

PERANCANGAN MODEL PENGENALAN CITRA DIGITAL TOMOGRAFI PARU-PARU UNTUK DETEKSI COVID-19 MENGGUNAKAN JARINGAN SARAF TIRUAN

MODEL DESIGN OF THE IMAGE RECOGNITION OF LUNG CT SCAN FOR COVID-19 DETECTION USING ARTIFICIAL NEURAL NETWORK

Maulana Akbar Dwijaya¹, Umar Ali Ahmad², Rudi Purwo Wijayanto³

^{1,2,3} Universitas Telkom, Bandung

maulanaakbardwijaya@student.telkomuniversity.ac.id¹, umar@telkomuniversity.ac.id²,
rudipurwo@365.telkomuniversity.ac.id³

Abstrak

Penyakit COVID-19 atau Coronavirus telah menjadi pandemi global dan menjadi masalah utama yang harus segera dikendalikan. Salah satu cara yang dapat dilakukan adalah memutus rantai penyebaran virus tersebut dengan melakukan deteksi dan melakukan karantina atau isolasi diri. Pencitraan digital dengan menggunakan pemindaian tomografi komputer (*CT scan*) dapat dijadikan alternatif dalam mempelajari COVID-19. Pencitraan digital dengan pemindaian tomografi komputer (*CT scan*) dianggap mampu menggambarkan kondisi paru-paru pada pasien terinfeksi COVID-19 dan dapat menjadi alat bantu diagnosa klinis. Pada penelitian ini, melakukan klasifikasi COVID-19 dengan mengenali gambar pada pemindaian tomografi komputer (*CT scan*) paru-paru dengan menggunakan teknik dalam pengolahan citra digital dan ekstraksi fitur GLCM dilanjutkan pembuatan model jaringan saraf tiruan sehingga citra *CT scan* paru-paru dapat diklasifikasi oleh model. Hasil pada penelitian ini mendapatkan model yang paling optimum untuk klasifikasi COVID-19 dengan performansi *accuracy* sebesar 93%, *precision* sebesar 90%, *recall* sebesar 96% dan *F1-score* sebesar 93%.

Kata Kunci: COVID-19, pemindaian tomografi komputer (*CT scan*) paru-paru, pengolahan citra digital, jaringan saraf tiruan.

Abstract

COVID-19 or Coronavirus disease has become a global pandemic and is a major problem that must be controlled immediately. One of the ways that can be done to stop it from spreading is to break the spreading chain of the virus by detecting it and carrying out self-quarantine or isolation. Digital imaging using computed tomography (*CT scan*) can be used as an alternative in studying COVID-19. Digital imaging with computed tomography (*CT scan*) is considered to be able to describe the condition of the lungs in patients infected with COVID-19 and can be a clinical diagnostic tool. In this research, classifying COVID-19 by recognizing images on a computer tomography scan (*CT scan*) of the lungs using techniques in digital image processing and GLCM feature extraction, followed by making an artificial neural network model so that *CT scan* images of the lungs can be classified by model. The results in this study obtained the most optimum model for the classification performance of COVID-19 with 93% accuracy, 90% precision, 96% recall and 93% *F1-score*.

Keywords: COVID-19, computer tomography scan (*CT scan*) of the lungs, digital image processing, artificial neural networks.

1. Pendahuluan

Perkembangan penelitian dengan menggunakan pengolahan citra digital (*image processing*) banyak diterapkan diberbagai bidang, salah satunya di bidang kesehatan atau biomedis. Di dalam bidang kesehatan atau biomedis pengolahan citra biasanya digunakan untuk mendeteksi suatu penyakit dari seseorang. COVID-19 (*Coronavirus Disease 2019*) adalah sebuah virus baru yang teridentifikasi di Wuhan, China untuk pertama kali, kasusnya terjadi ketika peningkatan kasus infeksi pneumonia atau radang paru-paru misterius [1]. Teknik pencitraan digital seperti pemindaian tomografi komputer (*CT scan*) dan X-ray memiliki peran penting dalam mendiagnosis awal dan proses pengobatan COVID-19 [2].

Didalam teknologi pengolahan citra digital dalam medis, pemindaian tomografi komputer (*CT scan*) merupakan teknologi non-invasif yang dipilih, karena dalam untuk deteksi porositas internal dan karakterisasi dianggap sebagai teknik yang menjanjikan [3]. Metode pencitraan digital dari pemindaian tomografi komputer (*CT scan*) untuk yang terinfeksi COVID-19 dapat dikenali dengan terekamnya *ground glass opacity* dan *consolidation*. Dalam penelitian ini penulis merancang dan menganalisis sebuah model pengenalan citra yang dapat mendeteksi COVID-19 menggunakan citra *CT scan* paru-paru dari seorang pasien. Pengenalan citra digital yang digunakan dalam penelitian ini menggunakan algoritma jaringan saraf tiruan.

2. Landasan Teori

2.1 Pencitraan Klinis COVID-19

COVID-19 sebuah penyakit virus baru yang juga dikenal dengan SARS-CoV-2 yang menyebabkan pandemi global dan menjadi *public health emergency of international concern* (PHEIC). Gejala COVID-19 yang paling umum biasanya adalah demam, kelelahan, dan batuk kering, gejala ini menunjukkan gejala yang ringan dan dimulai secara bertahap [4]. Orang yang terinfeksi COVID-19 dapat memberikan hasil tes positif asam nukleat virus atau protein virus dengan spesifisitas yang kuat dan sensitivitas yang lemah [5].

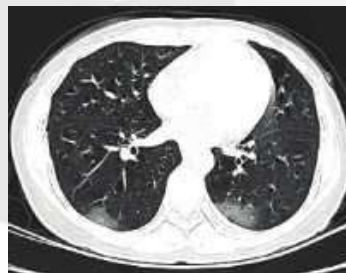
Pengujian dengan teknologi *CT scan* berfungsi sebagai skrining dan dasar diagnostik untuk deteksi seseorang menderita COVID-19, pencitraan dada bagian paru-paru pada tahap awal memperlihatkan bayangan *multiple* plak kecil dan perubahan interstitial, sebagian besar terlihat di *perifer* paru-paru dan *subpleural*, dan kemudian berkembang menjadi bayangan *multiple ground-glass* dan *infiltrate* di kedua paru. Pada kasus yang berat, dapat ditemukan konsolidasi paru bahkan “*white-lung*”, dengan efusi pleura dan pembengkakan kelenjar getah bening [5][6].

2.1.1 Image features COVID-19

Ketika pasien memiliki gejala COVID-19, seperti demam, batuk, atau sesak napas, mereka mungkin menjalani rontgen dada atau melakukan *CT scan* untuk mendiagnosis terinfeksi virus COVID-19 atau tidak. Temuan abnormal yang paling umum terlihat pada gambaran pada *CT scan* umumnya menunjukkan *ground glass opacity* (GGO), konsolidasi dan *air bronchogram*, *crazy-paving stone* [7]. Pemeriksaan *CT scan* digunakan untuk diagnosis COVID-19 dan selanjutnya dilakukannya tindakan terkait perawatan dan isolasi tepat waktu [8].

