. Skema Perancangan

III. HASIL DAN PEMBAHASAN

Berdasarkan hasil train pada setiap skenario, dapat dilihat bahwa perbandingan dari hasil klasifikasi dengan menggunakan *augmentasi* dan *grayscale* serta penggunaan *optimizer* seperti Adam dan SGD memiliki pengaruh pada tingkat *accuracy* model.

Operasi konvolusi *AlexNet* dilakukan dengan inialisasi awal kernel 3x3 yang dapat dilihat pada Gambar 7



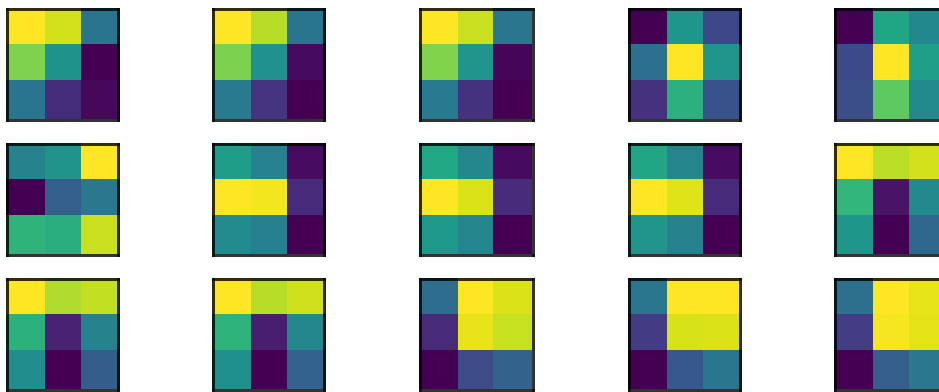
Gambar 7. Kernel 3x3 konvolusi AlexNet

Hasil dari operasi konvolusi *AlexNet* ini adalah citra yang dapat disebut sebagai *feature maps* yang dapat dilihat pada Gambar 8



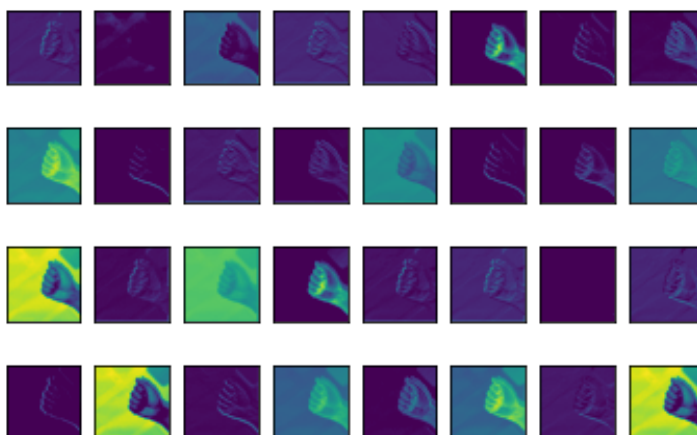
Gambar 8. Feature Maps Konvolusi AlexNet

Operasi konvolusi dilakukan dengan inisialisasi awal kernel 3x3 yang dapat dilihat pada Gambar 9



Gambar 9. Kernel 3x3 konvolusi VGG-16

Hasil dari operasi konvolusi pada VGG-16 ini adalah citra yang dapat disebut sebagai *feature maps* yang dapat dilihat pada Gambar 10

