

Penerapan VGG16 untuk Klasifikasi Pneumonia pada Citra X-ray

Ferani Christy^{#1}, Ken Ratri Retno Wardani, S.Kom., M.T.^{*2}

[#]Program Studi Informatika, Institut Teknologi Harapan Bangsa
Jalan Dipatiukur No. 80-84, Bandung, Indonesia 40132

¹christy.ferani@gmail.com

²ken_ratri@ithb.ac.id

Abstract— Pneumonia is a condition where a person experiences an infection that occurs in the air sacs of the lungs. Pneumonia is a dangerous disease and can cause death even in children. Therefore, we need a system that can classify pneumonia quickly and accurately so that pneumonia sufferers can immediately get the right treatment. This study will use the Convolutional Neural Network (CNN) with VGG16 architecture, dropout, and image enhancement Contrast Limited Adaptive Histogram Equalization (CLAHE) in classifying pneumonia. This research uses a dataset in the form of X-ray images called Chest X-ray Images (Pneumonia), which are obtained free of charge and can be used by the public from the Kaggle website. At the beginning of the study, all images will be preprocessed by resizing the image to a size of 256x256 pixels then the image contrast will be increased with image enhancement CLAHE. The VGG16 test is carried out by combining the batch size, learning rate, epoch, and rate values to find the best combination that produces the highest accuracy. Based on the tests carried out, the highest accuracy was obtained, which was 97.60% using CLAHE with a combination of a batch size of 5, the learning rate of 0.00001, the epoch of 20, and the rate of 0.4.

Keywords— Pneumonia, X-ray images, dropout, Convolutional Neural Network (CNN), VGG16, Contrast Limited Adaptive Histogram equalization (CLAHE).

Abstrak— Pneumonia (paru-paru basah) adalah kondisi dimana seseorang mengalami infeksi yang terjadi pada kantung udara paru-paru. Pneumonia merupakan salah satu penyakit yang membahayakan dan dapat menyebabkan kematian bahkan pada anak-anak. Oleh karena itu, diperlukan sistem yang dapat mengklasifikasikan pneumonia secara cepat dan akurat agar penderita pneumonia dapat segera mendapat penanganan yang tepat. Penelitian ini akan menggunakan metode Convolutional Neural Network (CNN) dengan arsitektur VGG16, dropout serta image enhancement Contrast Limited Adaptive Histogram Equalization (CLAHE) dalam melakukan klasifikasi pneumonia. Pada penelitian ini digunakan dataset berupa citra X-ray yang bernama Chest X-ray Images (Pneumonia) yang didapatkan secara gratis dan boleh dipakai untuk umum dari situs Kaggle. Pada awal penelitian, seluruh citra akan dilakukan *preprocessing* dengan cara melakukan *resize* citra menjadi ukuran 256x256 piksel kemudian kontras citra akan ditingkatkan dengan *image enhancement* CLAHE. Pengujian VGG16 dilakukan dengan mengkombinasikan nilai *batch size*, *learning rate*, *epoch* dan *rate* untuk mencari kombinasi terbaik yang menghasilkan akurasi tertinggi. Berdasarkan pengujian yang dilakukan didapatkan akurasi tertinggi yaitu sebesar 97.60% menggunakan CLAHE

dengan kombinasi *batch size* bernilai 5, *learning rate* bernilai 0.00001, *epoch* bernilai 20 dan *rate* bernilai 0.4.

Kata Kunci— Pneumonia, citra X-ray, dropout, Convolutional Neural Network (CNN), VGG16, Contrast Limited Adaptive Histogram equalization (CLAHE).

I. PENDAHULUAN

Pneumonia (paru-paru basah) adalah kondisi dimana seseorang mengalami infeksi yang terjadi pada kantung udara paru-paru. Infeksi yang diakibatkan dari pneumonia dapat terjadi pada salah satu sisi paru-paru maupun keduanya sehingga membuat kantung udara terisi oleh cairan. Infeksi virus, bakteri maupun jamur adalah penyebab utama pneumonia. Pneumonia merupakan salah satu penyakit yang membahayakan dan dapat menyebabkan kematian bahkan pada anak-anak. Menurut Badan Kesehatan Dunia (WHO) diperkirakan bahwa terdapat sekitar 15% kematian anak-anak berusia dibawah 5 tahun disebabkan oleh penyakit ini bahkan terdapat lebih dari 800.000 anak meninggal akibat pneumonia pada tahun 2017 [1]. Oleh karena itu, diperlukan sistem yang dapat mengklasifikasikan pneumonia secara cepat dan akurat agar penderita pneumonia dapat segera mendapat penanganan yang tepat.

