

Deteksi COVID-19 Berdasarkan Hasil Rontgen Dada (*Chest Xray*) Menggunakan Python

Pebri Andhi Herry Pratama¹⁾, Rony Teguh^{2*)}, Abertun Sagit Sahay³⁾, Valencia Wilentine⁴⁾

¹⁾²⁾³⁾ Teknik Informatika, Fakultas Teknik, Universitas Palangkaraya Kampus Tanjung Nyaho Jl. Yos Sudarso Palangka Raya 73112

⁴⁾ Laboratorium Kesehatan dan Kalibrasi Provinsi Kalimantan Tengah Palangka Raya
*corresponding author

¹⁾ nyahuidjabar@gmail.com

²⁾ ronnyteguh@gmail.com

³⁾ abertun@it.upr.ac.id

ABSTRAK

Rontgen Dada (*Chest X-Ray*) memiliki fungsi penting dalam tiga bidang kesehatan yaitu diagnosis, perawatan dan pemeriksaan ulang. Studi dari Cina menyatakan *Chest radiograph (X-Ray)* dan *Chest computed tomography (CT) scans* dapat membantu mendiagnosa COVID-19. Oleh hal tersebut maka *Chest radiographs (X-Ray)* dan *Chest computed tomography (CT) scans* adalah metode yang tepat untuk mendeteksi infeksi paru-paru akibat dari COVID-19. Berdasarkan hal tersebut penulis mencoba membuat sebuah model untuk klasifikasi citra digital hasil rontgen dada (*Chest X-Ray*) dengan label kelas Normal, Pneumonia, Tuberculosis (TBC), dan COVID-19. Melalui model yang dihasilkan tersebut akan dibandingkan model yang terbaik untuk digunakan.

Metode yang digunakan untuk membuat model tersebut adalah melalui pelatihan (*training*) dan pengujian (*testing*) dataset menggunakan model arsitektur Convolutional Neural Network (CNN) yaitu VGG19, ResNet50, dan InceptionV3. Jumlah citra yang digunakan sebanyak 1.000 Gambar *Chest X-Ray*. Dataset tersebut dibagi menjadi data latih (*train*) dan validasi (*validation*) dalam beberapa rasio 20% : 80%, 50% : 50%, dan 80% : 20%. Sedangkan data yang digunakan sebagai data uji (*test*) diambil 10% dari data latih (*train*) dan 4 gambar *chest x-ray* sebagai dataset prediksi.

Berdasarkan hasil dari penelitian yang telah dilakukan, didapatkan model terbaik yaitu VGG19 pada iterasi (*epoch*) 41 dari 61 *epoch* dengan rasio 20% : 80% menjadi model terbaik untuk penelitian ini. Dengan menghasilkan akurasi pelatihan mencapai 0,9444 atau 94,44% (dalam persentase) dan nilai Loss 0,1084. Pada pengujian menghasilkan Confusion Matrix dengan akurasi sebesar 0.96 dan kehilangan 0.0939 Lalu untuk pengujian prediksi data baru menghasilkan akurasi dengan nilai 98.97% untuk label Normal, 99.16% untuk COVID-19, 99.56% untuk Pneumonia dan 99.79% untuk Tuberculosis (TBC).

Kata kunci: *Deep Learning, Convolutional Neural Network, Python, Chest X-Ray, COVID-19*

Abstract

Chest X-ray have an important function in the three areas of healthcare, namely diagnosis, treatment and re-examination. Studies from China suggest Chest Radiographs (X-Ray) and Chest Computed Comography (CT) scans can help diagnose COVID-19. Therefore, chest x-rays (x-rays) and chest computed tomography (CT) are appropriate methods for lung infections due to COVID-19. Based on this, the authors tried to make a model for the classification of digital images of Chest X-Ray results with the labels of Normal, Pneumonia, Tuberculosis (TBC), and COVID-19. Through the resulting model, the best model to use will be compared.

Received: 28/06/2021

Accepted: 29/06/2021

The method used to create this model is through training and testing the dataset using the Convolutional Neural Network (CNN) architectural model, namely VGG19, ResNet50, and InceptionV3. The number of images used is 1,000 Chest X-Ray images. The dataset is divided into training and validation datasets in several ratios of 20% : 80%, 50%: 50%, and 80%: 20%. While testing uses 10% from train dataset chest x-ray images as a confusion matrix dataset and 4 chest x-ray images as a prediction dataset.