2.1.2 Ground Glass Opacity (GGO) sign

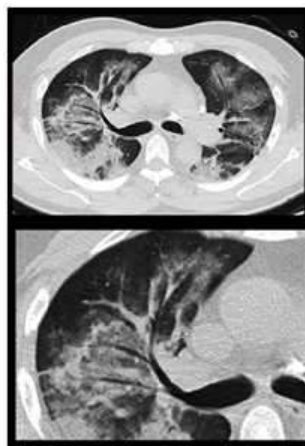
GGO (*Ground Glass Opacity*) adalah temuan yang tidak spesifik berupa gambaran buram dengan kepadatan yang sedikit lebih tinggi di paru-paru pada radiografi dan *computed tomography* (*CT scan*) scan yang dimana pembuluh darah pulmonalis terlihat. Karena invasi dari virus pada epitel alveolar dan bronkiolus berakibat hal ini terjadi. Kavitas alveolar bocor diakibatkan virus yang bereplikasi di sel epitel, sehingga dinding alveolus mengalami menebal dan inflamasi, Distribusinya di sekitar paru-paru dan di bawah pleura [9].



Gambar 2.1 GGO di lobus bawah bilateral [8]

2.1.3 Konsolidasi dan *air bronchogram* sign

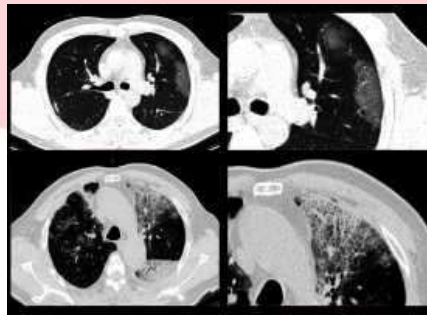
Di saat terjadi inflamasi berlangsung ada keterlibatan alveolus dan *mucosal ulcers*, diikuti oleh konsolidasi, saat tubuh bereaksi dengan *inflammatory storm*, muncul banyak eksudat dalam alveolus di paru-paru, sehingga tampak seperti paru-paru berwarna putih (*white lung*). *air bronchogram* menyebabkan pembengkakan pada dinding bronkial dan peradangan penebalan [9].



Gambar 2.2 GGO dengan konsolidasi dan *air-bronchogram* [8]

2.1.4 Paving Stone sign

Pada saat *CT scan* beresolusi tinggi, bayangan garis interlobular dan penebalan lobular tumpang tindih dengan ground glass opacity (GGO). Karena interval lobular dan penebalan interstitial interlobular hal ini terjadi, yang berakibat perubahan interstitial [9].



Gambar 2.3 GGO (atas) dan paving pattern (bawah) [10]

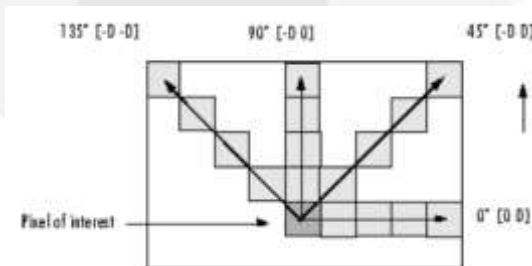
2.2 Image Processing

2.2.1 Citra Grayscale

Suatu citra *grayscale* merupakan suatu citra yang memiliki warna tingkat keabuan. Dalam 8 bit integer yang memberikan 256 kemungkinan yang mana dimulai dari level 0 sampai dengan 255 (0 untuk hitam dan 255 untuk putih dan nilai diantaranya adalah derajat keabuan) intensitas dari citra *grayscale* disimpan.

2.2.1 Gray-Level Co-occurrence Matrix (GLCM)

Gray-level co-occurrence matrix (GLCM) adalah matriks yang terdiri dari jumlah kolom dan baris yang sama dengan jumlah tingkat abu-abu dalam sebuah citra [11] digunakan sebagai metode statistik orde kedua untuk menganalisis tekstur pada sebuah citra [12]. Metode GLCM sebagai tabulasi dari data piksel citra yang digambarkan dengan seberapa sering kombinasi yang berbeda pada level nilai *grayscale* yang muncul pada suatu citra dengan cara menghitung probabilitas hubungan kedekatan antar 2 nilai piksel pada jarak (d) dan sudut (θ) tertentu [13].



Gambar 2.4 Orientasi sudut pada GLCM [14]

2.2.2 Features Extraction

Teknik ekstraksi fitur sangat penting didalam teknik pengolahan citra, karena supaya mendapatkan hasil deteksi bagian/bentuk (fitur) gambar yang diinginkan teknik ekstraksi fitur sangat menentukan hasil [15]. Digunakan empat nilai fitur tekstur yang diekstraksi dari matriks GLCM yaitu nilai fitur *contrast*, *entropy*, *homogeneity*, dan *energy*.

1. *Contrast* adalah besarnya sebaran nilai intensitas citra. Nilai kontras pada sebuah citra sangat dipengaruhi oleh level pencahayaan pada pengambilan citra [16].

$$Con = \sum_i \sum_j (i - j)^2 P_{(i,j)} \quad (2.1)$$

2. *Entropy* adalah nilai fitur yang menunjukkan level keteraturan struktur dan tekstur pada citra. Jika suatu struktur dan tekstur pada citra beraturan maka nilai *entropy* yang didapat besar, dan sebaliknya, jika struktur dan tekstur pada citra tidak beraturan maka nilai *entropy* akan kecil. [16].

$$E = \sum_i \sum_j P_{(i,j)} \log P_{(i,j)} \quad (2.2)$$

3. *Homogeneity* mempresentasikan ukuran kesamaan. *Homogeneity* akan bernilai tinggi jika semua piksel mempunyai nilai yang seragam [16].

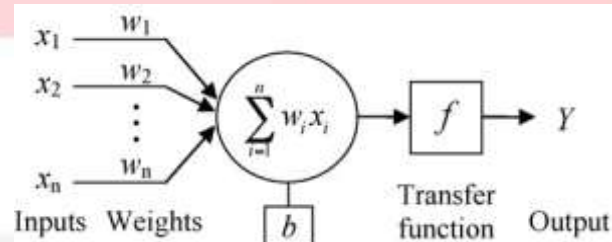
$$Hom = \sum_i \sum_j \frac{P_{(i,j)}}{1+|i-j|} \quad (2.3)$$

4. *Energy* mempresentasikan ukuran keseragaman pada citra. Semakin tinggi kemiripan citra maka akan semakin tinggi juga nilai energi [17]. *Energy* biasa juga disebut *Angular Second Moment* (ASM) jika nilai pikselnya sangat mirip maka nilai energi akan tinggi [16].

$$Eng = \sum_i \sum_j P_{(i,j)}^2 \quad (2.4)$$

2.3 Jaringan Saraf Tiruan

Jaringan saraf tiruan terdiri dari elemen proses sederhana (*neuron*) yang saling berhubungan satu sama lain dan juga terdiri dari berbagai lapisan. Sama seperti *neuron* biologis, *neuron* akan menerima masukan dari *neuron* yang lain [18]. Pada jaringan saraf tiruan beradaptasi terhadap masukan yang diberikan seperti melalui proses belajar didapatkan sebuah pengetahuan dan hubungan antar *neuron* yang biasa digunakan untuk menyimpan pengetahuan [19]. Masing-masing dari *neuron* tersebut akan melakukan transformasi informasi atau data yang diterima melalui sambungan antara keduanya menuju *neuron* lain. Hubungan yang akan disebut dengan bobot (*weight*).



Gambar 2.5 Model neuron [20]

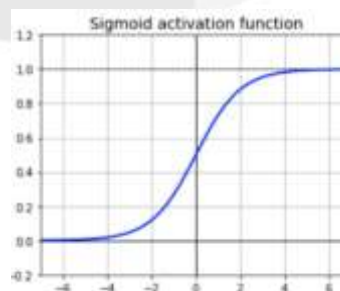
neuron secara teoritis yang dimodelkan, di mana $x = \{x_i, i = 1, 2, \dots, n\}$ adalah input ke *neuron* dan Y adalah data keluaran. Setiap masukan dikalikan dengan bobotnya $w = \{w_i, i = 1, 2, \dots, n\}$, b atau bias berkaitan dengan setiap *neuron* dan jumlahnya melewati *transfer function* [21]. Arsitektur *neural network* dapat dibagi berdasarkan jumlah lapisannya [22].