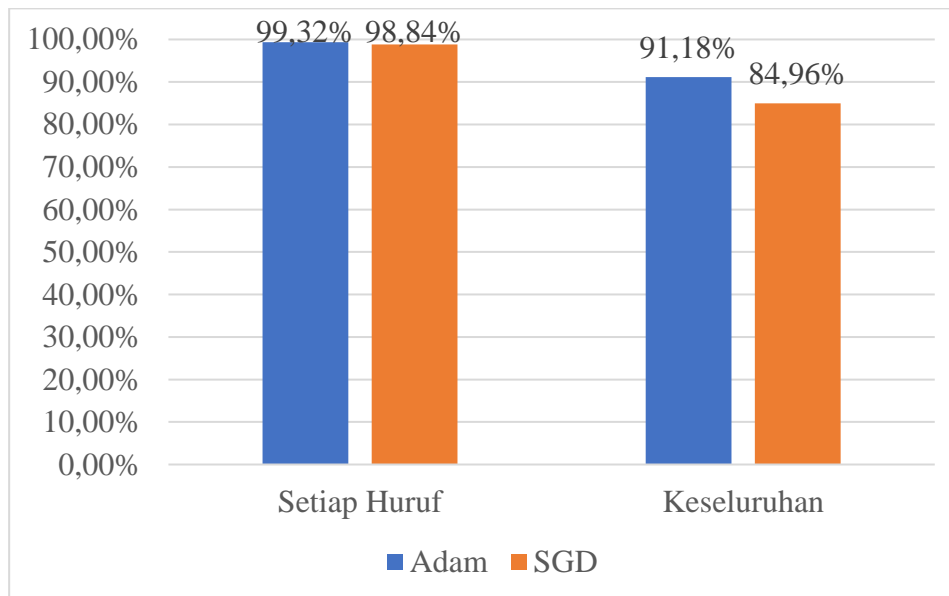


Gambar 10. Feature Maps Konvolusi VGG-16

Dari hasil pengujian VGG-16, dapat dilihat bahwa hasil dari klasifikasi yang menggunakan VGG-16 dengan *optimizer* Adam mendapatkan tingkat *accuracy* tertinggi yaitu sebesar 99,32% pada setiap huruf dan 91,18% pada keseluruhan. Sedangkan hasil dari klasifikasi yang menggunakan VGG-16 dengan *optimizer* SGD mendapatkan tingkat *accuracy* terendah yaitu sebesar 98,84% pada setiap huruf dan 84,96% pada keseluruhan. Tabel 1 merupakan Perbandingan tingkat akurasi VGG-16 untuk setiap skenario.

TABEL 1
PERBANDINGAN ACCURACY VGG-16

Skenario	Accuracy setiap huruf	Accuracy Keseluruhan
VGG-16 dengan <i>Optimizer</i> Adam	99,32%	91,18%
VGG-16 dengan <i>Optimizer</i> SGD	98,84%	84,96%



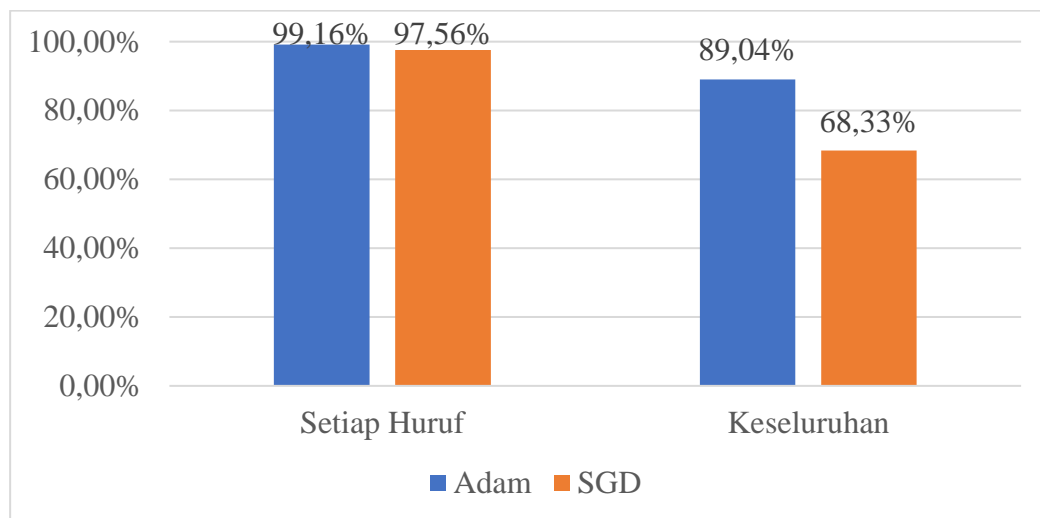
Gambar 11. Grafik Perbandingan Skenario VGG-16

Dari setiap skenario pada pengujian VGG-16 yang dilakukan, terdapat perbedaan hasil akurasi dari setiap skenario pengujian yang telah dilakukan yang dapat dilihat pada Gambar 11 yang menunjukkan grafik perbandingan skenario VGG-16.

Namun pada hasil pengujian AlexNet, dapat dilihat bahwa hasil dari klasifikasi yang menggunakan AlexNet dengan *optimizer* Adam mendapatkan tingkat *accuracy* tertinggi yaitu sebesar 99,16% pada setiap huruf dan 89,04% pada keseluruhan. Sedangkan hasil dari klasifikasi yang menggunakan AlexNet dengan *optimizer* SGD mendapatkan tingkat *accuracy* terendah yaitu sebesar 97,56% pada setiap huruf dan 68,33% pada keseluruhan. Pada Tabel 2 dapat dilihat Perbandingan tingkat akurasi AlexNet pada masing-masing skenario pengujian.

TABEL 2
PERBANDINGAN TINGKAT AKURASI PENGUJIAN ALEXNET

Skenario	Accuracy setiap huruf	Accuracy Keseluruhan
VGG-16 dengan <i>Optimizer</i> Adam	99,16%	89,04%
VGG-16 dengan <i>Optimizer</i> SGD	97,56%	68,33%



Gambar 12. Grafik Perbandingan Skenario AlexNet

Dari setiap skenario pada pengujian AlexNet yang dilakukan, terdapat perbedaan hasil akurasi dari setiap skenario pengujian yang telah dilakukan yang dapat dilihat pada Gambar 12 yang menunjukkan grafik perbandingan skenario AlexNet.