Di beberapa penelitian sebelumnya mengenai klasifikasi pneumonia, telah menggunakan metode yang berbeda yakni metode *K-Nearest Neighbor* (KNN) mendapatkan akurasi sebesar 79.62% [1], metode *Convolutional Neural Network* (CNN) mendapatkan akurasi sebesar 92.15% dengan arsitektur VGG16, 63.64% dengan arsitektur InceptionV3, 88.62% dengan arsitektur VGG19, 68.27% dengan arsitektur Xception, 62.50% dengan arsitektur DenseNet121, 62.50% dengan arsitektur ResNet50 [2], 92.16% dengan arsitektur VGG16 [3] dan 94% [4] dan metode *Support Vector Machine* (SVM) mendapatkan akurasi sebesar 62.66% [5].

KNN memiliki kelebihan pada waktu pelatihannya yang cepat, meskipun akurasi yang didapat dalam mengklasifikasikan pneumonia masih kurang baik [1]. SVM memiliki kelebihan dimana metode ini dapat dianalisis secara teoritis menggunakan konsep teori pembelajaran komputasi namun akurasi masih kurang baik dalam pengklasikasian pneumonia [5].

CNN merupakan salah satu metode *deep learning* yang ditujukan untuk klasifikasi citra. CNN ini memiliki kelebihan

dimana dapat menentukan dan mengekstrak fitur secara otomatis untuk mengklasifikasikan citra dan memiliki akurasi yang baik untuk kasus klasifikasi pneumonia namun metode ini dapat mengalami kondisi *overfit* dimana merupakan kondisi ketidakmampuan model dalam mengenali data pengujian sebaik data pelatihan. Kondisi ini terjadi karena pengaruh beberapa faktor, seperti kondisi dataset, formulasi metode CNN yang belum optimal dan jumlah *epoch* [3].

Karena itu, pada penelitian ini klasifikasi pneumonia akan menggunakan metode CNN dengan arsitektur VGG16, *dropout* untuk mengurangi *overfit* serta akan menggunakan *image enhancement* untuk memperbaiki kualitas citra.

II. METODOLOGI

A. Histogram Equalization

Histogram equalization merupakan salah satu teknik *image enhancement* yang digunakan untuk mengatur intensitas suatu citra dengan meningkatkan nilai kontras dari sebuah histogram. *Image enhancement* sendiri adalah sebuah proses memanipulasi citra sehingga hasilnya menjadi lebih baik daripada sebelumnya serta untuk meningkatkan pemahaman model terhadap data citra. Dengan *histogram equalization*, tingkat keabuan dalam sebuah citra akan diratakan sehingga bagian dengan kontras rendah dapat menjadi lebih tinggi dan citra menjadi lebih jelas. *Histogram equalization* ini cocok digunakan untuk citra yang terlihat buram atau gelap [6].

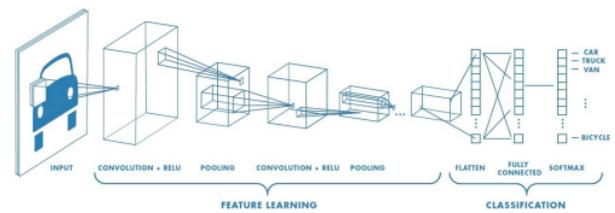
Meskipun dengan *histogram equalization* mampu meningkatkan kontras citra tetapi *histogram equalization* ini bekerja untuk semua piksel dalam citra secara global sehingga dapat meningkatkan *noise* yang ada pada citra. Masalah ini dapat diatasi dengan metode *Contrast Limited Adaptive Histogram Equalization*. Dengan pendekatan CLAHE, kontras sebuah citra dapat dibatasi sehingga dapat mengurangi masalah *noise*. CLAHE akan dibagi menjadi beberapa subcitra sesuai parameter *tile grid size*, lalu pada subcitra akan dilakukan perhitungan *Probability Distribution Function* (PDF). Jika nilai PDF melebihi nilai batas maka akan melakukan pemotongan histogram sesuai parameter *clip limit* dan nilai lebihnya akan didistribusikan secara merata. Lalu menghitung *Cumulative Distribution Function* (CDF) serta melakukan histogram equalization [7]. CLAHE memiliki beberapa parameter, diantaranya :

1. *Clip Limit* merupakan parameter untuk mengatur batas dari kontras citra.
2. *Tile Grid Size* merupakan parameter yang digunakan untuk mengatur ukuran dimensi subcitra.