2.2.1 Transfer Function

Transfer function (Fungsi aktivasi) adalah fungsi yang pada jaringan saraf tiruan yang digunakan untuk mengaktifkan atau tidak mengaktifkan *neuron*. *Transfer function* jaringan saraf tiruan memiliki karakteristik yang harus ada antara lain harus kontinu, terdiferensialkan, dan tidak menurun secara monoton (*monotonically non-decreasing*) [23].

1. *Sigmoid transfer function* juga dikenal sebagai fungsi logistik atau fungsi *squashing* di beberapa literatur [24]. *Transfer function* ini adalah *non-linear transfer function* mengambil nilai *real* sebagai masukan dan menghasilkan keluaran nilai antara 0 dan 1 yang dapat diturunkan terbatas, yang didefinisikan untuk nilai masukan *real*, dengan turunan yang positif di mana-mana [25]. Fungsi *sigmoid* diterapkan di lapisan keluaran arsitektur jaringan saraf tiruan, dan digunakan untuk memprediksi keluaran berbasis probabilitas dan digunakan aplikasi klasifikasi [26].

$$f(x) = \left(\frac{1}{1+e^{-x}} \right) \quad (2.5)$$



Gambar 2.6 Representasi *sigmoid transfer function*

2. *Rectified Linear Unit (ReLU)*, fungsi ini paling banyak digunakan untuk aplikasi *deep learning* dengan karya penelitian mutakhir hingga saat ini [27]. ReLU adalah *transfer function learning* yang cepat [28], yang telah terbukti sebagai fungsi yang paling sukses dan banyak digunakan [29]. Representasi dari fungsi ini mirip fungsi linier maka dari itu fungsi ini mempertahankan sifat-sifat fungsi linier yang membuatnya mudah untuk dioptimalkan, dengan metode *gradient-descent* [28].

$$f(x) = \max(0, x) = \begin{cases} x_i, & \text{jika } x_i \geq 0 \\ 0, & \text{jika } x_i < 0 \end{cases} \quad (2.6)$$



Gambar 2.7 Representasi ReLU *transfer function*

3. *Leaky Rectified Linear Unit (Leaky ReLU / LReLU)*, fungsi ini melakukan untuk mempertahankan pembaruan bobot tetap ada selama seluruh proses propagasi [30]. Untuk menyelesaikan permasalahan pada ReLU dibutlah LReLU. LReLU menghitung gradien dengan nilai konstanta yang sangat kecil untuk gradien negatif dalam kisaran 0,01 [26].

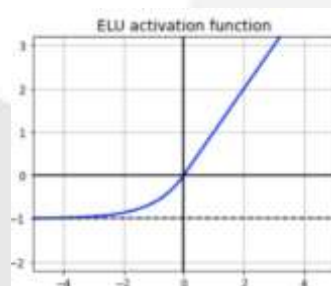
$$f(x) = ax + x = \begin{cases} x, & \text{jika } x > 0 \\ ax, & \text{jika } x \leq 0 \end{cases} \quad (2.7)$$



Gambar 2.8 Representasi *Leaky ReLU transfer function*

4. *Exponential Linear Units (ELU)*, fungsi ini memiliki nilai negatif yang memungkinkan untuk melakukan proses aktivasi unit rata-rata mendekati nol sehingga mengurangi kompleksitas komputasi sehingga meningkatkan kecepatan *learning* [31]. ELU representasi alternatif yang baik pada ReLU karena mengurangi pergeseran bias dengan mendorong proses aktivasi unit rata-rata menuju nol selama proses *training* [26].

$$f(x) = \begin{cases} x, & \text{jika } x > 0 \\ a(e^x - 1), & \text{jika } x \leq 0 \end{cases} \quad (2.8)$$



Gambar 2.9 Representasi *ELU transfer function*

2.2.2 Backpropagation

Backpropagation adalah algoritma jaringan saraf tiruan (JST) yang biasanya menggunakan *perceptron* yang mempunyai banyak *layer* untuk mengubah bobot-bobot yang terhubung dengan *neuron* yang terdapat di *hidden layer* [32]. Dalam *backpropagation* penggunaan *error* bertujuan untuk mengubah nilai setiap *weight* dalam arah mundur (*backward*). Sebelum didapatkan *error* ini, terdapat langkah awal yang harus dilakukan yaitu langkah perambatan maju (*forward*). Untuk menurunkan tingkat kesalahan yang terjadi maka dilakukan tahap memodifikasi nilai bobot [33].

2.2.2 Adam Optimization Algorithm

Tujuan pengoptimalan dari algoritma ini untuk menemukan parameter yang dapat meminimalkan *loss function* [34]. Adam berasal dari *Adaptive moment estimation*, dibandingkan *stochastic optimization method* lainn hasil empiris membuktikan bahwa *Adam Optimization Algorithm* bekerja dengan baik dalam praktiknya dan lebih baik [35].

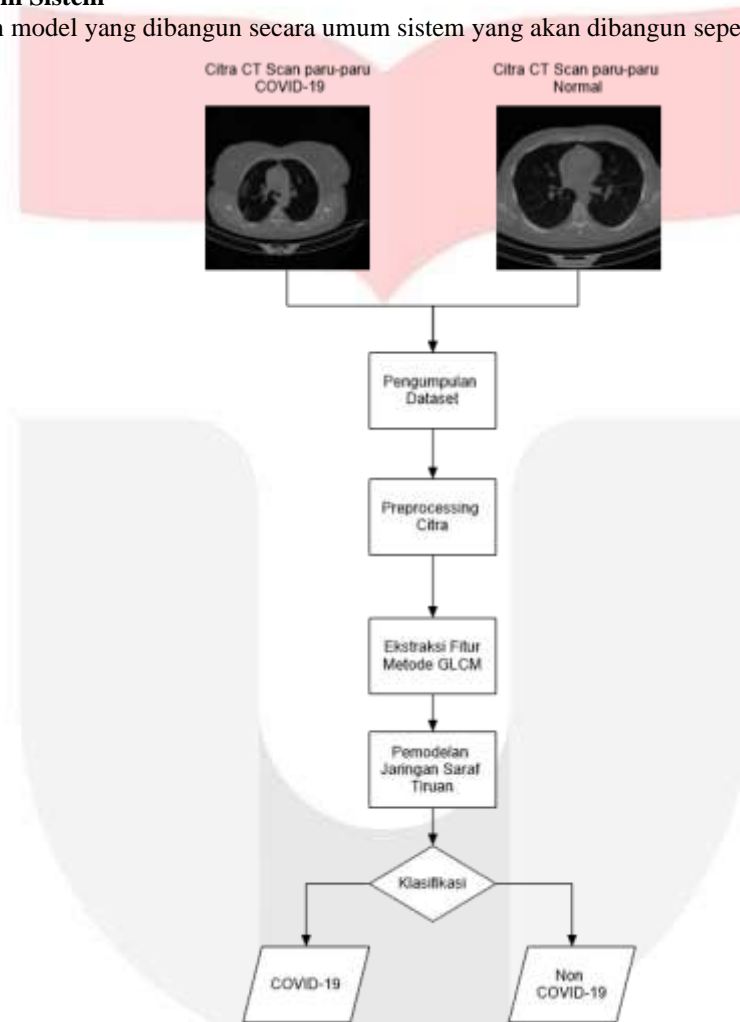
2.2.2 Dropout Layer

Dropout merupakan salah proses pada jaringan saraf tiruan untuk mengatasi/mencegah terjadinya masalah *overfitting* dan juga mempercepat proses *learning*. Dalam sistem kerjanya, *Dropout* menghilangkan sementara suatu *neuron* yang ada pada *hidden Layer* maupun *visibe Layer* yang berada didalam jaringan saraf tiruan. *Neuron* yang akan dihilangkan dipilih secara acak. Setiap *neuron* akan diberikan probabilitas p yang bernilai antara 0 dan 1 [36].

