

Pengaruh Reduksi Fitur Pada Klasifikasi Kanker Paru Menggunakan CNN Dengan Arsitektur *GoogLeNet*

Siti Nur Fadilah¹, Dian Candra Rini Novitasari², Lutfi Hakim³

^{1,2,3}Program Studi Matematika, Fakultas Sains dan Matematika, UIN Sunan Ampel Surabaya
Jl. Ahmad Yani No.117 Surabaya, Jawa Timur, Indonesia

Korespondensi; Siti Nur Fadilah, Email: sitinurfadilah576@gmail.com

Abstrak

Kanker paru merupakan jenis kanker dengan penyebab kematian terbanyak. Penelitian ini bertujuan untuk mengklasifikasikan jenis kanker paru apakah termasuk kedalam kelas *lung adenocarcinoma*, *benign lung tissue*, *lung squamous cell carcinoma* berdasarkan citra histopatologi menggunakan metode CNN arsitektur *GoogLeNet* serta reduksi fitur PCA. Evaluasi model yang digunakan pada penelitian ini menggunakan *confusion matrix*. Data yang digunakan dalam penelitian ini sejumlah 15000 data yang terbagi menjadi 3 kelas dengan masing-masing kelas berjumlah 5000 data. Pada penelitian ini parameter uji coba yang digunakan yaitu probabilitas *dropout* dan jumlah *batchsize*. Lalu, dilakukan uji coba dengan metode CNN *GoogLeNet* serta metode CNN *GoogLeNet* dengan reduksi fitur yang digunakan yaitu PCA. Berdasarkan uji coba yang telah dilakukan diperoleh hasil terbasik yaitu dengan menggunakan metode CNN *GoogLeNet* dengan reduksi fitur PCA. Hasil terbaik yang diperoleh yaitu pada pembagian data 90:10 menggunakan nilai probabilitas *dropout* 0.9 serta *batchsize* berjumlah 8 menghasilkan nilai akurasi, sensitivitas, spesifitas berturut-turut yaitu 99.95%, 99.97%, dan 99.86% serta membutuhkan waktu *training* selama 93 menit 27 detik.

Kata Kunci: Kanker Paru, CNN, *GoogLeNet*, Reduksi Fitur, PCA

Abstract

Lung cancer is the type of cancer with the most causes of death. The purpose of this research is to classify the types of lung cancer using the CNN *GoogLeNet* architecture method and PCA feature reduction. The evaluation of the model used in this study uses a confusion matrix. The data used in this study amounted to 15,000 data. In this study, the trial parameters used were the probability of dropout and the number of batch sizes. Then, trials were carried out using the CNN *GoogLeNet* method and the CNN *GoogLeNet* method with feature reduction used, namely PCA. Based on the trials that have been carried out, the best results were obtained, namely by using the *GoogLeNet* CNN method with PCA feature reduction. The best results obtained were when dividing the data 90:10 using a dropout probability value of 0.9 and a batch size of 8, producing accuracy, sensitivity and specificity values respectively, namely 99.95%, 99.97% and 99.86% and required a training time of 93 minutes 27 seconds.

Keywords: Lung Cancer, CNN, *GoogLeNet*, Feature Reduction, PCA

Pendahuluan

Kanker paru merupakan kanker yang selnya tumbuh dan berkembang pada jaringan epitel bronkus (*bronchogenic carcinoma*) serta merupakan penyebab kematian terbanyak [1]. WHO menyatakan pada tahun 2020 sebanyak 10 juta kasus kematian dimana penyebab terbanyak yaitu kanker paru [2]. Selain itu, WHO juga menyatakan bahwa pada tahun 2030 jumlah kasus kanker paru akan mengalami peningkatan khususnya di negara-negara berkembang [3]. Peningkatan jumlah kasus karena adanya

keterlambatan dalam memperoleh penanganan yang tepat. Penyebabnya karena adanya kesalahan hasil diagnosis.

Salah satu cara dalam mengantisipasi adanya kesalahan hasil diagnosis yaitu dengan bantuan Computer Aided Diagnosis (CAD). CAD merupakan teknik dalam melakukan diagnosis maupun deteksi terhadap suatu penyakit dengan bantuan komputer yang nantinya akan meningkatkan nilai akurasi hasil diagnosis [4]. Berbagai penelitian telah menggunakan CAD seperti penelitian oleh [5] yang mendeteksi nodul paru-paru dan memperoleh nilai akurasi sebesar 92,7%. Kemudian, penelitian oleh [6] terkait deteksi kanker payudara dengan nilai akurasi sebesar 90,0%. Kemudian, penelitian oleh [7] untuk mendeteksi kanker paru menggunakan CNN memperoleh akurasi sebesar 96,1%.

CNN (*Convolutional Neural Network*) yaitu salah satu sistem CAD dan juga merupakan metode *deep learning* yang digunakan dalam mengolah data berupa data dua dimensi [8]. Metode CNN ini juga lebih efisien dikarenakan tidak memerlukan bantuan metode lain lagi dalam melakukan proses ekstraksi fitur. Metode CNN ini memiliki berbagai macam arsitektur, salah satunya yaitu *GoogLeNet*. *GoogLeNet* merupakan arsitektur CNN yang memiliki inception modules yang terdiri dari convolution kecil yang digunakan untuk mereduksi [9]. CNN telah banyak digunakan dalam berbagai penelitian seperti penelitian yang dilakukan oleh [10] dengan menggunakan *GoogLeNet* memperoleh akurasi sebesar 90.6%. Kemudian, penelitian yang dilakukan oleh [11] terkait deteksi kanker paru menggunakan *GoogLeNet* memperoleh akurasi sebesar 94.3%.

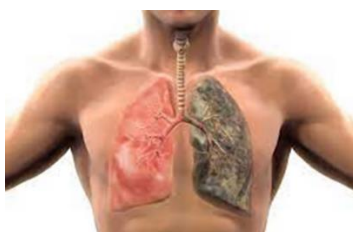
Metode CNN jenis arsitektur *GoogLeNet* ini membutuhkan waktu komputasi yang cukup lama. Salah satu cara yang dapat dilakukan yaitu dengan melakukan reduksi fitur. Hal ini tidak akan menghilangkan nilai keaslian dari data tersebut. Selain dapat mengurangi waktu komputasi, reduksi fitur juga dapat meningkatkan nilai akurasi [12]. Salah satu metode reduksi fitur yaitu Principal Component Analysis (PCA). PCA juga telah banyak digunakan dalam berbagai penelitian seperti penelitian yang dilakukan oleh [13] terkait klasifikasi penyakit jantung dengan menggunakan reduksi fitur PCA memperoleh akurasi sebesar 98.7%. Kemudian, pada penelitian yang dilakukan oleh [14] terkait klasifikasi kanker payudara dengan menggunakan reduksi fitur PCA memperoleh akurasi sebesar 97.0%.

