



COMPLETE | Journal of Computer, Electronic, and Telecommunication | Vol.4 No.2 December 2023

COMPLETE

Journal of Computer, Electronic,
and Telecommunication

Vol.4 No.2 December 2023



JURNAL

KEMAJUAN TEKNIK
KOMPUTER, ELEKTRO, & TELEKOMUNIKASI

ISSN: 2723-4371 E-ISSN: 2723-5912



Website: journal.ittelkom-sby.ac.id/complete | E-mail: complete@ittelkom-sby.ac.id

Institut Teknologi Telkom Surabaya
Ketintang Str. No. 156, Surabaya, 60231,
East Java, Indonesia
E-mail: complete@ittelkom-sby.ac.id;
Web: journal.ittelkom-sby.ac.id/complete



EDITORIAL TEAM

Editor in Chief:

Dr. Chaironi Latif, S.Si, *Institut Teknologi Telkom Surabaya, Indonesia*

Editorial Members:

Dr. Filbert Hilman Juwono, Xi'an Jiaotong-Liverpool University, China

Dr. M. V. Reddy, Hydro-Québec, Canada

Dr. Eng. Muhammad Abdillah, S.T., M.T., *Universitas Pertamina, Indonesia*

Dr. Isa Hafidz., ST., MT, *Institut Teknologi Telkom Surabaya, Indonesia*

Nisa Isrofi., ST., MT, *Institut Teknologi Telkom Surabaya, Indonesia*

Aulia Rahma Annisa, S.ST., M.T., *Institut Teknologi Telkom Surabaya, Indonesia*

Billy Montolalu, S.Kom., M.Kom., *Institut Teknologi Telkom Surabaya, Indonesia*

Peer Reviewers:

Dr. Purba Daru Kusuma, ST, MT, *Telkom University, Indonesia*

Dr. Susijanto Tri Rasmana, S.Kom., M.T, *Institut Teknologi Telkom Surabaya, Indonesia*

Mohamad Ridwan. ST., M.T., *Electronic Eng. Polytechnic Institute of Surabaya, Indonesia*

Rahmat Febrianto Wijanarko, *Institut Teknologi Perusahaan Listrik Negara, Indonesia*

Dimas Adiputra, ST., M.Phil, Ph.D, *Institut Teknologi Telkom Surabaya, Indonesia*

Anifatul Faricha, ST., M.Sc, Ph.D, *Institut Teknologi Telkom Surabaya, Indonesia*

Muhsin, ST., MT, Ph.D, *Institut Teknologi Telkom Surabaya, Indonesia*

Journal of Computer, Electronic, and Telecommunication

Faculty of Electrical Engineering - Institut Teknologi Telkom Surabaya

Ketintang Str. No. 156, Surabaya, 60231, East Java, Indonesia

Phone. +6285 234852155; E-mail: complete@ittelkom-sby.ac.id;

Web: journal.ittelkom-sby.ac.id/complete

(This page is intentionally left blank)

TABLE OF CONTENTS

COVER PAGE

EDITORIAL TEAM	1
TABLE OF CONTENTS	3
ABOUT THE JOURNAL	5
PREFACE	7

ARTICLES

1. Design of 3-Phase Load Dividing Automatic System with Naive Bayes Method	9-22
2. Pengembangan Cyber Physical System untuk Presensi dengan Deteksi Multi Wajah	23-30
3. Rancang Bangun Aplikasi Intelligent Visual Scanner berbasis CNN untuk Identifikasi Cacat pada Hasil Pengelasan	31-42
4. Perbandingan Performa Algoritma VGG16 dan VGG19 Melalui Metode CNN Untuk Klasifikasi Varietas Beras	43-58
5. Enhancing Heart Disease Detection Using Convolutional Neural Networks and Classic Machine Learning Methods	59-74
6. Review : Pengenalan Pola Objek Satu Dimensi	75-90

(This page is intentionally left blank)

ABOUT THE JOURNAL

COMPLETE : Journal of Computer, Electronic, and Telecommunication is an official journal of the Faculty of Electrical Engineering, Institut Teknologi Telkom Surabaya. It publishes research or review articles in Computer, Electronic, and Telecommunication technology. This journal provides a platform for research lecturers, reviewers, practitioners, industry, and observers across Indonesia and overseas to promote, share, and discuss new issues and technology development. The scope of the journal includes:

- Technology utilization of maritime resources
- Strengthening infrastructure maritime
- Technology and management safety transportation
- Industrial strengthening technology transportation
- Supporting infrastructure and transportation system
- Operational efficiency
- Electronics Technology
- Telecommunication Technology
- Computer Technology
- System Security
- Advanced Robotics Technology
- Technology and disaster management
- Advanced Power Electronics
- Application of Power System
- Renewable Energy
- Chips Technology
- Sensor Technology
- Smart IoT Devices
- 5G Technology and Ecosystems
- Technology and management environment

COMPLETE published twice a year. Editors invite research lecturers, the reviewer, practitioners, industry, and observers to contribute to this journal. The language used in the form of Indonesian and English. All accepted manuscripts will receive individual digital object identifier (DOI) and indexed by Sinta, Arjuna, and Google Scholar. The online PDF version of the journal is open access from journal.ittelkom-sby.ac.id/complete. Subscription of the hard copy can be requested by email to complete@ittelkom-sby.ac.id.

(This page is intentionally left blank)

PREFACE

Welcome to the Journal of Computer, Electronic, and Telecommunication, Vol. 4, Issue No. 2. It is my privilege and pleasure to present the fourth volume of this peer-reviewed journal under Institut Teknologi Telkom Surabaya. This journal aims to accommodate the results of research publications through national and worldwide journals as part of continuous improvement.

As the chairman of COMPLETE, I would like to thank many people who supported this journal, especially Research and Community Service Units (LPPM). Furthermore, as the editor-in-chief, I would like to extend my sincere thanks to all members of the editorial and the advisory boards from Institut Teknologi Telkom Surabaya, whose service, dedication, and commitment have made the creation of this journal possible. We work together to improve the quality and excellence of articles published continuously.

We hope that COMPLETE could deliver valuable and interesting information and stimulate further research to the worldwide telecommunications, electrical, and computer engineering communities.

Surabaya, December 2023

Editor-In-Chief of Journal

(This page is intentionally left blank)

Design of 3-Phase Load Dividing Automatic System with Naive Bayes Method

Faris Anugrah Pratama¹, Hendro Agus Widodo², Anggara Trisna Nugraha^{3*}, Rama Arya Sobhita⁴, Widi Nur Hidayat⁵

¹⁻⁵ Shipbuilding Institute of Polytechnic Surabaya, Surabaya, Indonesia

* Correspondence: anggaranugraha@ppns.ac.id

Abstract: The electric power distribution system is a process for distributing electricity from a 150 kV electric power transmission system to consumers, both 20 kV consumers and 380/220 V consumers. The problem that often occurs in the electric power distribution system at PT.X is load imbalance due to consumption power on each phase is not the same so that a neutral current appears. The current flowing in the neutral phase will cause power losses. Based on these problems, a load balancing device was made for a three-phase network with the standard SPLN No. 17 of 2014, namely the magnitude of the value of the current imbalance between the phases may not exceed 25%. This tool is also equipped with an overcurrent protection system and a tapping phase limitation system to support the performance of the load balancing device on a three-phase network. The method used is the Naive Bayes method. The naïve Bayes method is used to calculate the current parameters in the R, S, T, and N phases. The classification results will be used as a reference for activating the tapping phase system and the protection system. This system uses Internet of Things technology as a media monitoring of neutral current, current and voltage values in phases R, S and T as well as power losses values. Based on the test results, when there is an imbalance of 30.20%, a neutral current of 1.20A appears, causing a power loss of 0.17W. After the tapping phase is carried out, the percentage of unbalance is reduced to 12.20%, the neutral current is 0.45A, and the power losses are 0.02W.

Keywords: Load Unbalance, Neutral Current, Tapping Phase

1. Introduction

The most important system in the electric power industry is the distribution system since it has a continuous connection with electricity consumers [1]. The process of sending ten-gauge electric energy from the 150-kV transmission system to consumers (whether 20-kV or 380/220-volt consumers) is known as the system distribution. The most often used distribution system is a three-phase with a four-wire system that consists of three-phase cables and one global cable [2], [3].

Unbalanced load on each phase (R, S, and T) is a non-technical problem often occurring in power distribution systems [4]. The low voltage distribution system network has extensive network coverage, which often causes the low voltage distribution system to become unbalanced. This is because household customers use single-phase power, causing load imbalance in the low-voltage distribution network [5]-[7].

An unbalanced low-voltage distribution system will have an impact on a number of things, including transformer operation, overheating of the overloaded phase, current flowing in the neutral wire, and a drop in end voltage on the overloaded phase network. So that the electric power at the consumer level becomes worse [8]. Transferring a large one-phase load to a lower one is one method of reducing engineering losses [9]. The solution to this load imbalance is to balance the load on the transformer [10].

According to Sinaga, loads affect the quality of the electric power system's electrical energy [11]. The quality of electrical energy changes when there is a load imbalance. Voltage unbalance is one such change. This occurs due to load imbalance, which occurs when one of the phases carries a greater load on the lines of a three-phase system [12].

In the paper entitled "Load Balancing Analysis of the Low Voltage Network of the AMH02 Distribution Substation Amahai Feeder at PT. PLN (Persero) Masohi," written by C. de Queljoe et al. in 2020, the procedure for loading on traffic is explained. The current procedure is being carried out to address issues that frequently arise in traffic, such as traffic jams, excess points at lanes that are closer together, currents flowing at highway intersections, and drops of voltage at lanes that are closer together. Balancing the load is carried out via the method of moving the house's tap [13].

In 2020, Hilmi Dzakwan conducted a study titled "Load Balancing Design on Secondary Distribution Channels by Monitoring Power Losses Due to Neutral Currents Based on IoT." It describes the design and load balancing devices on secondary distribution lines that can be constructed to reduce load balancing by as much as 20% [14]. If a load imbalance occurs, the load will automatically switch from the phase with the high current load to the phase with the lower current load [15].

A study entitled "Auto feeder Design and Water Quality Monitoring for Vaname Shrimp Ponds at Ibab Banjar Kemuning Using the Naive Bayes Method Based on Wireless Sensor Networks" explained how pond water conditions were classified using the naive Bayes method [16]. The naive Bayes method uses pH, temperature, ammonia, and salinity sensors to calculate water quality [17]. The android application receives classification results in real time and is used as a reference to activate water pumps and DC motors to monitor water conditions and regulate shrimp feeding according to pond conditions. The author wants adaptation of this technique to simplify the classification of transformer conditions and simplify the calculation of current parameters in the R, S, T, and N [18], [19]. The results of the classification will be used as a reference to start the phase tapping system and protection system.

The phase current parameters R, S, T, and N are calculated using the Naïve Bayes method to create a prototype Design of a Load Sharing Phase Tapping System in a Three-Phase Network Using the Naïve Bayes Method. The results of the classification will be used as a reference to activate the tapping phase and protection system. In accordance with the standard SPLN No. 17 of 2014, the limit for imbalance in this system is 25% [20]. This system categorizes imbalances based on three-time intervals. The first interval lasts from 1 second to 4 seconds and will turn on the indicator light. The second interval lasts from 5 seconds to 10 seconds, and the buzzer will sound [21].

If the second interval time is longer than the first interval time, the system will release the load. If the imbalance is more than 25%, the relay will convey load information on the phase with the largest load current and move the phase with the smallest load current. In addition, the system has an overcurrent protection system [22], [23]. If the current on one of the phases exceeds the tolerance limit, Arduino will order the SSR to trip. Using Internet of Things technology, this system tracks the R, S, and T phase current and voltage values as well as the neutral current and power loss values [24], [25]. It is expected that this load current balancing prototype can reduce the neutral phase current and power loss.

2. Materials and Methods

Figure 2.1 explains the workflow concept of the entire system implemented in the form of a flowchart. Starting from the initialization of the current and voltage sensors, the data can then be processed by the Arduino Mega 2560 microcontroller. Data from the Arduino Mega 2560 microcontroller will enter the data grouping stage using the Naïve Bayes method. The grouped data will be displayed on the 20x4 LCD and the Blynk application.

In this system there are two condition scenarios, if the two scenarios are not running then the system is indicated to be normal and will be displayed in the monitoring data. The first condition is a condition where there is an overload current, if the first condition occurs then the overcurrent

protection system will work. The second condition is a condition where the imbalance of the R, S, and T phase currents is more than 25%. If the second condition occurs, the tapping phase system and the unbalanced time classification system will work.

The following is the hardware design that will be used in this Final Project, the design can be seen in Figure 2 is the overall design of the prototype. Table 1 show the remarks on hardware design based on Figure 2. The design of the wiring diagram is used to make it easier when assembling the required electrical components. Electrical equipment or components that have been prepared are arranged in such a way as to form a system that has a certain work function. All components are assembled or installed according to the working principle of the tool used. The design of the tool to be made refers to the block diagram shown in Figure 3.

