



Klasifikasi Covid-19 menggunakan Arsitektur *DarkCovidNet* pada Citra Radiografi X-ray Dada

Rima Tri Wahyuningrum^{#1}, Wahyu Zainur Putra^{#2}, Budi Dwi Satoto^{#3}, Amillia Kartika Sari^{#4}, Anggraini Dwi Sensusiaty^{#5}

[#]Jurusan Teknik Informatika, Universitas Trunojoyo Madura
Jl. Raya Telang, Kamal, Bangkalan, 69162

¹rimatriwahyuningrum@trunojoyo.ac.id

²wahyu.zainurputra@gmail.com

³budids@trunojoyo.ac.id

*Prodi Teknologi Radiologi dan Pencitraan, Fakultas Vokasi, Universitas Airlangga
Jl. Srikana 65 Surabaya 60286

⁴amillia.kartika.sari@vokasi.unair.ac.id

⁵anggraini-d-s@fk.unair.ac.id

Abstrak— Covid-19 adalah penyakit *severe acute respiratory syndrome*. Coronavirus menjadi penyebab gangguan pernapasan dan infeksi paru paru, sehingga dapat menyebabkan kematian. Penyakit Covid-19 sudah tersebar ke seluruh negara termasuk negara Indonesia. Meskipun saat ini, Covid-19 telah mereda namun pencegahan maupun penanganannya tetap dibutuhkan. Oleh karena itu sangat diperlukan alat untuk mendiagnosis penyakit Covid-19 pada X-ray dada. Penggunaan klasifikasi citra berfungsi untuk memproses penggabungan piksel pada suatu citra ke dalam kelompok untuk diinterpretasikan sebagai bentuk properti yang spesifik. Dengan klasifikasi citra, mampu mempermudah pengelompokan individu untuk mewakili fitur kelas citra. Pada penelitian citra radiografi X-ray dada ini, menggunakan *multiclass-classification* yang terdiri dari 3 kelas yaitu: Covid-19, Normal (*No-Findings*), dan *Pneumonia*. Dataset yang diperoleh berjumlah 4.945 citra X-ray. Pertama, dilakukan proses *input* citra dan *resize image*. Setelah itu dilakukan pembagian data yaitu 80% sebagai data *train* dan 20% sebagai data *test*. Pada proses pelatihan (*train*) akan menggunakan model *DarkCovidNet*. Arsitektur yang diusulkan terdiri dari 19 *convolutional layer* dan 5 *maxpooling*. Model ini terdapat proses *DarkNet* (DN). DN terdiri dari proses *convolutional*, *batch normalization* dan *LeakyReLU*. Pada skenario uji coba menggunakan optimasi Adam, *reduce learning rate*, dan menambahkan 3 *hidden layer*. Hasil uji coba terbaik terdapat pada uji coba keempat dengan hasil akurasi sebesar 95,85%, *F1-score* 95,89%, *AUC* 99,48%. Dengan demikian model *DarkCovidNet* tersebut sangat bagus dalam melakukan klasifikasi citra X-ray dada.

Kata kunci— Covid-19, *Multiclass-Classification*, Model *DarkCovidNet*, Akurasi, *F1-Score*.

I. PENDAHULUAN

Virus Corona atau biasa disebut Covid-19 adalah penyakit *severe acute respiratory syndrome*. Virus ini menyebabkan gangguan pernapasan maupun infeksi paru paru, sehingga dapat juga menyebabkan kematian. Penyakit Covid-19 ini telah tersebar ke seluruh negara. Dari hasil laporan WHO, penyebab pneumonia yang tidak dikenali di Wuhan-Cina pada 31 Desember 2019, dengan cepat menjadi pandemik [1]. Penyakit ini diberi nama COVID-19 dan virusnya disebut SARS-Cov-2. Virus ini menyebar dari Wuhan ke sebagian besar Cina dalam waktu 30 hari [2].

Selama ini, Covid-19 didiagnosis menggunakan tes *Reverse Transcription-Polymerase Chain Reaction* (RT-PCR). Namun selain menggunakan tes tersebut, Covid-19 juga dapat didiagnosis menggunakan metode pencitraan radiologi pada dada dengan bantuan komputer untuk dilakukan tomografi *Cycle Threshold* (CT) dan X-Ray. Karena sensitivitas RT-PCR yang diperoleh rendah, yaitu sebesar 60-70% maka bisa disimpulkan bahwa hasil negatif yang diperoleh menghasilkan gejala yang dapat dideteksi dengan memeriksa citra radiologi pasien. Selain itu, penggunaan RT-PCR juga membutuhkan biaya yang mahal. Oleh karena itu, jumlah pemakaian yang bisa digunakan menjadi terbatas. Gejala yang diketahui dalam citra radiologis pasien, menyatakan bahwa CT adalah metode sensitif untuk mendeteksi pneumonia, Covid-19, sehingga dianggap sebagai alat *skinning* seperti RT-PCR [3], [4], [5].

Pada awal pandemi, pusat klinis Tiongkok di Cina tidak memiliki alat tes yang cukup dan memperoleh hasil optimal untuk mengatasi pneumonia penyakit tersebut [2]. Ketika

dilakukan pemeriksaan pada alat tes tersebut, menghasilkan tingkat *false-negative*. *False-negative* adalah kondisi hasil PCR negatif yang bisa disebut pasien tersebut terinfeksi virus corona [4]. Sehingga dokter-dokter Tiongkok didorong untuk membuat hasil diagnosis berdasarkan hasil klinis X-ray dada dan CT dada [5].

Saat ini, penerapan metode *deep learning* telah diterapkan seperti pada deteksi *pneumonia* dari citra X-ray dada, citra fundus dan segmentasi paru-paru [6]. Peningkatan pandemik Covid-19 menimbulkan kekhawatiran terutama dalam bidang medis. Sehingga dibutuhkan alat untuk dapat mengklasifikasikan citra X-ray dada yang dapat membedakan pasien-pasien Covid-19 [7]. Dengan terbatasnya tenaga medis yang diperoleh, akan lebih dipermudah ketika menggunakan teknologi radiologi untuk dapat memberikan bantuan diagnosis yang akurat [8].

Beberapa penelitian telah banyak menggunakan citra radiologi untuk mendeteksi Covid-19. Pada tahun 2020, Ozturk et al. [4] mengusulkan model *deep learning* untuk mendeteksi Covid-19 menggunakan DarkCovidNet model yang memperoleh akurasi 98,08% dalam mengklasifikasikan kelas normal, dan Covid-19. Sementara itu, Marium Malik et al. [3] mengembangkan model *deep learning* menggunakan CNN untuk menganalisis citra X-ray dada dan menunjukkan akurasi 85,11%. Pada tahun yang sama Linda Wang et al. [9] menggunakan metode Covid-Net memberikan hasil sensitivitas yang baik dengan hasil sebesar 91,0% untuk kasus Covid-19. Salah satu kelemahan utama dari analisis radiografi dada adalah inefisiensi untuk deteksi tahap awal Covid-19, karena di rumah sakit Cina tidak memiliki yang memadai dalam deteksi Covid-19 [10]. Namun, metode *deep learning* yang terlatih dengan baik dapat berfokus pada poin-poin yang tidak dapat terlihat oleh mata.