From the results of the research that has been done, the best model is VGG19 at 41 of 61 epoch and a ratio of 20% : 80%. Where the VGG19 model produces 94.44% for accuracy and 0,1084 loss value for training. Whereas at the testing stage with a configuration matrix, 95% accuracy value was obtained. Then for testing the new data prediction produces the best accuracy with 98.97% accuracy for the Normal label, 99.16% for COVID-19, 99.56% for Pneumonia, and 99.79% for Tuberculosis (TBC).

Keywords: Deep Learning, Convolutional Neural Network, Python, Chest X-Ray, COVID-19.

1. PENDAHULUAN

Metode Chest Radiograph atau Chest X-Ray telah lama digunakan untuk mendeteksi anatomi tubuh manusia. Pemeriksaan Chest X-Ray adalah salah satu pemeriksaan pencitraan medis yang paling sering digunakan karena lebih terjangkau. Pembacaan foto Chest X-Ray memiliki keterbatasan, yaitu dalam menentukan hasilnya masih menggunakan metode manual dan sulit terdeteksinya penyakit, sehingga memerlukan waktu lama sebelum tenaga medis atau dokter mendiagnosis penyakit yang diderita oleh pasien. Salah satu metode untuk mengatasi masalah tersebut adalah dengan mengklasifikasikan citra Chest X-Ray ke dalam kelas atau label tertentu dengan menggunakan Deep Learning. Data yang berupa citra Chest X-Ray, merupakan data gambar dua dimensi yang dapat diproses menggunakan deep learning dengan metode Convolutional Neural Network (CNN). CNN digunakan untuk mengklasifikasi data yang terlabel dengan menggunakan metode Supervised Learning, yang mana cara kerja cari supervised learning adalah terdapat data yang dilatih dan terdapat variabel yang ditargetkan sehingga tujuan dari metode ini adalah mengklasifikasikan suatu data ke dalam data yang sudah ada. Kemampuan CNN di klaim sebagai model terbaik untuk memecahkan permasalahan object detection dan object recognition pada data citra sebuah gambar hasil rontgen dada.

Penelitian mengenai object recognition telah dilakukan oleh beberapa peneliti. Salah satunya penelitian yang berjudul penelitian Implementasi Deep Learning Menggunakan Convolutional Neural Network (CNN) Untuk Klasifikasi Alat Tulis. Pada penelitian tersebut bertujuan untuk pengenalan objek dengan kelas ballpoint, penghapus, dan penggaris. Data sample yang digunakan sebanyak 300 gambar merupakan hasil crawling dari google image. Akurasi yang dihasilkan model untuk data train adalah 100% dan untuk data test sebesar 95%. Kemudian dilakukan percobaan untuk 3 data baru, didapatkan 1 gambar masuk kategori yang salah [1].

Berdasarkan permasalahan tersebut penulis ingin menerapkan penelitian yang telah dilakukan peneliti sbelumnya menggunakan metode yang sama dengan kasus yang berbeda. Penulis mengusulkan sebuah model yang dapat membaca hasil rontgen dada (Chest X-Ray) untuk diagnosis COVID-19 dengan menggunakan tiga model Convolutional Neural Network (CNN), yaitu model VGG19, ResNet50, dan InceptionV3. Dari ketiga model CNN tersebut akan dibandingkan model yang cocok digunakan untuk diagnosa beberapa label atau kelas dataset yaitu normal, COVID-19, Tuberculosis, dan Pneumonia dari hasil rontgen dada. . Lalu model yang menghasilkan akurasi terbaik akan di implementasikan ke sebuah aplikasi berbasis Web, sehingga dapat digunakan secara umum. Sehingga teknologi deep learning dapat diterapkan dalam bidang kesehatan dan menjadi alat bantu seorang radiologis ataupun dokter dalam hal diagnosis hasil rontgen dada (Chest X-Ray):.

2. TINJAUAN PUSTAKA

Penelitian Pertama dengan jurnal penelitian yang berjudul “Implementasi Sistem Pakar Menggunakan Metode Certainty Factor Untuk Mendiagnosa Dini Corona Virus Disease (COVID-19)”. Dari 152 data pasien yang di input pada penelitian ini mendapatkan hasil 114 ODP dengan rata-rata nilai CF 91,38% , 36 PDP dengan rata-rata nilai CF 98,25% dan 2 NON dengan rata-rata nilai CF 40%. CF dengan percobaan perhitungan sistem yaitu data yang mewakili pasien mendapatkan nilai CF 0.998848 atau 99.88% menjadi PDP [2].