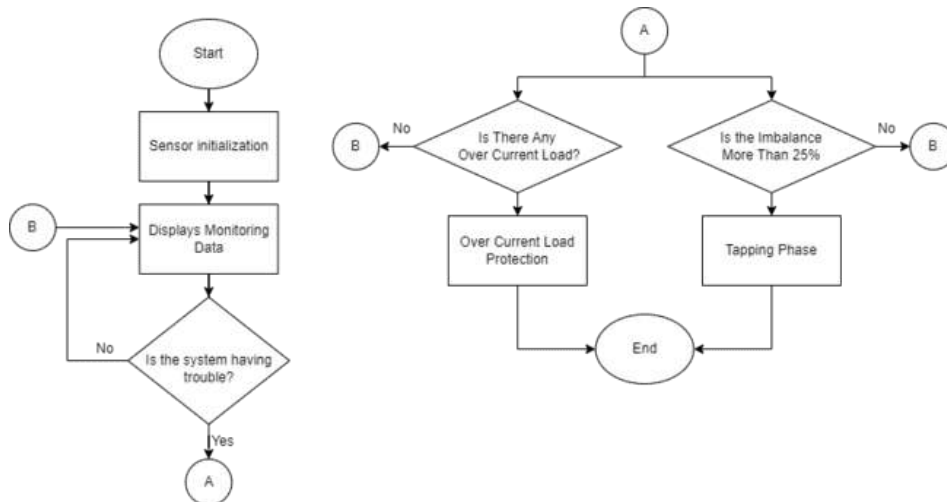


Figure 1. Entire Flowchart of Load Dividing Automatic System

Table 1. Remarks on Hardware Design

No	Component	No	Component
1	Plywood board	8	Buzzer
2	Panel box	9	LCD 16 x 2
3	Bulb	10	Power supply unit 5VDC
4	Switch	11	Relay 5VDC
5	Fitting lamp	12	ESP32
6	Indicator lamp of R, S, and T phase	13	Arduino Mega 2560
7	Plywood board	14	PZEM 004T

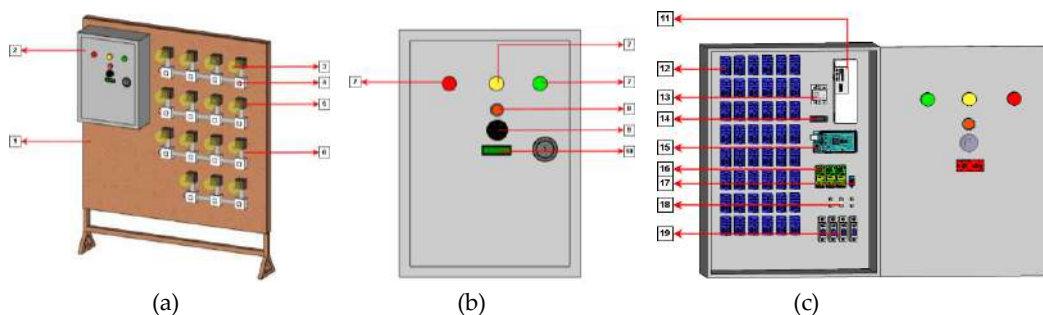


Figure 2. (a) Entire Hardware Design (b) Outside part of box panel (c) Inside part of box panel

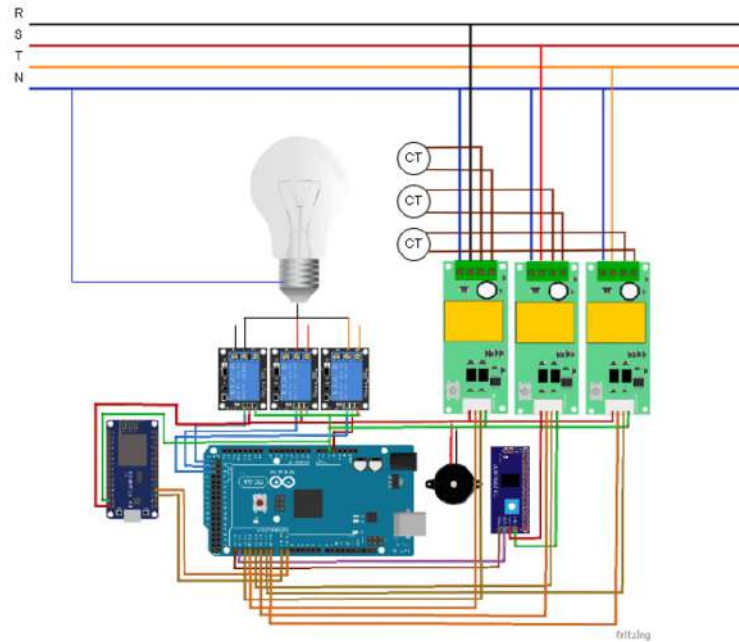


Figure 3. Entire Wiring of Arduino

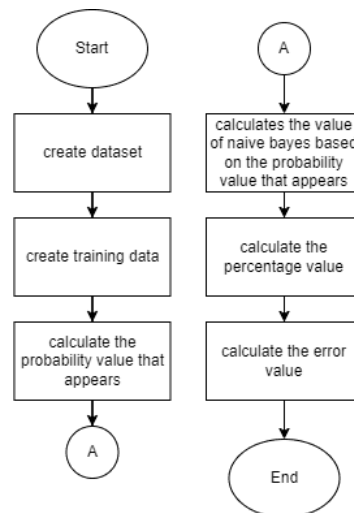


Figure 4 Method Flowchart

To carry out the functions in the design of this tool, in this final project there are several programs that need to be made, namely:

1. Programming PZEM-004T sensor readings on Arduino Mega.
2. Automatic relay performance programming and control on Arduino Mega.
3. Auto buzzer performance programming and control on Arduino Mega.
4. Auto indicator light performance programmer and control on Arduino Mega.
5. Programming to display PZEM-004T sensor readings from Arduino Mega to LCD.
6. Programming to display PZEM-004T sensor readings from Arduino Mega to the Blynk application on a smartphone.

This final project uses the naïve Bayes algorithm to help classify transformer conditions. In this final project, Naïve Bayes testing will be tested using manual calculations, using machine learning applications, and will be tested factually.

Based on Figure 4, the first thing to do is create a dataset. The dataset is divided into 2 variables, namely input variables and class variables. Then proceed with creating training data and calculating the probability value of each variable. After calculating the probability value, proceed

with calculating the bayes value. After knowing the probability value that has been calculated, it is continued by calculating the percentage of the predicted value. After getting the percentage of the predicted value then proceed by comparing the predicted value with the training data that has been made so that the error value is obtained

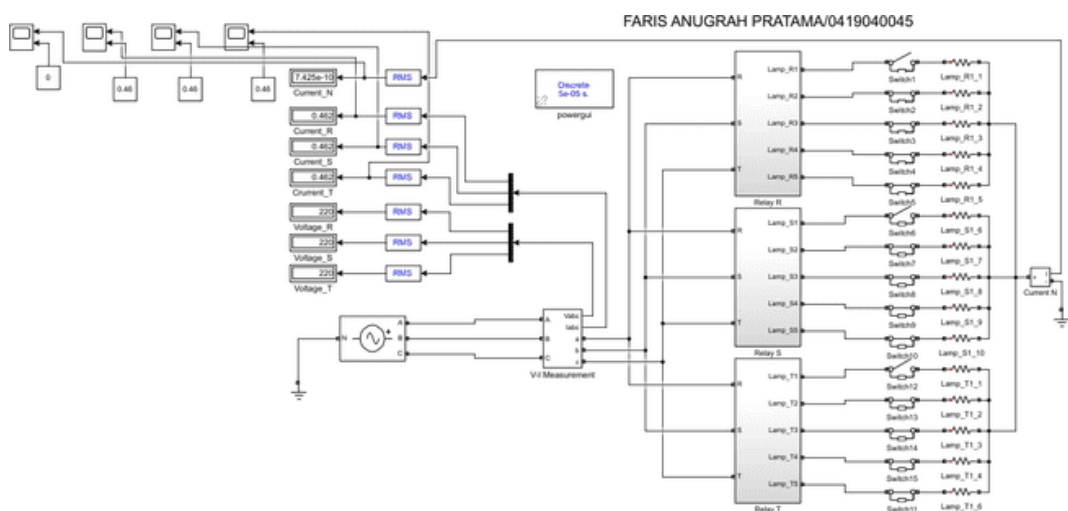
3. Results

3.1. Simulation

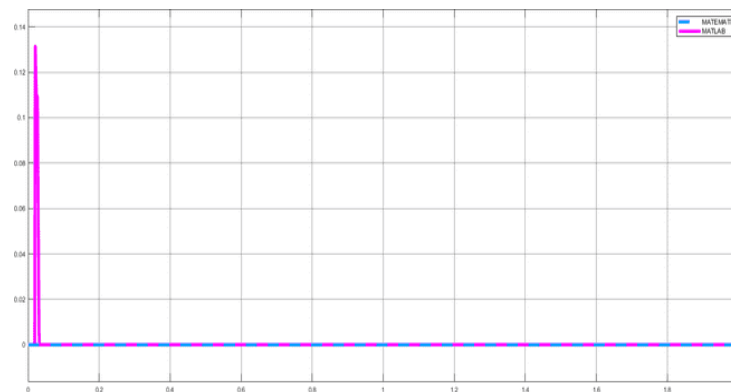
Integrated simulation testing consists of unbalanced load testing and neutral current testing. This circuit has 15 loads with a power of 100 watts where each load is connected to a switch and from each switch it is connected to 3 relays, namely relays R, S, and T. The workings of this system are to turn on the switches on each phase according to with the scenarios that have been created. If the current is read in a balanced state, the relay will not make a switch and if there is an imbalance in the current due to a difference in the amount of power on each phase, the relay will move from the phase with the large current to the phase with the small current.

For phase R, the switch on the load will be connected to 3 relays, namely relay R which is connected to NC to the load and 2 other relays namely relay S and T connected to NO. Vice versa, for loads on phase S, the relay S is connected to NC to the load and the other 2 relays namely relays R and T are connected to NO to the load.

3.1.1 Testing 1 Condition 1



(a)



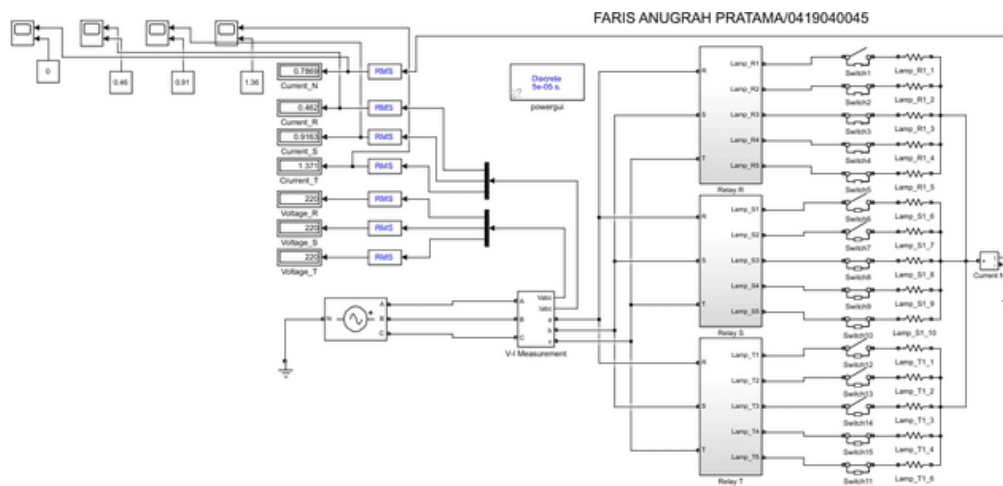
(b)

Figure 5. (a) Overall System Series (b) Circuit on R Phase Load

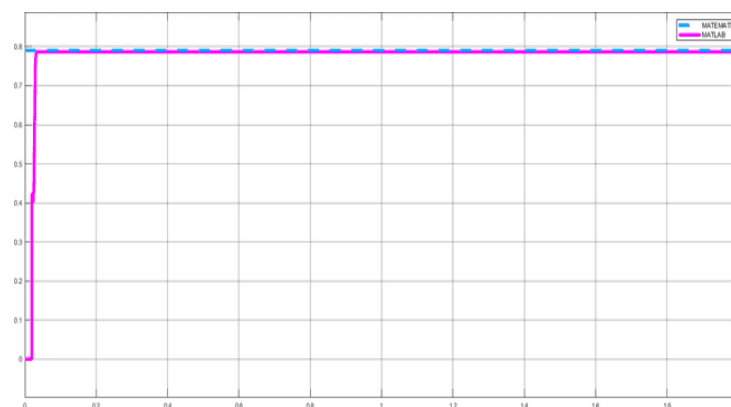
The first test is to simulate a balanced load using 3 loads of 100 watts each connected to the R, S, and T phases. The values from the simulation results are 220V for the voltages on the R, S, and T phases. 0.462A for the currents on the R, S, and T phases. R, S, and T. The value of the neutral current is very small due to balanced loads. Then it can be seen in the signal image that there is a spike resulting from switching on from the relay

3.1.2 Testing 1 Condition 2

The second test is to simulate an unbalanced load using 1 load connected to phase R, 2 loads connected to phase S, and 3 loads connected to phase T. The initial value of the simulation results is 220V for the voltage on phases R, S, and T. 0.462A for the current in phase R, 0.9163A for the current in phase S, and 1.371A for the current in phase T. The value of the neutral current is 0.7869A. In step 1, relay switching is carried out so that the load becomes balanced. The value after balancing is 220V for the voltage on phase R, S, and T. 0.9169A for the current on phase R, 0.9163A for the current on phase S, and 0.9158A for the current on phase T. The value of the neutral current is 0.0003809A. Then it can be seen in the signal image that there are signal spikes in the R, S, and T phases resulting from the switching on of the relay in step 0 and small spikes in the R and T phase signals as a result of switching the relay in step 1.