Struktur CNN yang khas memiliki *convolutional layer* dengan mengekstrak fitur dari *input* citra sehingga difilter untuk dapat diterapkan. *Pooling layer* untuk mengurangi ukuran kinerja komputasi dan *fully-connected-layer* tersebut yang merupakan jaringan saraf [3]. Model CNN dibuat sesuai dengan parameter internalnya seperti klasifikasi atau pengenalan objek. Oleh karena itu dalam penelitian ini menggunakan model Darknet-19. Darknet-19 merupakan pengklasifikasi model yang membentuk dasar dari sistem deteksi objek real-time yang disebut *YOLO (You Only Look Once)* [4]. Sistem ini memiliki arsitektur canggih untuk bisa merancang deteksi objek. Maka dari itu arsitektur model Darknet-19 atau biasa disebut *DarkCovidNet* dapat digunakan untuk menyesuaikan klasifikasi atau pengenalan objek tertentu dan bisa untuk mengklasifikasikan citra X-ray dada.

Oleh karena itu diperlukan metode model terbaik untuk bisa mendapatkan hasil akurasi terbaik dalam mengklasifikasi citra X-ray dada. Sehingga penelitian ini menerapkan *DarkCovidNet* dari pilihan metode model yang dirujuk. Tujuan penelitian yaitu mendapatkan sistem kinerja terbaik pada *Convolutional Neural Network (CNN)* menggunakan Arsitektur *DarkCovidNet* untuk dapat mengklasifikasikan citra X-ray dada. Sehingga penelitian

ini dapat bermanfaat sebagai alat bantu dalam mempercepat diagnosis penyakit Covid-19 pada pasien. Dengan demikian lebih mempermudah dalam mengkategorikan pasien yang terkena penyakit Covid-19. Batasan permasalahan dalam penelitian ini adalah *dataset* difokuskan pada 3 kelas yaitu kelas Normal, Covid-19 dan *Pneumonia*. *Dataset* yang digunakan berjumlah 4.945, dengan rincian kelas normal sejumlah 1.800 citra, kelas Covid-19 sejumlah 1.800 citra dan *Pneumonia* sejumlah 1.345 citra.

II. TINJAUAN PUSTAKA

A. Klasifikasi Citra

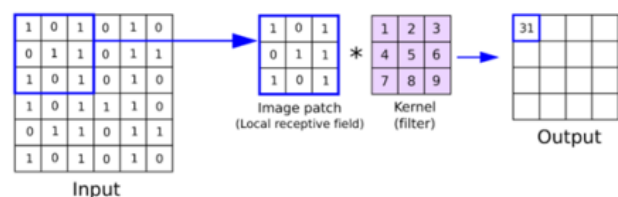
Klasifikasi Citra adalah suatu proses penggabungan atau pengelompokan seluruh *pixel* pada suatu citra ke dalam kelompok untuk diinterpretasikan sebagai bentuk properti yang spesifik [12]. Dengan adanya klasifikasi citra, mempermudah pengelompokan individu atau set *pixel* untuk mewakili fitur, dan kelas sehingga dapat dianalisis, kemudian dapat dikelompokkan sesuai fitur klasifikasi citra lainnya. Klasifikasi citra, yang akan digunakan yaitu *Multi Classification*. *Multi Classification*, yang digunakan lebih dari dua *output* kelas yang diklasifikasi.

B. Convolutional Neural Network (CNN)

Convolutional Neural Network adalah algoritma *Deep Learning* dalam *input* citra, yang bisa ditetapkan untuk pembobotan dan basis yang dapat dipelajari dalam berbagai aspek/objek citra sehingga dapat membedakan satu dari berbagai objek lainnya [13]. CNN memiliki kemampuan untuk mempelajari karakteristik objek secara efisien. Arsitektur CNN terdiri dari 3 tahapan yaitu *convolutional layer* bisa mengekstrak fitur dari *input* citra yang difilter untuk dapat diterapkan, *pooling layer* untuk mengurangi ukuran kinerja komputasi dan *fully-connected-layer* tersebut yang merupakan *neural network*. Berikut penjelasan detailnya:

1. Convolutional Layer

Convolutional layer menerapkan operasi konvolusi ke *input*, untuk meneruskan hasilnya ke lapisan berikutnya. Konvolusi dilakukan untuk mengubah semua piksel dalam bidang reseptif menjadi suatu nilai [13]. Konvolusi juga disebut sebagai apikasi sederhana yang dapat memfilter *input* menjadi aktivasi untuk bisa menunjukkan fitur yang terdeteksi pada suatu objek atau citra. Pada pemrosesan klasifikasi di fitur yang telah diekstrak dari citra, akan membentuk *convolutional layer* sehingga diaplikasikan untuk memperoleh *feature map*.

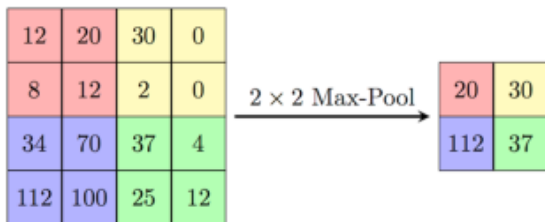


Gambar. 1 Convolutional layer

Setelah melakukan proses *convolution*, fungsi aktivasi yang digunakan adalah ReLu. Fungsi ReLu adalah nilai *output* dari neuron, yang dinyatakan sebagai 0 jika nilai *input* adalah negatif. Jika nilai *input* dinyatakan positif maka *output* adalah nilai input aktivasi itu sendiri. Fungsi aktivasi berada pada tahapan sebelum melakukan proses *pooling layer* dan setelah melakukan proses *convolution* [14].

2. Pooling Layer

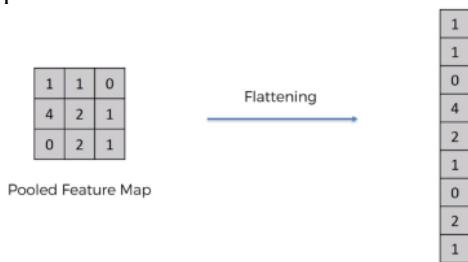
Pooling layer biasa berada pada tahap setelah melakukan proses *convolutional layer*. Pada prinsip *pooling layer* terdiri dari sebuah *filtering* ukuran tertentu untuk bisa melakukan pergeseran pada seluruh area *feature map*. *Pooling* yang biasa digunakan untuk adalah *Max Pooling* dan *Average Pooling* [13]. *Pooling layer* digunakan untuk mengurangi dimensi pada peta fitur (*feature map*).



Gambar. 2 Pooling layer

3. Flatten

Setelah proses *Pooling layer/ Max-pooling* dilakukan proses *flatten*, yang digunakan untuk mengkonversi *feature map* menjadi vektor satu dimensi agar mempermudah proses *fully - connected layer*. Dapat dilihat contoh proses *flatten* pada Gambar 3.