Berdasarkan pemaparan dari latar belakang di atas dengan menggunakan data citra kanker paru, maka penelitian ini bertujuan untuk mengetahui bagaimana pengaruh reduksi fitur pada klasifikasi kanker paru menggunakan CNN dengan arsitektur *GoogLeNet*.

Landasan Teori

1. Kanker Paru

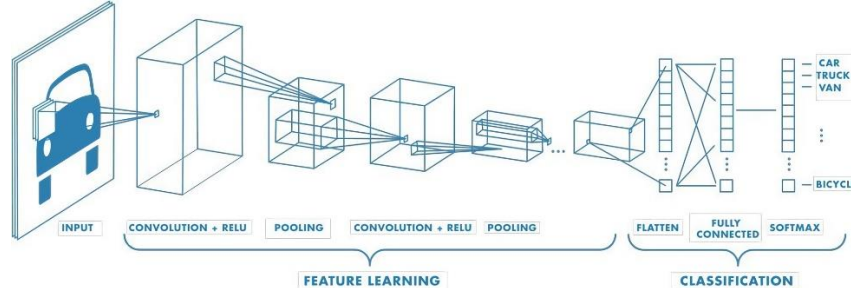
Kanker paru merupakan *neoplasma* ganas yang berasal dari *epitel bronkus* yakni sel epitelium respirasi. Sel kanker tersebut akan mengalami invasi serta bermetastasis pada organ lain, seperti otak. Umumnya kanker paru disebabkan karena kebiasaan merokok dimana pada rokok terdapat suatu zat yang dapat membahayakan yakni zat karsinogen [15]. Selain itu, penyebab lainnya seperti adanya pengaruh genetik dan status imunologi, serta predisposisi karena adanya penyakit lain. Kanker paru dapat dideteksi dengan melakukan *X-Ray*, *CT Scan*, *PET-CT Scan*, dan juga pemeriksaan bronkoskopi [16].



Gambar 1. Kanker Paru

2. CNN

Convolutional Neural Network (CNN) merupakan salah satu jenis *deep learning* yang sering digunakan dalam mengolah serta mengklasifikasikan suatu citra. CNN memiliki kedalaman jaringan yang tinggi serta cara kerjanya mencontoh sistem pengenalan citra pada *visual cortex* manusia [17]. CNN memiliki tiga komponen utama yaitu *input*, *feature learning*, serta *classification*. Pada CNN juga terdapat *neuron* yang memiliki bobot serta bias [18]. Ilustrasi CNN ditunjukkan oleh Gambar 2.



Gambar 2. Ilustrasi CNN

2.1. Convolution Layer

Convolution Layer merupakan komponen utama pada CNN yang memiliki tugas menerima *input* citra untuk kemudian melakukan ekstraksi fitur pada citra tersebut. Pada *convolution layer* ini terjadi proses konvolusi, dimana proses ini akan menghasilkan citra baru. Citra baru tersebut ini menyatakan fitur dari citra *input* [19]. Pada proses konvolusi rumus yang digunakan ditunjukkan oleh Persamaan 1.

$$Q_{(i,j)} = \left(\sum_{i=1}^N I_{(i,j)} \odot F_{(i,j)} \right) + B_j \quad (1)$$

Keterangan:

$Q_{(i,j)}$: hasil *convolution* baris ke- i dan kolom ke- j

N : banyaknya *channel*

$I_{(i,j)}$: matriks *input* baris ke- i dan kolom ke- j

$F_{(i,j)}$: matriks filter baris ke- i dan kolom ke- j

B_j : bias

2.2. Activation Function

Fungsi aktivasi merupakan suatu fungsi yang digunakan untuk mentransformasikan dengan menggunakan perhitungan fungsi non-linier. Salah satu fungsi aktivasi yang digunakan pada CNN yaitu fungsi *Rectified Linear Unit* (ReLU). Fungsi ReLU digunakan untuk menghilangkan nilai negatif pada citra serta dapat membantu mempercepat waktu komputasi [20]. Persamaan fungsi aktivasi ReLU ditunjukkan oleh Persamaan 2.

$$R(z) = \max(0, z) \quad (2)$$

2.3. Pooling Layer

Pooling layer merupakan suatu *layer* yang digunakan untuk memperkecil ukuran dimensi dari *feature map* yang nantinya akan membantu untuk mempercepat komputasi. Terdapat dua macam *pooling* yang sering digunakan yaitu *max pooling* dan *average pooling*. Jika *max pooling* mengambil nilai maksimal dari setiap piksel citra, maka *average pooling* mengambil nilai rata-rata [21]. Ilustrasi *pooling layer* ditunjukkan oleh Gambar 3.

				Max Pooling	
2	3	5	4	6	8
5	6	7	8	3	4
3	2	1	0	Average Pooling	
1	2	3	4	4	8
				8	8

Gambar 3. Ilustrasi *Pooling Layer*

2.4. Cross Channel Normalization

Cross Channel Normalization merupakan layer yang digunakan untuk mengatasi adanya nilai yang tidak terbatas yang diperoleh dari fungsi aktivasi ReLU. Selain itu, *cross channel normalization* digunakan menormalisasikan nilai tidak terbatas tersebut berdasarkan nilai elemen dari fitur terdekat baik berasal dari *feature map* yang sama ataupun tidak [22]. Rumus yang digunakan pada operasi konvolusi ditunjukkan pada Persamaan 3.

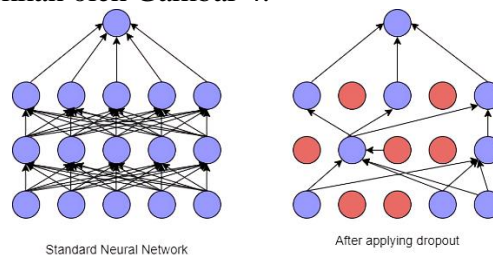
$$x_{i,j}^p = \frac{a_{i,j}^p}{\left(k + \frac{\alpha \sum_{q=\max(1, p-\frac{n}{2})}^{\min(N-1, p+\frac{n}{2})} (a_{i,j}^q)^2}{n} \right)^\beta} \quad (3)$$

Keterangan:

- p : filter ke- p
- $x_{i,j}$: piksel sesudah dinormalisasi
- $a_{i,j}$: piksel sebelum dinormalisasi
- k, α, β, n : *hyperparameter*

2.5. Dropout

Dropout merupakan suatu teknik dalam algoritma jaringan saraf yang digunakan untuk mengantisipasi adanya *overfitting* dengan menggabungkan beberapa jenis arsitektur. Jika semakin tinggi nilai bobot maka proses *testing* akan membutuhkan waktu yang semakin lama sehingga nantinya akan mempengaruhi kinerja CNN dalam melakukan proses *testing* [23]. Ilustrasi *dropout* ditunjukkan oleh Gambar 4.