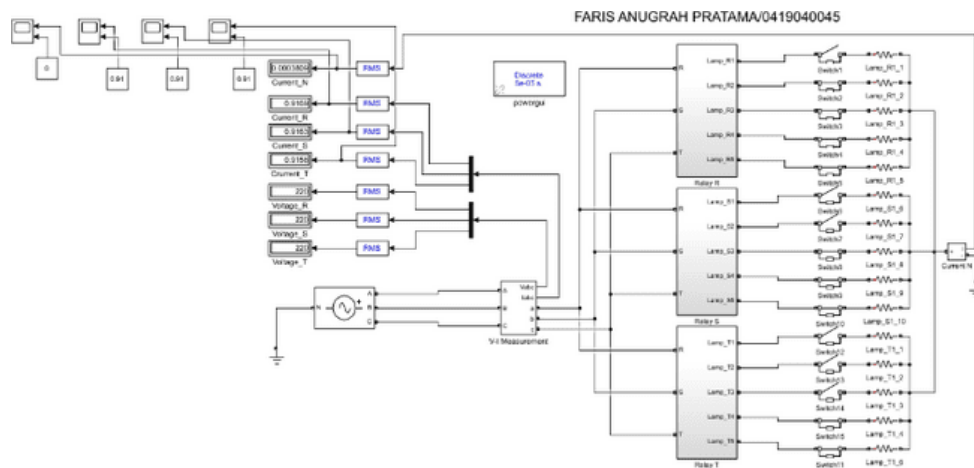


(a)

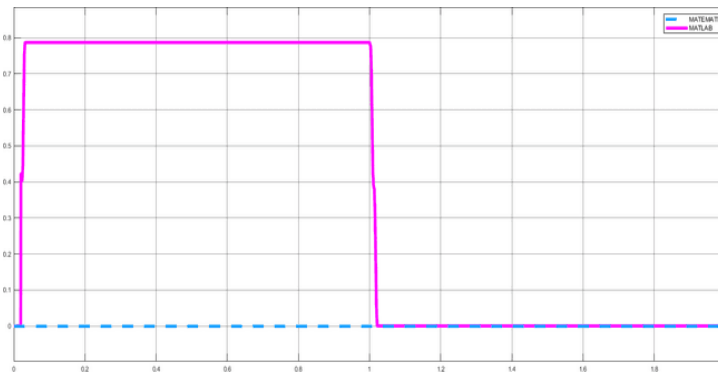


(b)

Figure 6. (a) Tapping Phase Simulation Series Test 1 Condition 1 (b) Signal on Neutral Phase



(a)

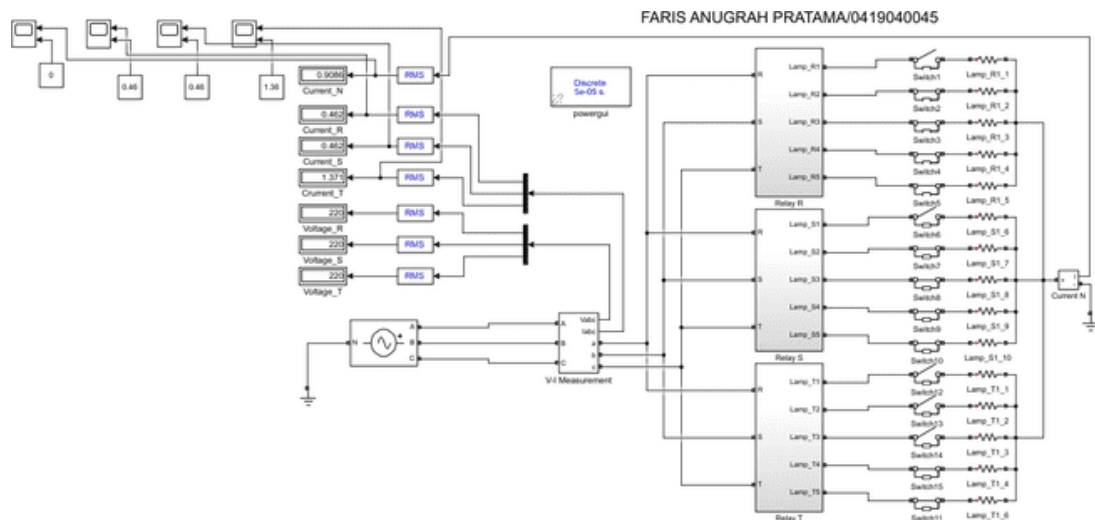


(b)

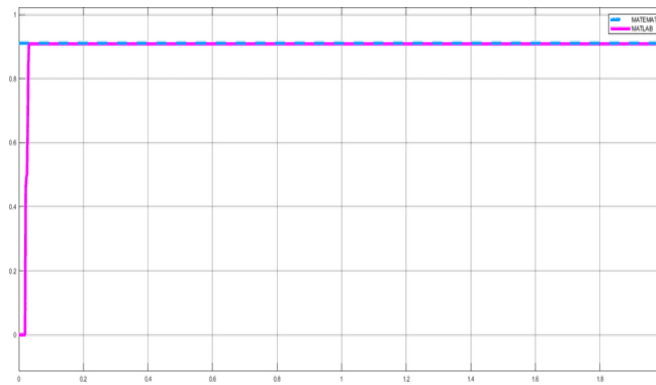
Figure 7. (a) Tapping Phase Simulation Series Test 1 Condition 2, before switching (b) Signal on Neutral Phase, before switching

3.1.3 Testing 1 Condition 3

The third test is to simulate an unbalanced load using 1 load connected to phase R, 1 load connected to phase S, and 3 loads connected to phase T. The initial value of the simulation results is 220V for the voltage on phases R, S, and T. 0.462A for current on phase R, 0.462A for current on phase S, and 1.371A for current on phase T. The value of the neutral current is 0.9086A. In step 1, relay switching is carried out with the aim of balancing the load. The value after balancing is 220V for the voltages on the R, S, and phases T. 0.9169A for the current in phase R, 0.462A for the current in phase S, and 0.9158A for the current in phase T. The value of the neutral current is 0.4543A. It can be seen even though switching has occurred in the relay, the current in the neutral phase remains high.



(a)



(b)

Figure 8. (a) Tapping Phase Simulation Series Test 1 Condition 3, before switching (b) Signal on Neutral Phase, before switching

3.2. Result of Calculation

3.2.1 Calculation Testing 1 Condition 1

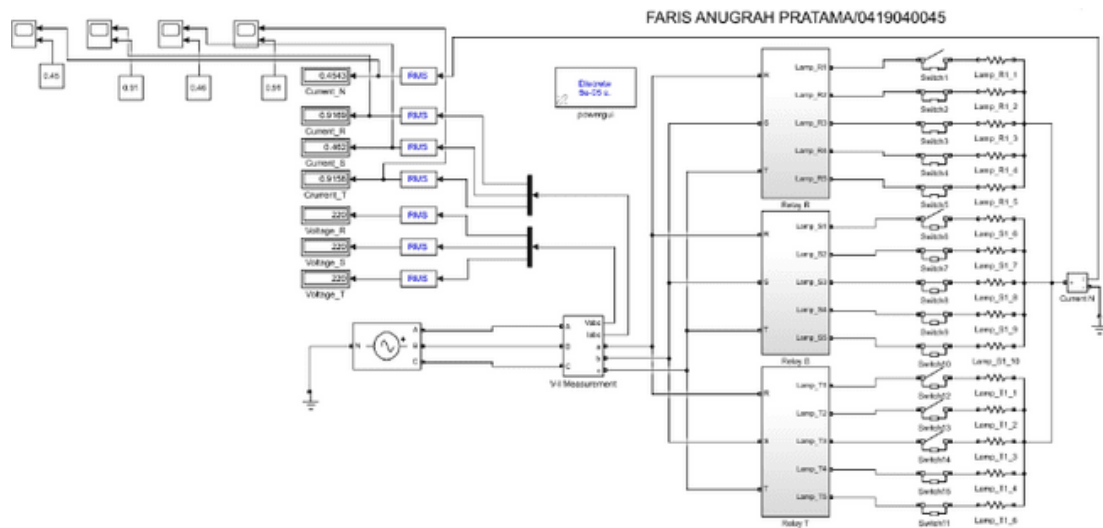
Neutral current calculation

$$\text{Total Sin} = 0,45 \times \sin (0) + 0,45 \times \sin (240) + 0,45 \sin (120)$$

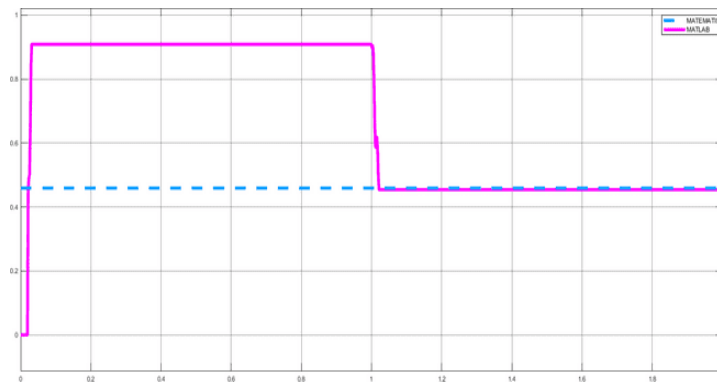
$$\text{Total Cos} = 0,45 \times \cos (0) + 0,45 \times \cos (240) + 0,45 \cos (120)$$

$$IN = (02 + 02)0,5$$

$$IN = 0 \text{ A}$$



(a)



(b)

Figure 9. (a) Tapping Phase Simulation Series Test 1 Condition 3, after switching (b) Signal on Neutral Phase, after switching

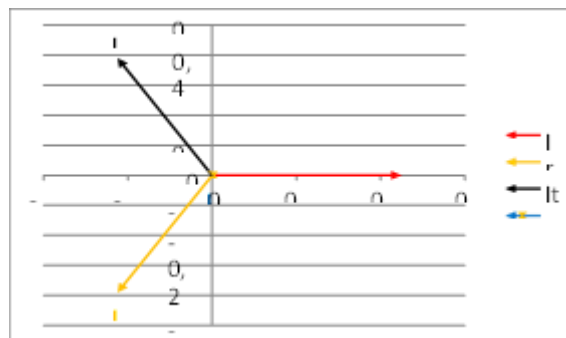


Figure 10 Interphase Current Graph

Based on the graph in Figure 9, there is no difference in the value of each phase and when added together it is zero. This situation does not generate a neutral current.

3.2.2 Calculation Testing 1 Condition 2

Neutral Current Calculation

$$\text{Total Sin} = 0,45\text{Sin} (0) + 0,91\text{Sin} (240) + 1,36\text{Sin} (120)$$

$$\text{Total } \cos = 0,45\cos (0) + 0,91\cos (240) + 1,36\cos (120)$$

$$IN = (-0,685)2 + (0,39)2)0,5$$

$$IN = 0,79 \text{ A}$$

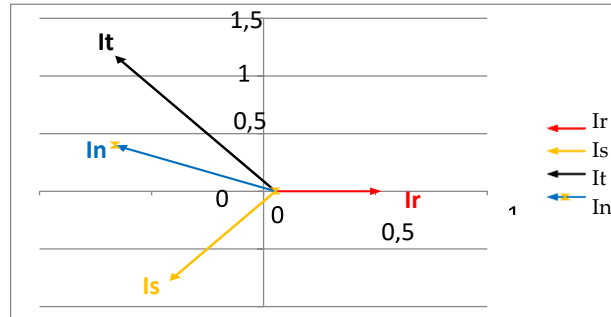


Figure 11 Interphase Current Graph

There is a difference in the value of the current in each phase which in theory will give rise to a neutral current. Based on calculations, the value of the neutral current is 0.79A. Head and shoulders shots of authors that appear at the end of our papers.

3.2.3 Calculation Testing 1 Condition 3

Neutral current calculation

$$\text{Total } \sin = 0,45\sin (0) + 0,45\sin (240) + 1,36\sin (120)$$

$$\text{Total } \cos = 0,45\cos (0) + 0,45\cos (240) + 1,36\cos (120)$$

$$IN = (-0,455)2 + (0,79)2)0,5$$

$$IN = 0,91 \text{ A}$$

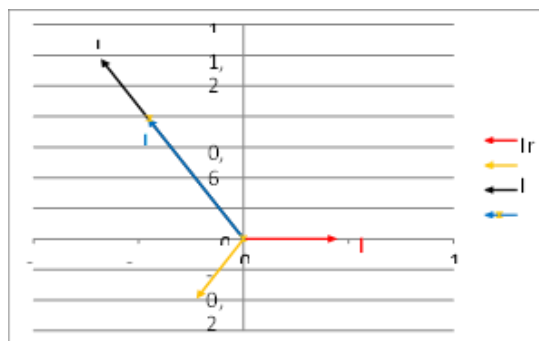


Figure 12 Interphase Current Graph

There is a difference in the value of the current in each phase which in theory will give rise to a neutral current. Based on calculations, the value of the neutral current is 0.91A.