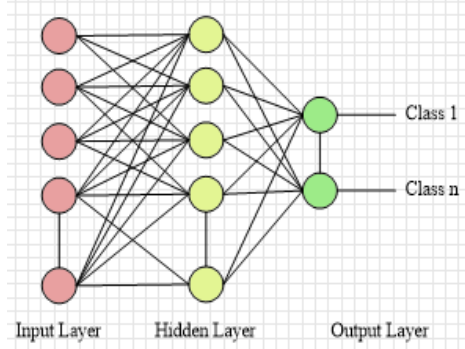


Gambar. 3 Flatten

4. Fully - Connected Layer

Setelah melakukan proses *flattening*, selanjutnya adalah tahapan proses *fully-connected-layer*. Dari proses *flattening* sebelumnya yang sudah menjadi nilai vektor agar bisa langsung digunakan sebagai *input* dari *fully-connected-layer*. *Fully-connected-layer* adalah lapisan yang saling terhubung dari lapisan *input* pertama diteruskan ke setiap unit aktivasi lapisan berikutnya [14]. Lapisan ini bisa disebut sebagai lapisan terhubung penuh yang dapat mengkompilasi data yang diekstraksi oleh lapisan sebelumnya untuk membentuk *output* akhir [12]. Setelah ekstraksi fitur, hasil klasifikasi data yang tergolong

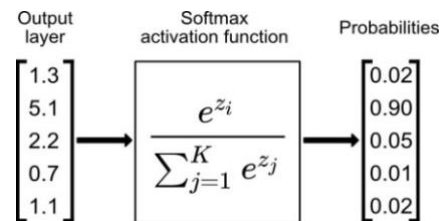
dalam berbagai kelas, dapat dilakukan dengan menggunakan *Fully-Connected (FC) neural network*.



Gambar. 4 Fully connected layer

C. Softmax

Fungsi aktivasi *Softmax*, juga dikenal sebagai *SoftArgMax* atau Fungsi Eksponensial Ternormalisasi. *Softmax* adalah fungsi aktivasi yang mengambil vektor bilangan real sebagai *input*, dan menormalkannya menjadi distribusi probabilitas yang sebanding dengan eksponensial dari bilangan *input*. Sebelum menerapkan, beberapa data *input* bisa negatif atau lebih besar dari 1, juga bisa tidak berjumlah 1. Setelah menerapkan *Softmax*, setiap elemen akan berada dalam kisaran 0 hingga 1, dan elemen akan bertambah hingga 1 [15]. Dengan cara ini bisa ditafsirkan sebagai distribusi probabilitas.



Gambar. 5 Softmax

D. Arsitektur DarkCovidNet

Arsitektur *DarkCovidNet* atau dapat disebut sebagai *Darknet-19* adalah pengklasifikasi model pembentuk dasar dari sistem deteksi waktu nyata yang bernama *YOLO (You Only Look Once)*. Sistem ini dirancang memiliki arsitektur canggih untuk dapat mendeteksi objek. *Darknet-19* terdiri dari 19 konvolusi lapisan dan 5 *pooling layer* menggunakan *Maxpool* [4]. Setiap lapisan memiliki hasil *filtering* angka yang berbeda dari jumlah nomor, ukuran dan nilai langkah yang berbeda. *Darknet-19* memiliki tata letak lapisan sebagai berikut:

$$C_1-M_1-C_2-M_2-C_3-C_4-C_5-M_3-C_6-C_7-C_8-M_4-C_9-C_{10}-C_{11}-C_{12}-C_{13}-M_5-C_{14}-C_{15}-C_{16}-C_{17}-C_{18}-C_{19}$$

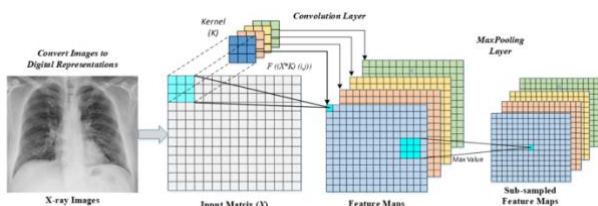
Huruf *C* menunjukkan *convolutional layer* dan *M* menunjukkan *Maxpooling layer*. Untuk sinyal *input X* (citra) dan kernel *K*, operasi konvolusi dua dimensi dapat didefinisikan, di persamaan 1:

$$(X * K)(i, j) = \sum_m \sum_n K(m, n)X(i - m, j - n) \tag{1}$$

dimana * mewakili operasi konvolusi diskrit. Matriks K merupakan matriks *input* dengan parameter langkah. Unit linier yang diperbaiki adalah *Leaky (Leaky ReLu)* digunakan sebagai fungsi aktivasi dalam arsitektur DarkNet. Perhitungan fungsi *Leaky ReLu* diberikan pada persamaan 2:

$$f(x) = \begin{cases} 0.01x & \text{for } x < 0 \\ x & \text{for } x \geq 0 \end{cases} \quad (2)$$

Presentasi skema untuk aliran data *input* dari *convolutional layer (C)* dan *Max-pooling (M)* dapat digambarkan seperti pada Gambar 6.

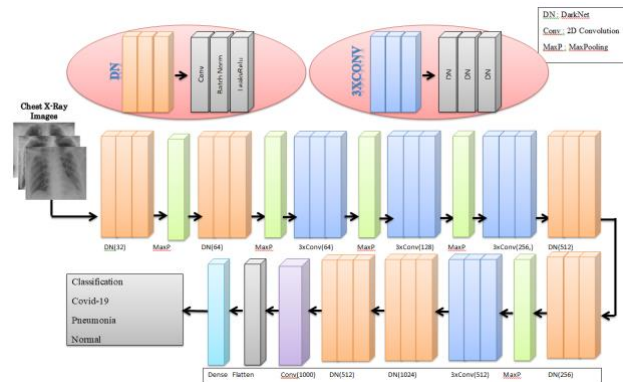


Gambar. 6 Presentasi skematis konvolusi dan max-pooling

Arsitektur yang diusulkan pada penelitian terdiri dari 19 *convolutional layer* dan 5 *maxpooling*. Tujuan menggunakan model ini adalah untuk bisa memberikan pengembangan model sehingga dapat memberikan diagnostik yang akurat untuk *multiclassification*. Diagram arsitektur pada Gambar 7, dimana lapisan *DN (DarkNet)* memiliki satu lapisan konvolusi yang diikuti oleh operasi *BatchNorm* dan *LeakyReLU*. Lapisan 3 x *Conv Layer* memiliki pengaturan yang sama sampai 3 kali [4].

Operasi *Batch Normalization* digunakan untuk standar *input* sehingga operasi ini memiliki manfaat untuk menyamakan distribusi nilai *input* yang selalu berubah. Contohnya untuk mengurangi waktu pelatihan dan meningkatkan stabilitas model. Dengan menggunakan *Batch Normalization* pada *convolutional layer* dapat meningkatkan akurasi pelatihan sehingga mempercepat nilai *loss* menuju konvergen [16].

LeakyReLU adalah variasi operasi *ReLU* yang digunakan untuk mencegah kematian neuron. Tidak seperti fungsi aktivasi *ReLU* atau *sigmoid* yang memiliki nilai 0 di bagian negatif dari hasil turunan. Penggunaan *LeakyReLU* sudah digunakan dari sejumlah penelitian dengan menggunakan $\alpha = 0,01$ dan memiliki nilai epsilon dengan kapabilitas kecil untuk bisa mengatasi masalah neuron yang kritis [16]. *Maxpooling* mengecilkan *input* dengan mengambil nilai maksimum suatu wilayah, dan menentukan jumlah filter. Pada proses ekstraksi fitur, semua *convolutional layer DarkNet* menggunakan fungsi aktivasi *Leaky Relu* $\alpha = 0,01$ kecuali pada *layer* terakhir menggunakan fungsi aktivasi *Flatten* [17].