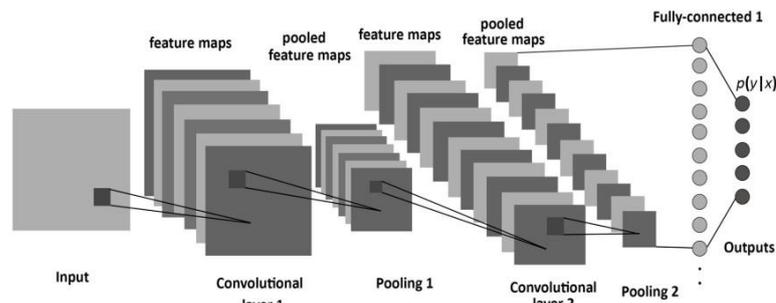
Penelitian kedua dengan jurnal penelitian yang berjudul Implementasi Metode Forward Chaining Untuk Diagnosa Penyakit COVID-19 di Rsud Berkah Pandeglang Banten. Berdasarkan hasil penelitian ini metode forward chaining sangat cocok untuk penelitian diagnosis Virus COVID 19 dikarenakan algoritma forward chaining menggunakan algoritma runtu maju yang setiap langkah mendiagnosis gejala dan langkah selanjutnya yaitu membuat system aplikasi agar penelitian ini dapat berkembang menjadi sistem pakar deteksi Virus COVID 19 dengan metode forward chaining kedepannya [3].

Penelitian Ketiga dengan penelitian yang berjudul penelitian Implementasi Deep Learning Menggunakan Convolutional Neural Network Untuk Klasifikasi Alat Tulis. Akurasi yang dihasilkan model untuk data train adalah 100% dan untuk data test sebesar 95%. Kemudian dilakukan percobaan untuk 3 data baru, didapatkan 1 gambar masuk kategori yang salah [1].

3. METODE PENELITIAN

3.1 Convolutional Neural Network (CNN)

Metode yang digunakan dalam penelitian adalah melalui pelatihan (*training*) dan pengujian (*testing*). CNN pada dasarnya adalah susunan banyak layer yang terdiri dari *convolution layer*, *pooling layer* yang disebut *feature layer*, dan *fully connected layer* yang disebut *classification layer*. Model arsitektur *Convolutional Neural Network* (CNN) yang digunakan pada penelitian yaitu VGG19, ResNet50, dan InceptionV3 yang telah dimodifikasi. Model-model yang digunakan tersebut dirancang untuk digunakan untuk klasifikasi ribuan label. Modifikasi dilakukan pada *classification layer*, karena jumlah label yang digunakan pada penelitian hanya empat kelas yaitu normal, *tuberculosis*(TBC), *pneumonia*, dan COVID-19. Setelah model melalui tahap pelatihan dan menghasilkan model untuk klasifikasi citra rontgen dada ke dalam 4 kelas. Tahap selanjutnya adalah menguji model tersebut dengan prediksi data baru menggunakan model yang menghasilkan nilai akurasi dan kehilangan paling optimal.



Gambar 1. CNN Concept [4]

3.2 Populasi dan Sampel

Populasi dalam penelitian ini adalah citra hasil rontgen dada (*Chest X-Ray*). Sedangkan sampel yang digunakan dalam penelitian ini adalah dataset citra rontgen dada yaitu normal, *tuberculosis*(TBC), *pneumonia*, dan COVID-19.

Tabel 1. Definisi Operasional

No	Variabel	Definisi
1	Normal	Dataset citra rontgen dada paru-paru normal yang tidak terkena infeksi pneumonia, <i>Tuberculosis</i> (TBC) ataupun COVID-19.
2	<i>Tuberculosis</i> (TBC)	Dataset citra rontgen dada terinfeksi <i>Tuberculosis</i> (TBC).
3	<i>Pneumonia</i>	Dataset citra rontgen dada infeksi paru-paru.
4	COVID-19	Dataset citra rontgen dada terinfeksi positif COVID-19.

3.3 Jenis dan Sumber Data

Jenis data yang digunakan untuk pelatihan (*training*) dan pengujian (*testing*) pada penelitian ini adalah data sekunder. Data yang digunakan berjumlah 1.000 gambar *Chest Xray* yang digunakan untuk pelatihan (*training*), dengan pembagian rasio 20% : 80%, 50% : 50%, dan 80% : 20% untuk data uji (*train*) dan data validasi (*validation*). Sedangkan data uji (uji)yang digunakan untuk pengujian (*testing*) sebesar 10% data yang di bagi dari data uji (*train*).