4. Discussion

4.1. Tapping Phase Testing

Integrated testing is testing the entire system which consists of a combination of components that are bound to each other to form a single unit. In this final project, what is meant by integrated testing is a combination of testing between tapping phases. After testing the supporting components, the next step is to test the function of the system as a whole. The actual integrated test without a method is a test that is carried out regardless of the simulation results from the method that has been simulated. This test aims to determine the overall function of the system can work properly by analyzing the load imbalance protection system.

Based on table 2 there are 4 data showing the percentage of imbalance exceeding 25%. In a row data numbers 4, 5, 7, 8, and 9 have an Unbalanced Load Percentage (UL) of 33.30%, 32.60%, 30.20%, 26.50% and 32.80%. Test data that has an imbalance percentage above 25% will automatically experience a tapping phase.



Figure 13. Integrated Tool Testing

Table 2 Tapping Phase Testing Table

No	Testing						
	Ir (A)	Is (A)	It (A)	UL (%)	In (A)	Losses (W)	Condition
1	1,67	0,85	1,26	21,90	0,80	0,08	Balanced
2	1,64	1,66	1,65	0,50	0,00	0,00	Balanced
3	1,24	2,07	1,65	16,80	0,80	0,08	Balanced
4	0,84	2,10	1,26	33,30	1,20	0,17	Unbalanced
5	0,86	0,44	0,44	32,60	0,40	0,02	Unbalanced
6	1,71	1,29	1,28	13,10	0,45	0,02	Balanced
7	0,86	1,72	2,14	30,20	1,20	0,17	Unbalanced
8	0,86	0,81	1,71	26,50	0,90	0,10	Unbalanced
9	0,44	0,87	1,29	32,80	0,79	0,07	Unbalanced
10	0,44	0,44	0,44	0,00	0,00	0,00	Balanced



Figure 14. Result of Method Simulation

Table 3. Result of Method Simulation Table

No	Classifier Output		
1	Correctly Classified Instances	17	94.4444%
2	Incorrectly Classified Instances	1	5.5556%
3	Kappa Statistic	0.9159	
4	Mean Absolute Error	0.124	
5	Root Mean Squared Error	0.1728	
6	Relative Absolute Error	36.4054%	
7	Root Relative Squared Error	42.0418%	
8	Total Number of Instances	18	

4.2 Naïve Bayes Method Testing

As can be seen in Figure 4.50, it is known that the probability values for each class (pointed by arrows) successively from normal to trip are 0.41, 0.23, 0.27, 0.09. After looking at the probability value of each class, then in Figure 4.51 it is known that the accuracy of the predicted value of the reference data (assumptions). Overall accuracy is shown in the Correctly Classified Instances chart, the chart shows data that matches the reference values that have been made as many as 17 out of 18 data or 94.44%. Incorrectly Classified Instances (data that do not match) in this chart shows data that does not match the reference data as much as 1 out of 18 data or 5.56%. When the load is balanced ideally the neutral current will not appear. In calculations and simulations when the load is balanced, the neutral current is 0.00A. It can be seen that the average error value in the calculation of the simulation is the smallest, namely 1%, this is because each data is almost not influenced by other factors such as hardware and other factors, or in other words, both data are in ideal conditions

5. Conclusions

Based on the experiments that have been carried out in balanced conditions, the unbalance percentage value is 0.50% and the neutral current read is 0 A, when the unbalanced condition is unbalanced, the percentage value is 33.30% and the neutral current read is 1.20A. After tapping the phase, the percentage of imbalance becomes 13.40% and the neutral current that is read is 0.50A. The neutral current value arises due to an imbalance in the load. Based on testing, when the percentage unbalance value is 33.30%, a neutral current of 1.20A arises and when the percentage unbalance value is 0.00% (balanced), the neutral current value that is read is 0%.

References

1. Adnan, Muhammad Hasil. "Gangguan yang terjadi pada saluran kabel tegangan menengah 20 KV pada sistem Jointing di PT. PLN (Persero) Area Bandengan." (2017).
2. Aghenta, Lawrence Oriaghe, and Mohammad Tariq Iqbal. "Low-cost, open source IoT-based SCADA system design using thinger. IO and ESP32 thing." *Electronics* 8.8 (2019): 822.
3. Widianoro, Anang, and Achmad Nur Hidayat. "Rancang Bangun Sistem Proteksi Arus Listrik Dan Tegangan Pada Kabel Line Listrik Berbasis Mikrokontroler Arduino Mega." *Jurnal Teknik Elektro dan Komputer TRIAC* 5.1 (2018): 22-29.
4. Asif, Rao M., et al. "Smart underground wireless cable fault detection and monitoring system." 2020 International Conference on Engineering and Emerging Technologies (ICEET). IEEE, 2020.
5. Mahdianto Arsyad, Billy, Aghus Sofwan, and Agung Nugroho. "Perancangan Sistem Kontrol Over/Under Voltage Relay Berbasis Mikrokontroler Pada Saluran Tegangan 220VAC."
6. Amir, D., and Indrawati Indrawati. "Rancang Bangun Radar Pendeteksi Saluran Kabel Listrik di Bawah Tanah." *Jurnal Litek: Jurnal Listrik Telekomunikasi Elektronika* 13.1 (2016): 1-7.
7. Darmawansyah, Darmawansyah, M. Khairul Amri Rosa, and Ika Novia Anggraini. "Sistem Proteksi Motor Induksi 3 Fasa Terhadap Berbagai Gangguan Menggunakan Mikrokontroler." *Jurnal Amplifier: Jurnal Ilmiah Bidang Teknik Elektro dan Komputer*. 10.1 (2020): 9-17.
8. Sofyan, M., Syaifudin, S., Pudji, A., Nugraha, A. T., & Utomo, B. (2023). Comparative Analysis of Water and Oil Media on Temperature Stability in PID Control-Based Digital Thermometer Calibrator. *Indonesian Journal of Electronics, Electromedical Engineering, and Medical Informatics*, 5(2), 73–78. <https://doi.org/10.35882/ijeeemi.v5i2.274>
9. Sugianto, Moh. G. P. A., & Nugraha, A. T. (2022). Implementasi sensor cahaya sebagai level bahan bakar pada tangki harian kapal. *Journal of Computer Electronic and Telecommunication*, 3(1). <https://doi.org/10.52435/complete.v2i1.191>
10. Supiyadi, R. A., Asri, P., & Nugraha, A. T. (2023). Rancang Bangun Prototipe Sistem Kontrol UV Conveyor dan Monitoring Kadar Air Cacahan Plastik Berbasis Mikrokontroler. *Elektriase: Jurnal Sains dan Teknologi Elektro*, 13(01), 34–41. <https://doi.org/10.47709/elektriase.v13i01.2349>
11. Utomo, A. P., Apriani, M., Suhardjito, G., Arta, I. P., Nugraha, A. T., Cahyono, L., Iswantoro, D., Afri, G., Arifina, N. F. P., Syarif, R., Risky, M., & Febianto, M. A. (2023). Pelatihan Pembuatan Miniatur Kapal Berdasar Standar Desain Berbahan Fiberglass Reinforced Plastic (FRP) untuk Kelompok Pengrajin di Wilayah Pantai Situbondo. 7(2).
12. Agna, D. I. Y. ., Yuniza, S. I. ., & Nugraha, A. T. . (2022). The Single-Phase Controlled Half Wave Rectifier with Single-Phase Generator Circuit Model to Establish Stable DC Voltage Converter Result. *International Journal of Advanced Electrical and Computer Engineering*, 3(3). Retrieved from <http://www.amcspress.com/index.php/ijaece/article/view/1119>
13. Shiddiq, M. J. ., Yuniza, S. I. ., & Nugraha, A. T. . (2022). The Design of Uncontrolled Rectifier Three Phase Half Wave with Single Phase AC Generator Source. *International Journal of Advanced Electrical and Computer Engineering*, 3(2). Retrieved from <http://amcspress.com/index.php/ijaece/article/view/1114>
14. Yuniza, S. I. ., Agna, D. I. Y. ., & Nugraha, A. T. . (2022). The Design of Effective Single-Phase Bridge Full Control Resistive Load Rectifying Circuit Based on MATLAB and PSIM. *International Journal of Advanced Electrical and Computer Engineering*, 3(3). Retrieved from <http://amcspress.com/index.php/ijaece/article/view/1120>
15. As'ad, R. F. ., Yuniza, S. I. ., & Nugraha, A. T. . (2022). The Effect of 3 Phase Full Wave Uncontrolled Rectifier on 3 Phase AC Motor. *International Journal of Advanced Electrical and Computer Engineering*, 3(2). Retrieved from <http://amcspress.com/index.php/ijaece/article/view/1117>
16. Achmad, I. & Anggara Trisna Nugraha. (2022). Implementation of Voltage Stabilizers on Solar Cell System Using Buck-Boost Converter. *Journal of Electronics, Electromedical Engineering, and Medical Informatics*, 4(3), 154–160. <https://doi.org/10.35882/ijeeemi.v4i3.246>
17. Anggara Trisna Nugraha, Purwidi Asri, Perwi Darmajanti, Diego Ilham Yoga Agna, & Muhammad 'Ubaid Amrullah. (2023). Rancang Bangun Monitoring Kualitas Air Tambak Udang Vaname Dengan Kontrol Paddle Wheel Berbasis Mikrokontroler. *Jurnal 7 Samudra*, 8(2). <https://doi.org/10.54992/7samudra.v8i2.132>

18. Apriani, M., Nindyapuspa, A., Cahyani, F. D. A. F., & Nugraha, A. T. (2023). Recovery of sugarcane bagasse as adsorbent for chromium (Cr) (III) removal. IOP Conference Series: Earth and Environmental Science, 1265(1), 012006. <https://doi.org/10.1088/1755-1315/1265/1/012006>
19. As'ad, R. F., & Nugraha, A. T. (2022). Rancang Bangun Penstabil Kinerja Panel Hubung Bagi Tegangan Rendah. Journal of Computer Electronic and Telecommunication, 3(1). <https://doi.org/10.52435/complete.v2i1.187>
20. Asri, P., Widodo, H. A., Nugraha, A. T., Rachman, I., & Poertro, J. E. (2022). Desain Hybrid Panel Surya dan Generator Set pada Kapal Ikan Pesisir Selatan Jawa. 12(1).
21. Cahyono, L., Apriani, M., Nugraha, A. T., & Prasetyo Utomo, A. (2022). Potensi Risiko Waktu Pelaksanaan Proyek Swakelola di Kabupaten Pasuruan melalui Program Kotaku Kementerian PUPR. Jurnal Spektran, 10(2), 118. <https://doi.org/10.24843/SPEKTRAN.2022.v10.i02.p08>
22. Cahyono, L., Apriani, M., Utomo, A. P., Nugraha, A. T., Setiawan, A., Fatoni, A., Qurani, V. F., Firtsanti, A. A., Prasetyo, R. M., & Wulandari, I. (2022). Pelatihan Pembuatan Lilin Aromaterapi dari Minyak Jelantah Sebagai Sarana Peduli Lingkungan Perairan dan Implementasi Konsep Ekonomi Sirkular Warga Bumi Suko Indah. Dharma Raflesia : Jurnal Ilmiah Pengembangan dan Penerapan IPTEKS, 20(1), 53–67. <https://doi.org/10.33369/dr.v20i1.19271>
23. Department of Marine Electrical Engineering, Faculty of Shipbuilding Electrical Engineering, Shipbuilding Institute of Polytechnic Surabaya, Nugraha, A. T., Rinaldi, D., Department of Marine Electrical Engineering, Faculty of Shipbuilding Electrical Engineering, Shipbuilding Institute of Polytechnic Surabaya, Messiah, M. S., Department of Automation Engineering, Faculty of Shipbuilding Electrical Engineering, Shipbuilding Institute of Polytechnic Surabaya, Shiddiq, M., Department of Automation Engineering, Faculty of Shipbuilding Electrical Engineering, Shipbuilding Institute of Polytechnic Surabaya, Ramadhan, M., Department of Automation Engineering, Faculty of Shipbuilding Electrical Engineering, Shipbuilding Institute of Polytechnic Surabaya, Ainudin, F., & Department of Marine Electrical Engineering, Faculty of Shipbuilding Electrical Engineering, Shipbuilding Institute of Polytechnic Surabaya. (2022). Implementation of Line of Sight Algorithm Design Using Quadcopter on Square Tracking. JPSE (Journal of Physical Science and Engineering), 7(2), 99–107. <https://doi.org/10.17977/um024v7i22022p099>
24. Fahmi Ivannuri, & Nugraha, A. T. (2022). Implementation Of Fuzzy Logic On Turbine Ventilators As Renewable Energy. Journal of Electronics, Electromedical Engineering, and Medical Informatics, 4(3), 178–182. <https://doi.org/10.35882/jeeemi.v4i3.236>
25. Intyanto, G. W., Trihatmojo, A. A., Yuniarti, D. A. F., & Nugraha, A. T. (2023). ELDOC - Design of Electric Dolly Camera for Video Recording Using the Omni-Direction Wheel. Walisongo Journal of Information Technology, 5(1), 41–52. <https://doi.org/10.21580/wjit.2023.5.1.16463>



© 2019 by the authors. Submitted for possible open access publication under the terms and conditions of the Creative Commons Attribution (CC BY) license (<http://creativecommons.org/licenses/by/4.0/>).