B. Convolutional Neural Network (CNN)

Convolutional Neural Network (CNN) merupakan salah satu jenis *neural network* yang populer dan digunakan untuk data berdimensi tinggi seperti gambar dan video. CNN ini telah menghasilkan performa yang baik dalam masalah pengenalan dan klasifikasi citra. CNN dapat melatih kumpulan data yang besar dengan masukan citra 2D sebagai masukan serta menggabungkannya dengan *kernel* untuk menghasilkan keluaran yang diinginkan. Pada CNN, terdapat

tiga jenis lapisan dasar yaitu *convolutional layer*, *pooling layer* dan *fully connected layer* [8].



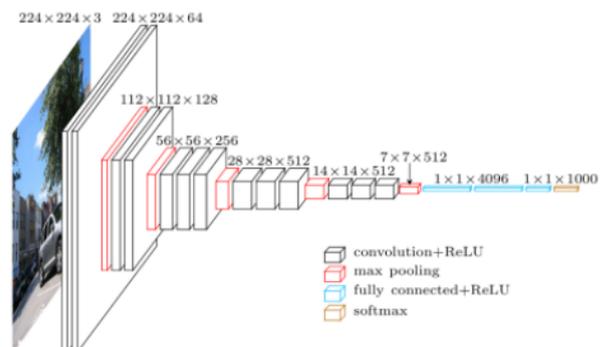
Gambar 1. Convolutional Neural Network [9]

1) VGG16

VGG16 merupakan salah satu arsitektur CNN yang populer karena kesederhanaan modelnya dan menggunakan *kernel* berukuran kecil yang diperkenalkan pada tahun 2014. Arsitektur VGG16 menggunakan 3×3 *kernel* dengan *Max Pooling* dan *stride* bernilai 1 untuk *feature extraction* dan 3 *fully connected layer* pada akhir arsitektur. Dengan ukuran *kernel* tersebut, kedalaman *neural network* dapat ditambahkan sehingga dapat menghasilkan akurasi yang lebih akurat daripada arsitektur lainnya. Setiap lapisan konvolusi akan diikuti dengan fungsi aktivasi ReLU untuk mengurangi kondisi *overfit* [8].

Citra masukan bawaan pada arsitektur VGG16 adalah citra dengan dimensi 224×224 dengan 3 *channel* yaitu R, G dan B. Kemudian, citra ini akan dimasukkan ke dua *convolutional layer* dengan 3×3 *kernel* yang kemudian diikuti dengan fungsi aktivasi ReLU. Proses konvolusi ini akan menggunakan *stride* bernilai 1 dan *padding* bernilai 1 sehingga ukuran keluaran *feature map* akan sama dengan dimensi citra masukan. *Feature map* sebelumnya akan dimasukkan ke *Max Pooling layer* dengan *spatial extent* berukuran 2×2 serta *stride* bernilai 2 sehingga dimensi *feature map* akan berkurang menjadi $112 \times 112 \times 64$. Kemudian *feature map* akan dimasukkan ke *stack* yang serupa, sehingga proses ini akan berulang dan menghasilkan keluaran *feature map* akhir dengan dimensi $7 \times 7 \times 512$ [10].

Arsitektur VGG16 ini merupakan arsitektur yang baik digunakan untuk memecahkan masalah seperti pengenalan dan klasifikasi citra serta untuk deteksi dan lokalisasi citra. Selain itu, VGG16 memiliki kelebihan pada kesederhanaan arsitekturnya dan akurasinya yang baik.