Gambar 4. Ilustrasi *Dropout*

2.6. Fully Connected Layer

Fully connected layer merupakan layer terakhir setelah *convolutional layer* dan *pooling layer* yang digunakan sebagai layer penyatuan akhir yang melakukan klasifikasi ke dalam kelas pada data *training*. Pada layer ini nilai-nilai piksel yang semula berupa matriks diubah kedalam bentuk 1 dimensi [24]. Rumus yang digunakan pada operasi konvolusi ditunjukkan pada Persamaan 4.

$$y_j = B_j + \sum_i w_{i,j} x_i \quad (4)$$

Keterangan:

y_i : output

B_j : bias

$w_{i,j}$: bobot

x_i : input

2.7. Softmax Layer

Softmax merupakan salah satu fungsi aktivasi yang digunakan untuk mengklasifikasikan suatu data dengan melihat nilai probabilitas tertinggi. Nilai *softmax* atau nilai probabilitas tiap data dengan interval 0 hingga 1 [25]. Rumus yang digunakan pada operasi konvolusi ditunjukkan pada Persamaan 5.

$$p_j(x) = \frac{e^{x_j}}{\sum_{i=1}^k e^{x_i}} \quad (5)$$

3. GoogLeNet

GoogLeNet merupakan salah satu jenis arsitektur yang ada pada CNN dan pertama kali dikenalkan oleh Google pada tahun 2014. Pada tahun 2014 pula arsitektur ini memperoleh penghargaan sebagai arsitektur terbaik pada ILSVRC14. CNN model ini memiliki inception modules yang terdiri atas sejumlah *convolution* kecil. Algoritma CNN model *GoogLeNet* memiliki 144 layer dimana terdiri atas *convolutional layer*, *ReLU layer*, *Pooling layer*, serta klasifikasi [26].

4. Reduksi Fitur

Reduksi fitur merupakan salah satu teknik yang digunakan untuk mengurangi dimensi dari suatu data yang akan diklasifikasikan. Salah satu metode yang dapat digunakan untuk melakukan reduksi fitur. PCA merupakan salah satu metode reduksi fitur yang digunakan dalam pemilihan dan ekstraksi data, dimana data yang dipilih adalah data-data yang sangat berpengaruh namun tetap tidak menghilangkan keaslian data tersebut [27]. Pada metode ini dilakukan perhitungan nilai mean dan variansi dari citra, kemudian dilanjutkan dengan perhitungan matriks kovarian yang ditunjukkan pada Persamaan 6.

$$A = Y^T \times Y \quad (6)$$

Keterangan:

A : matriks kovarian

Y : mean

5. Confusion matrix

Confusion matrix merupakan salah satu metode yang digunakan untuk menyatakan hasil klasifikasi data yang teruji benar dan salah. Pada penelitian ini *confusion matrix* yang digunakan adalah *confusion matrix multiclass* dimana tabel *confusion matrix* ditunjukkan oleh Gambar 5.

Kelas Prediksi	Kelas Aktual		
	Adenocarcinoma	Benign	Squamous Cell
Adenocarcinoma	TP	FN	FN
Benign	FP	TN	TN
Squamous Cell	FP	TN	TN

a. Kelas Adenocarcinoma

Kelas Prediksi	Kelas Aktual		
	Adenocarcinoma	Benign	Squamous Cell
Adenocarcinoma	TN	FP	TN
Benign	FN	TP	FN
Squamous Cell	TN	FP	TN

b. Kelas Benign

Kelas Prediksi	Kelas Aktual		
	Adenocarcinoma	Benign	Squamous Cell
Adenocarcinoma	TN	TN	FP
Benign	TN	TN	FP
Squamous Cell	FN	FN	TP

c. Kelas Squamous Cell

Gambar 5. *Confusion matrix Multiclass*

Pada gambar diatas terdapat 3 tabel *confusion matrix* dimana pada tabel tersebut terdapat 4 istilah yaitu *True Positive* (TP) dimana sejumlah data yang terklasifikasi benar pada kedua kelas, *True Negative* (TN) dimana sejumlah data yang terklasifikasi salah pada kedua kelas, *False Positive* (FP) dimana sejumlah data yang terklasifikasi salah pada kelas aktual dan benar pada kelas target, dan *False Negative* (FN) dimana sejumlah data yang terklasifikasi benar pada kelas aktual dan salah pada kelas target [28]. Berdasarkan istilah-istilah tersebut maka dapat digunakan untuk menghitung akurasi, sensitivitas, dan spesifisitas yang ditunjukkan oleh Persamaan 7.

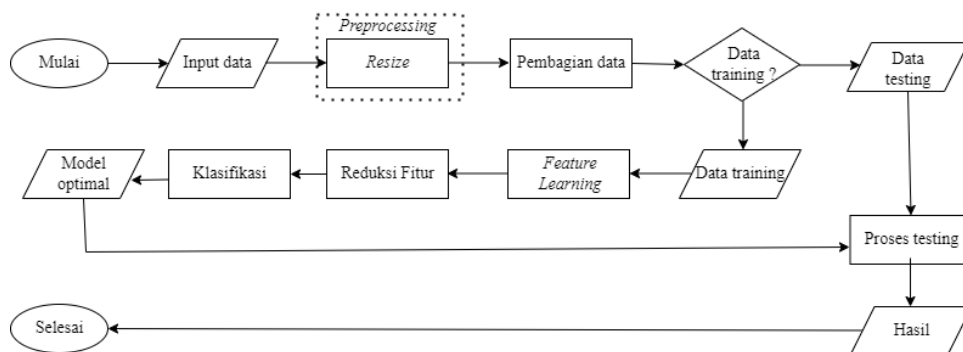
$$\text{Akurasi} = \frac{TP_{all}}{n_{all}} \times 100\%$$

$$\text{Sensitivitas} = \frac{\sum \frac{TP_{all}}{TP_{all} + FN_{all}}}{n_{all}} \times 100\% \quad (7)$$

$$\text{Spesifisitas} = \frac{\sum \frac{TN_{all}}{TN_{all} + FP_{all}}}{n_{all}} \times 100\%$$

Bahan dan Metode

Penelitian ini memiliki beberapa tahapan yaitu *preprocessing*, *feature learning*, reduksi fitur, klasifikasi, dan evaluasi model. Diagram alir penelitian akan ditunjukkan pada Gambar 6.



Gambar 6. Flowchart

Hasil dan Pembahasan

Pada penelitian ini dengan menggunakan data citra histopatologi kanker paru yang diperoleh dari web *Cornell University* sejumlah 15000 *dataset* yang terdiri dari 3 kelas yaitu *Adenocarcinoma*, *Benign*, dan *Squamous Cell Carcinoma*. Data citra histopatologi yang diperoleh merupakan citra RGB yang berukuran 768×768 piksel. Citra RGB merupakan suatu citra yang memiliki 3 *channel* warna yaitu *Red* (R), *Green* (G), dan *Blue* (B). Citra ini perlu dilakukan *resizing* untuk menyamakan ketentuan dari metode yang digunakan yakni metode CNN model *GoogLeNet* dengan ketentuan ukuran *input* citra sebesar 224×224 .