Tabel 2. Sumber Dataset Pelatihan (*training*)

No	Variabel	Judul	Sumber (link)	Jumlah
1	Normal	Chest X-Ray Images (pneumonia) [5]	https://www.kaggle.com/paultimothymooney/chest-xray-pneumonia	250
2	<i>Tuberculosis</i> (TBC)	Pulmonary Chest X-Ray Abnormalities [6]	https://www.kaggle.com/kmader/pulmonary-chest-xray-abnormalities?select=Montgomery	250
3	<i>Pneumonia</i>	Chest X-Ray Images [5]	https://www.kaggle.com/paultimothymooney/chest-xray-pneumonia	250
4	COVID-19	COVID-19 Radiography Database [7] COVID-19 chest xray [8]	https://www.kaggle.com/tawsi-furrahman/covid19-radiography-database https://www.kaggle.com/bachr/covid-chest-xray	219 31
Jumlah				1.000

3.4 Rancangan Dataset

Dataset dibagi berdasarkan 4 label yang dapat dilihat pada Gambar 2 yaitu gambar hasil rontgen dada Normal, *Tuberculosis* (TBC), *Pneumonia*, dan positif COVID-19.

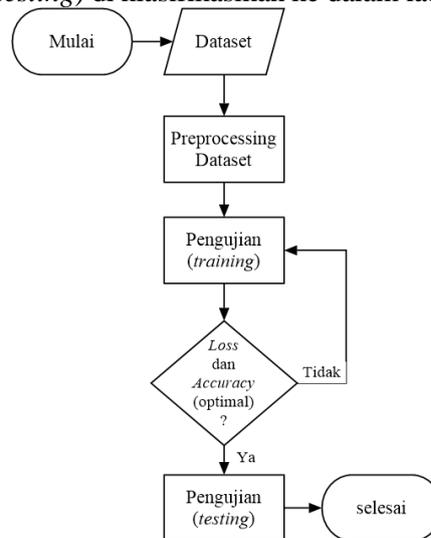


Gambar 2. Label Dataset (Normal, COVID-19, *Pneumonia*, dan *Tuberculosis*)

3.5 Rancangan Deep Learning

Rancangan *deep learning* terdiri dari beberapa tahapan, dimulai dari pengumpulan data citra Normal, *Tuberculosis* (TBC), *Pneumonia* dan COVID-19 yang akan digunakan untuk citra latih (*train*) dan dan citra uji (*test*). Lalu masuk ke tahap *preprocessing*, kemudian pelatihan

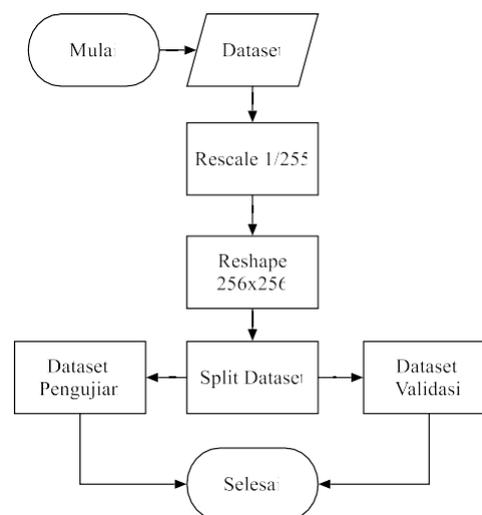
menggunakan model-model *Convolutional Neural Network* (CNN), lalu terakhir data citra yang digunakan sebagai pengujian (*testing*) di klasifikasikan ke dalam label dataset yang telah dilatih.