Gambar 2. Arsitektur VGG16 [11]

2) Convolutional Layer

Convolutional Layer merupakan lapisan terpenting dalam CNN yang digunakan untuk mendeteksi fitur seperti tepi, garis, warna dan elemen visual lainnya. Semakin banyak *kernel* yang diberikan pada CNN, maka semakin banyak fitur yang dapat dideteksi. *Kernel* sendiri adalah sebuah objek berbentuk persegi yang digunakan untuk memindai sebuah citra. Pada *convolutional layer* terdapat parameter yang digunakan untuk menentukan keluaran dari *convolutional layer*, diantaranya [12]:

- Jumlah *kernel* akan memengaruhi banyaknya *feature map* yang dihasilkan pada *convolutional layer*.
- Ukuran *kernel* akan memengaruhi dimensi keluaran yang dihasilkan.
- *Stride* merupakan langkah pergeseran *kernel* secara horizontal pada sebuah proses konvolusi. Semakin kecil *stride*, maka semakin banyak informasi yang didapatkan.
- *Padding* merupakan sebuah piksel bernilai 0 yang ditambahkan pada sisi dari piksel masukan. *Padding* ini berfungsi untuk memanipulasi dimensi keluaran yang dihasilkan.
- Fungsi aktivasi ini digunakan untuk menghasilkan keluaran yang nonlinier

3) Pooling Layer

Sama seperti *convolutional layer*, *pooling layer* merupakan lapisan yang berfungsi untuk mengurangi dimensi. Hal ini untuk mengurangi daya komputasi yang diperlukan untuk memproses data. Selain itu, *pooling layer* berfungsi untuk mengekstrak fitur yang dominan sehingga dapat memproses *training* secara efektif [9]. *Pooling layer* memiliki beberapa parameter, yaitu [12]:

1. *Spatial extent* merupakan luas area yang ingin dilakukan proses pooling.
2. *Stride* merupakan perpindahan area pooling.

Operasi *Pooling* yang banyak digunakan adalah operasi *Max Pooling*. Operasi *Max Pooling* ini adalah proses pemilihan nilai dari suatu area pooling, dimana akan diambil nilai yang paling besar dari area tersebut.

4) Fully Connected Layer

Fully connected layer merupakan lapisan yang terhubung dengan semua neuron di lapisan sebelumnya. Pada lapisan ini, fitur tingkat tinggi akan diubah menjadi skor atau probabilitas kelas untuk menghasilkan kelas keluaran. Pada *fully connected layer* ini, keluaran yang dihasilkan dari *convolutional layer* akan dilakukan proses *flattening*, yaitu proses untuk mengkonversi *feature map* 2 dimensi menjadi larik 1 dimensi sehingga mempermudah proses klasifikasi. Beberapa arsitektur CNN menggunakan beberapa *fully connected layer* di akhir jaringan [8].

5) Dropout Layer

Dropout Layer adalah sebuah lapisan yang digunakan untuk mencegah *overfit* dengan menghubungkannya ke

convolutional layer atau *fully connected layer*. *Dropout layer* biasanya ditambahkan setelah *fully connected layer* dan sebelum *layer* keluaran. *Dropout* sendiri adalah teknik regularisasi yang bekerja dengan cara menonaktifkan beberapa neuron tertentu secara acak untuk mengurangi jumlah neuron yang saling terhubung. Tindakan ini dilakukan agar mencegah suatu neuron bergantung pada neuron lainnya [13].

Dropout memiliki *hyperparameter* yang disebut *rate* atau *probability*. *Hyperparameter* ini dapat diatur dalam rentang 0 sampai 1. Jika nilai *rate* yang digunakan adalah 0 maka tidak ada neuron yang akan dinonaktifkan, sebaliknya jika nilai *rate* yang digunakan adalah 1 maka seluruh neuron akan dinonaktifkan.

C. Confusion Matrix

Confusion matrix adalah sebuah tabel baris dan kolom yang mewakili prediksi dan hasil aktual dari klasifikasi. *Confusion matrix* dapat digunakan untuk mengevaluasi performa sebuah model atau klasifikasi berdasarkan jawaban yang benar pada sebuah hasil klasifikasi [14].