Setelah melakukan proses *resizing* maka selanjutnya dengan melakukan ekstraksi fitur yang hasilnya diperoleh dari *feature learning*. Kemudian, setelah diperoleh fitur-fitur pada proses sebelumnya dimana fitur yang dihasilkan sangat banyak. Oleh karena itu, dilakukan proses reduksi fitur dengan menggunakan PCA untuk mengurangi fitur yang digunakan. Setelah dilakukan reduksi PCA maka dilanjutkan dengan klasifikasi. Kemudian, diperoleh model yang optimal yang nantinya akan dilakukan beberapa uji coba dengan beberapa parameter yaitu pembagian data, probabilitas *dropout*, serta *batchsize*. Berikut hasil uji coba yang ditunjukkan oleh Tabel 1.

Tabel 1. Hasil Uji Coba

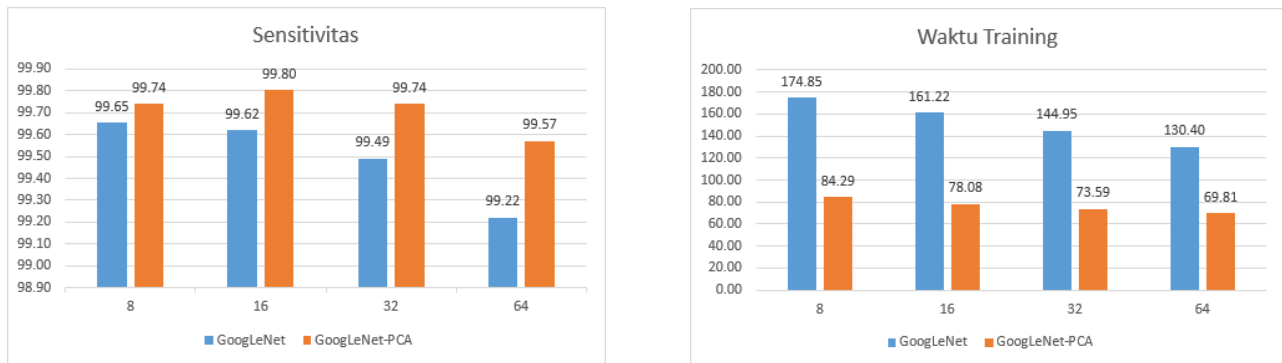
Pem	Prob	Nilai	GoogLeNet				GoogLeNet-PCA			
			Batch				Batch			
			8	16	32	64	8	16	32	64

60:40	0.6	Acc	99.30	99.58	99.55	99.15	99.40	99.60	99.65	99.35
		Sen	99.31	99.59	99.55	99.13	99.45	99.69	99.75	99.40
		Spe	99.65	99.79	99.78	99.57	99.85	99.80	99.88	99.67
		Time	177m 34s	159m 13s	142m 52s	129m 32s	82m 32s	76m 18s	74m 28s	68m 28s
	0.7	Acc	99.75	99.68	99.57	99.33	99.8	99.78	99.67	99.43
		Sen	99.75	99.68	99.57	99.34	99.81	99.80	99.60	99.54
		Spe	99.88	99.84	99.78	99.67	99.9	99.85	99.88	99.77
		Time	186m 33s	174m 13s	131m 21s	121m 45s	87m 31s	77m 28s	76m 23s	72m 1s
	0.8	Acc	99.78	99.43	99.57	99.30	99.81	99.83	99.70	99.53
		Sen	99.78	99.44	99.57	99.30	99.88	99.85	99.75	99.53
		Spe	99.89	99.72	99.78	99.65	99.9	99.75	99.78	99.69
		Time	172m 20s	154m 16s	152m 57s	125m 49s	85m 13s	80m 46s	72m 44s	70m 43s
	0.9	Acc	99.72	99.77	99.27	99.10	99.82	99.87	99.87	99.71
		Sen	99.77	99.77	99.27	99.11	99.82	99.87	99.86	99.81
		Spe	99.86	99.88	99.63	99.55	99.86	99.88	99.63	99.75
		Time	163m 51s	157m 44s	153m 51s	145m 32s	82m 4s	78m 4s	71m 4s	68m 42s
70:30	0.6	Acc	99.91	99.82	99.71	99.38	99.30	99.58	99.55	99.15
		Sen	99.91	99.82	99.71	99.38	99.81	99.89	99.85	99.83
		Spe	99.96	99.91	99.86	99.69	99.65	99.79	99.78	99.57
		Time	201m 29s	172m 1s	171m 3s	170m 48s	85m 3s	77m 28s	70m 2s	68m 2s
	0.7	Acc	99.76	99.47	99.42	99.22	99.75	99.68	98.93	98.87
		Sen	99.76	99.47	99.43	99.22	99.78	99.78	99.77	99.74
		Spe	99.88	99.73	99.71	99.61	99.88	99.84	99.78	99.67
		Time	169m 26s	152m 0s	148m 19s	148m 4s	89m 1s	79m 8s	71m 8s	65m 34s
	0.8	Acc	99.82	99.78	99.73	98.8	99.51	99.47	99.02	97.93
		Sen	99.82	99.78	99.73	98.83	99.88	99.84	99.87	99.83
		Spe	99.91	99.89	99.87	99.41	99.89	99.72	99.78	99.65
		Time	181m 34s	165m 27s	151m 31s	149m 59s	96m 43s	76m 1s	74m 3s	73m 19s
	0.9	Acc	99.76	99.73	99.47	99.38	99.11	99.2	98.87	97.82
		Sen	99.76	99.73	99.48	99.38	99.82	99.87	99.87	99.81
		Spe	99.88	99.87	99.74	99.69	99.86	99.88	99.63	99.55
		Time	169m	164m	159m	141m	103m	106m	65m	53m