Gambar 3. Flowchart Desain Deep Learning

3.6 Rancangan Image Processing

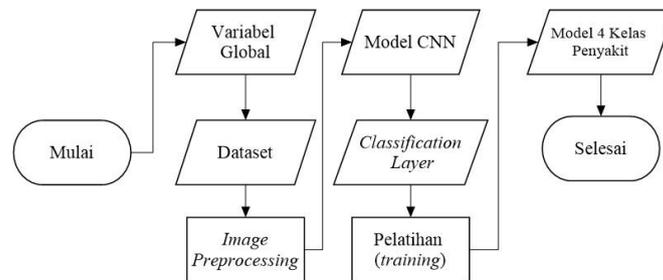
Perancangan *preprocessing* merupakan langkah yang wajib dilakukan sebelum masuk perancangan *Convolutional Neural Network* (CNN). *Preprocessing* berfungsi untuk menyiapkan citra agar dapat digunakan pada proses selanjutnya. Hal-hal yang dilakukan dalam perancangan *preprocessing* yang pertama yaitu dilakukan proses *thresholding* yang merupakan proses untuk menjadikan nilai citra menjadi 0-1 dari 0-255. Kemudian proses *resize* menjadi ukuran 256x256 untuk model VGG19, 244x224 untuk ResNet50, dan 299x299 untuk InceptionV3 seperti pada gambar 3.6 yang berguna untuk normalisasi citra agar seluruh dalam dataset sama. Setelah itu dilakukan proses *split* untuk membagi data dalam beberapa rasio (20% : 80% ; 50% : 50% ; 80% : 80%) yang telah dikumpulkan menjadi dua, yaitu data uji (*train*) dan data validasi (*validation*) yang digunakan untuk proses pengujian (*training*). Kemudian dari data uji (*train*) diambil sebesar 10% untuk digunakan sebagai data uji (*test*) pada proses pengujian (*testing*).



Gambar 4. Design flowchart preprocessing dataset

3.7 Rancangan Pelatihan (*training*)

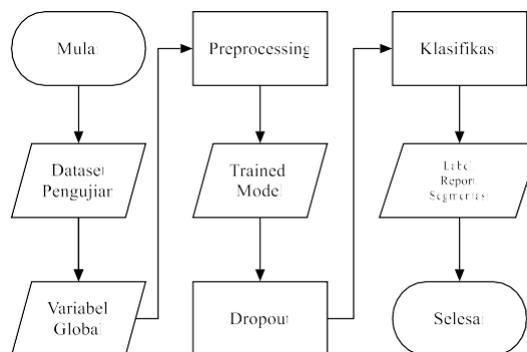
Dataset latih (*train*) dan data validasi (*validation*) yang telah melewati *preprocessing* dapat dilanjutkan untuk proses proses pelatihan (*training*). Proses *training* ini bertujuan untuk melatih algoritma *Convolutional Neural Network* (CNN) dalam mempelajari pola setiap gambar pada dataset yang telah melewati *image processing* dan membentuk sebuah model baru. Rancangan *training* yang diajukan untuk pada penelitian ini terdiri dari beberapa tahapan seperti pada Gambar 5.



Gambar 5. Flowchart Proses Pelatihan (*training*)

3.8 Rancangan Pengujian (*testing*)

Proses ini mencocokkan antara data masukkan dengan data yang telah melalui proses *training*. Setelah gambar *Chest X-Ray* telah melewati *preprocessing* Tahap selanjutnya adalah memilih file *training* yang telah disimpan untuk digunakan sebagai klasifikasi. Setelah proses klasifikasi dijalankan, maka akan ditampilkan Gambar *Chest X-Ray* beserta dengan hasil klasifikasi berupa label dan persentase akurasi.



Gambar 6. Flowchart Pengujian (*Testing*)

4. PEMBAHASAN

4.1 *Preprocessing*

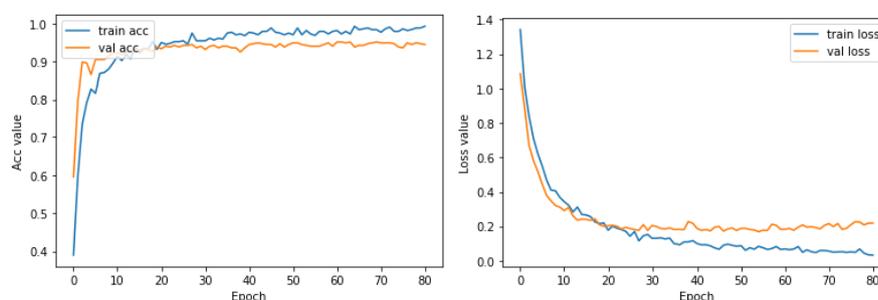
Pada tahap *peporcessing* ini, ukuran keseluruhan dataset dinormalisasi ukuran 256x256 untuk model VGG19, 244x224 untuk ResNet50, dan 299x299 untuk InceptionV3. Lalu dataset dilakukan proses *rescale* yang merupakan proses untuk menjadikan nilai citra menjadi 0-1 dari 0-255 yang gunanya mempermudah dalam proses input *training* dan *testing*. Setelah itu dilakukan proses *split* untuk membagi data dalam beberapa rasio (20% : 80% ; 50% : 50% ; 80% : 80%) yang telah dikumpulkan menjadi dua, yaitu data uji (*train*) dan data validasi (*validation*) yang digunakan untuk proses pengujian (*training*). Kemudian dari dari data uji (*train*) diambil sebesar 10% untuk digunakan sebagai data uji (*test*) pada proses pengujian (*testing*).