3. Perancangan

3.1 Gambaran Umum Sistem

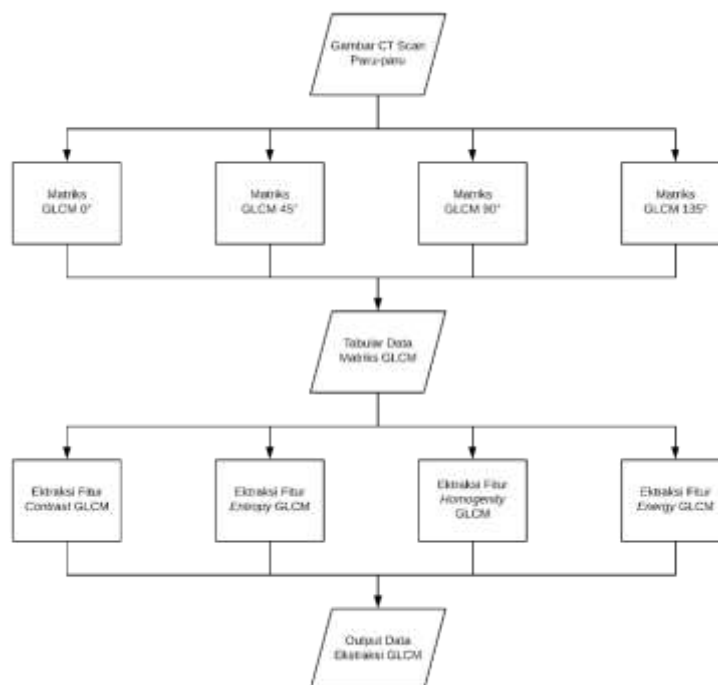
Pada perancangan model yang dibangun secara umum sistem yang akan dibangun seperti gambar berikut :



Gambar 3.1 Gambaran Umum Sistem

3.2 Perancangan Ekstraksi Fitur GLCM

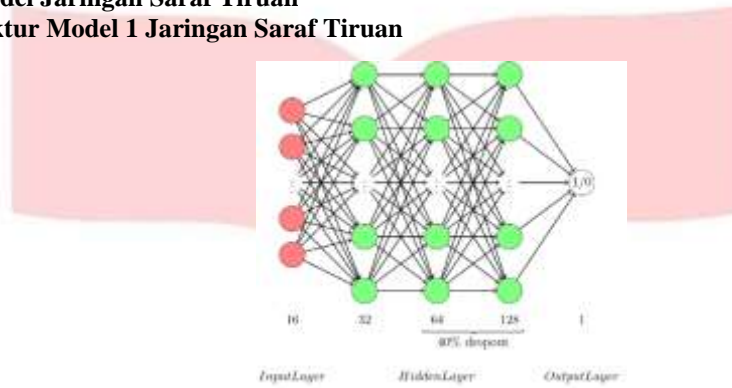
Gambaran secara umum untuk proses ekstraksi fitur gambar untuk pendeteksian citra *CT scan* paru-paru dengan metode GLCM digambarkan dengan *flowchart* sebagai berikut :



Gambar 3.2 Flowchart Ekstraksi Fitur GLCM

3.3 Perancangan Model Jaringan Saraf Tiruan

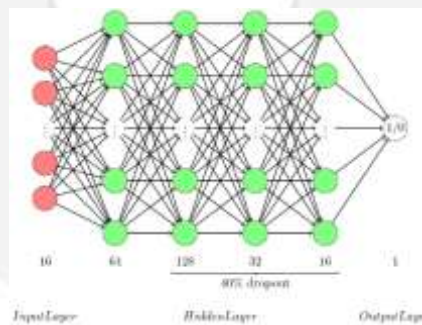
3.3.1 Arsitektur Model 1 Jaringan Saraf Tiruan



Gambar 3.3 Arsitektur Model 1 Jaringan Saraf Tiruan

Arsitektur model 1 jaringan saraf tiruan. Dapat dilihat arsitektur terdiri dari *input layer*, tiga *hidden layer*, dan *output layer*. Pada bagian *input layer* di arsitektur model pertama ini memiliki *neuron* yang berjumlah 16. Pada bagian *hidden layer* ada tiga *hidden layer*. Pada *hidden layer* pertama ada *neuron* yang berjumlah 32, *hidden layer* kedua ada *neuron* yang berjumlah 64, dan *hidden layer* ketiga ada *neuron* yang berjumlah 128. Pada *hidden layer* model ini juga dilakukan proses *droupout layer* sebesar 40%. Pada bagian *output layer* di arsitektur model pertama ini memiliki *neuron* yang berjumlah 1. Pada bagian *output layer transfer function* yaitu *sigmoid*.

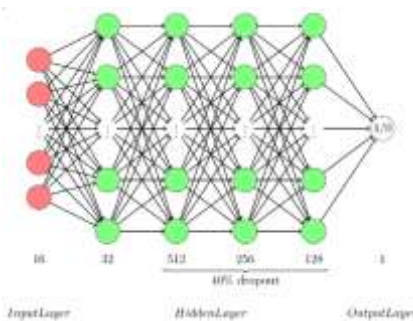
3.3.2 Arsitektur Model 2 Jaringan Saraf Tiruan



Gambar 3.4 Arsitektur Model 2 Jaringan Saraf Tiruan

Arsitektur model 2 jaringan saraf tiruan. Dapat dilihat arsitektur terdiri dari *input layer*, tiga *hidden layer*, dan *output layer*. Pada bagian *input layer* di arsitektur model pertama ini memiliki *neuron* yang berjumlah 16. Pada bagian *hidden layer* ada empat *hidden layer*. Pada *hidden layer* pertama ada *neuron* yang berjumlah 64, *hidden layer* kedua ada *neuron* yang berjumlah 128, *hidden layer* ketiga ada *neuron* yang berjumlah 32, dan dan *hidden layer* keempat ada *neuron* yang berjumlah 16. Pada *hidden layer* model ini juga dilakukan proses *droupout layer* sebesar 40%. Pada bagian *output layer* di arsitektur model pertama ini memiliki *neuron* yang berjumlah 1. Pada bagian *output layer transfer function* yaitu *sigmoid*.