			28s	29s	29s	49s	0s	4s	34s	21s
80:20	0.6	Acc	99.77	99.33	99.60	99.63	99.30	99.58	99.55	99.15
		Sen	99.77	99.34	99.60	98.67	99.61	99.69	99.65	99.63
		Spe	99.88	99.67	99.80	99.33	99.65	99.79	99.78	99.57
		Time	181m 3s	176m 46s	162m 38s	162m 38s	104m 3s	90m 39s	87m 46s	81m 17s
	0.7	Acc	99.80	99.70	99.00	99.87	99.75	99.68	99.57	99.33
		Sen	99.80	99.70	99.02	99.89	99.85	99.88	99.87	99.84
		Spe	99.90	99.85	99.51	99.44	99.88	99.84	99.78	99.67
		Time	191m 47s	158m 48s	150m 44s	148m 28s	108m 5s	93m 46s	88m 58s	86m 48s
	0.8	Acc	99.77	99.33	99.60	98.63	99.78	99.43	99.57	99.30
		Sen	99.77	99.34	99.60	98.67	99.88	99.84	99.87	99.83
		Spe	99.88	99.67	99.80	99.33	99.89	99.72	99.78	99.65
		Time	189m 4s	171m 8s	163m 27s	139m 38s	113m 2s	74m 53s	71m 48s	65m 4s
	0.9	Acc	99.87	99.80	99.40	99.43	99.72	99.77	99.27	99.10
		Sen	99.87	99.80	99.41	99.43	99.82	99.87	99.87	99.81
		Spe	99.93	99.90	99.70	99.72	99.86	99.88	99.63	99.55
		Time	198m 27s	175m 27s	162m 22s	160m 56s	87m 53s	79m 5s	74m 24s	74m 8s
90:10	0.6	Acc	99.80	99.87	99.53	99.60	99.30	99.58	99.55	99.15
		Sen	99.80	99.87	99.54	99.60	99.71	99.79	99.75	99.73
		Spe	99.90	99.93	99.77	99.80	99.65	99.79	99.78	99.57
		Time	209m 49s	187m 59s	188m 45s	173m 1s	103m 7s	91m 2s	87m 4s	80m 51s
	0.7	Acc	99.93	99.73	99.73	99.33	99.72	99.68	99.57	99.33
		Sen	99.93	99.73	99.73	99.35	99.85	99.88	99.87	99.84
		Spe	99.97	99.87	99.87	99.67	99.88	99.84	99.78	99.67
		Time	190m 2s	174m 11s	168m 46s	153m 26s	109m 2s	86m 9s	86m 0s	81m 12s
	0.8	Acc	99.87	99.80	99.73	99.47	99.78	99.43	99.57	99.30
		Sen	99.87	99.80	99.73	99.47	99.78	99.74	99.77	99.73
		Spe	99.93	99.9	99.87	99.73	99.89	99.72	99.78	99.65
		Time	208m 9s	184m 55s	172m 8s	163m 53s	92m 13s	89m 5s	83m 56s	83m 56s
	0.9	Acc	99.87	99.87	99.53	99.80	99.95	99.77	99.27	99.10
		Sen	99.09	99.80	99.60	99.80	99.97	99.77	99.27	99.11
		Spe	99.54	99.90	99.80	99.90	99.86	99.88	99.63	99.55

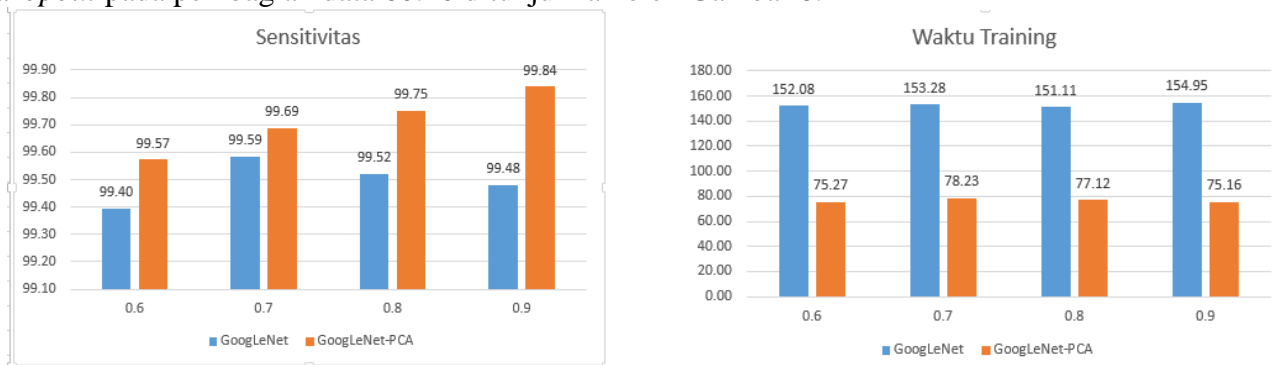
		Time	202m	181m	194m	176m	93m	96m	96m	93m
			26s	4s	54s	22s	27s	13s	27s	1s

Berdasarkan hasil uji coba yang dilakukan dengan pembagian data 60:40, pembagian data 70:30, pembagian data 80:20, dan pembagian data 90:10 yang ditunjukkan oleh Tabel 1. maka diperoleh rata-rata sensitivitas dan waktu *training* berdasarkan jumlah *batchsize* pada pembagian data 60:40 yang ditunjukkan oleh Gambar 7.



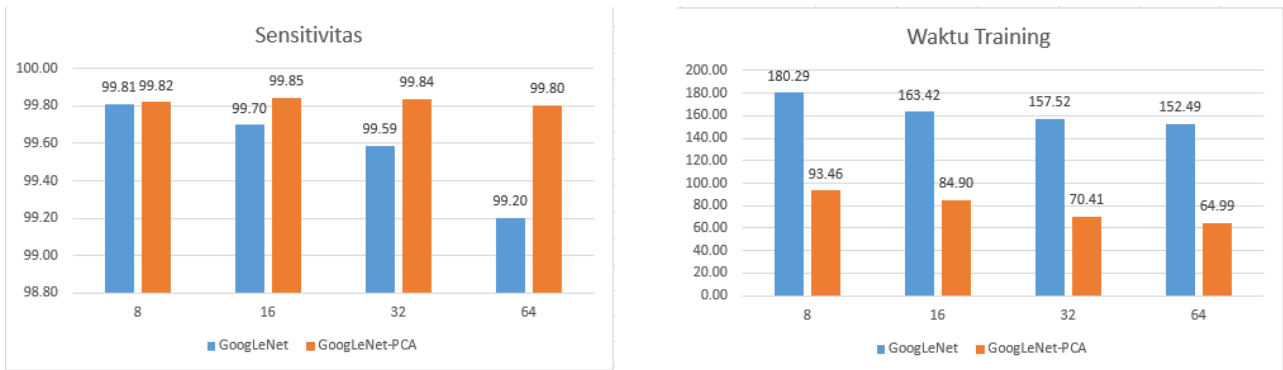
Gambar 7. Rata-Rata Sensitivitas dan Waktu *Training* Berdasarkan Jumlah *Batchsize* Pada Pembagian 60:40

Berdasarkan Gambar 7. dapat diketahui bahwa nilai sensitivitas yang diperoleh *GoogLeNet-PCA* tertinggi sebesar 99.80%, sedangkan pada metode *GoogLeNet* saja sebesar 99.74%. Kemudian, pada waktu *training* juga membutuhkan waktu yang lebih sedikit dengan menggunakan *GoogLeNet-PCA* yaitu sebesar 69 menit 81 detik dibandingkan tanpa PCA waktu yang paling cepat yaitu sebesar 130 menit 40 detik. Sedangkan, rata-rata sensitivitas serta waktu *training* berdasarkan nilai probabilitas *dropout* pada pembagian data 60:40 ditunjukkan oleh Gambar 8.