4.2 Implementasi Model

Setelah melakukan *preprocessing* pada dataset, tahap selanjutnya adalah menentukan model yang digunakan untuk proses *training*. Pada penelitian ini menggunakan beberapa model dari CNN, yaitu VGG19, ResNet50, dan InceptionV3. Model tersebut digunakan sebagai *feature layer*, sedangkan *classification layer* di modifikasi menyesuaikan jumlah label yang digunakan. Dikarenakan jumlah kelas pada imagenet adalah 1.000 maka ini tidak cocok dengan jumlah kelas yang kita miliki yaitu hanya 4. Maka pada penelitian menggunakan beberapa *layer* sebagai model output.

4.3 Pelatihan (*training*)

Setelah membuat model selanjutnya adalah melakukan proses *training*, dimana proses *training* ini merupakan proses dimana *machine learning* bekerja sehingga algoritma yang sudah kita definisikan dapat mengingat pola dari masing masing kelas pada data yang kita *train*. Pada proses *training* penelitian ini menggunakan proses pengulangan atau iterasi (*epoch*) *training* yang berhenti otomatis saat akurasi sudah tidak dapat naik dari akurasi dari beberapa *epoch* sebelumnya.



Gambar 7. Grafik Akurasi (*Accuracy*) dan Kehilangan (*Loss*) VGG19 Rasio 50%:50%

Gambar 7 menunjukkan grafik pergerakan nilai akurasi (*acc/accuracy*) dan nilai *loss* dari hasil pelatihan (*training*) dan validasi (*validation*). Berdasarkan gambar tersebut, garis berwarna biru menunjukkan pergerakan nilai akurasi dan kehilangan untuk data pelatihan, sedangkan garis berwarna orange menunjukkan pergerakan nilai akurasi dan kehilangan untuk data validasi.

Pada penelitian ini saat proses pelatihan, setiap terjadi peningkatan akurasi maka pelatihan dilanjutkan, sedangkan jika tidak ada peningkatan akurasi dalam beberapa *epoch* maka pelatihan akan otomatis berhenti. Gambar 4.4 pelatihan berhenti pada *epoch* 81, karena tidak ada peningkatan akurasi dari *epoch* 61. Sehingga akurasi tertinggi dihasilkan pada *epoch* 61. Begitu juga dengan model lainnya jika proses pelatihan tidak terdapat peningkatan pada beberapa *epoch* maka proses pelatihan akan berhenti otomatis.

Berdasarkan pelatihan model-model yang telah dijalankan, didapatkan variasi waktu untuk setiap model dengan iterasi (*epoch*) dan rasio dataset yang berbeda.

Tabel 4. Waktu Pelatihan

Rasio	Model	VGG19	ResNet50	InceptionV3
20%	Epoch	49	21	33
	Waktu	00:25:08	0:10:09	00:11:53
50%	Epoch	81	21	27
	Waktu	00:39:05	0:10:09	00:14:48
80%	Epoch	71	27	35
	Waktu	00:34:27	0:15:54	0:21:17

4.4 Validasi

Untuk mengetahui akurasi (*accuracy*) dan kehilangan (*loss*) rata-rata pada setiap model maka dilakukan evaluasi model menggunakan dataset *validation*. Tabel 4 adalah Tabel evaluasi hasil *training* setiap model yang telah dilakukan berulang-ulang untuk mendapatkan nilai akurasi (*accuracy*) dan kehilangan (*loss*) yang optimal untuk digunakan pada proses pengujian (*testing*).

Tabel 5. Tabel Evaluasi Hasil *Training*

Rasio	Model	VGG19	ResNet50	InceptionV3
20%	<i>Epoch</i>	41	1	13
	<i>Accuracy</i>	0,9444	0,25	0,6222
	<i>Loss</i>	0,1084	4,72	1,4805
50%	<i>Epoch</i>	61	1	7
	<i>Accuracy</i>	0.9335	0.2695	0,6328
	<i>Loss</i>	0.2354	3,1717	1,0025
80%	<i>Epoch</i>	51	7	15
	<i>Accuracy</i>	0.9335	0,2695	0,6523
	<i>Loss</i>	0,2329	3,7394	1,1784