3.3.3 Arsitektur Model 3 Jaringan Saraf Tiruan



Gambar 3.4 Arsitektur Model 3 Jaringan Saraf Tiruan

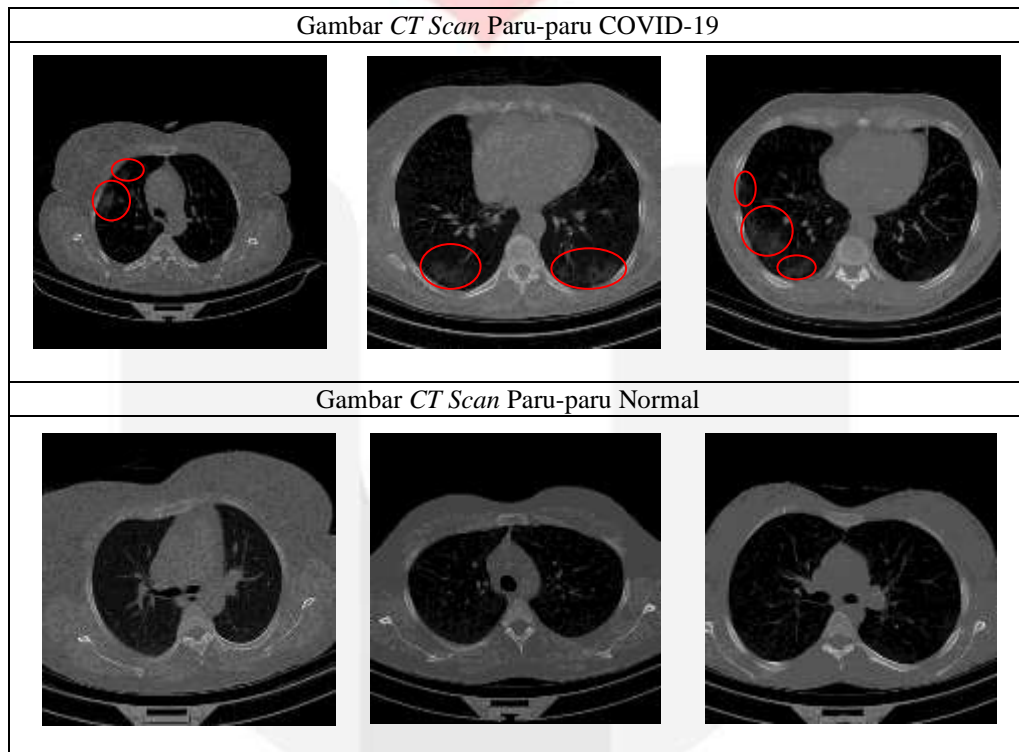
Arsitektur model 2 jaringan saraf tiruan. Dapat dilihat arsitektur terdiri dari *input layer*, tiga *hidden layer*, dan *output layer*. Pada bagian *input layer* di arsitektur model pertama ini memiliki *neuron* yang berjumlah 16, *neuron-neuron* ini adalah nilai-nilai ekstraksi fitur GLCM. Pada bagian *hidden layer* ada empat *hidden layer*. Pada *hidden layer* pertama ada *neuron* yang berjumlah 32, *hidden layer* kedua ada *neuron* yang berjumlah 512, *hidden layer* ketiga ada *neuron* yang berjumlah 256, dan dan *hidden layer*

kempat ada *neuron* yang berjumlah 128. Pada *hidden layer* model ini juga dilakukan proses *droupout layer* sebesar 40%. Pada bagian *output layer* di arsitektur model pertama ini memiliki *neuron* yang berjumlah 1. Pada bagian *output layer transfer function* (Fungsi aktivasi) yaitu *sigmoid*.

4. Implementasi dan Pengujian

4.1 Pengumpulan Dataset

Data yang digunakan adalah data citra dari *Kaggle.com* yaitu "*Large COVID-19 CT scan slice dataset*" [37]. Jumlah data yang digunakan yaitu 500 sampel untuk citra *CT scan* paru-paru yang terkena COVID-19 dan 500 sampel untuk citra *CT scan* paru-paru yang normal, total citra *CT scan* paru-paru yang digunakan berjumlah 1000 citra.



Gambar 4.1 CT Scan Paru-paru COVID-19 dan Normal

4.1 Preprocessing Dataset

Preprocessing dataset ini meliputi :

1. Proses konversi jenis citra ke citra *grayscale*
Citra jenis citra *grayscale* untuk mendapatkan nilai fitur-fitur pada metode GLCM. Oleh karena itu, data masukkan citra yang masih bukan dalam bentuk citra *grayscale* perlu diubah jenisnya menjadi citra *grayscale*.
2. *Resize* citra
Proses pengubahan ukuran citra menjadi ukuran tertentu untuk mendapatkan ukuran semua data citra menjadi seragam, maka dilakukan *resize* citra menjadi ukuran data citra yaitu 128×128.

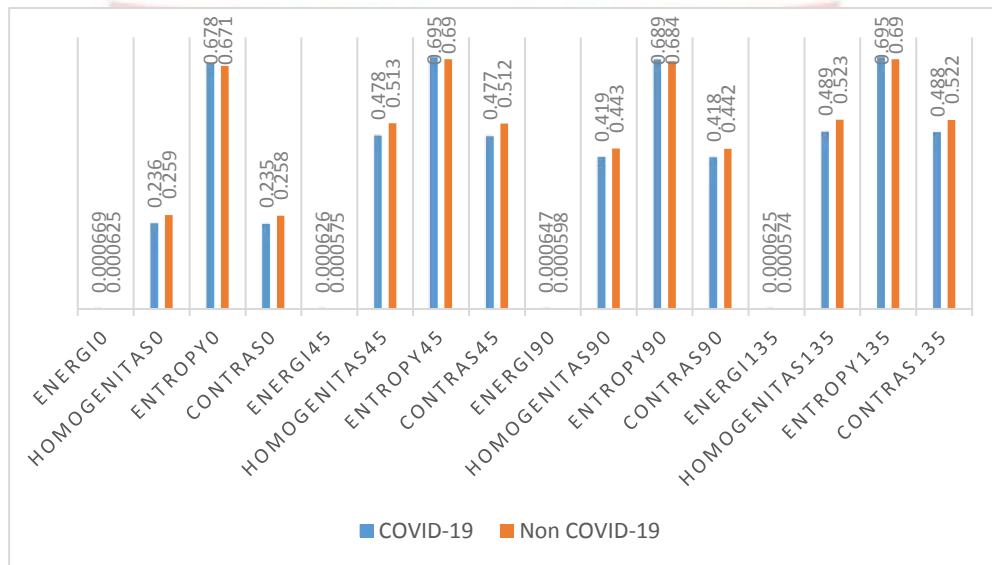
4.2 Ekstraksi Fitur Dataset

Langkah-langkah dalam mencari ekstraksi fitur metode GLCM sebagai berikut :

1. Pembuatan matriks GLCM
Dalam pembuatan matriks GLCM ini dipakai orientasi sudut *co-occurrence* matriks yaitu 0°, 45°, 90°, dan 135°.
2. Pengambilan nilai fitur GLCM
Nilai fitur yang akan diambil nilai fitur GLCM yaitu *contrast*, *entropy*, *homogeneity*, dan *energy*.
3. Standarisasi data
Nilai ekstraksi fitur GLCM dilakukan normalisasi data dengan teknik *decimal scaling* dengan cara membagi tiap kolom pada data dengan bilangan berpangkat basis 10.

$$new_data = \frac{data}{10^j} \quad (2.9)$$

4.3 Pengujian Parameter Ekstraksi Fitur GLCM



Gambar 4.2 Pengujian Parameter Ekstraksi Fitur GLCM

Dapat dilihat dari setiap nilai fitur GLCM memiliki selisih antar kelas citra COVID-19 dan Non COVID-19 sehingga dapat menunjukkan perbedaan yang berarti data ekstraksi yang didapat bisa dilakukan proses *learning* untuk klasifikasi.

4.3 Pengujian Performasi Partisi Data

Pengujian ini dilakukan untuk mengukur besar tingkat keberhasilan sistem yang dirancang dengan melakukan analisis terhadap partisi data dan untuk mengetahui performansi sistem yang telah dirancang. Berdasarkan pengujian menggunakan parameter partisi data, ada banyak hasil performansi yang dihasilkan oleh 5 skenario data *training* dan data *testing*.