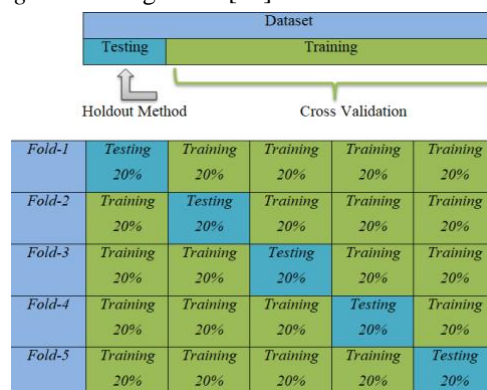
Gambar. 7 Arsitektur model DarkCovidNet

E. Keras

Keras digunakan untuk *library* pada *Python* yang bersifat *open source*, agar mudah digunakan untuk pengembangan dan penerapan *deep learning* [18]. *Library* pada Keras ini biasa digunakan untuk membungkus *library* perhitungan numerik yang lebih efisien. Sedangkan penerapan *TensorFlow* yang bisa digunakan untuk mendefinisikan dan melatih model *neural network* yang digunakan dalam beberapa baris *code*. *Tensorflow* adalah *opensource library* untuk *Machine Learning* yang digunakan dalam berbagai tugas.

F. K-Fold Cross Validation

K-fold merupakan salah satu metode *Cross Validation*, dimana bisa disebut sebagai sistem pembagian data yang menyilang (*cross*). Sistem pembagian data ini adalah salah satu dari berbagai teknik validasi yang digunakan untuk menilai bagaimana hasil statistik analisis yang digeneralisasi ke kumpulan data (*dataset*) [4]. Data *K-fold Cross Validation* ini menggunakan metode dengan memvalidasi data *training* untuk di *Cross Validation* sehingga data tersebut ketika sudah dilakukan validasi maka akan menjadi data *testing*. Data sebanyak *K* akan dilipat gandakan sesuai dengan iterasi jumlah *K* yang akan dipakai. Dalam teknik *K-fold Cross Validation*, *dataset* akan terbagi dalam data *training* dan data *testing*. Kemudian satu bagian dari data akan digunakan sebagai data *testing* dan data lainnya akan digunakan sebagai data *training* secara bergantian [19].



Gambar. 8 Cross validation

G. Hyperparameter Model

Hyperparameter adalah pengaturan nilai parameter sebelum proses pelatihan. Sama pentingnya dengan pembersihan data dan fitur ekstraksi, penyetelan parameter sangat penting untuk bisa mengatur fungsi aktivasi, penentuan efisiensi dan akurasi dalam pelatihan model. Hyperparameter sangat sensitif terhadap perubahan kecil dalam tingkat pembelajaran atau penaksiran yang akan mengarah pada perubahan besar dalam akurasi model [20]. Pemilihan parameter yang sesuai akan sangat mempengaruhi hasil yang diberikan pada pelatihan model. Maka dari itu akan bisa mencapai kinerja yang optimal. Pada pembangunan model DarkCovidNet, hyperparameter yang dilakukan yaitu jumlah dataset, jumlah epochs, batch size, algoritma optimasi Adam, loss, learning rate, reduce learning rate, earlystopping, dan callbacks. Epoch digunakan untuk menentukan berapa kali jumlah model neural network masuk ke dalam dataset. Batch Size pada hyperparameter termasuk faktor terpenting karena digunakan untuk menentukan jumlah bagian dataset/ jumlah data poin yang ada dalam satu batch/ bagian (subset). Semakin tinggi ukuran batch maka semakin banyak ruang memori yang akan diperlukan [21].

Learning rate (LR) adalah istilah yang digunakan dalam Machine Learning dan statistik. LR mengacu pada tingkat dimana suatu algoritma menyatu dengan memberikan solusi [22]. Learning rate juga termasuk faktor penting dari salah hyperparameter yang digunakan untuk train data.

H. Optimasi Adam

Algoritma optimasi yang digunakan pada penelitian ini adalah Adam. Adam merupakan algoritma optimasi yang berguna sebagai pengganti prosedur penurunan gradien stokastik klasik untuk memperbarui bobot jaringan iteratif berdasarkan data pelatihan dengan lebih efisien [23]. Metode ini menghitung tingkat pembelajaran adaptif individu untuk parameter yang berbeda dari perkiraan momen pertama dan kedua dari gradien. Optimasi Adam sangat sesuai untuk diterapkan secara langsung untuk model apapun, terkait dari banyaknya data dan parameter yang besar. Penyetelan yang digunakan mudah untuk disesuaikan tingkat pembelajaran adaptif tarif terpisah dari setiap parameter, sehingga dapat mempertahankan tingkat pembelajaran tunggal untuk semua bobot pembaruan yang tidak berubah selama pelatihan [24].

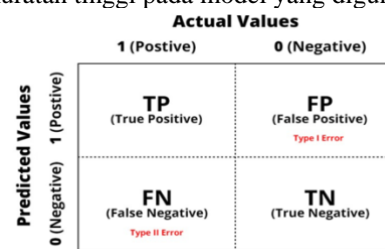
I. Reduce Learning Rate

Reduce learning rate digunakan untuk matriks yang dilakukan pemberhentian sehingga dapat memberikan peningkatan performansi model. Untuk itu model reduce learning rate ini berguna untuk memberikan manfaat dalam pengurangan tingkat pembelajaran (learning rate) dengan faktor 2-10 setelah pembelajaran meningkat [25]. Pada Reduce learning rate atau bisa disebut sebagai ReduceLRonPlateau, sangat bisa menyesuaikan dengan data 5-fold cross validation karena data bisa menyatu lebih

cepat sambil mempertahankan loss score (skor kerugian) yang serupa.

J. Matrik Evaluasi

Evaluasi model dilakukan untuk mengetahui perbandingan hasil klasifikasi sistem model dengan hasil klasifikasi sebenarnya. Evaluasi matrik termasuk pada evaluasi dengan spesifikasi tinggi dalam mengukur nilai F1-Score. Pada Gambar 9 merupakan confusion matrix yang digunakan untuk memeriksa kinerja berdasarkan klasifikasi model untuk Machine Learning [26]. Confusion matrix digunakan sebagai alat ukur kinerja Accuracy dan F1-Score. F1-Score adalah salah satu pengukur kinerja model yang menggabungkan precision dan recall sehingga dapat meningkatkan tingkat prediksi dengan memberikan hasil keakuratan tinggi pada model yang digunakan [27].



Gambar. 9 Binary confusion matrix

Akurasi digunakan untuk mengetahui seberapa tingkat keakuratan sistem yang dapat mengklasifikasikan data tersebut dengan benar. Akurasi dapat disebut sebagai nilai banding antara data yang sudah diklasifikasi benar dengan keseluruhan data. Selain itu, akurasi juga sebagai ukuran berapa banyak prediksi yang benar dari model yang digunakan [26]. Akurasi merupakan matrik dasar yang baik untuk mengukur performa model. Yaitu nilai akurasi dapat diperoleh pada persamaan 3.

$$Akurasi\ Binary = \frac{TP+TN}{TP+TN+FP+FN} \times 100\% \tag{3}$$

Untuk bisa menghitung akurasi multiclass confusion matrix digunakan persamaan 4.

$$Akurasi\ Multi = \frac{\sum T}{\sum T + \sum F} \tag{4}$$

$$Precision = \frac{TP_i}{TP_i + FP_i} \tag{5}$$

$$Recall = \frac{TP_i}{TP_i + FN_i} \tag{6}$$

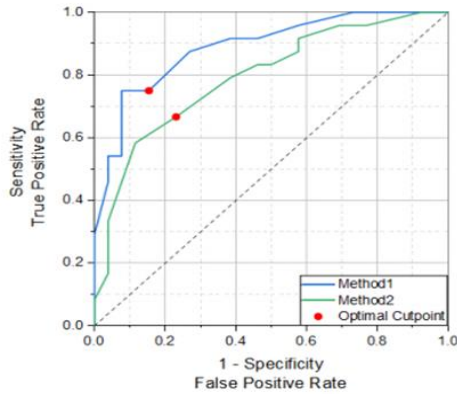
$$F1 - Score = \frac{2 \times precision \times recall}{precision + recall} \tag{7}$$

Nilai terbaik dari F1-Score adalah 1, sedangkan nilai terendah adalah 0.