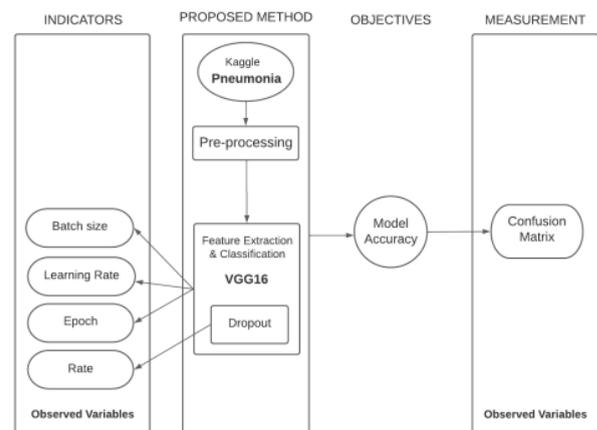
Pada *confusion matrix* terdapat empat komponen sebagai berikut [14]:

1. *True Positive*(TP) merupakan kondisi dimana data yang sebenarnya adalah positif dan prediksinya juga positif.
2. *True Negative*(TN) merupakan kondisi dimana data yang sebenarnya adalah positif tetapi prediksinya negatif.
3. *False Positive*(FP) merupakan kondisi dimana data yang sebenarnya adalah negatif tetapi prediksinya positif.
4. *False Negative*(FN) merupakan kondisi dimana data yang sebenarnya adalah negatif dan prediksinya juga negatif.

D. Perancangan Sistem

1) Kerangka Pemikiran

Kerangka pemikiran sistem klasifikasi pneumonia dapat dilihat pada Gambar 2 berikut.

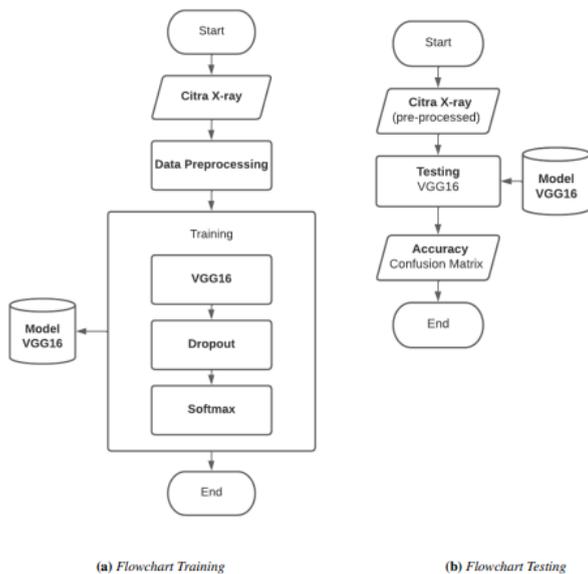


Gambar 3. Kerangka Pemikiran

Berdasarkan Gambar 3, penelitian ini akan dimulai dengan mengambil dataset *X-ray* pneumonia yang didapatkan dari situs Kaggle. Data ini kemudian akan masuk ke tahap *preprocessing* lalu dimasukkan ke model VGG16 untuk ekstraksi fitur dan klasifikasi. Terdapat beberapa variabel indikator yang akan diuji pada penelitian ini, yaitu *batch size*, *learning rate*, *epoch* dan *rate*. Hasil dari klasifikasi akan diukur berdasarkan metode confusion matrix.

2) Flowchart Global

Penelitian ini akan dibagi menjadi 2 proses utama, yaitu proses *training* dan proses *testing*. Proses *training* adalah proses dimana model akan dilatih dengan data yang ada untuk menemukan pola, sedangkan *testing* adalah proses untuk pengukuran keberhasilan suatu model. Gambar 3.2 menunjukkan urutan proses global pada penelitian ini dalam bentuk flowchart.



Gambar 4. Flowchart Global

Berikut adalah uraian proses *training* yang dilakukan pada penelitian ini sesuai Gambar 4(a):

1. Citra masukan merupakan citra *X-ray* dengan 1 channel (grayscale)
2. Citra akan diubah ukurannya menjadi 256x256 piksel.
3. Data akan dibagi menjadi 90% *train* data dan 10% *test* data.
4. Kualitas citra akan ditingkatkan menggunakan algoritma CLAHE. Penelitian ini akan menggunakan *clip limit* bernilai 1 dan *tile grid size* bernilai 8x8.
5. *Training* menggunakan VGG16
6. Terapkan *dropout layer* dengan menguji hyparameter *rate*.
7. Terapkan teknik *Softmax* untuk klasifikasi.
8. Hasil keluaran dari proses *training* adalah model VGG16 yang disimpan di memori