Berdasarkan hasil dari perbandingan pada seluruh skenario, maka terdapat hasil terbaik pada seluruh skenario adalah VGG-16 dengan Optimizer Adam dengan *accuracy* setiap hurufnya adalah 99,32% dan *accuracy* keseluruhannya adalah 91,18% yang dapat dilihat pada Tabel 3. VGG-16 memiliki jumlah *convolutional layer* dibandingkan dengan AlexNet sehingga pada saat melakukan klasifikasi, *accuracy* yang diperoleh pada VGG-16 lebih baik dibandingkan dengan AlexNet. Hal ini dapat dilihat baik *accuracy* keseluruhan maupun *accuracy* setiap huruf. Optimizer Adam baik untuk arsitektur VGG-16 maupun AlexNet lebih baik dibandingkan dengan SGD.

TABEL 3
HASIL SELURUH SKENARIO

Skenario	<i>Accuracy</i> setiap huruf	<i>Accuracy</i> Keseluruhan
VGG-16 dengan Optimizer Adam	99,32%	91,18%
AlexNet dengan Optimizer Adam	99,16%	89,04%
VGG-16 dengan Optimizer SGD	98,84%	84,96%
AlexNet dengan Optimizer SGD	97,56%	68,33%

IV. SIMPULAN

Dari penelitian ini diketahui hasil perbandingan hasil klasifikasi menggunakan CNN dengan arsitektur VGG-16 dan AlexNet. Arsitektur yang memberikan hasil terbaik yaitu VGG-16. *Accuracy* setiap huruf pada arsitektur VGG-16 menggunakan optimizer Adam memperoleh hasil yang terbaik dengan *accuracy* 99,32% dan yang terendah adalah arsitektur AlexNet dengan optimizer SGD yaitu 97,56%. Jika dilihat dari optimizer yang digunakan, optimizer Adam lebih baik dibandingkan dengan optimizer SGD. Jika dilihat dari arsitektur yang digunakan baik VGG-16 maupun AlexNet *accuracy* per huruf yang dihasilkan memiliki perbedaan hasil yang kecil. Secara keseluruhan VGG-16 dengan optimizer Adam adalah yang terbaik sedangkan yang terendah adalah AlexNet dengan optimizer SGD.