Pengembangan *Cyber Physical System* untuk Presensi dengan Deteksi Multi Wajah

Sopian Alviana^{1*}, Bobi Kurniawan²

¹ Program Studi Teknik Informatika, Universitas Komputer Indonesia, Bandung, Indonesia

² Program Studi Teknik Elektro, Universitas Komputer Indonesia, Bandung, Indonesia

* Korespondensi: sopian.alviana@email.unikom.ac.id

Abstrak: Sistem presensi menggunakan pengenalan wajah memiliki keuntungan efisiensi waktu dan tenaga. Saat ini presensi dengan menggunakan wajah sudah banyak diterapkan, tetapi masih terbatas pada penggunaan di suatu lokasi tertentu. Keterbatasan mesin presensi saat ini dalam mendeteksi dan melakukan verifikasi hanya dapat dilakukan satu wajah dalam satu waktu. Maka, dibutuhkan pengembangan dengan peningkatan komputasi menggunakan *cyber physical system* untuk proses presensi. Pemanfaatan *cyber physical system* dilakukan dengan memanfaatkan akses *close circuit television* (CCTV) yang digunakan untuk membantu proses presensi. Hal ini memberikan keuntungan agar dapat melakukan verifikasi presensi dengan banyak wajah dalam satu waktu, sehingga lebih efisien terhadap proses presensi.

Kata kunci: CCTV, *Cyber physical system*, multi wajah, presensi

1. Pendahuluan

Sistem presensi dapat dikategorikan menjadi beberapa jenis, yaitu sistem dengan menggunakan aplikasi dan penggunaan biometrik. Penggunaan biometrik untuk sistem presensi dilakukan dengan pengenalan bagian tubuh seperti sidik jari dan pengenalan wajah [1]. Proses identifikasi wajah dilakukan dengan bantuan kamera, hal ini memiliki keuntungan efisiensi waktu dan tenaga, serta mudah dalam proses pencatatan [2]. *Cyber Physical System* (CPS) merupakan kerangka dasar dalam pengembangan sistem cerdas. Skala CPS saat ini berkembang menjadi terbuka, besar, dan bersifat otonom [3]. CPS merupakan sistem berbasis kemampuan industri 4.0 yang mampu menawarkan pengumpulan data, analisis dan transparansi secara *real time*. Penerapan CPS [4], *artificial intelligence* [5], atau penggunaan *digital twin* [6] memunculkan penerapan teknologi revolusi industri 4.0.

Close circuit television (CCTV) merupakan perangkat yang digunakan pada bidang keamanan untuk melakukan pemantauan keadaan sebuah lokasi [7]. CCTV tidak hanya digunakan untuk keamanan, tetapi dapat digunakan untuk proses pengenalan wajah menggunakan metode *image recognition* dari data video yang diperoleh [8]. Sehingga CCTV dapat digunakan untuk media presensi pada setiap sudut fasilitas publik [9].

Sistem presensi saat ini menggunakan sidik jari dan wajah memiliki keterbatasan diantaranya proses presensi hanya dapat dilakukan pada lokasi tertentu, kesulitan proses deteksi dan verifikasi satu wajah dalam satu waktu. Selain itu, data yang dihasilkan masih berupa data statis. Maka, berdasarkan permasalahan tersebut untuk memudahkan proses deteksi dan presensi diusulkan untuk menambahkan proses identifikasi sistem presensi dengan menggunakan CCTV dengan meningkatkan komputasi menggunakan pendekatan *cyber physical system*.

2. Metode dan Pengembangan Sistem

Tahapan penelitian yang dilakukan meliputi beberapa tahapan diantaranya identifikasi masalah, tujuan penelitian, analisis kebutuhan sistem, desain sistem, implementasi sistem, hasil penelitian, serta kesimpulan. Gambaran tahapan penelitian terdapat pada Gambar 1.

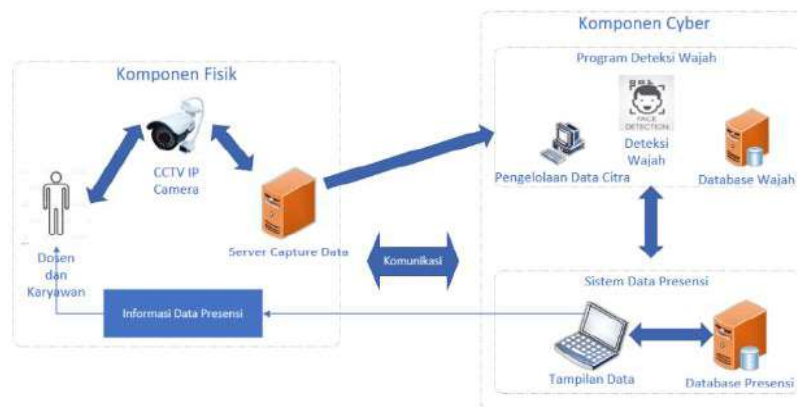


Gambar 1. Tahapan penelitian

Gambar 1 merupakan tahapan penelitian yang akan ditempuh yaitu identifikasi masalah yang meliputi tahap awal penelitian melakukan observasi dan wawancara. Tujuan penelitian ditentukan untuk dapat menyelesaikan permasalahan yang telah diidentifikasi. Analisis kebutuhan sistem merupakan tahapan untuk menentukan kebutuhan sistem yang akan dibangun baik secara perangkat keras dan lunak untuk sistem presensi yang akan digunakan. Desain sistem merupakan tahapan untuk merancang desain sistem yang akan dibangun. Implementasi dilakukan untuk menguji sistem yang telah dibangun. Hasil penelitian merupakan tahapan akhir untuk melihat kesesuaian sistem yang diusulkan. Kesimpulan merupakan tahap akhir untuk tujuan yang telah dicapai.

Dalam pelaksanaan penelitian terdapat metode dalam pengembangan sistem yang diusulkan. Pengembangan yang diusulkan mengacu kepada komponen yang terdapat pada *cyber physical system* (CPS) yaitu komponen komputasi, komunikasi, dan kontrol. Dalam pemetaan komponen tersebut dibagi menjadi tiga yaitu komponen fisik, *cyber*, dan komunikasi. Komponen CPS pembentuk sistem presensi terdapat pada Gambar 2. CPS yang digunakan pada penelitian ini merupakan skema CPS dengan semi *autonomous system*. Proses otonom dilakukan dalam mendeteksi wajah pada saat proses presensi.

Pengembangan sistem dengan menggunakan CPS terdapat tiga bagian utama yaitu komponen fisik, *cyber*, dan komunikasi seperti ditunjukkan pada Gambar 2. Komponen fisik merupakan komponen untuk proses akuisisi data yaitu proses untuk mengumpulkan data presensi. Komponen terdiri dari sensor yang akan digunakan yaitu kamera dan server untuk mengumpulkan data dari kamera. Komponen *cyber* merupakan komponen untuk melakukan proses komputasi yaitu proses deteksi wajah dan proses pengiriman data ke sistem presensi. Komponen komunikasi merupakan komponen proses komunikasi dalam pengiriman data. Dalam sistem ini diusulkan skema dengan menggunakan *Wireless Sensor Network* (WSN) untuk pertukaran data.



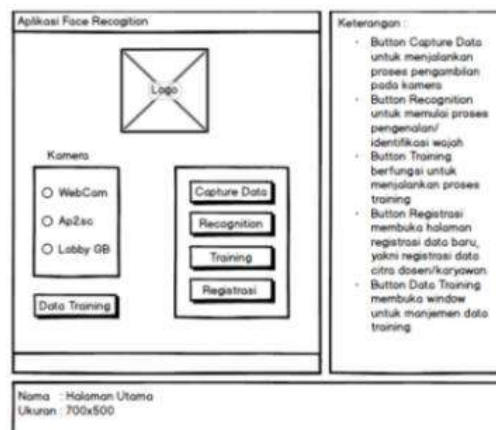
Gambar 2. Pengembangan sistem

Pada bagian komponen *cyber* untuk dapat mendeteksi wajah kami menggunakan pendekatan algoritma *Histogram of Oriented Gradients* (HOG). HOG merupakan algoritma yang digunakan untuk mendeteksi objek dan wajah dan melakukan klasifikasi dengan memanfaatkan bidang – bidang pemrosesan gambar. Tahapan HOG terdiri dari preprocessing, menghitung gradien vector dan melakukan *spatial binning*. Metode preprocessing merupakan tahapan dalam melakukan penyesuaian ukuran yang dikonversi kedalam model *grayscale*. Tahapan kedua yaitu menghitung nilai setiap piksel dan magnitude dari vector tersebut. Tahapan akhir merupakan nilai gradien untuk mencari nilai histogram.

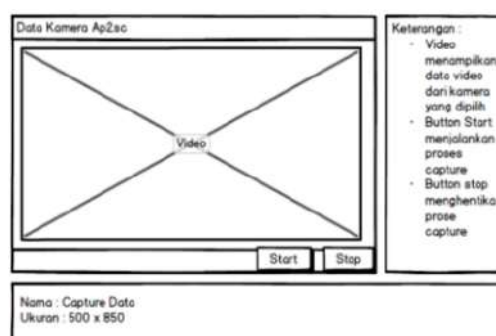
3. Hasil

3.1. Perancangan Antarmuka

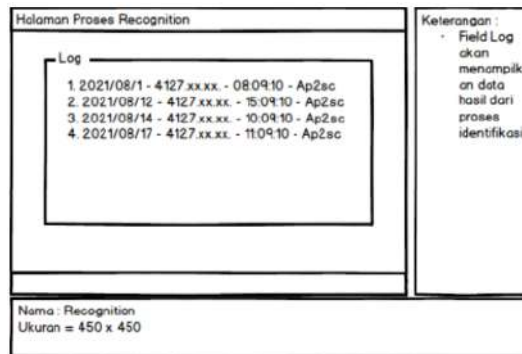
Perancangan antarmuka bertujuan untuk memberikan gambaran tentang aplikasi yang akan dibangun. Perancangan ini diimplementasikan menjadi sebuah program utuh yang dapat digunakan oleh pengguna sistem. Rancangan antarmuka pembangunan aplikasi presensi dengan wajah adalah sebagai berikut :



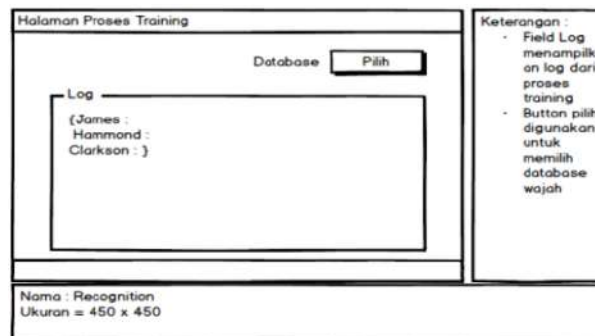
Gambar 3. Rancangan antarmuka



Gambar 4. Capture data



Gambar 5. Halaman proses rekognisi

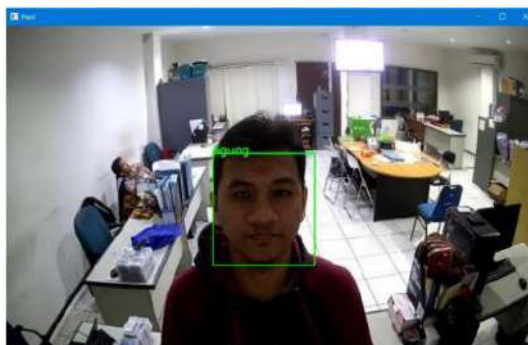


Gambar 6. Halaman data latih

Gambar 3 merupakan tampilan antarmuka utama yang diusulkan dalam proses presensi dengan multi wajah. Gambar 4 merupakan halaman untuk melakukan penangkapan data wajah. Gambar 5 merupakan halaman proses rekognisi yang berfungsi untuk menampilkan log data hasil deteksi selama proses berjalan sedangkan Gambar 6 merupakan halaman untuk melakukan pelatihan data wajah yang digunakan untuk proses presensi.

3.2. Implementasi Presensi

Pada proses implementasi presensi sistem yang sudah dibangun, dilakukan beberapa tahapan pengujian untuk memastikan sistem berjalan dengan baik. Pengujian yang dilakukan diantaranya menguji deteksi wajah dengan berbagai jarak dan jumlah wajah yang dapat dilakukan deteksi. Gambar 7 hingga 9 menunjukkan hasil dari pengujian deteksi wajah dengan menggunakan beberapa jarak yaitu 50, 150, dan 200 cm. Hasil lengkap pengujian ditunjukkan pada Tabel 1. Hasil pengujian deteksi wajah terhadap jarak pada Tabel 1 menunjukkan bahwa sistem dapat mendeteksi wajah dengan maksimal jarak 200 cm dengan akurasi 98.8%.