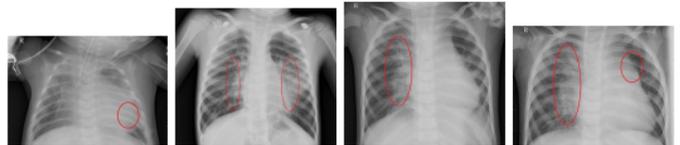
Berikut adalah uraian proses *testing* yang dilakukan pada penelitian ini sesuai Gambar 4(b) :

1. Citra masukan merupakan citra *X-ray* yang telah dilakukan perubahan ukuran menjadi 256x256 serta operasi CLAHE
2. Dari citra tersebut, akan dilakukan tahap *testing* dengan model VGG16 yang didapatkan dari tahap *training*.
3. Setelah didapatkan hasil klasifikasi dari proses *testing*, model akan di evaluasi dengan menggunakan confusion matrix dan classification report.

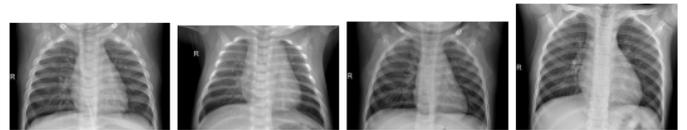
Pada penelitian ini akan membandingkan model dengan menggunakan image enhancement, *dropout layer* dengan model yang tidak menggunakannya.

3) Dataset

Sampel data citra *X-ray* yang digunakan untuk penelitian ini didapatkan dari situs Kaggle yang memuat data citra *X-ray* paru-paru pneumonia dan normal. Dataset sudah diatur ke dalam 3 folder yaitu *train*, *test* dan *val* dan setiap folder berisi subfolder pneumonia dan normal. Terdapat 5856 data citra *X-ray* dengan 1583 citra normal dan 4273 citra pneumonia atau 5216 citra untuk *train*, 624 citra *test* dan 16 citra *val*.



Gambar 5. Citra *X-ray* Pneumonia



Gambar 6. Citra *X-ray* Normal

Pada Gambar 5 dan 6 dapat dilihat bahwa pada citra *X-ray* paru-paru pneumonia, bayangan putih akan terlihat lebih banyak dan meluas daripada citra *X-ray* paru-paru normal. Lingkaran merah pada citra *X-ray* pneumonia menandakan bahwa bagian pada paru-paru tersebut sedang dipenuhi oleh cairan atau nanah akibat infeksi sedangkan paru-paru *X-ray* normal yang dipenuhi oleh oksigen akan menghasilkan citra *X-ray* yang terlihat berwarna hitam. Data citra memiliki ukuran yang berbeda-beda mulai dari rentang piksel 842 × 572 sampai 1823 × 1389.

III. HASIL DAN PEMBAHASAN

A. Hasil Pengujian

Pada penelitian ini, hasil pengujian dengan menggunakan VGG16 untuk klasifikasi pneumonia memperoleh hasil terbaik berdasarkan *preprocessing* yang dilakukan yaitu:

TABEL I
HASIL PENGUJIAN TERBAIK UNTUK KLASIFIKASI CITRA X-RAY

Preprocessing	Batch Size	Learning Rate	Epoch	Rate (Dropout)	Accuracy
Tanpa CLAHE	5	0.00001	20	0.4	97.43%
CLAHE	5	0.00001	20	0.4	97.60%

Berdasarkan Tabel I, dapat disimpulkan bahwa model VGG16 sudah dapat memberikan kinerja yang baik untuk klasifikasi pneumonia yaitu memiliki akurasi sekitar 95% sampai 97%. Model dengan menggunakan *preprocessing* CLAHE mendapatkan akurasi terbaik yaitu 97.60% dengan kombinasi *hyperparameter batch size* bernilai 5, *learning rate* bernilai 0.00001, *epoch* bernilai 20 dan *rate* bernilai 0.4. Sedangkan model tanpa CLAHE mendapatkan akurasi terbaik sebesar 96.43%. Berikut merupakan hasil analisis pengaruh *preprocessing* berdasarkan hasil penelitian yang dilakukan:

1. Model tanpa CLAHE sudah memiliki performa yang baik yaitu sekitar 97.43% dengan model VGG16.
2. Model dengan menggunakan *preprocessing* CLAHE memiliki performa yang terbaik daripada pengujian lainnya yaitu dengan akurasi 97.60%. Penggunaan *preprocessing* CLAHE dapat memperbaiki kontras dan mengurangi *noise* pada citra sehingga dapat meningkatkan pemahaman model dan akurasi.