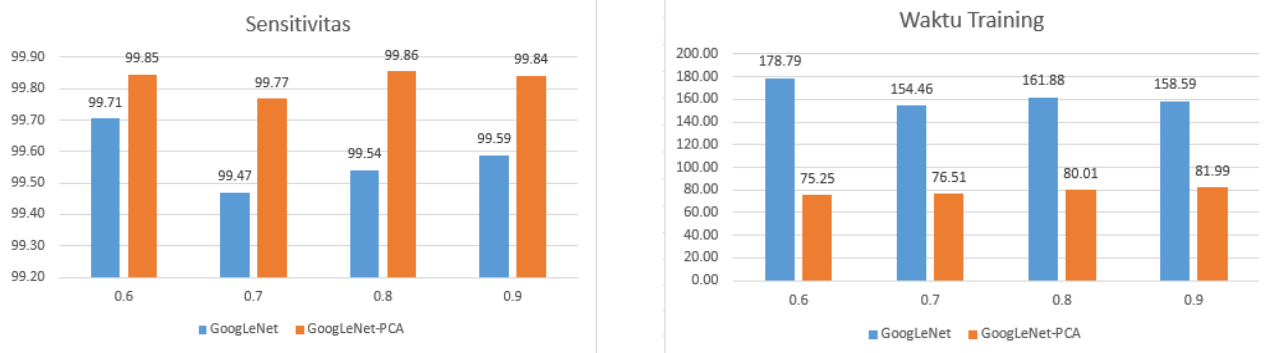
Gambar 8. Rata-Rata Sensitivitas dan Waktu *Training* Berdasarkan Probabilitas *Dropout* Pada Pembagian 60:40

Pada Gambar 8. Diketahui bahwa nilai sensitivitas tertinggi berdasarkan nilai probabilitas *dropout* pada *GoogLeNet-PCA* lebih tinggi yaitu sebesar 99.84%, sedangkan pada *GoogLeNet* saja yakni sebesar 99.59%. Selain itu, waktu *training* yang dibutuhkan lebih sedikit dengan menggunakan *GoogLeNet-PCA* yaitu sebesar 75 menit 15 detik sedangkan pada *GoogLeNet* saja membutuhkan waktu sebanyak 155 menit 11 detik. Kemudian, pada pembagian data 70:30 rata-rata sensitivitas dan waktu *training* berdasarkan jumlah *batchsize* pada ditunjukkan oleh Gambar 9.



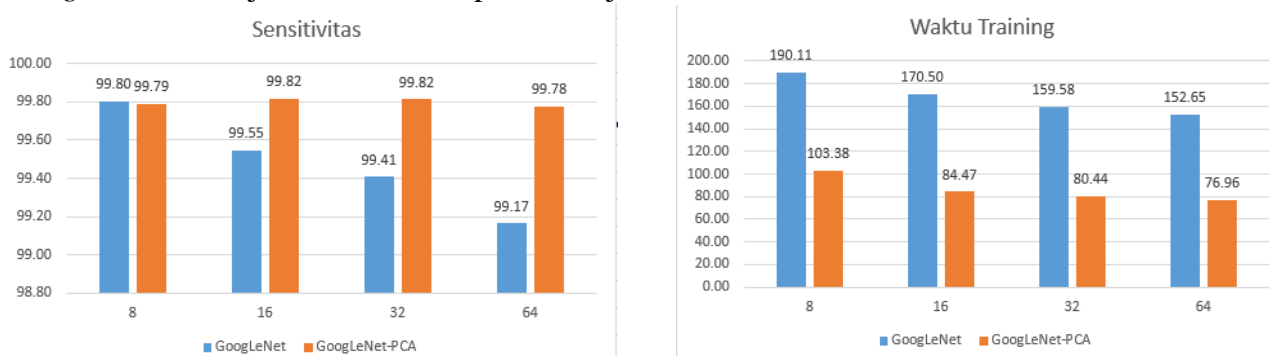
Gambar 9. Rata-Rata Sensitivitas dan Waktu *Training* Berdasarkan Jumlah *Batchsize* Pada Pembagian 70:30

Berdasarkan Gambar 9. Diketahui bahwa nilai sensitivitas tertinggi yaitu diperoleh menggunakan *GoogLeNet-PCA* yaitu sebesar 99.85% sedangkan menggunakan *GoogLeNet* saja nilai sensitivitas tertinggi sebesar 99.81%. Kemudian, waktu *training* yang dibutuhkan juga lebih sedikit dengan menggunakan *GoogLeNet-PCA* yaitu sebesar 64 menit 99 detik. Sedangkan, rata-rata sensitivitas serta waktu *training* berdasarkan nilai probabilitas *dropout* pada pembagian data 70:30 ditunjukkan oleh Gambar 10.



Gambar 10. Rata-Rata Sensitivitas dan Waktu *Training* Berdasarkan Probabilitas *Dropout* Pada Pembagian 70:30

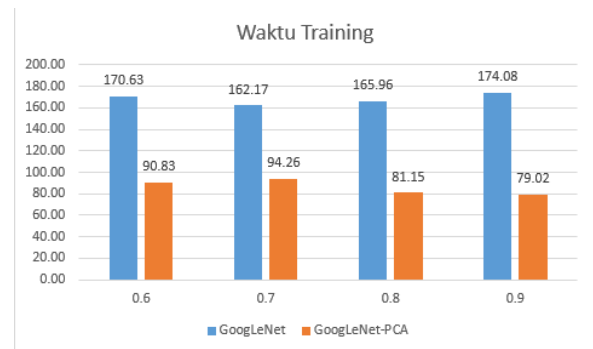
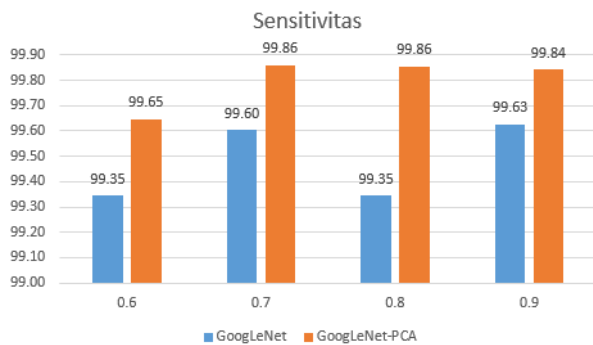
Pada Gambar 10. Menunjukkan bahwa nilai sensitivitas tertinggi berdasarkan nilai probabilitas *dropout* diperoleh menggunakan *GoogLeNet-PCA* yaitu sebesar 99.86%. Selain itu, waktu yang dibutuhkan lebih sedikit dibandingkan menggunakan *GoogLeNet* yakni hanya membutuhkan waktu sebanyak 75 menit 25 detik. Kemudian, pada pembagian data 80:20 rata-rata sensitivitas dan waktu *training* berdasarkan jumlah *batchsize* pada ditunjukkan oleh Gambar 11.