Gambar 7. Deteksi wajah jarak 50 cm



Gambar 8. Deteksi wajah jarak 150 cm



Gambar 9. Deteksi wajah jarak 200 cm



Gambar 10. Deteksi multi wajah jarak 50 cm



Gambar 11. Deteksi multi wajah jarak 200 cm

Tabel 1. Hasil pengujian jarak

No	Jarak (Cm)	Deteksi Wajah	Akurasi (%)	Error (%)
1	50	Ya	100	0
2	100	Ya	99.7	0.3
3	150	Ya	99.2	0.8
4	200	Ya	98.8	1.2
5	250	Tidak	0	100
6	300	Tidak	0	100

Tabel 2. Hasil pengujian multi wajah

No	Jarak (Cm)	Jumlah Wajah	Jumlah Deteksi	Akurasi (%)	Error (%)
1	50	3	3	100	0
2	150	3	3	100	0
3	200	3	3	100	0
4	250	3	0	0	100
5	300	3	0	0	100

Gambar 10 dan 11 merupakan hasil pengujian deteksi wajah menggunakan multi wajah dengan berbagai jarak. Hasil lengkap pengujian multi wajah terhadap jarak ditunjukkan pada Tabel 2. Hasil pengujian deteksi multi wajah terhadap jarak pada Tabel 2 menunjukkan bahwa sistem dapat mendeteksi wajah dengan maksimal 3 wajah dengan jarak maksimal adalah 200 cm.

Hasil pengujian menunjukkan bahwa system dapat mendeteksi dan melakukan identifikasi citra dengan jarak 50, 100, dan 150 cm dengan akurasi yang berkisar pada 99.2% hingga 100%. Nilai akurasi dapat berubah sesuai dengan kondisi citra pada masing – masing jarak. Pada pengembangan CPS yang telah dilakukan terdapat peningkatan kemampuan system yaitu proses presensi yang digunakan dengan penggunaan mesin presensi dapat digantikan system menggunakan CCTV. Dengan kemampuan tersebut system dapat bekerja dari mana saja tanpa terpaku terhadap satu tempat untuk melakukan presensi. Untuk performa presensi secara keseluruhan hanya dilakukan berfokus pada tingkat akurasi dengan multi wajah.

4. Kesimpulan

Berdasarkan analisis perancangan, dan implementasi yang telah dilakukan dalam pembangunan presensi menggunakan cyber physical system dengan multi deteksi wajah, didapatkan bahwa sistem mampu mendeteksi wajah hingga tiga wajah dalam satu waktu sehingga mempermudah proses presensi. Akurasi deteksi mencapai 98.8% dengan jarak deteksi 250 cm dari posisi kamera.

Referensi

1. S. Sawhney, K. Kacker, S. Jain, S. N. Singh, and R. Garg, "Real-time smart attendance system using face recognition techniques," *Proc. 9th Int. Conf. Cloud Comput. Data Sci. Eng. Conflu.* 2019, pp. 522–525, 2019, doi: 10.1109/CONFLUENCE.2019.8776934.
2. N. T. Son *et al.*, "Implementing CCTV-based attendance taking support system using deep face recognition: A case study at FPT polytechnic college," *Symmetry (Basel)*, vol. 12, no. 2, 2020, doi: 10.3390/sym12020307.
3. A. Ahmad, A. Paul, M. M. Rathore, and H. Chang, "Smart cyber society: Integration of capillary devices with high usability based on Cyber-Physical System," *Futur. Gener. Comput. Syst.*, vol. 56, no. 2016, pp. 493–503, 2016, doi: 10.1016/j.future.2015.08.004.
4. L. Ribeiro, "Cyber-physical production systems' design challenges," *IEEE Int. Symp. Ind. Electron.*, pp. 1189–1194, 2017, doi: 10.1109/ISIE.2017.8001414.
5. A. D. Rocha, P. Lima-Monteiro, M. Parreira-Rocha, and J. Barata, "Artificial immune systems based multi-agent architecture to perform distributed diagnosis," *J. Intell. Manuf.*, vol. 30, no. 4, pp. 2025–2037, 2019, doi: 10.1007/s10845-017-1370-y.
6. Y. H. Son, K. T. Park, D. Lee, S. W. Jeon, and S. Do Noh, "Digital twin-based cyber-physical system for automotive body production lines," *Int. J. Adv. Manuf. Technol.*, vol. 115, no. 1–2, pp. 291–310, 2021, doi: 10.1007/s00170-021-07183-3.
7. A. Vinay, A. Lokesh, V. R. Kamath, K. N. B. Murty, and S. Natarajan, *Enhancement of Degraded CCTV Footage for Forensic Analysis*, vol. 1165. Springer Singapore, 2021. doi: 10.1007/978-981-15-5113-0_50.
8. R. D. Hefni Al-Fahsi, A. Patar Jiwandono Pardosi, K. A. Winanta, T. Kirana, O. F. Suryani, and I. Ardiyanto, "Laboratory Attendance Dashboard Website Based on Face Recognition System," *IES 2019 - Int. Electron. Symp. Role Techno-Intelligence Creat. an Open Energy Syst. Towar. Energy Democr. Proc.*, pp. 19–23, 2019, doi: 10.1109/ELECSYM.2019.8901615.
9. S. Bahri and H. Kusindaryadi, "Rancang Bangun Pemantauan Absensi Mahasiswa dengan Menggunakan Sidik Wajah secara Simultan Melalui CCTV Ruang Kelas," *Resist. (elektRONika kEndali Telekomun. tenaga List. kOmputeR)*, vol. 3, no. 1, p. 37, 2020, doi: 10.24853/resistor.3.1.37-44.



© 2019 oleh penulis. Dikirim untuk kemungkinan publikasi akses terbuka di bawah syarat dan ketentuan lisensi Creative Commons Attribution (CC BY) (<http://creativecommons.org/licenses/by/4.0/>).

Rancang Bangun Aplikasi *Intelligent Visual Scanner* berbasis CNN untuk identifikasi cacat pada hasil pengelasan

Mochammad Karim Al Amin ¹, Ryan Yudha Adhitya ², Mohammad Miftachul Munir ³, Mohammad Thoriq Wahyudi ⁴, Dika Anggara ⁵, Eka Cahya Septian ⁶, Muhammad Ainul Yaqin ⁷, Muhammad Ilham Safrudin ⁸, Aulia Rahma Annisa ⁹, Zindhu Maulana Ahmad Putra ¹⁰

^{1,3,4,5} Program Studi Teknik Pengelasan, Politeknik Perkapalan Negeri Surabaya, Surabaya, Indonesia

^{2,7,8} Program Studi Teknik Otomasi, Politeknik Perkapalan Negeri Surabaya, Surabaya, Indonesia

⁹ Program Studi Teknik Komputer, Institut Teknologi Telkom Surabaya, Surabaya, Indonesia

¹⁰ Program Studi Teknik Kelistrikan Kapal, Politeknik Perkapalan Negeri Surabaya, Surabaya, Indonesia

⁶ Chief Executive Officer, PT. Kintomo Engineering Group, Klaten, Indonesia

* Korespondensi: ranjudhaadhitya@ppns.ac.id

Abstrak: Pengelasan merupakan salah satu sektor yang berperan penting dalam mendukung pembangunan infrastruktur yang semakin pesat. Namun dalam proses pengelasan banyak muncul kesalahan - kesalahan atau cacat yang terkadang luput dari inspeksi dan menyebabkan kerusakan pada benda kerja. Tujuan diadakannya penelitian ini adalah untuk merancang aplikasi *Intelligent Visual Scanner* dengan menggunakan metode CNN untuk mendeteksi kelayakan hasil pengelasan berdasarkan 3 kondisi, yaitu *Normal*, *Excess Reinforcement*, dan *Undercut*. Daerah yang akan dideteksi merupakan bagian dari hasil pengelasan pada plat besi dengan ketebalan 4mm. Pemotretan dilakukan menggunakan kamera ponsel beresolusi 48 MP. Gambar yang diambil berukuran 3024 x 3024 piksel, dengan terlebih dahulu diproses dengan konversi RGB ke *grayscale*. Kemudian gambar tersebut diperkecil (diubah ukurannya) ke skala yang lebih kecil yaitu 128 x 128 piksel untuk mempercepat proses training dan pada akhirnya proses testing menggunakan model CNN. Model CNN yang diusulkan pada penelitian ini menggunakan *optimizer* Adam. Untuk deteksi yang optimal, diperlukan dataset gambar sebanyak 300 gambar, dengan rincian 100 gambar normal, 100 gambar *Excess Reinforcement* dan 100 gambar *Undercut*. Data *split* training saat ini dibagi menjadi 75% data training dan 25% data validasi. Dari 300 dataset yang tersedia didapatkan rata-rata akurasi training sebesar 98% dengan rata-rata waktu training 1 menit 46 detik. Sedangkan untuk proses testing, akurasi deteksi pada *smartphone* 1 sebesar 84,8% dan *smartphone* 2 sebesar 72,7%. Pada saat proses testing dengan menggunakan *smartphone*, sistem lebih baik dalam mendeteksi hasil pengelasan dengan kondisi normal daripada kondisi cacat.

Kata kunci: Cacat; CNN; *Intelligent Visual Inspection Scanner*; Pengelasan.

1. Pendahuluan

Pengelasan merupakan salah satu metode untuk penyambungan antar benda kerja, namun pada proses pengelasan terkadang terdapat kecacatan pada hasil pengelasan yang mengakibatkan penurunan kekuatan [1]. Identifikasi hasil pengelasan perlu dilakukan untuk mengetahui kekuatan dari pengelasan tersebut. Metode identifikasi pengelasan terbagi menjadi 2 yaitu: *Destructive Testing* (DT) dan *Non Destructive Testing* (NDT) [2].

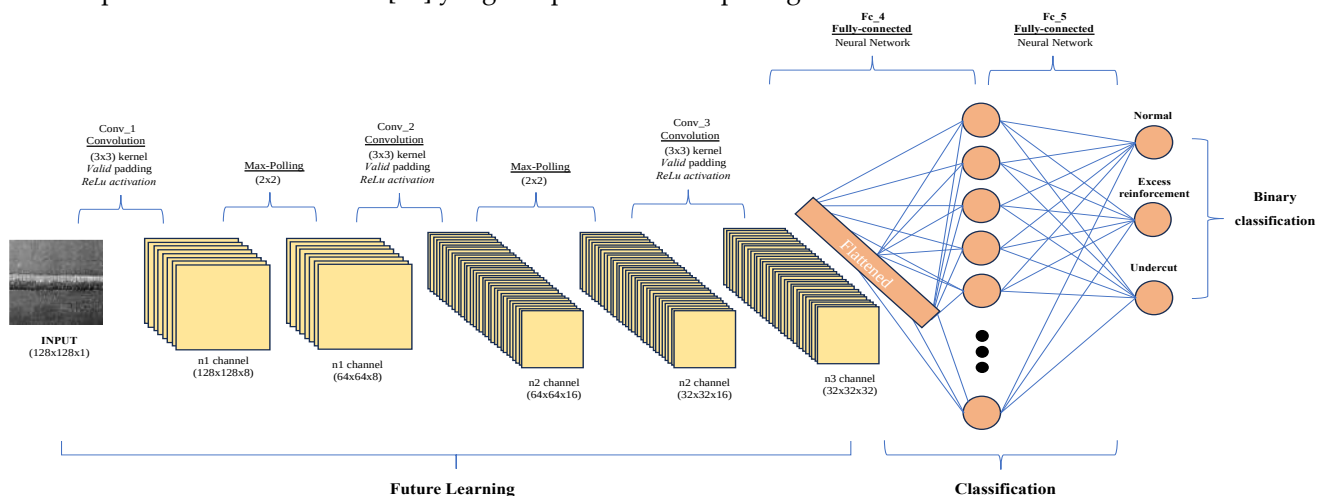
Keberadaan cacat las dapat menyebabkan kekuatan dan tingkat mutu hasil pengelasan berkurang [3]. Kecacatan yang sering terjadi adalah *Excess Reinforcement* dan *Undercut*. Tonjolan berlebih pada hasil pengelasan adalah ciri dari cacat las *Excess Reinforcement* [4]. *Excess Reinforcement* rawan terjadi retak atau patah pada pangkalnya, karena terlalu tingginya kondisi *surface* atau permukaan pengelasan [5]. Hasil pengelasan dengan bentuk berupa repetisi kontur yang sama pada base metal merupakan ciri cacat las *Undercut* [6]. *Undercut* dapat menjadi titik potensial terjadinya retakan dan perambatan retakan karena kekuatan sambungan yang berkurang [7].