TABEL II
HASIL PENGUJIAN HYPERPARAMETER UNTUK KLASIFIKASI CITRA X-RAY

Hyperparameter		Rata-rata akurasi	Rata-rata generalization gap
Batch Size	5	96.78%	9.09%
	100	96.62%	7.10%
Learning Rate	0.0001	96.68%	7.59%
	0.00001	96.72%	8.60%
Epoch	10	96.64%	4.44%
	20	96.76%	11.75%
Rate Dropout (0)	Tanpa	96.64%	8.19%
	0.2	96.88%	8.06%
	0.3	96.71%	8.07%
	0.4	96.69%	7.61%
	0.5	96.58%	7.91%

Pada penelitian ini, *hyperparameter* yang terdapat pada model VGG16 juga turut memengaruhi kinerja dan performa dalam melakukan klasifikasi pneumonia. Berikut merupakan hasil analisis pengaruh *hyperparameter* berdasarkan hasil penelitian yang dilakukan:

1. *Batch size* merupakan *hyperparameter* yang memengaruhi penyebaran data pada satu *epoch*. Semakin besar nilai *batch size* maka akan semakin kecil iterasi dalam 1 *epoch*. Penelitian ini menguji 2 nilai *batch size* yaitu 5 dan 100. Keduanya dapat melatih model dengan cukup baik pada penelitian ini. Nilai *batch size* 5 memberikan hasil dengan rata-rata akurasi terbaik yaitu 96.78% tetapi masih kurang mencapai konvergensi.

2. *Learning rate* merupakan salah satu *hyperparameter* penting yang memengaruhi seberapa cepat model beradaptasi saat proses *training*. Secara keseluruhan, *learning rate* bernilai 0.0001 dan 0.00001 sudah memberikan hasil yang cukup baik pada penelitian ini. Nilai *learning rate* yang rendah dapat meningkatkan nilai akurasi dan memberikan hasil yang optimal namun memerlukan waktu yang cukup lama dalam proses *training*. Pada penelitian ini, *learning rate* dengan nilai 0.00001 memberikan hasil yang optimal yaitu dengan rata-rata akurasi mencapai 96.71%.
3. *Epoch* merupakan *hyperparameter* yang menentukan jumlah iterasi yang dilakukan pada model. Pada penelitian ini, *epoch* bernilai 10 dan 20 sudah dapat memberikan hasil yang cukup baik. *Epoch* dengan nilai 20 memberikan hasil terbaik dengan rata-rata akurasi 96.76%.
4. *Rate* merupakan *hyperparameter* pada regularisasi *dropout* yang digunakan untuk mengatur probabilitas penonaktifan suatu neuron agar mengurangi kondisi *overfit*. Penelitian ini menguji model tanpa menggunakan *dropout* dan *dropout* dengan nilai *rate* 0.2, 0.3, 0.4 dan 0.5. Jika dibandingkan dengan rata-rata data tanpa *dropout*, data yang menggunakan *dropout* memiliki peningkatan rata-rata akurasi dan penurunan rata-rata *generalization gap* sehingga mengurangi kondisi *overfit*. Nilai *rate* 0.2 mendapatkan rata-rata akurasi terbaik daripada nilai lainnya yaitu 96.88%. Rata-rata *generalization gap* terendah yang didapatkan adalah 7.61% dengan *rate* bernilai 0.4.

IV. SIMPULAN

Metode VGG16 dengan CLAHE sudah mendapatkan akurasi yang baik yaitu sebesar 96.84%, *precision* bernilai 96.83%, *recall* bernilai 96.83% dan *F1-score* bernilai 96.83% dengan *batch size* bernilai 5, *learning rate* bernilai 0.00001, *epoch* bernilai 20 dan *rate* bernilai 0.4. Penerapan CLAHE dalam model mendapatkan peningkatan rata-rata akurasi daripada penerapan model tanpa CLAHE yaitu sebesar 0.13%.