Gambar 11. Rata-Rata Sensitivitas dan Waktu *Training* Berdasarkan Jumlah *Batchsize* Pada Pembagian 80:20

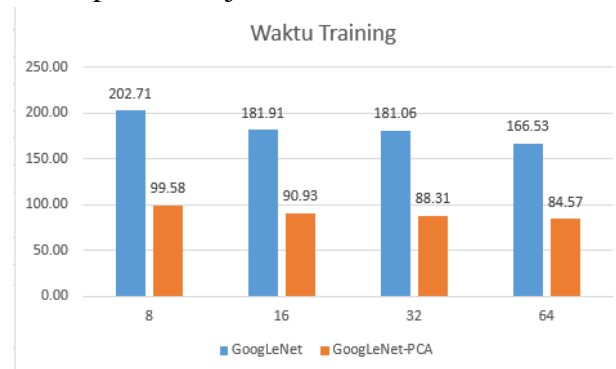
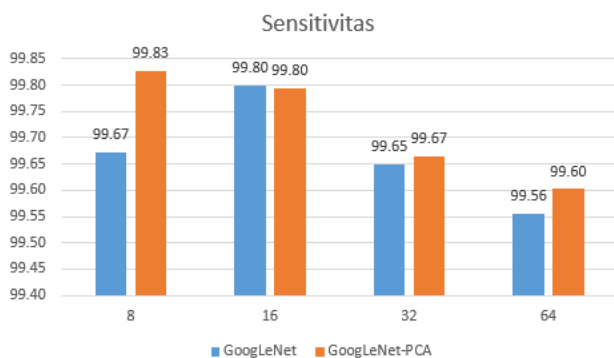
Berdasarkan Gambar 11. Diketahui bahwa nilai sensitivitas tertinggi pada *GoogLeNet-PCA* yaitu sebesar 99.82% sedangkan pada *GoogLeNet* saja sebesar 99.80%. Kemudian, waktu *training* yang dibutuhkan juga lebih sedikit dengan menggunakan *GoogLeNet-PCA* yaitu sebesar 76 menit 96 detik.

Sedangkan, rata-rata sensitivitas serta waktu *training* berdasarkan nilai probabilitas *dropout* pada pembagian data 80:20 ditunjukkan oleh Gambar 12.



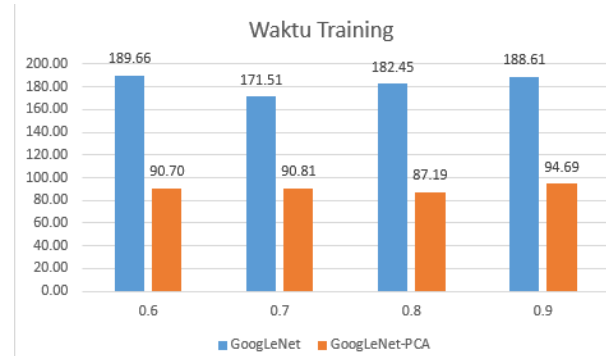
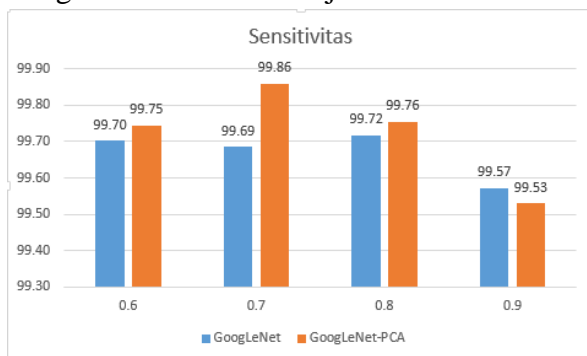
Gambar 12. Rata-Rata Sensitivitas dan Waktu *Training* Berdasarkan Probabilitas *Dropout* Pada Pembagian 80:20

Pada Gambar 12. Menunjukkan bahwa nilai sensitivitas berdasarkan nilai probabilitas *dropout* pada *GoogLeNet-PCA* lebih tinggi yaitu sebesar 99.86% dibandingkan dengan *GoogLeNet* saja yakni sebesar 99.63%. Kemudian, pada waktu *training* juga membutuhkan waktu yang lebih sedikit dengan menggunakan *GoogLeNet-PCA* yaitu sebesar 79 menit 2 detik. Pada pembagian data 80:20 rata-rata sensitivitas dan waktu *training* berdasarkan jumlah *batchsize* pada ditunjukkan oleh Gambar 13.



Gambar 13. Rata-Rata Sensitivitas dan Waktu *Training* Berdasarkan Jumlah *Batchsize* Pada Pembagian 90:10

Berdasarkan Gambar 11. Diketahui bahwa nilai sensitivitas tertinggi pada *GoogLeNet-PCA* yaitu sebesar 99.83% sedangkan pada *GoogLeNet* saja sebesar 99.67%. Kemudian, pada waktu *training* yang dibutuhkan juga lebih sedikit dengan menggunakan *GoogLeNet-PCA* yaitu sebesar 84 menit 57 detik. Sedangkan, rata-rata sensitivitas serta waktu *training* berdasarkan nilai probabilitas *dropout* pada pembagian data 90:10 ditunjukkan oleh Gambar 14.



Gambar 14. Rata-Rata Sensitivitas dan Waktu *Training* Berdasarkan Probabilitas *Dropout* Pada Pembagian 90:10

Berdasarkan Gambar 8. Diketahui bahwa nilai sensitivitas berdasarkan nilai probabilitas *dropout* pada *GoogLeNet-PCA* lebih tinggi yaitu sebesar 99.86% dibandingkan dengan *GoogLeNet* saja yakni

sebesar 99.72%. Kemudian, pada waktu *training* juga membutuhkan waktu yang lebih sedikit dengan menggunakan *GoogLeNet*-PCA yaitu sebesar 94 menit 69 detik.

Oleh karena itu, klasifikasi kanker paru menggunakan *GoogLeNet*-PCA memiliki nilai sensitivitas yang lebih tinggi dibandingkan menggunakan *GoogLeNet* saja. Hal ini menunjukkan bahwasanya reduksi fitur menggunakan PCA sangat memiliki pengaruh dalam nilai sensitivitas yang diperoleh. Selain itu, waktu *training* yang dibutuhkan juga lebih sedikit dibandingkan tanpa PCA yaitu rata-rata dibawah 100 menit. Hal ini juga menunjukkan semakin sedikit waktu yang dibutuhkan maka semakin sedikit pula memory sistem yang dibutuhkan.

Kesimpulan

Berdasarkan hasil penelitian ini yakni pengaruh reduksi fitur pada klasifikasi kanker paru menggunakan metode CNN model *GoogLeNet* memperoleh hasil yang lebih unggul dibandingkan menggunakan *GoogLeNet* saja. Hasil terbaik yang diperoleh menggunakan *GoogLeNet*-PCA yaitu pada pembagian data 90:10 dengan nilai probabilitas *dropout* 0.9 dan jumlah *batchsize* 8 dengan memperoleh nilai akurasi, sensitivitas, spesifitas berturut-turut yaitu 99.95%, 99.97%, dan 99.86%. Hasil ini hanya membutuhkan waktu *training* selama 93 menit 27 detik.