DAFTAR PUSTAKA

- [1] S. Shivashankara and S. Srinath, "American Sign Language Recognition System: An Optimal Approach," *Int. J. Image, Graph. Signal Process.*, vol. 10, no. 8, p. 18–30, 2018.
- [2] R. S. Legowo, "Klasifikasi gerakan tangan SIBI (sistem isyarat bahasa indonesia) menggunakan leap motion dengan metode klasifikasi naive bayes," November 2017. [Online]. Available: https://repository.its.ac.id/46150/1/2913100028-Undergraduate_Thesis.pdf.
- [3] I. Fareza, R. Busdin, M. E. A. Rivan and H. Irsyad, "Pengenalan Alfabet Bahasa Isyarat Amerika Menggunakan Edge Oriented Histogram dan Image Matching," *J. Tek. Inform. dan Sist. Inf.*, vol. 4, p. 82–92, 2018.
- [4] Y. Pratama, E. Marbun, Y. Parapat and A. Manullang, "Deep convolutional neural network for hand sign language recognition using model e," *Bull. Electr. Eng. Informatics*, vol. 9, no. 5, p. 1873–1881, 2020.
- [5] M. E. A. Rivan, H. Irsyad, K. Kevin and A. T. Narta, "Pengenalan Alfabet American Sign Language Menggunakan K-Nearest Neighbors Dengan Ekstraksi Fitur Histogram Of Oriented Gradients," *J. Tek. Inform. dan Sist. Inf.*, vol. 5, no. 3, 2020.
- [6] M. E. A. Rivan and M. T. Noviardy, "Klasifikasi American Sign Language Menggunakan Ekstraksi Fitur Histogram of Oriented Gradients dan Jaringan Syaraf Tiruan," *J. Tek. Inform. dan Sist. Inf.*, vol. 6, no. 3, p. 442–451, 2020.
- [7] N. F. P. Setyono and E. Rakun, "Recognizing Word Gesture in Sign System for Indonesian Language (SIBI) Sentences Using DeepCNN and BiLSTM," *International Conference on Advanced Computer Science and Information Systems (ICACSIS)*, p. 199–204, 2019.
- [8] S. Nugraha, D. H. Murti and W. N. Khotimah, "Penggunaan Dual Sensor Leap Motion Controller untuk Sistem Isyarat Bahasa Indonesia (SIBI)," *INFORMAL Informatics J.*, vol. 4, no. 2, p. 48, 2019.
- [9] A. R. T. H. Ririd, Y. Yunhasnawa and Y. G. Buata, "Sistem Pengenalan Huruf Bahasa Isyarat Menggunakan Adaptive Learning Vector Quantization," *J. Inform. Polinema*, vol. 4, no. 2, p. 145, 2018.
- [10] D. Darmatasia, "Pengenalan Sistem Isyarat Bahasa Indonesia (Sibi) Menggunakan Gradient-Convolutional Neural Network," *J. INSTEK (Informatika Sains dan Teknol.)*, vol. 6, no. 1, p. 56, 2021.
- [11] M. E. A. Rivan and A. G. Riyadi, "Perbandingan Arsitektur LeNet dan AlexNet Pada Metode Convolutional Neural Network Untuk Pengenalan American Sign Language," *J. Komput. Terap.*, vol. 7, no. 1, p. 53–61, 2021.
- [12] T. Shanthi and R. S. Sabeenian, "Modified Alexnet architecture for classification of diabetic retinopathy images," *Comput. Electr. Eng.*, vol. 76, p. 56–64, 2019.
- [13] M. E. A. Rivan and A. Setiawan, "Pengenalan Gestur Angka Pada Tangan Menggunakan Arsitektur AlexNet Dan LeNet Pada Metode Convolutional Neural Network," *Komputika J. Sist. Komput.*, vol. 11, no. 1, p. 19–28, 2022.
- [14] S. Tammina, "Transfer learning using VGG-16 with Deep Convolutional Neural Network for Classifying Images," *Int. J. Sci. Res. Publ.*, vol. 9, no. 10, p. 9420, 2019.
- [15] R. Rismiyati and A. Luthfiarta, "VGG16 Transfer Learning Architecture for Salak Fruit Quality Classification," *Telematika*, vol. 18, no. 1, p. 37, 2021.
- [16] Y. Achmad, R. C. Wihandika and C. Dewi, "Klasifikasi emosi berdasarkan ciri wajah menggunakan convolutional neural network," *J. Pengemb. Teknol. Inf. dan Ilmu Komput.*, vol. 3, no. 11, p. 10595–10604, 2019.
- [17] M. Peng, C. Wang, T. Chen, G. Liu and X. Fu, "Dual temporal scale convolutional neural network for micro-expression recognition," *Front. Psychol.*, vol. 8, p. 1–12, 2017.
- [18] I. A. Sabilla, "Arsitektur Convolutional Neural Network (Cnn) Untuk Klasifikasi Jenis Dan Kesegaran Buah Pada Neraca Buah," 2020. [Online]. Available: https://repository.its.ac.id/73567/1/05111850010020-Master_Thesis.pdf.
- [19] I. C. Buulolo and H. A. Sihombing, "Pengenalan Buah Kopi Berdasarkan Parameter Warna Menggunakan Algoritma Backpropagation Dan Algoritma Support Vector Machine (Svm)," *Seminastika*, vol. 3, no. 1, p. 26–32, 2021.
- [20] T. A. Kurnia, J. Endrasmono and R. Y. Adhitya, "Pengembangan Sistem Identifikasi Alat Pelindung Diri (APD) Menggunakan Metode Convolutional Neural Network (CNN)," *J. Conf. Autom. Eng. Its Appl.*, vol. 1, no. 2809, p. 107–114, 2021.
- [21] I. H. Wafli, J. Jondri and A. Rizal, "Klasifikasi Suara Paru-Paru Menggunakan Convolutional Neural Network (CNN)," *e-Proceeding of Engineering*, vol. 8, no. 2, p. 3218–3223, 2021.
- [22] A. Hibatullah and I. Maliki, "Penerapan Metode Convolutional Neural Network Pada Pengenalan Pola Citra Sandi Rumput," 2019. [Online]. Available: <https://elibrary.unikom.ac.id/id/eprint/1529/>.
- [23] A. Krizhevsky, I. Sutskever and G. E. Hinton, "ImageNet Classification with Deep Convolutional Neural Networks," *Advances in Neural Information Processing Systems*, vol. 25, pp. 1–9, 2012.
- [24] N. D. Miranda, L. Novamizanti and S. Rizal, "Convolutional Neural Network Pada Klasifikasi Sidik Jari Menggunakan Resnet-50," *J. Tek. Inform.*, vol. 1, no. 2, p. 61–68, 2020.
- [25] M. L. Afkaar, "Datasets SIBI Sign Language Alphabets," Datasets SIBI Sign Language Alphabets, 2020. [Online]. Available: <https://www.kaggle.com/datasets/mlanangafkaar/datasets-lemlitbang-sibi-alphabets>. [Accessed 25 March 2022].