K. Receiver Operating Characteristic (ROC)

Receiver Operating Characteristic (ROC) Curve yang didefinisi sebagai plot dari uji sensitivitas sebagai koordinat y sebagai True Positive Rate (TPR) yang membandingkan dengan 1 - spesifisitas atau False Positive

Rate (FPR) sebagai koordinat x , adalah metode yang efektif untuk mengevaluasi kinerja dari tes diagnostic [28].



Gambar. 10 Contoh ROC curve

Dari beberapa indeks pengukuran terkait dengan ROC Curve, salah satu pengukuran yang banyak digunakan adalah *Area Under ROC Curve* (AUC) yang merupakan pengukuran gabungan dari sensitivitas dan spesifisitas [28]. *Area Under ROC* (AUC) adalah ukuran dari kinerja keseluruhan dari tes diagnostik dan ditafsirkan sebagai nilai rata-rata sensitivitas untuk semua nilai yang mungkin dari kekhususan nilai diantara nilai 0 dan 1.

Rumus Sensitivity dan Specificity dapat dilihat pada persamaan 5 dan 6.

$$Sensitivity = \frac{TP}{TP+FN} \tag{5}$$

$$Specificity = \frac{TN}{TN+FP} \tag{6}$$

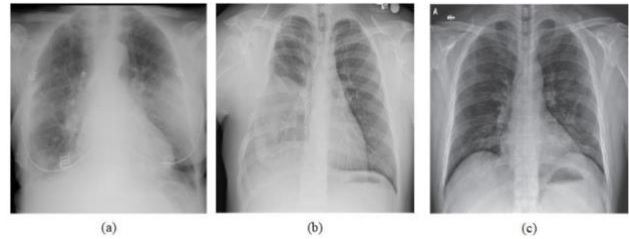
III. METODE PENELITIAN

A. Dataset

Dataset penelitian ini menggunakan dataset citra X-ray dada dari radiografi yang diambil dari website: <https://www.kaggle.com/datasets/tawsifurrahman/covid19-radiography-database>. Data citra yang diambil merupakan data file citra dengan ekstensi png dan total jumlah data yang diambil sebanyak 4.945 dengan 3 kelas, yaitu Covid-19, Normal dan *Pneumonia*. Tabel 1 menunjukkan jumlah data yang ada di Kaggle dan data yang digunakan pada penelitian ini. Dengan menggunakan data Normal 1800 citra, dan Covid-19 1.800 citra supaya dilakukan penyeimbangan data dengan data *Pneumonia* sebanyak 1.345 citra. Selain itu pada Tabel 1 juga menunjukkan keseluruhan data sebesar 18081 citra, tetapi untuk data *Pneumonia* hanya 1345 citra. Untuk itu bisa diasumsikan penyeimbangan data yang ada di data Covid-19 yaitu 10192 citra diambil sekian persen untuk bisa menyesuaikan data dengan data *Pneumonia*. Oleh karena itu supaya bisa mendapatkan hasil yang optimal data Covid-19 dan *Pneumonia* diambil dari distribusi yang bisa menyesuaikan dengan data *Pneumonia* [29].

TABEL I
DATASET DETAIL

Kelas	Jumlah data	Jumlah digunakan
Normal	3616	1800
<i>Pneumonia</i>	1345	1345
Covid-19	10192	1800
Total	18081	4945



Gambar. 11 Citra X-ray radiografi dada

Pada Gambar 11 menunjukkan contoh citra X-ray dada. Data diklasifikasikan menjadi 3 kelas: bagian (a) adalah dataset citra Covid-19, bagian (b) merupakan dataset citra Normal (No-Findings), dan bagian (c) yaitu dataset citra *Pneumonia*.

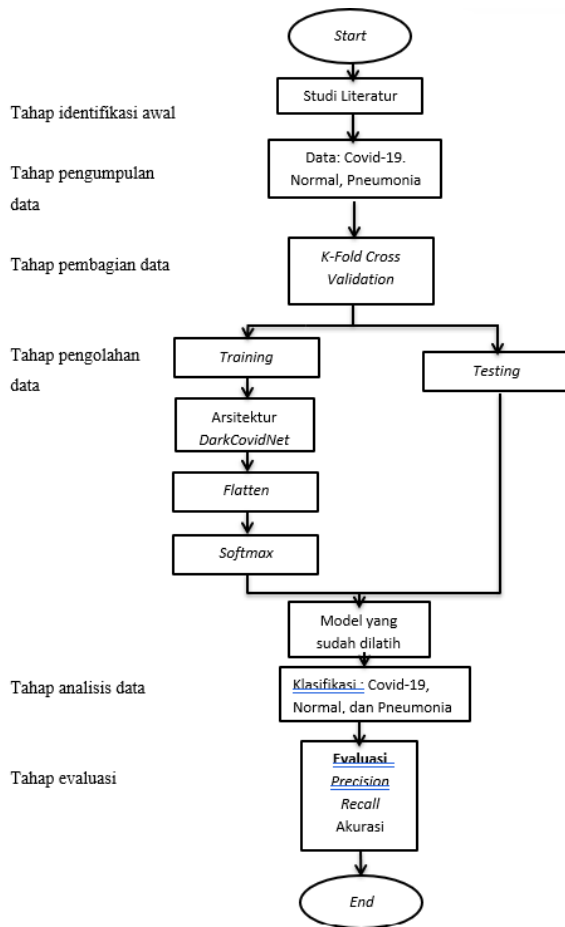
B. Pembagian Dataset

Input data yang digunakan berasal dari *dataset* citra dari <https://www.kaggle.com/tawsifurrahman/covid19-radiography-database> yang tentunya sudah dilakukan oleh sejumlah peneliti sebelumnya. Citra *input* berukuran 256x256 yang terbagi menjadi 3 kelas yaitu Covid-19, *Pneumonia* dan Normal. Untuk mendapatkan hasil evaluasi terbaik, pada penelitian ini menggunakan *5-fold Cross Validation* artinya data diacak menjadi 5 bagian untuk bisa melakukan proses *train* dan *test* supaya diuji kinerja modelnya agar bisa lebih efisien [19]. Pembagian *dataset* yang dilakukan memberikan hasil *validation* dan *validation loss*. Dari hasil tersebut dapat diketahui bahwa nilai *trainloss*, *validation*, dan *accuracy* memberikan nilai terbaik [4][6]. Pada Tabel 2 menunjukkan persebaran data citra yang digunakan dalam uji coba penelitian ini.