Tabel 4.2 Jumlah Data Setelah dipartisi *training* dan *testing*

Pengujian	Training	Testing	Hasil Partisi	
			Training	Testing
1	50%	50%	500	1
2	60%	40%	600	2
3	70%	30%	700	3
4	80%	20%	800	4
5	90%	10%	900	5

Tabel 4.3 Parameter Model Jaringan Saraf Tiruan

	Trainable parameters	Batch size	Optimizer	Learnig Rate	Transfer function
Model 1	11.105	32	Adam	0.001	ReLU
					Leaky ReLU
					ELU
Model 2	14.081	32	Adam	0.001	ReLU
					Leaky ReLU
					ELU
Model 3	181.793	32	Adam	0.001	ReLU
					Leaky ReLU
					ELU

Parameter yang digunakan untuk pengujian partisi data yaitu *Trainable parameters* adalah jumlah elemen/neuron yang dapat dilatih pada arsitektur model jaringan saraf tiruan, *Batch size* adalah jumlah sampel data yang disebarkan ke model jaringan saraf tiruan, *Optimizer* model jaringan saraf tiruan digunakan algoritma adam, nilai *Learnig Rate* untuk menghitung nilai koreksi bobot pada waktu proses *training* untuk pengujian ini digunakan nilai *default* dari algoritma adam pada *framework* Keras yaitu 0.001, dan untuk *Transfer function* adalah fungsi aktivasi yang digunakan di bagian *hidden layer* pada arsitektur setiap model.

Selanjutnya dilakukan proses klasifikasian model jaringan saraf tiruan sesuai paramater yang diatur. Pada proses didefinisikan terlebih dahulu *true positive* (TP), *false positive* (FP), *false negative* (FN) dan *true negative* (TN) seperti yang ditunjukkan pada *confusion matrix*. TP adalah data positif yang diprediksi sebagai positif dan TN adalah data negatif yang di prediksi sebagai negatif. FP yaitu data negatif yang diprediksi positif, Sedangkan FN yaitu data positif

yang diprediksi sebagai negatif. Proses *confusion matrix* untuk mendapatkan hasil perhitungan untuk *accuracy*, *precision*, *recall*, dan *F1-score*.

Tabel 4.4 Confusion Matrix

		Kelas Sebenarnya	
		Positif	Negatif
Prediksi	Positif	True Positive (TP)	False Positive (FP)
	Negatif	False Negative (FN)	True Negative (TN)

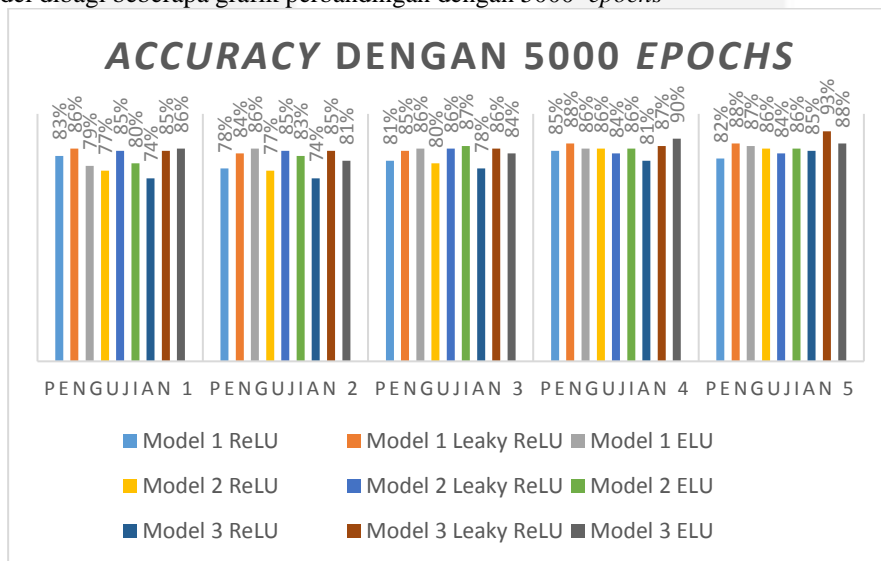
$$Accuracy = \frac{TP+TN}{TP+TN+FP+FN} \tag{4.1}$$

$$Precision = \frac{TP}{(TP+FP)} \tag{4.2}$$

$$Recall = \frac{TP}{(TP+FN)} \tag{4.3}$$

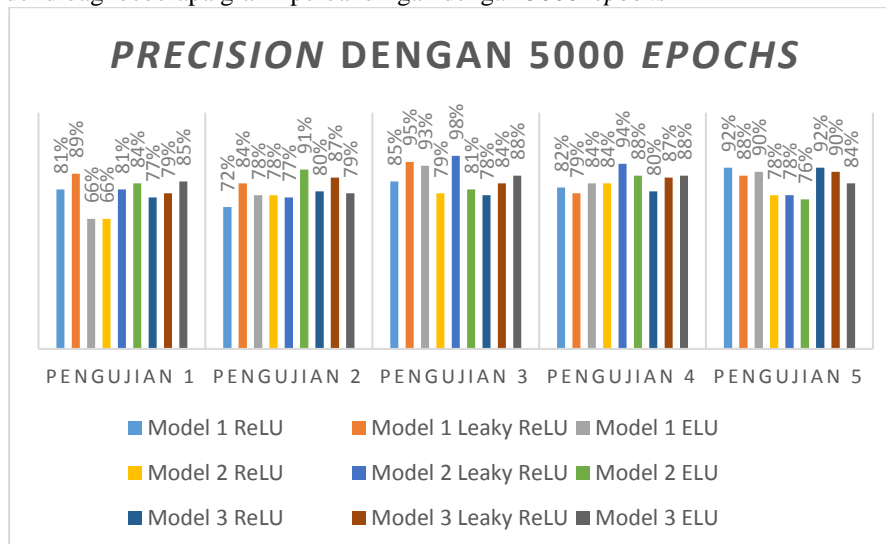
$$F1-score = \frac{2 \times Precision \times Recall}{Precision + Recall} \tag{4.4}$$

Berikut hasil perbandingan *accuracy* antar model jaringan saraf tiruan yang dibangun dengan proses pembangunan model dibagi beberapa grafik perbandingan dengan 5000 *epochs*



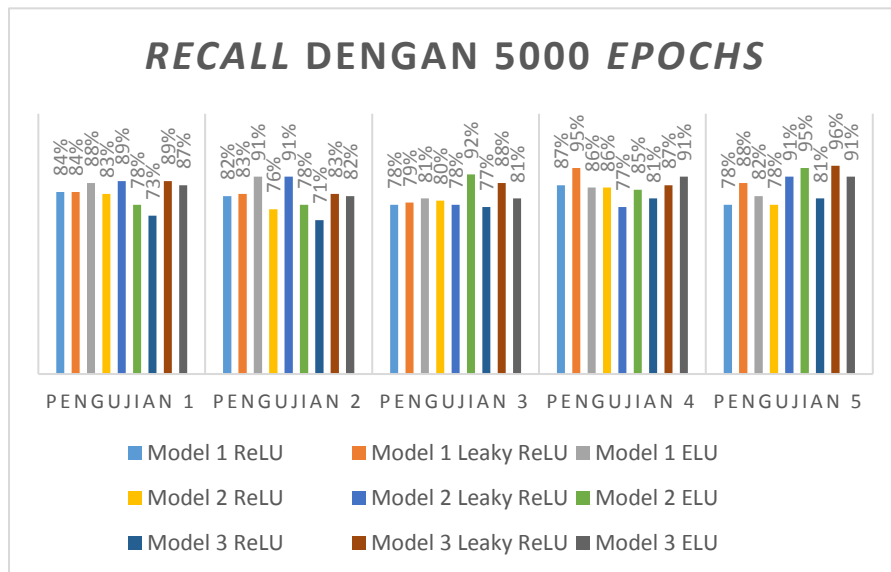
Gambar 4.3 Pengujian Data Dengan Accuracy Model dengan 5000 Epochs