Karena keberadaan cacat las maka penelitian ini mengusulkan alat bantu inspeksi yang dinamakan *Intelligent Visual Inspection Scannner* menggunakan metode CNN untuk mengidentifikasi hasil pengelasan. Pada penelitian sebelumnya yang dilakukan oleh R. Miao pada tahun 2022 [8] mendapatkan nilai akurasi sebesar 96,94% dengan metode CNN dimana penelitian tersebut mendapatkan nilai akurasi yang lebih tinggi 10% daripada metode konvensional, sedangkan pada penelitian lain yang dilakukan oleh M. Karim Al Amin pada tahun 2023 [9] mendapatkan akurasi sebesar 100% dengan waktu puncak akurasi pada iterasi ke-2 dengan menggunakan *optimizer* ADAM. Maka dengan penggunaan *optimizer* ADAM pada metode CNN diharapkan mendapatkan akurasi yang baik untuk identifikasi pengelasan. Penelitian ini dilakukan untuk merancang aplikasi *Intelligence Visual Scanner* dengan menggunakan metode CNN untuk membantu pekerja melakukan identifikasi hasil pengelasan secara visual untuk kategori *Normal*, *Excess Reinforcement*, dan *Undercut*. Perbedaan mendasar dengan penelitian sebelumnya [9] adalah modifikasi dari proses arsitektur pada proses *feature maps* dengan pengulangan proses konvolusi hingga tiga kali dan proses maxpooling sebanyak dua kali. Modifikasi arsitektur ini dilakukan untuk mendapatkan informasi yang representatif dari masukan citra asli, sehingga akan didapatkan ciri khusus untuk matriks flattening yang dapat mempermudah dan mempercepat proses training. Modifikasi arsitektur ini menjadi nilai keterbaruan dari penelitian ini.

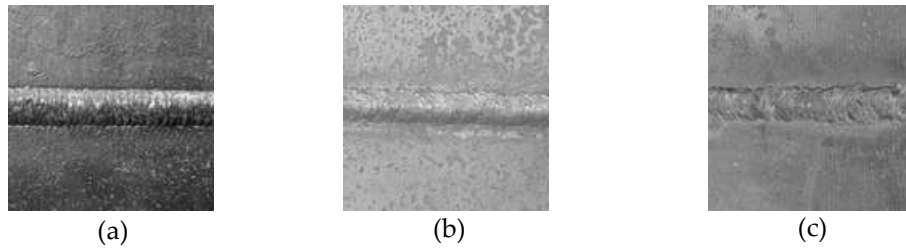
2. Metode

A. Convolutional Neural Network

Convolutional merupakan bagian dari jaringan saraf tiruan yang dapat mengambil keputusan klasifikasi dengan melatihnya, serta mengimplementasikan jaringan multi-layer supaya dapat mengevaluasi gambar visual [10]. Karena kedalaman jaringan yang tinggi, CNN termasuk kedalam Deep Neural Network yang banyak diaplikasikan pada data citra [11]. Arsitektur *Convolutional Neural Network* dimulai dengan masukan citra digital dengan ukuran yang telah ditentukan, kemudian diteruskan ke lapisan konvolusional, lapisan penghubung keseluruhan (*fully connected*), dan menghasilkan output berupa label hasil klasifikasi [12] yang direpresentasikan pada gambar 1.



Gambar 1. Struktur Convolutional Neural Network



Gambar 2. (a) *Normal*; (b) *Excess Reinforcement*; dan (c) *Undercut*

B. Spesifikasi *software* dan *hardware*

Pada penelitian ini untuk proses training dan running menggunakan software MATLAB R2023b dengan spesifikasi hardware Macbook M1 2020 dengan processor M1, RAM 8GB, SSD 246GB serta *Smartphone* Samsung seri A32, RAM 8GB, ROM 128GB, 48MP dan *Smartphone* Samsung seri A20, RAM 4GB, ROM 32GB, 13MP.

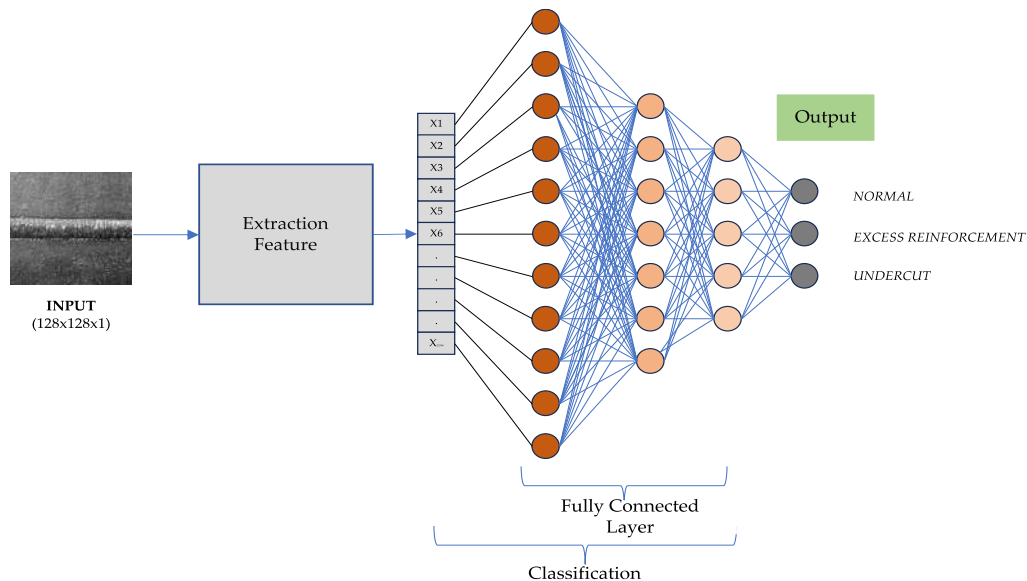
C. Alur penelitian

1. Pengumpulan dataset

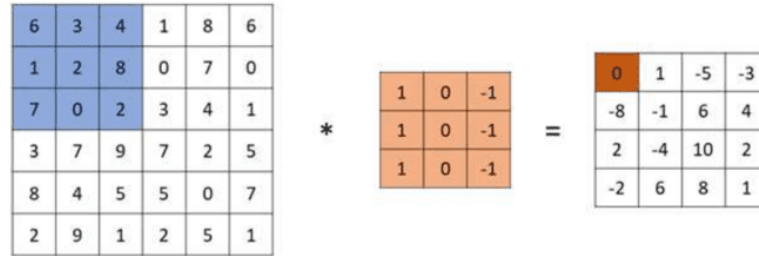
Pengumpulan dataset dilakukan dengan cara mengambil gambar hasil pengelasan menggunakan kamera *smartphone*. Data gambar yang sudah terkumpul diberi label berdasarkan kelas *Normal*, *Excess Reinforcement*, dan *Undercut* dengan jumlah masing-masing katagori berjumlah 100 gambar, sebagaimana yang ditunjukkan pada gambar 2.

2. Penyusunan arsitektur CNN

Untuk memperkaya model gambar, persiapan arsitektur dimulai dengan membuat parameter dan generator data termasuk pengubahan ukuran, rotasi gambar, *crop range*, *fill mode*, dan *range micro zoom* [13]. Pada penelitian ini menggunakan penyusunan model *Sequential*, yaitu mengurutkan dan menumpuk lapisan jaringan secara linier [14].



Gambar 3. Arsitektur *Convolutional Neural Network*

Gambar 4. Operasi *convolutional*

3. Proses data

Tahap proses data terdiri dari *convolutional layer*, *activation function*, *subsampling* atau *polling layer*, *fully connected layer*, dan *softmax function*.

1. *Convolutional layer*

Lapisan ini melakukan proses konvolusi antara filter dan gambar untuk menentukan tekstur/pola pada gambar 3 dan gambar 4 [15]. Dengan menggunakan operasi perkalian antar filter pada posisi atau titik yang sama sehingga menghasilkan ukuran dimensional feature map output yang lebih kecil. Penulisan persamaan perhitungan operasi konvolusi dapat dituliskan sebagai berikut (2.1):

$$TM[s]a, b = \sum c \sum d N[a - c, b - d] F[m, n] + Rt \quad (1)$$

Keterangan:

TM[s] : Feature map dari matriks ke-i

a,b : Koordinat pixel pada input

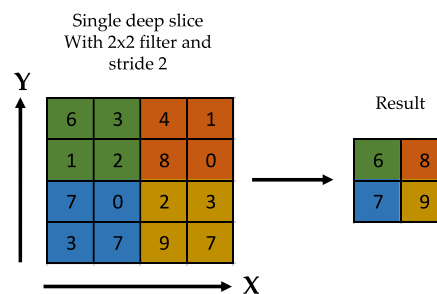
c,d : Koordinat pixel convolutional filters

N : Input matriks

T : Matriks filters

2. *Activation function*

Fungsi aktivasi dapat berupa sistem linear maupun nonlinear dengan mengukur fungsi penjumlahan dengan tujuan menentukan neuron aktif atau tidak [16]. Pelatihan data menggunakan fungsi aktivasi ReLu (Rectified Linear Unit) dapat mempercepat proses pembentukan data berdasarkan nilai terbesar (0, maksimum) yang dimasukkan.



Gambar 5. Struktural Pooling layers

3. *Subsampling* atau *polling layers*

Subsampling merupakan pengurangan jumlah elemen pada gambar yang dihasilkan dengan tujuan untuk mempercepat proses komputasi [17]. Pada penelitian ini menggunakan *max pooling* atau nilai terbesar dari tiap filter sebagai proses pooling layers seperti pada gambar 5.

4. *Fully connected layers*

Lapisan ini berfungsi untuk menghasilkan keluaran dengan mengambil seluruh masukan setiap neuron dari lapisan sebelumnya untuk melakukan operasi pada lapisan dengan masing-masing individu neuron [18].

5. *Softmax function*

Fungsi softmax sering digunakan pada lapisan keluaran jaringan saraf multilayer untuk menormalkan titik distribusi probabilitas dari hasil pengambilan vektor titik bernilai nyata yang telah diubah menjadi distribusi probabilitas agar interpretasi awal menjadi lebih mudah [19].

4. *Optimizer*

Proses optimasi penting untuk meningkatkan nilai akurasi dan juga meminimalkan overfitting pada saat pelatihan model. Dalam penelitian ini, Adam *optimizer* digunakan untuk mengoptimalkan model. *Adaptive Moment Estimation* atau ADAM merupakan metode gabungan antara RMSProp dan momentum, dua momen gradien yang dirata-ratakan dan juga mengambil gradien sebelumnya untuk mempercepat pembelajaran [20]. Penulisan persamaan perhitungan dirumuskan sebagai berikut (2.2):

$$\theta_t = \theta_t - 1 - \frac{\alpha}{\sqrt{V_t^{corrected} + \epsilon}} \cdot mt^{corrected} \quad (2)$$

Keterangan

θ_t : Parameteter yang diperbarui pada waktu t

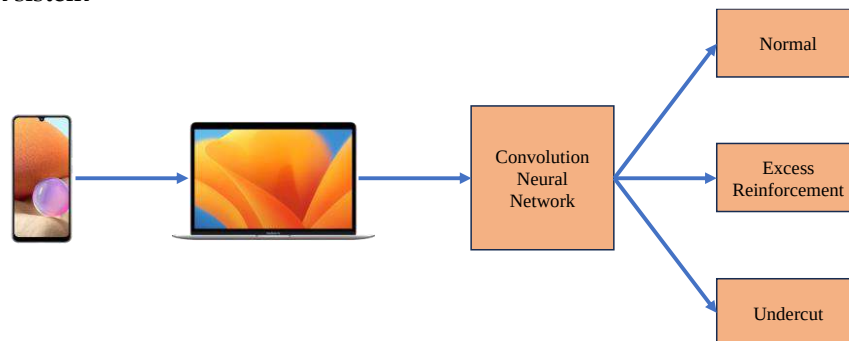
α : Laju pembelajaran

V_t : Rata-rata gradien pada waktu iterasi

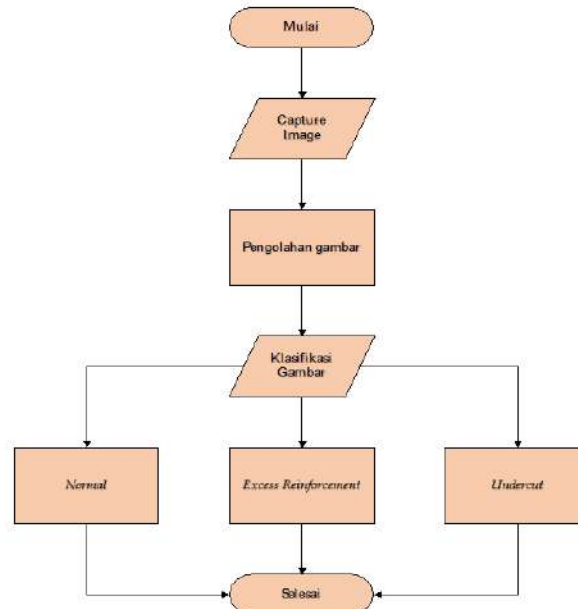
ϵ : Faktor kecil untuk mencegah pembagian oleh 0

mt : Moment gradien pada waktu iterasi

D. Perancangan sistem



Gambar 6. Blok diagram rancang bangun aplikasi *Intelligent Visual Inspection Scanner*



Gambar 7. Flow chart sistem kerja aplikasi *Intelligent Visual Scanner*

Tahapan perancangan sistem dibuat dengan aplikasi MATLAB menggunakan Bahasa C, dimulai dari *capture image* menggunakan bantuan kamera smartphone dengan mengambil ip camera menggunakan aplikasi *ip webcam*, dapat diilustrasikan pada gambar 6 dan gambar 7. Gambar yang didapat diproses untuk mendapatkan hasil klasifikasi dari bobot hasil training. Pada program klasifikasi dilakukan pemrosesan gambar dan training untuk mendapatkan model bobot dari proses training, model tersebut digunakan sebagai bobot pada program utama untuk klasifikasi hasil pengelasan. Pada sistem utama terdapat program untuk tampilan aplikasi GUI sebagai interface pada aplikasi *Intelligent Visual Inspection Scanner* yang dapat ditunjukkan pada gambar 8.