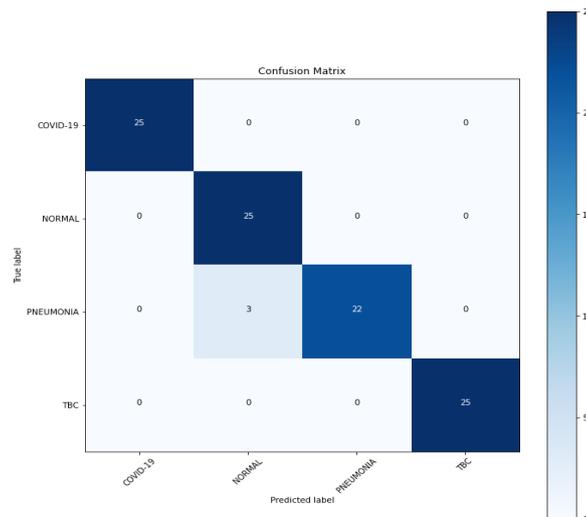
Berdasarkan evaluasi model tersebut didapatkan model optimal masing-masing model yaitu VGG19, ResNet50, dan InceptionV3. Model VGG19 menghasilkan nilai akurasi 0,9444 dan nilai Loss 0,1084 pada iterasi (*epoch*) terhenti pada 61 *epoch* karena tidak ada peningkatan dari 41 *epoch* dengan rasio 20% : 80%, ResNet50 menghasilkan akurasi optimal pada iterasi (*epoch*) terhenti pada 21 *epoch* karena tidak ada peningkatan dari 1 *epoch* dengan rasio 50% : 50% yaitu dengan nilai akurasi 0.2695 dan nilai Loss 3,1717. InceptionV3 menghasilkan akurasi optimal pada iterasi (*epoch*) terhenti pada 35 *epoch* karena tidak ada peningkatan dari 15 *epoch* dengan rasio 80% : 20% yaitu dengan nilai akurasi 0.6523 dan nilai Loss 1,1784.

4.5 Pengujian (*testing*)

Algoritma *Convolutional Neural Network* (CNN) membutuhkan proses *training* dan *testing*. Proses *testing* bertujuan menguji sebuah model yang dibentuk pada saat proses *training*. Pada pengujian (*testing*) ini dilakukan dengan menggunakan model terbaik saat proses pelatihan (*training*) yaitu VGG19 dengan nilai akurasi tertinggi berhenti di 41 iterasi (*epoch*) pada rasio 20% : 80%, ResNet50 dengan nilai akurasi tertinggi berhenti di 1 iterasi (*epoch*) pada rasio 50% : 50%, dan untuk model InceptionV3 dengan nilai akurasi tertinggi berhenti di 15 iterasi (*epoch*) pada rasio 80% : 20%.

4.5.1 Confussion Matrix

Confussion Matrix adalah matriks N x N yang digunakan untuk mengevaluasi kinerja model klasifikasi, di mana N adalah jumlah kelas target. *Confussion Matrix* membandingkan nilai target aktual dengan yang diprediksi oleh model *machine learning*. Ditampilkan dengan tabel yang menyatakan jumlah data uji yang benar diklasifikasikan dan jumlah data uji yang salah diklasifikasikan [9].



Gambar 8. Confussion Matrix

Pada penelitian ini penulis menggunakan perintah *Classification report* pada python yang akan menampilkan nilai *Accuracy* dan *Loss* dari pengujian yang telah dilakukan. Berikut merupakan *classification report* Model VGG19 dengan iterasi (*epoch*) 41 pada rasio 20% : 80%, ResNet50 dengan iterasi (*epoch*) 1 pada rasio 50% : 50%, dan untuk model InceptionV3 dengan iterasi (*epoch*) 15 pada rasio 80% : 20%.

Tabel 6. Classification Report

Model	Accuracy	Loss
VGG19 (Rasio 20% : 80%, epoch 41)	0.96	0.0939
ResNet50 (Rasio 50%:50% , epoch 1)	0.25	1.6600
InceptionV3 (Rasio 50%:50% , epoch 15)	0.63	1.1077

Dari Tabel 7 dapat dilihat bahwa pengujian data menggunakan model VGG19 pada rasio 20% : 80% dengan iterasi (*epoch*) 41 menghasilkan akurasi sebesar 0.96 atau 96% dan Kehilangan 0,0939. Model ResNet50 pada rasio 50% : 50% dengan iterasi (*epoch*) 1 menghasilkan akurasi sebesar 0.25 atau 25% dan Kehilangan 1,660. Sementara itu model InceptionV3 pada rasio 80% : 20% dengan iterasi (*epoch*) 15 menghasilkan akurasi sebesar 0.63 atau 63% dan Kehilangan 1,1077.