Berikut hasil perbandingan *precision* antar model jaringan saraf tiruan yang dibangun dengan proses pembangunan model dibagi beberapa grafik perbandingan dengan 5000 *epochs*



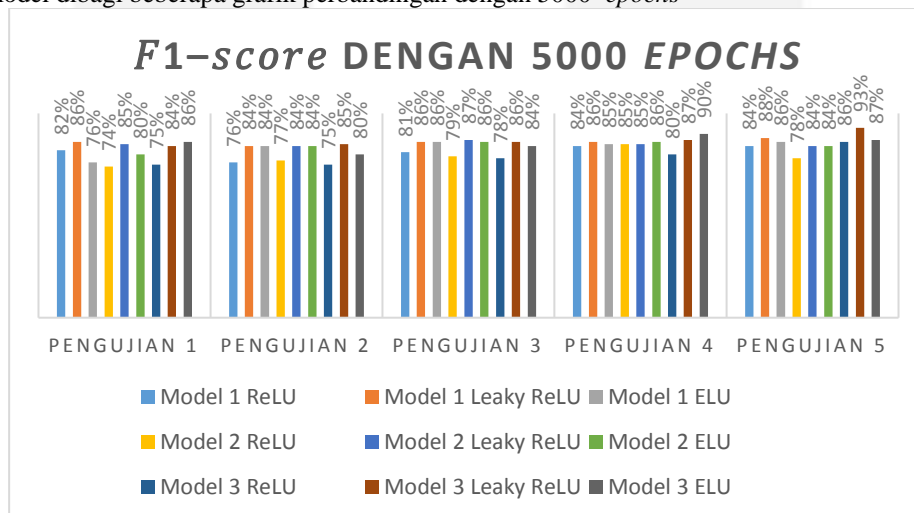
Gambar 4.4 Pengujian Data Dengan Precision Model dengan 5000 Epochs

Berikut hasil perbandingan *recall* antar model jaringan saraf tiruan yang dibangun dengan proses pembangunan model dibagi beberapa grafik perbandingan dengan 5000 *epochs*



Gambar 4.5 Pengujian Data Dengan *Recall* Model dengan 5000 Epochs

Berikut hasil perbandingan *F1-score* antar model jaringan saraf tiruan yang dibangun dengan proses pembangunan model dibagi beberapa grafik perbandingan dengan 5000 *epochs*



Gambar 4.6 Pengujian Data Dengan *F1-score* Model dengan 5000 Epochs

Berdasarkan grafik untuk pengujian partisi dapat dilihat pada pengujian 5 yang memiliki jumlah data partisi training dan testing dengan perbandingan 90 : 10. Pada pengujian 5 dengan *epochs* sampai 5000 memiliki *accuracy*, *precision*, *recall*, dan *F1-score* yang paling optimal untuk melakukan klasifikasi karena semakin banyak data *training* dan semakin sedikit data *testing* maka performansi model akan semakin tinggi, karena model lebih banyak belajar dengan data daripada untuk mengenali data *testing*. Dan didapatkan juga model yang performansi paling optimum yaitu model 3 dengan *hidden layer* menggunakan *transfer function* Leaky ReLU dengan nilai *accuracy* sebesar 93%, *precision* sebesar 90%, *recall* sebesar 96% dan *F1-score* sebesar 93%.

5. Kesimpulan dan Saran

5.1 Kesimpulan

Dari Pembahasan yang telah diuraikan sebelumnya, maka peneliti mengambil kesimpulan seperti berikut:

1. Dengan menggunakan metode *Gray-level co-occurrence matrix* (GLCM) dapat mendapatkan karakteristik ekstraksi fitur dari data citra *greyscale CT scan* paru-paru yang terinfeksi COVID-19 dan yang tidak.
2. Dengan membuat rancangan arsitektur model 1, 2, dan 3 dengan *hidden layer* menggunakan *transfer function* ReLU, Leaky ReLU, dan ELU dapat menghasilkan hasil klasifikasi citra digital *CT scan* paru-paru terinfeksi COVID-19 atau tidak COVID-19.
3. Model dengan rancangan arsitektur model 3 dengan *hidden layer* menggunakan *transfer function* Leaky ReLU memiliki performansi paling optimum dengan nilai *accuracy* sebesar 93%, *precision* sebesar 90%, *recall* sebesar 96% dan *F1-score* sebesar 93%, dengan melakukan partisi data *training* 90% dan data *testing* 10% dan *epochs* sampai 5000.

5.2 Saran

Tentunya hasil dari tugas akhir yang penulis buat masih jauh dari sempurna, dimana kekurangan dan kesalahan masih terdapat di dalamnya. Adapun hal yang perlu dikembangkan lagi adalah :

1. Mempersiapkan dataset yang baik dalam proses identifikasi citra CT scan paru-paru terinfeksi COVID-19 agar dapat mengoptimalkan proses identifikasi.
2. Menggunakan metode ekstraksi fitur yang berbeda untuk mendapatkan hasil perbandingan metode ekstraksi fitur yang lebih maksimal dalam mengklasifikasikan ekstraksi fitur citra, sehingga dapat dibandingkan metode mana yang lebih baik.
3. Menggunakan metode klasifikasi yang berbeda untuk mendapatkan hasil perbandingan performa prediksi klasifikasi yang lebih maksimal dalam prediksi citra, sehingga dapat dibandingkan performa prediksi metode klasifikasi mana yang lebih baik.