TABEL II
PEMISAHAN DATA TRAIN DAN TEST

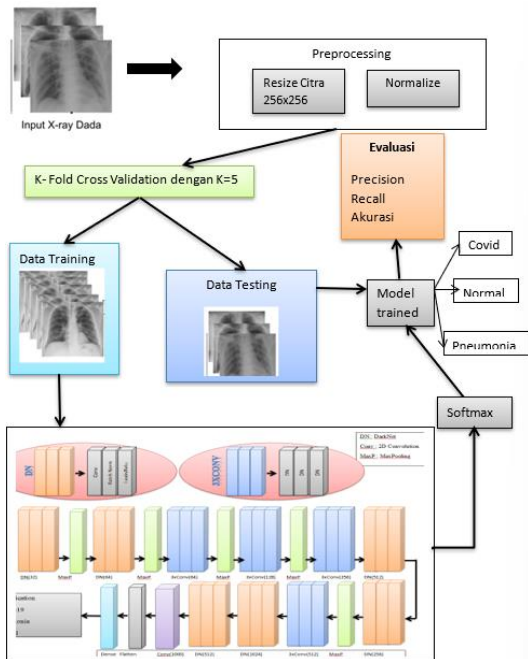
Data	Covid	Normal	<i>Pneumonia</i>	Jumlah	Rasio
<i>Train</i>	1440	1440	1076	3956	80%
<i>Test</i>	330	330	269	989	20%

C. Tahapan Penelitian



Gambar. 12 Flowchart tahapan penelitian

D. Arsitektur Sistem Training dan Testing



Gambar. 13 Arsitektur training dan testing

Model DarkCovidNet dapat dilihat pada Gambar 7 sedangkan ilustrasi lengkap tahapan penelitian ini ditunjukkan pada Gambar 12. Tahapan arsitektur sistem terdiri dari beberapa proses, yaitu:

1. Melakukan *load data* untuk bisa memasukkan citra X-ray yang digunakan.
2. Selanjutnya dilakukan tahapan *preprocessing*. Setiap citra X-ray di *resize* menjadi ukuran 256x256.
3. Sesuai dengan persentase pada *5-fold Cross Validation*, citra dari *dataset* X-ray dada yang telah diberi label kelas, dibagi menjadi data *training* dan *testing*.
4. Pada proses *training*, dilakukan beberapa tahapan diantaranya:
 - a. Pada proses pelatihan akan dilakukan proses uji coba, yang hanya menggunakan proses optimasi Adam dengan *batch size* 64, *learning rate* 0,001 dan *Categorical Cross-Entropy loss* untuk pengukuran keakuratan model selama proses *train* yang berguna untuk pengukuran *multi-classification*. Pada uji coba lainnya juga menggunakan penggabungan optimasi Adam dan algoritma *reduce learning rate* dimana akan dijalankan proses *reduce* pada *learning rate* yang digunakan, sesuai dengan estimasi yang ditetapkan. Pada estimasi algoritma *reduce learning rate* ini, berjalan ketika tingkat *patience* (kesabaran) sama dengan 3, atau bisa disebut 3 *epochs*. Ketika 3 *epochs* tersebut tidak ada *improvement* (perkembangan) maka *reduce learning rate* ini akan berjalan sesuai dengan estimasi yang sudah ditentukan. Faktor yang digunakan 0,5 maka *new learning rate* sama dengan 0,001 dikalikan dengan 0,5 dan ini akan terus berjalan sesuai dengan algoritma penetapan *reduce learning rate* yang digunakan.
 - b. Setelah proses pelatihan, dilanjutkan pada citra yang nantinya akan diproses menggunakan arsitektur *DarkCovidNet* untuk menentukan klasifikasi.
 - c. Hasil dari proses tersebut akan membentuk model yang digunakan dalam pengujian. Model yang dilatih akan digabungkan dengan *fully-connected-layer*.
5. Pada proses *testing*, dilakukan beberapa tahapan diantaranya:
 - a. Hasil dari proses uji coba menggunakan arsitektur *DarkCovidNet* akan diproses kembali dengan *trained* model pada tahapan *training* untuk membentuk 3 luaran kelas klasifikasi.
 - b. Hasil akhir yang didapatkan berupa klasifikasi kelas citra Covid-19, Normal dan *Pneumonia*.

E. Skenario Uji Coba

Skenario uji coba dilakukan untuk mengetahui jawaban dari masalah terkait dengan akurasi klasifikasi Covid-19 dari pemodelan arsitektur *DarkCovidNet* pada citra X-ray dada. Skenario ini dilakukan untuk memperoleh kinerja model dengan akurasi yang terbaik. Untuk *hyperparameter* yang digunakan pada skenario tersebut menggunakan *batch size* 64, dengan *learning rate default* 0,001 [20], [21].

Berikut hasil skenario uji coba yang dapat dilihat pada Tabel 3.

TABEL III
SKENARIO UJI COBA

Skenario	Arsitektur	Optimasi	Learning rate	HL
1	DarkCovidNet	Adam	-	1
2	DarkCovidNet	Adam	RL	1
3	DarkCovidNet	Adam	RL	2
4	DarkCovidNet	Adam	RL	3

Keterangan:

(-): Default Learning rate (0,001)

RL: Reduce Learning Rate

HL: Hidden Layer

IV. HASIL DAN PEMBAHASAN

Dari uji coba pada model yang telah dilakukan, didapatkan hasil uji coba menggunakan penggabungan model Arsitektur *DarkCovidNet*, Optimasi Adam, *Reduce Learning rate*, dan *Hidden Layer* yang terdapat pada skenario uji coba 1 - 4. Tabel 4 – 7 menunjukkan hasil uji setiap skenario pengujian.

TABEL IV
SKENARIO UJI COBA 1

Hasil (%)	fold 1	fold 2	fold 3	fold 4	fold 5	Rata-rata
FI-score	90,22	85,25	89,27	90,35	88,53	88,72
Accuracy	89,58	84,32	88,77	90,29	88,27	88,25
AUC	97,6	94,2	96,9	98,1	96,8	96,68

TABEL V
SKENARIO UJI COBA 2

Hasil (%)	fold 1	fold 2	fold 3	fold 4	fold 5	Rata-rata
FI-score	94,54	95,36	95,04	96,26	95,13	95,26
Accuracy	94,43	95,34	95,04%	96,25	95,04	95,22
AUC	99,1	99,4	99,3	99,4	99,0	99,24

TABEL VI
SKENARIO UJI COBA 3

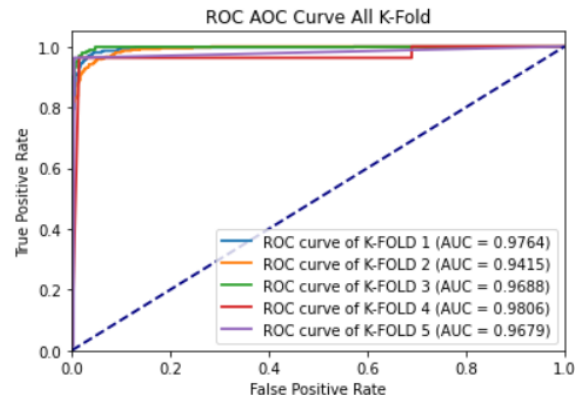
Hasil (%)	fold 1	fold 2	fold 3	fold 4	fold 5	Rata-rata
FI-score	96,35	96,29	95,13	95,76	94,51	95,61
Accuracy	96,25	96,25	95,04	95,75	94,44	95,35
AUC	99,6	99,5	99,3	99,5	99,2	99,42