4.5.2 Prediksi

Proses ini bertujuan untuk mengetahui label data baru yang belum diketahui berdasarkan pola gambar dengan menampilkan segmentasi dan persentasi akurasinya.pada penelitian ini dilakukan prediksi menggunakan 4 data baru menghasilkan *output* seperti pada Tabel 7.

Tabel 7. Hasil Prediksi VGG19

Label	Normal	COVID-19	Pneumonia	TBC
Akurasi (%)	98.97%	99.16%	99.56%	99.79%
Output				

Tabel 7 merupakan hasil prediksi dari model VGG19 dengan iterasi (*epoch*) 41 pada rasio 20% : 80%. Berdasarkan tabel tersebut menghasilkan prediksi seluruh label dengan benar. Dimana label normal diprediksi ke normal dengan akurasi 98,97%, label COVID-19 diprediksi ke COVID-19 dengan akurasi 99,16%, label *Pneumonia* diprediksi ke *Pneumonia* dengan akurasi 99,56%, sedangkan label TBC diprediksi ke TBC dengan akurasi 99,79.

5. KESIMPULAN

Berdasarkan hasil dari penelitian yang telah dilakukan dapat disimpulkan bahwa model VGG19 pada iterasi (*epoch*) 41 dari 61 epoch dengan rasio 20% : 80% menjadi model terbaik untuk penelitian ini. Dengan menghasilkan akurasi pelatihan mencapai 0,9444 atau 94,44% (dalam persentase) dan nilai Loss 0.1084. Pada pengujian menghasilkan *Confussion Matrix* dengan sebesar 0.96 atau 96% dan Kehilangan 0,0939, sedangkan prediksi menghasilkan prediksi keseluruhan label dengan benar dengan label NORMAL dengan nilai akurasi tertinggi yaitu 99,97%.

DAFTAR PUSTAKA

- [1] J. Pujoseno, "IMPLEMENTASI DEEP LEARNING MENGGUNAKAN CONVOLUTIONAL NEURAL NETWORK UNTUK KLASIFIKASI ALAT TULIS," 2018.
- [2] M. F. Suryana, F. Fauziah, and R. T. K. Sari, "Implementasi Sistem Pakar Menggunakan Metode Certainty Factor Untuk Mendiagnosa Dini Corona Virus Desease (COVID-19)," *J. Media Inform. Budidarma*, vol. 4, no. 3, p. 559, 2020, doi: 10.30865/mib.v4i3.2132.
- [3] R. Rizky, M. Ridwan, and Z. Hakim, "Implementasi Metode Forward Chaining Untuk Diagnosa Penyakit Covid 19 Di Rsud Berkah Pandeglang Banten," *J. Teknol. Inf.*, vol. 4, no. 1, pp. 1–4, 2020.
- [4] V. Nigam, "Understanding Neural Networks. From neuron to RNN, CNN, and Deep Learning," *medium.com*, 2018. <https://medium.com/analytics-vidhya/understanding-neural-networks-from-neuron-to-rnn-cnn-and-deep-learning-cd88e90e0a90>.
- [5] P. Mooney, "Chest X-Ray Images (Pneumonia)," 2018. <https://www.kaggle.com/paultimothymooney/chest-xray-pneumonia>.
- [6] K. S. Mader, "Pulmonary Chest X-Ray Abnormalities," 2018. <https://www.kaggle.com/kmader/pulmonary-chest-xray-abnormalities?select=Montgomery>.
- [7] T. Rahman, M. Chowdhury, and A. Khandakar, "COVID-19 Radiography Database," 2020. <https://www.kaggle.com/tawsifurrahman/covid19-radiography-database>.
- [8] Bachir, "COVID-19 chest xray," 2020. <https://www.kaggle.com/bachrr/covid-chest-xray>.
- [9] A. Bhandari, "Everything you Should Know about Confusion Matrix for Machine Learning," *Analytics Vidhya*, 2020. <https://www.analyticsvidhya.com/blog/2020/04/confusion-matrix-machine-learning/>.
- [10] R. Teguh, A. S. Sahay, and F. F. Adji, "Pemodelan Penyebaran Infeksi Covid-19 Di Kalimantan, 2020," *J. Teknol. Inf. J. Keilmuan dan Apl. Bid. Tek. Inform.*, vol. 14, no. 2, pp. 171–178, 2020, doi: 10.47111/jti.v14i2.1229.