Referensi

- [1] K. Xu *et al.*, "Progress of exosomes in the diagnosis and treatment of lung cancer," *Biomed. Pharmacother.*, vol. 134, p. 111111, 2021.
- [2] H. Sung *et al.*, "Global cancer statistics 2020: GLOBOCAN estimates of incidence and mortality worldwide for 36 cancers in 185 countries," *CA. Cancer J. Clin.*, vol. 71, no. 3, pp. 209–249, 2021.
- [3] L. Listyalina, E. L. Utari, and D. E. Puspaningtyas, "Penentuan Penyakit Paru Dengan Menggunakan Jaringan Saraf Tiruan," *Simetris J. Tek. Mesin, Elektro dan Ilmu Komput.*, vol. 11, no. 1, pp. 233–240, 2020.
- [4] M. A. Al-Antari, C.-H. Hua, J. Bang, and S. Lee, "Fast deep learning computer-aided diagnosis of COVID-19 based on digital chest x-ray images," *Appl. Intell.*, vol. 51, no. 5, pp. 2890–2907, 2021.
- [5] O. L. Istifarinta, P. Prajitno, and D. S. Soejoko, "Computer-Aided Detection (CAD) Deteksi Nodul Paru-Paru dari Computed Tomography (CT)," *J. Med. Phys. Biophys.*, vol. 9, no. 1, pp. 10–17, 2022.
- [6] E. L. Henriksen, J. F. Carlsen, I. M. M. Vejborg, M. B. Nielsen, and C. A. Lauridsen, "The efficacy of using computer-aided detection (CAD) for detection of breast cancer in mammography screening: a systematic review," *Acta radiol.*, vol. 60, no. 1, pp. 13–18, 2019.
- [7] B. K. Hatuwal and H. C. Thapa, "Lung cancer detection using convolutional neural network on histopathological images," *Int. J. Comput. Trends Technol.*, vol. 68, no. 10, pp. 21–24, 2020.
- [8] C. Yi, "Application of Convolutional Networks in Clothing Design from the Perspective of Deep Learning," *Sci. Program.*, vol. 2022, 2022.
- [9] D. Haritha, N. Swaroop, and M. Mounika, "Prediction of COVID-19 Cases Using CNN with X-rays," in *2020 5th International Conference on Computing, Communication and Security (ICCCS)*, IEEE, 2020, pp. 1–6.
- [10] P. Salavati and H. M. Mohammadi, "Obstacle detection using GoogleNet," in *2018 8th International Conference on Computer and Knowledge Engineering (ICCKE)*, IEEE, 2018, pp. 326–332.
- [11] M. S. AL-Huseiny and A. S. Sajit, "Transfer learning with GoogLeNet for detection of lung cancer," *Indones. J. Electr. Eng. Comput. Sci.*, vol. 22, no. 2, pp. 1078–1086, 2021.
- [12] A. K. Idrees and A. K. M. Al-Qurabat, "Energy-efficient data transmission and aggregation protocol in periodic sensor networks based fog computing," *J. Netw. Syst. Manag.*, vol. 29, no. 1, pp. 1–24, 2021.

- [13] A. K. Gárate-Escamila, A. H. El Hassani, and E. Andrès, “Classification models for heart disease prediction using feature selection and PCA,” *Informatics Med. Unlocked*, vol. 19, p. 100330, 2020.
- [14] B. Sahu, S. Mohanty, and S. Rout, “A hybrid approach for breast cancer classification and diagnosis,” *EAI Endorsed Trans. Scalable Inf. Syst.*, vol. 6, no. 20, 2019.
- [15] R. T. Prasetyo and S. Susanti, “Prediksi Harapan Hidup Pasien Kanker Paru Pasca Operasi Bedah Toraks Menggunakan Boosted k-Nearest Neighbor,” *J. Responsif Ris. Sains dan Inform.*, vol. 1, no. 1, pp. 64–69, 2019.
- [16] N. P. Nugroho and F. F. Wati, “Koeksistensi Kanker Paru Dan Tuberkulosis,” *Syifa’Med J Kedokt dan Kesehat*, vol. 11, no. 1, p. 49, 2020.
- [17] P. Wang, E. Fan, and P. Wang, “Comparative analysis of image classification algorithms based on traditional machine learning and deep learning,” *Pattern Recognit. Lett.*, vol. 141, pp. 61–67, 2021, doi: 10.1016/j.patrec.2020.07.042.
- [18] A. Khan, A. Sohail, U. Zahoor, and A. S. Qureshi, “A survey of the recent architectures of deep convolutional neural networks,” *Artif. Intell. Rev.*, vol. 53, no. 8, pp. 5455–5516, 2020.
- [19] J. Gupta, S. Pathak, and G. Kumar, “A hybrid optimization-tuned deep convolutional neural network for bare skinned image classification in websites,” *Multimed. Tools Appl.*, pp. 1–23, 2022.
- [20] N. D. Miranda, L. Novamizanti, and S. Rizal, “Convolutional Neural Network pada klasifikasi sidik jari menggunakan RESNET-50,” *J. Tek. Inform.*, vol. 1, no. 2, pp. 61–68, 2020.
- [21] S. Ilahiyah and A. Nilogiri, “Implementasi Deep Learning Pada Identifikasi Jenis Tumbuhan Berdasarkan Citra Daun Menggunakan Convolutional Neural Network,” *JUSTINDO (Jurnal Sist. Dan Teknol. Inf. Indones.)*, vol. 3, no. 2, pp. 49–56, 2018.
- [22] L. Tsochatzidis, L. Costaridou, and I. Pratikakis, “Deep learning for breast cancer diagnosis from mammograms—a comparative study,” *J. Imaging*, vol. 5, no. 3, p. 37, 2019.
- [23] C. Garbin, X. Zhu, and O. Marques, “Dropout vs. batch normalization: an empirical study of their impact to deep learning,” *Multimed. Tools Appl.*, vol. 79, no. 19, pp. 12777–12815, 2020.
- [24] F. Demir, D. A. Abdullah, and A. Sengur, “A new deep CNN model for environmental sound classification,” *IEEE Access*, vol. 8, pp. 66529–66537, 2020.
- [25] S. P. Kannoja and G. Jaiswal, “Effects of varying resolution on performance of CNN based image classification: An experimental study,” *Int. J. Comput. Sci. Eng.*, vol. 6, no. 9, pp. 451–456, 2018.
- [26] F. Yuesheng *et al.*, “Circular fruit and vegetable classification based on optimized GoogLeNet,” *IEEE Access*, vol. 9, pp. 113599–113611, 2021.
- [27] S. Bhattacharya *et al.*, “A novel PCA-firefly based XGBoost classification model for intrusion detection in networks using GPU,” *Electronics*, vol. 9, no. 2, p. 219, 2020.
- [28] M. A. S. Yudono, E. A. Z. Hamidi, A. H. K. Jumadi, A. De, and W. M. Sidik, “Jaringan Syaraf Tiruan Perambatan Balik untuk Klasifikasi Covid-19 Berbasis Tekstur Menggunakan Orde Pertama Berdasarkan Citra Chest X-Ray,” *J. Teknol. Inf. dan Ilmu Komput.*, vol. 9, no. 4, pp. 799–808, 2022.