Referensi

- [1] Q. Li *et al.*, "Early Transmission Dynamics in Wuhan, China, of Novel Coronavirus-Infected Pneumonia," *N. Engl. J. Med.*, 2020, doi: 10.1056/nejmoa2001316.
- [2] Z. Y. Zu *et al.*, "Coronavirus Disease 2019 (COVID-19): A Perspective from China," *Radiology*. 2020, doi: 10.1148/radiol.2020200490.
- [3] P. Hermanek and S. Carmignato, "Porosity measurements by X-ray computed tomography: Accuracy evaluation using a calibrated object," *Precis. Eng.*, 2017, doi: 10.1016/j.precisioneng.2017.03.007.
- [4] WHO, "Media Statement Knowing the risks for COVID-19," *Media Statement: Knowing the risks for COVID-1*, 2020. <https://www.who.int/indonesia/news/detail/08-03-2020-knowing-the-risk-for-covid-19> (accessed Nov. 25, 2020).
- [5] C. Huang *et al.*, "Clinical features of patients infected with 2019 novel coronavirus in Wuhan, China," *Lancet*, 2020, doi: 10.1016/S0140-6736(20)30183-5.
- [6] Z. Wang, W. Qiang, and H. Ke, "A Handbook of 2019-nCoV Pneumonia Control and Prevention," *Hubei Sci. Technol. Press*, 2020.
- [7] B. Yanti and U. Hayatun, "Peran pemeriksaan radiologis pada diagnosis Coronavirus disease 2019," *J. Kedokt. Syiah Kuala Vol.*, vol. 20, no. 1, pp. 53–57, 2020.
- [8] L. Hu and C. Wang, "Radiological role in the detection, diagnosis and monitoring for the coronavirus disease 2019 (COVID-19)," *Eur. Rev. Med. Pharmacol. Sci.*, 2020, doi: 10.26355/eurrev_202004_21035.
- [9] H. Chen, L. Ai, H. Lu, and H. Li, "Clinical and imaging features of COVID-19," *Radiol. Infect. Dis.*, 2020, doi: 10.1016/j.jrid.2020.04.003.
- [10] M. Durrani, I. U. Haq, U. Kalsoom, and A. Yousaf, "Chest x-rays findings in covid 19 patients at a university teaching hospital-a descriptive study," *Pakistan J. Med. Sci.*, 2020, doi: 10.12669/pjms.36.COVID19-S4.2778.
- [11] P. Mohanaiah, P. Sathyanarayana, and L. Gurukumar, "Image Texture Feature Extraction Using GLCM Approach," *Int. J. Sci. Res. Publ.*, vol. 3, no. 5, pp. 1–5, 2013.
- [12] T. Sutojo, D. R. I. M. Setiadi, P. S. Tirajani, C. A. Sari, and E. H. Rachmawanto, "CBIR for classification of cow types using GLCM and color features extraction," *Proc. - 2017 2nd Int. Conf. Inf. Technol. Inf. Syst. Electr. Eng. ICITISEE 2017*, vol. 2018-Janua, pp. 182–187, 2018, doi: 10.1109/ICITISEE.2017.8285491.
- [13] R. A. Pramunendar, C. Supriyanto, D. H. Novianto, I. N. Yuwono, G. F. Shidik, and P. N. Andono, "A classification method of coconut wood quality based on Gray Level Co-occurrence matrices," *Proc. 2013 Int. Conf. Robot. Biomimetics, Intell. Comput. Syst. ROBIONETICS 2013*, no. November, pp. 254–257, 2013, doi: 10.1109/ROBIONETICS.2013.6743614.
- [14] S. Debbarma, "FPGA Implementation of GLCM," *Science (80-.)*, vol. 00, no. 1, pp. 883–888, 2005.
- [15] A. Chaudhary and S. S. Singh, "Lung cancer detection on CT images by using image processing," 2012, doi: 10.1109/ICCS.2012.43.
- [16] C. Irawan, E. N. Ardyastiti, D. R. I. M. Setiadi, E. H. Rachmawanto, and C. A. Sari, "A survey: Effect of the number of GLCM features on classification accuracy of lasem batik images using K-nearest neighbor," *2018 Int. Semin. Res. Inf. Technol. Intell. Syst. ISRITI 2018*, pp. 33–38, 2018, doi: 10.1109/ISRITI.2018.8864443.
- [17] O. R. Indriani, E. J. Kusuma, C. A. Sari, E. H. Rachmawanto, and D. R. I. M. Setiadi, "Tomatoes classification using K-NN based on GLCM and HSV color space," *Proc. - 2017 Int. Conf. Innov. Creat. Inf. Technol. Comput. Intell. IoT, ICITech 2017*, vol. 2018-Janua, pp. 1–6, 2018, doi: 10.1109/INNOCIT.2017.8319133.
- [18] G. P. Zhang, "Neural networks for classification: A survey," *IEEE Trans. Syst. Man Cybern. Part C Appl. Rev.*, vol. 30, no. 4, pp. 451–462, 2000, doi: 10.1109/5326.897072.
- [19] S. Kusmaryanto, "Jaringan Saraf Tiruan Backpropagation untuk Pengenalan Wajah Metode Ekstraksi Fitur Berbasis Histogram," *J. EECCIS Vol. 8, No. 2, Desember 2014*, 2014.
- [20] K. Kumar and J. D. Haynes, "Forecasting credit ratings using an ANN and statistical techniques," vol. 11, pp. 91–108, 2003.
- [21] J. Jiang, P. Trundle, and J. Ren, "Medical image analysis with artificial neural networks," *Comput. Med. Imaging Graph.*, 2010, doi: 10.1016/j.compmedimag.2010.07.003.
- [22] D. Puspitaningrum, *Pengantar Jaringan Syaraf Tiruan*. 2004.

- [23] Julpan, E. B. Nababan, and M. Zarlis, "Analisis Fungsi Aktivasi Sigmoid Biner Dan Sigmoid Bipolar Dalam Algoritma Backpropagation Pada Prediksi Kemampuan Siswa," *J. Teknovasi*, vol. 02, no. 1, pp. 103–116, 2015.
- [24] J. Turian, J. Bergstra, and Y. Bengio, "Quadratic features and deep architectures for chunking," no. June, p. 245, 2009, doi: 10.3115/1620853.1620921.
- [25] J. Han and C. Moraga, "The influence of the sigmoid function parameters on the speed of backpropagation learning," *Lect. Notes Comput. Sci. (including Subser. Lect. Notes Artif. Intell. Lect. Notes Bioinformatics)*, vol. 930, pp. 195–201, 1995, doi: 10.1007/3-540-59497-3_175.
- [26] C. Nwankpa, W. Ijomah, A. Gachagan, and S. Marshall, "Activation Functions: Comparison of trends in Practice and Research for Deep Learning," pp. 1–20, 2018, [Online]. Available: <http://arxiv.org/abs/1811.03378>.
- [27] V. Nair and G. E. Hinton, "Rectified Linear Units Improve Restricted Boltzmann Machines," *ICML*, 2010.
- [28] Y. Lecun, Y. Bengio, and G. Hinton, "Deep learning," *Nature*, vol. 521, no. 7553, pp. 436–444, 2015, doi: 10.1038/nature14539.
- [29] B. Zoph and Q. V Le, "Searching for activation functions," *6th Int. Conf. Learn. Represent. ICLR 2018 - Work. Track Proc.*, pp. 1–13, 2018.
- [30] A. L. Maas, A. Y. Hannun, and A. Y. Ng, "Rectifier nonlinearities improve neural network acoustic models," *ICML Work. Deep Learn. Audio, Speech Lang. Process.*, vol. 28, 2013.
- [31] D. A. Clevert, T. Unterthiner, and S. Hochreiter, "Fast and accurate deep network learning by exponential linear units (ELUs)," *4th Int. Conf. Learn. Represent. ICLR 2016 - Conf. Track Proc.*, pp. 1–14, 2016.
- [32] M. A. Bustomi, H. Bisri, and E. Purwanti, "Desain Perangkat Lunak Berbasis Jaringan Syaraf Tiruan Backpropagation untuk Klasifikasi Citra Rontgen Paru-paru," *J. Fis. dan Apl.*, 2015, doi: 10.12962/j24604682.v10i1.819.
- [33] A. Jumarwanto, "APLIKASI JARINGAN SARAF TIRUAN BACKPROPAGATION UNTUK MEMPREDIKSI PENYAKIT THT DI RUMAH SAKIT MARDI RAHAYU KUDUS," *J. Tek. Elektro*, 2009, doi: 10.15294/jte.v1i1.1601.
- [34] S. Ruder, "An overview of gradient descent optimization algorithms," pp. 1–14, 2016, [Online]. Available: <http://arxiv.org/abs/1609.04747>.
- [35] D. P. Kingma and J. L. Ba, "Adam: A method for stochastic optimization," *3rd Int. Conf. Learn. Represent. ICLR 2015 - Conf. Track Proc.*, pp. 1–15, 2015.
- [36] N. Srivastava, G. Hinton, A. Krizhevsky, I. Sutskever, and R. Salakhutdinov, "Dropout : A Simple Way to Prevent Neural Network from Overfitting," *J. Mach. Learn. Res.*, 2014.
- [37] M. Maftouni, A. Chung, C. Law, B. Shen, and Z. J. Kong, "A Robust Ensemble-Deep Learning Model for COVID-19 Diagnosis based on an Integrated CT Scan Images Database," no. June, 2021.