TABEL VII
SKENARIO UJI COBA 4

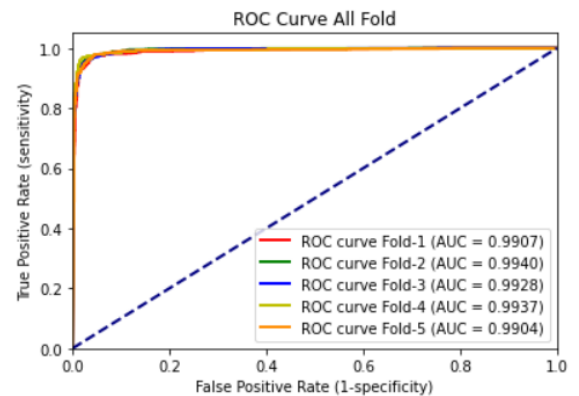
Hasil (%)	fold 1	fold 2	fold 3	fold 4	fold 5	Rata-rata
FI-score	94,97	94,86	96,29	96,53	96,74	95,89
Accuracy	94,94	94,84	96,25	96,56	96,66	95,85
AUC	99,2	99,5	99,5	99,7	99,5	99,48

Dari Tabel 7, dapat diketahui bahwa skenario uji coba ke 4 memberikan hasil yang lebih optimal dibandingkan skenario 1 - 3. Skenario uji coba ke-4 menunjukkan hasil rata-rata *FI-Score* sebesar 95,89% dan nilai terbaik pada *fold 5* yaitu 96,74%. Sementara itu, hasil akurasi rata-rata sebesar 95,85% dan akurasi terbaik pada *fold 5* yaitu

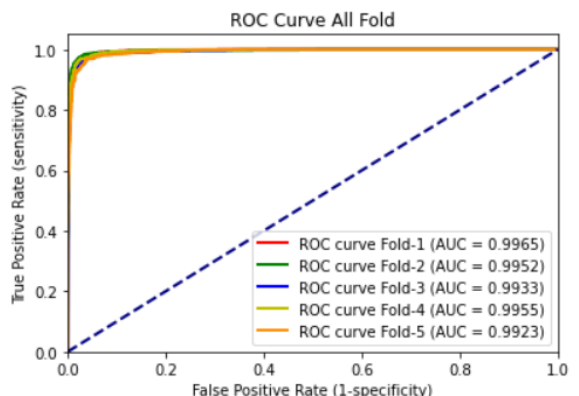
96,66%. Sedangkan hasil rata-rata AUC adalah 99,48% dan AUC terbaik pada *fold 4* yaitu sebesar 99,7%. Oleh karena itu skenario ke 4 yang menggunakan arsitektur *DarkCovidNet* dengan optimasi Adam, *reduce learning rate*, dan 3 *hidden layer* memberikan hasil yang optimal dalam skenario uji coba yang dilakukan. Sementara untuk melihat performa uji coba pada setiap skenario model dapat dilihat pada Gambar 14 - 17.



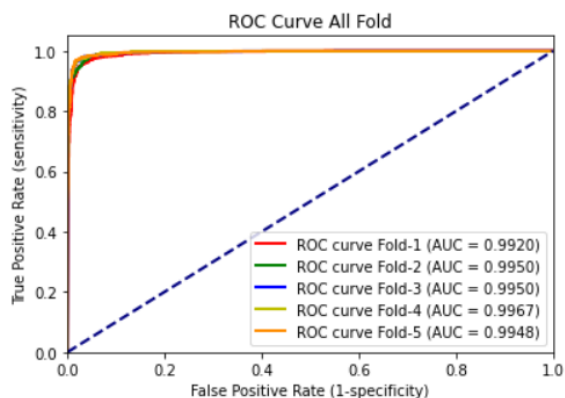
Gambar. 14 Kurva AUC skenario 1



Gambar. 15 Kurva AUC skenario 2



Gambar 16 Kurva AUC skenario 3



Gambar 17 Kurva AUC skenario 4

Pada hasil kurva AUC dapat dilihat bahwa koordinat x menunjukkan *False Positive Rate* atau biasa disebut sebagai *specificity* sedangkan koordinat y menunjukkan *True Positive Rate* atau biasa disebut dengan *sensitivity*. Masing-masing skenario memberikan hasil rata-rata akurasi diatas 90%. Tetapi untuk hasil yang terbaik dari rata-rata *F1-Score*, *Accuracy*, dan AUC adalah skenario ke-4 yang menggunakan penggabungan optimasi Adam, *reduce learning rate*, dan 3 *hidden layer*. Hasil uji coba tersebut memperoleh nilai rata-rata akurasi, *F1-score* dan AUC sebesar 95,85%, 95,89%, dan 99,48%, Sehingga dapat diketahui bahwa performa uji model tersebut dikatakan baik dan optimal.

V. KESIMPULAN

Dalam penelitian ini metode *DarkCovidNet* telah mampu mengklasifikasikan Covid-19 dengan hasil rata-rata akurasi dari 5-fold *Cross Validation* sebesar 95,85%. Nilai ini diperoleh menggunakan penggabungan optimasi Adam, *reduce learning rate*, dan 3 *hidden layer*. Selain itu, dengan *tuning* parameter tersebut diperoleh rata-rata *F1-score* dan AUC masing-masing sebesar 95,89% dan 99,48%. Oleh karena itu penggunaan algoritma optimasi Adam, *reduce learning rate*, dan jumlah *hidden layer* yang tepat akan sangat memberikan dampak pada kinerja model yang diterapkan. Dengan melakukan *reduce learning rate*, semakin mengalami penurunan (*reduce*) maka akan menambahkan jumlah komputasi yang dibutuhkan. Dengan kata lain, memberikan proses waktu yang lebih lama untuk estimasi waktu yang dijalankan ketika melakukan proses *training*.

Saran untuk penelitian selanjutnya antara lain dengan melakukan perubahan parameter sehingga akan mempengaruhi hasil training data, dan perbandingan antara *optimizer* yang digunakan untuk bisa mengetahui optimasi apa yang lebih baik pada model tersebut. Serta perubahan jumlah *hidden layer* yang nantinya sudah ditentukan akan sangat berpengaruh dalam menghitung jumlah hasil akurasi yang dihasilkan.

UCAPAN TERIMA KASIH / ACKNOWLEDGMENT

Ucapan terima kasih disampaikan kepada Rumah Sakit Universitas Airlangga (RSUA), RSPHC Surabaya dan repositori Kaggle yang telah menyediakan dataset *x-ray* dada untuk uji coba metode pada penelitian ini. Ucapan terimakasih juga disampaikan kepada Direktorat Riset, Teknologi dan Pengabdian kepada Masyarakat – Pendidikan Tinggi (DRTPM-Dikti) dengan Nomor kontrak (induk) 069/E5/PG.02.00.PL/2023 dan kontrak (turunan) 2630/UN46.4.1/PT.01.03/2023.