Implementasi *dropout* pada VGG16 juga memiliki pengaruh terhadap peningkatan akurasi dan penurunan *generalization gap* yang menjadi pertanda *overfit*. Berdasarkan penelitian yang sudah dilakukan, *rate* bernilai 0.4 mendapatkan rata-rata *generalization gap* terendah yaitu 7.61% sedangkan *rate* bernilai 0.2 mendapatkan rata-rata akurasi tertinggi yaitu 96.88%

Dalam penelitian lanjutan, dapat menerapkan VGG16 dengan mengkombinasikan metode *cross-validation*, *early stopping*, *data augmentation* dan teknik regularisasi lainnya untuk mencegah terjadinya *overfit* pada jaringan. Selain itu, dapat juga menambahkan *dropout layer* dengan susunan yang berbeda seperti menambahkan setelah *convolutional layer* atau *pooling layer* sehingga dapat mengurangi kondisi *overfit* dengan lebih baik.

DAFTAR REFERENSI

- [1] F. Antony, H. Irsyad and M. Al Rivan, "KNN Dan Gabor Filter Serta Wiener Filter Untuk Mendiagnosis Penyakit Pneumonia Citra X-RAY Pada Paru-Paru", *Jurnal Algoritme*, vol. 1, no. 2, hlm. 147-155, 2 April 2021.
- [2] R. Hidayatullah and S. Violina, "Convolutional Neural Network Architecture and Data Augmentation for Pneumonia Classification from Chest X-Rays Images", *International Journal of Innovative Science and Research Technology*, vol. 5, no. 2, hlm. 158-164, 2 February 2020.
- [3] I. Maysanjaya, "Klasifikasi Pneumonia pada Citra X-rays Paru-paru dengan Convolutional Neural Network", *Jurnal Nasional Teknik Elektro dan Teknologi Informasi*, vol. 9, no. 2, hlm. 190-195, Mei 2020.
- [4] Z. Knok, K. Pap and M. Hrnčić, "Implementation of intelligent model for pneumonia detection", *Tehnicki glasnik*, vol. 13, no. 4, hlm. 315-322, Desember 2019.
- [5] R. Wati, H. Irsyad and M. Rivan, "Klasifikasi Pneumonia Menggunakan Metode Support Vector Machine", *Jurnal Algoritme*, vol. 1, no. 1, hlm. 21- 32, 1 Oktober 2020.
- [6] R. C. Gonzalez and R. E. Woods, *Digital Image Processing*, 4th ed. Upper Saddle River, NJ: Pearson, 2018.
- [7] A. Distanto and C. Distanto, *Handbook of image processing and computer vision: Volume 1: From energy to image*, 2020th ed. Cham, Switzerland: Springer Nature, 2020.
- [8] S. Khan, H. Rahmani, S. Shah and M. Bennamoun, *A Guide to Convolutional Neural Networks for Computer Vision*, Morgan & Claypool Publishers, 2018.
- [9] S. Saha. (2018, 16 Desember). "A Comprehensive Guide to Convolutional Neural Networks—the ELI5 way" *Medium*. [Daring]. Tersedia : <https://towardsdatascience.com/a-comprehensive-guide-to-convolutional-neural-networks-the-eli5-way-3bd2b1164a53>. [17 Februari 2022]
- [10] A. Thite. (2021, 1 Oktober). "What is VGG16" *My Great Learning*. [Daring] Tersedia: <https://www.mygreatlearning.com/blog/introduction-to-vgg16/> [3 Maret 2022]
- [11] T. Rohit . (2019, 6 Agustus). "Step by step VGG16 implementation in Keras for beginners" *Medium*. [Daring]. Tersedia : <https://towardsdatascience.com/step-by-step-vgg16-implementation-in-keras-for-beginners-a833c686ae6c> [29 Januari 2022]
- [12] J. Heaton, *Artificial Intelligence for humans. Volume 3 : Deep Learning and Neural Networks*, vol. 3. St. Louis: Heaton Research, Inc, 2015.
- [13] H. Habibi Aghdam and E. Jahani Heravi, *Guide to Convolutional Neural Networks : A Practical Application to Traffic-Sign Detection and Classification*, 1 st ed. Switzerland: Springer International Publishing AG, 2017.
- [14] J. Patterson and A. Gibson, *Deep Learning*. Sebastopol, CA: O'Reilly Media, 2017