REFERENSI

- [1] W. Swastika, P. Studi, T. Informatika, and P. Korespondensi, "Studi Awal Deteksi Covid-19 Menggunakan Citra CT Berbasis Deep Preliminary Study of Covid-19 Detection Using CT Image Based on," *J. Teknol. Inf. dan Ilmu Komput.*, vol. 7, no. 3, pp. 629–634, 2020, doi: 10.25126/jtiik.202073399.
- [2] N. Yudistira, A. W. Widodo, and B. Rahayudi, "Deteksi Covid-19 pada Citra Sinar-X Dada Menggunakan Deep Learning yang Efisien," *J. Teknol. Inf. dan Ilmu Komput.*, vol. 7, no. 6, p. 1289, 2020, doi: 10.25126/jtiik.2020763651.
- [3] M. Malik, T. Masood, and M. Kamran, "Towards Covid-19 Detection in X-ray images using Convolutional Neural Network," *Int. J. Adv. Trends Comput. Sci. Eng.*, vol. 10, no. 3, pp. 2501–2507, 2021, doi: 10.30534/ijatcse/2021/1411032021.
- [4] T. Ozturk, M. Talo, E. A. Yildirim, U. B. Baloglu, O. Yildirim, and U. Rajendra Acharya, "Automated detection of COVID-19 cases using deep neural networks with X-ray images," *Comput. Biol. Med.*, vol. 121, no. April, 2020, doi: 10.1016/j.compbiomed.2020.103792.
- [5] X. Yang, X. He, J. Zhao, Y. Zhang, S. Zhang, and P. Xie, "COVID-CT-Dataset: A CT Scan Dataset about COVID-19," no. March, 2020, [Online]. Available: <http://arxiv.org/abs/2003.13865>
- [6] M. Z. Islam, M. M. Islam, and A. Asraf, "A combined deep CNN-LSTM network for the detection of novel coronavirus (COVID-19) using X-ray images," *Informatics Med. Unlocked*, vol. 20, p. 100412, 2020, doi: 10.1016/j.imu.2020.100412.
- [7] B. D. Satoto, M. I. Utoyo, R. Rulaningtyas, and E. B. Koendhori, "Custom convolutional neural network with data augmentation to predict Pneumonia COVID19," *IBIOMED 2020 - Proc. 37th Int. Conf. Biomed. Eng.*, pp. 71–76, 2020, doi: 10.1109/IBIOMED50285.2020.9487567.
- [8] T. D. Pham, "A comprehensive study on classification of COVID-19 on computed tomography with pretrained convolutional neural networks," *Sci. Rep.*, vol. 10, no. 1, pp. 1–8, 2020, doi: 10.1038/s41598-020-74164-z.
- [9] L. Wang, Z. Q. Lin, and A. Wong, "COVID-Net: a tailored deep convolutional neural network design for detection of COVID-19 cases from chest X-ray images," *Sci. Rep.*, vol. 10, no. 1, pp. 1–12, 2020, doi: 10.1038/s41598-020-76550-z.
- [10] S. Samsir, J. H. P. Sitorus, Zulkifli, Z. Ritonga, F. A. Nasution, and R. Watrionthos, "Comparison of machine learning algorithms for chest X-ray image COVID-19 classification," *J. Phys. Conf. Ser.*, vol. 1933, no. 1, 2021, doi: 10.1088/1742-6596/1933/1/012040.
- [11] M. M. Krishna, M. Neelima, M. Harshali, and M. V. G. Rao, "Image classification using Deep learning," *Int. J. Eng. Technol.*, vol. 7, no. August, pp. 614–617, 2018, doi: 10.14419/ijet.v7i2.7.10892.
- [12] S. Indolia, A. K. Goswami, S. P. Mishra, and P. Asopa, "Conceptual Understanding of Convolutional Neural Network- A Deep Learning Approach," *Procedia Comput. Sci.*, vol. 132, pp. 679–688, 2018, doi: 10.1016/j.procs.2018.05.069.
- [13] K. O'Shea and R. Nash, "An Introduction to Convolutional Neural Networks," no. December, 2015, [Online]. Available: <http://arxiv.org/abs/1511.08458>
- [14] G. T. Indrawan, A. Nilgiri, and H. A. Al Faruq, "Diagnosis COVID-19 Berdasarkan Citra X-ray Paru-Paru Menggunakan Metode Convolutional Neural Network," *Diagnose COVID-19*

- Based X-ray Image Lungs Using Convolutional Neural Netw.*, 2020.
- [16] F. T. A. Muhammad Alfin Jimly Asshiddiqie, Basuki Rahmat, "Deteksi Tanaman Tebu Pada Lahan Pertanian Menggunakan Metode Convolutional Neural Network," *Inform. dan Sist. Inf.*, vol. 1, no. 1, pp. 229–237, 2020.
- [17] L. Agustien *et al.*, "Real-time Deteksi Masker Berbasis Deep Learning menggunakan Algoritma CNN YOLOv3," vol. 8, no. 2, pp. 129–137, 2021.
- [18] A. Santoso and G. Ariyanto, "Implementasi Deep Learning berbasis Keras untuk Pengenalan Wajah," *Emit. J. Tek. Elektro*, vol. 18, no. 1, pp. 15–21, 2018, doi: 10.23917/emitor.v18i01.6235.
- [19] R. Mohammadizi and M. M. Bilec, "Application of machine learning for predicting building energy use at different temporal and spatial resolution under climate change in USA," *Buildings*, vol. 10, no. 8, 2020, doi: 10.3390/BUILDINGS10080139.
- [20] P. P. Ippolito, "Hyperparameter Tuning," no. August, pp. 231–251, 2022, doi: 10.1007/978-3-030-88389-8_12.
- [21] I. Kandel and M. Castelli, "The effect of batch size on the generalizability of the convolutional neural networks on a histopathology dataset," *ICT Express*, vol. 6, no. 4, pp. 312–315, 2020, doi: 10.1016/j.ict.2020.04.010.
- [22] G. S. Na, "Efficient learning rate adaptation based on hierarchical optimization approach," *Neural Networks*, vol. 150, pp. 326–335, 2022, doi: 10.1016/j.neunet.2022.02.014.
- [23] Rizwan Muhammad, "Adam Optimization Algorithm | engMRK," no. June, pp. 4–6, 2018, [Online]. Available: <https://engmrk.com/adam-optimization-algorithm/>
- [24] D. P. Kingma and J. Lei Ba, "15iclr-ADAM," *Iclr*, pp. 1–15, 2015, [Online]. Available: <https://arxiv.org/pdf/1412.6980.pdf> %22 entire document
- [25] A. Al-Kababji, "Scheduling Techniques for Liver Segmentation: ReduceLRonPlateau Vs OneCycleLR", [Online]. Available: <http://www.qf.org.qa>
- [26] K. M. Ting, "Confusion Matrix," *Encycl. Mach. Learn. Data Min.*, no. October, pp. 260–260, 2017, doi: 10.1007/978-1-4899-7687-1_50.
- [27] M. R. Lascu, "Deep Learning in Classification of Covid-19 Coronavirus, Pneumonia and Healthy Lungs on CXR and CT Images," *J. Med. Biol. Eng.*, vol. 41, no. 4, pp. 514–522, 2021, doi: 10.1007/s40846-021-00630-2.
- [28] S. H. Park, J. M. Goo, and C. H. Jo, "Receiver operating characteristic (ROC) curve: Practical review for radiologists," *Korean J. Radiol.*, vol. 5, no. 1, pp. 11–18, 2004, doi: 10.3348/kjr.2004.5.1.11.
- [29] A. Saifudin, U. Pamulang, R. S. Wahono, U. Dian, and N. Semarang, "Pendekatan Level Data untuk Menangani Ketidakseimbangan Kelas pada Prediksi Cacat Software," *J. Softw. Eng.*, vol. 1, no. 2, pp. 76–85, 2015.