Gambar 8. Tampilan aplikasi *Intelligent Visual Inspection Scanner*

Keterangan pada gambar 8 terdiri dari beberapa bagian:

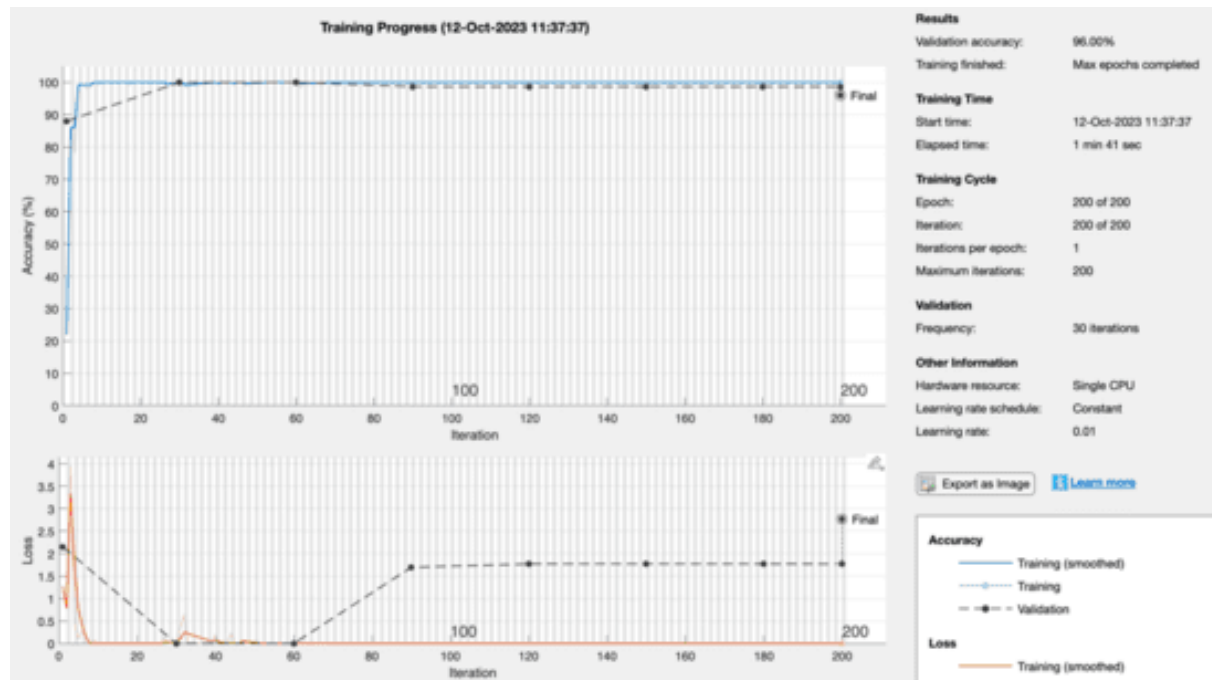
1. Setting IP Camera yang terdapat pada *smartphone*.
2. Tombol untuk memulai koneksi dengan kamera *smartphone*.
3. Tombol untuk menghentikan koneksi dengan kamera *smartphone*.
4. Tampilan data klasifikasi dalam bentuk tabel.
5. Tampilan kamera dan hasil klasifikasi.

3. Hasil

Hasil dari penelitian ini menunjukkan keberhasilan dalam mengidentifikasi hasil pengelasan secara *real-time*. Pengujian dilakukan pada plat *test piece* dengan menggunakan *smartphone* dan bantuan tripod untuk menjaga jarak kamera dengan object deteksi.

A. Hasil training dataset

Training dataset dilakukan menggunakan foto berjumlah 300 buah dengan rincian 100 foto normal, 100 foto *Excess reinforcement*, dan 100 foto *Undercut*. Pengujian training dilakukan untuk mengetahui akurasi dan lost dari dataset yang dilakukan, ditunjukkan pada gambar 9. Proses training CNN menghasilkan data sebagai berikut:



Gambar 9. Grafik akurasi dan training loss

Tabel 1. Akurasi training

No.	Akurasi	Waktu	
		Menit	Detik
1	96%	2	3
2	98%	1	58
3	98%	1	54
4	97%	1	57
5	98%	1	48
6	96%	1	52
7	99%	1	47
8	98%	1	47
9	97%	1	49
10	99%	1	41

Dengan menggunakan 300 dataset didapatkan rata- rata akurasi training sebesar 98% dengan rata – rata waktu training 1 menit 46 detik, secara keseluruhan dideskripsikan pada tabel 1. Proses training yang memakan waktu hingga lebih dari satu menit disebabkan oleh karakteristik input neuron pada *flattening layer* yang relatif banyak sehingga membutuhkan waktu komputasi proses yang tidak sedikit. Waktu pemrosesan training memiliki respon yang bervariasi karena kecepatan komputasi *personal computer* yang dinamis dengan hasil akurasi tiap sampling yang relative konstan dengan kisaran akurasi 96% hingga 99%. Waktu komputasi sedikit banyak juga dipengaruhi oleh metode propagasi balik ADAM *optimization* yang digunakan, proses kalkulasi metode ADAM memiliki matriks komputasi yang kompleks yang akan memakan ruang penyimpanan sementara atau *temporary space* pada komputer.

B. Hasil perbandingan dengan menggunakan dua buah *smartphone*

Tabel 2. Hasil *smartphone* Samsung A32

No.	Normal	Excess Reinforcement	Undercut
1	Berhasil	Berhasil	Berhasil
2	Berhasil	Berhasil	Berhasil
3	Berhasil	Berhasil	Tidak
4	Berhasil	Berhasil	Berhasil
5	Berhasil	Tidak	Berhasil
6	Berhasil	Berhasil	Berhasil
7	Berhasil	Tidak	Berhasil
8	Berhasil	Berhasil	Berhasil
9	Tidak	Berhasil	Berhasil
10	Berhasil	Berhasil	Tidak
11	Berhasil	Berhasil	Berhasil
Akurasi	90.9%	81.8%	81.8%
Rerata		84.8%	

Tabel 3. Hasil *smartphone* Samsung A20

No.	Normal	Excess Reinforcement	Undercut
1	<u>Berhasil</u>	<u>Berhasil</u>	<u>Tidak</u>
2	<u>Berhasil</u>	<u>Berhasil</u>	<u>Berhasil</u>
3	<u>Tidak</u>	<u>Tidak</u>	<u>Tidak</u>
4	<u>Tidak</u>	<u>Berhasil</u>	<u>Berhasil</u>
5	<u>Berhasil</u>	<u>Berhasil</u>	<u>Berhasil</u>
6	<u>Berhasil</u>	<u>Berhasil</u>	<u>Berhasil</u>
7	<u>Berhasil</u>	<u>Berhasil</u>	<u>Berhasil</u>
8	<u>Berhasil</u>	<u>Tidak</u>	<u>Tidak</u>
9	<u>Berhasil</u>	<u>Tidak</u>	<u>Berhasil</u>
10	<u>Berhasil</u>	<u>Berhasil</u>	<u>Berhasil</u>
11	<u>Berhasil</u>	<u>Berhasil</u>	<u>Tidak</u>
<u>Akurasi</u>	81.8%	72.7%	63.6%
<u>Rerata</u>		72.7%	

Dari tabel hasil akurasi didapatkan nilai akurasi pada *smartphone* 1 mendapatkan nilai akurasi pada kondisi normal sebesar 90.9%, pada kondisi cacat *Excess reinforcement* sebesar 81.8%, dan pada kondisi cacat *Undercut* sebesar 81.8% sedangkan pada *smartphone* 2 mendapatkan nilai akurasi pada kondisi normal sebesar 81.8%, pada kondisi cacat *Excess reinforcement* sebesar 72.7%, dan pada cacat undercut sebesar 63.6%. *Smartphone* 1 dalam hal ini adalah Samsung A23 memiliki kondisi akurasi rata – rata yang lebih baik dengan rata – rata akurasi 84.8% jika dibanding dengan *Smartphone* 2 atau Samsung A20 dimana akurasi rata – rata yang dihasilkan adalah 72.7%. Hal ini disebabkan kemampuan autofocus yang tidak dimiliki oleh semua *smartphone* dan faktor pencahayaan eksternal.

4. Kesimpulan

Berdasarkan penelitian yang telah dilakukan maka dapat disimpulkan bahwa sistem berhasil melakukan identifikasi hasil pengelasan dengan menggunakan kamera *smartphone* untuk mendeteksi kelayakan hasil pengelasan berdasarkan 3 kondisi, yaitu *Normal*, *Excess Reinforcement*, dan *Undercut*. Ketepatan akurasi deteksi *smartphone* 1 sebesar 83,3% dan *smartphone* 2 sebesar 70%, sementara itu terdapat perbedaan akurasi antara kondisi normal dan kondisi cacat yang dapat diindikasikan bahwa *smartphone* 1 dan *smartphone* 2 lebih baik dalam mendeteksi hasil pengelasan dengan kondisi normal daripada kondisi cacat. Tampilan GUI mampu menampilkan parameter tanggal secara *real-time* dengan benar dan juga menampilkan parameter objek yang masuk didalam bounding box pada hasil tangkapan kamera seperti *size, format width, height, bitdept, colortype*. Pada sistem ini tingkat keberhasilan dipengaruhi oleh spesifikasi kamera dan kemampuan sebuah kamera untuk memfokuskan ke sebuah objek hasil pengelasan secara dekat. Hal ini menunjukkan bahwa perangkat keras (kamera) memiliki peran yang penting akurasi deteksi.

Referensi

1. H. Pan, Z. Pang, Y. Wang, Y. Wang, and L. Chen, "A New Image Recognition and Classification Method Combining Transfer Learning Algorithm and MobileNet Model for Welding Defects," *IEEE Access*, vol. 8, pp. 119951–119960, 2020, doi: 10.1109/ACCESS.2020.3005450.
2. A. Rizal Lucky Pradana, S. Khumaidi, M. T. St, R. Andiana, and S. St, "Identifikasi Penyebab Cacat Pada Hasil Pengelasan Dengan Image Processing Menggunakan Metode YOLO", [Online]. Available: <https://journal.trunojoyo.ac.id/triac>
3. F. Kurniadi, F. Handoko, T. Priyasmanu, and P. Studi Teknik Industri S-, "ANALISIS WELDING DEFECT RATE DAN PENANGANANNYA DENGAN METODE SIX SIGMA DAN FMEA (STUDI KASUS: PT. MEINDO ELANG INDAH, MUARA JAWA, KUTAI KARTANEGARA, KALTIM)," *Jurnal Mahasiswa Teknik Industri*, vol. 5, no. 1, 2022.
4. N. V. R. Maganti and R. R. Potturi, "Investigation on Mechanical and Machinability Properties of Aluminium Metal Matrix Composite Reinforced with Titanium Oxide (TiO₂) and Graphite (Gr) Particles," *Trends in Sciences*, vol. 20, no. 11, p. 5682, Aug. 2023, doi: 10.48048/tis.2023.5682.
5. M. Bin Afan, B. Rahim Jurusan Teknik Mesin, F. Teknik, and U. Negeri Padang Jl Hamka, "Pengaruh Suhu Penyimpanan Elektroda Low Hydrogen E7016 terhadap Hasil Uji Tekuk Sambungan Las Pelat Baja Karbon SS400," 2020. [Online]. Available: <https://jurnal.polines.ac.id/index.php/rekayasa>
6. A. Khatammi and A. W. Rizqi, "Analisis Kecacatan Produk Pada Hasil Pengelasan dengan Metode Failure Mode Effect Analysis," *Serambi Engineering*, vol. VII, no. 2, 2022.
7. A. O. Igbinake, "FUPRE Journal of Scientific and Industrial Research Estimation of Undercuts in Mild Steel Weldment using Artificial Neural Network," *Journal Research Article / Review Article FUPRE JOURNAL*, vol. 7, no. 1, pp. 138–147, 2023, [Online]. Available: <http://fupre.edu.ng/journal>
8. R. Miao *et al.*, "Real-time defect identification of narrow overlap welds and application based on convolutional neural networks," *J Manuf Syst*, vol. 62, pp. 800–810, Jan. 2022, doi: 10.1016/j.jmsy.2021.01.012.
9. M. Karim Al Amin *et al.*, "Analysis of Optimizer Effects on CNN Model for Defect Identification in Welding Results of A 106 Grade B Carbon Steel Pipe.", ICEEI 2023.
10. M. Defriani and I. J. Jaelani, "Recognition of Regional Traditional House in Indonesia Using Convolutional Neural Network (CNN) Method," *Journal of Computer Networks, Architecture and High Performance Computing*, vol. 4, no. 2, pp. 104–115, Jul. 2022, doi: 10.47709/cnahpc.v4i2.1562.
11. P. Adi Nugroho, I. Fenriana, and R. Arijanto, "IMPLEMENTASI DEEP LEARNING MENGGUNAKAN CONVOLUTIONAL NEURAL NETWORK (CNN) PADA EKSPRESI MANUSIA," *JURNAL ALGOR*, vol. 2, no. 1, 2020, [Online]. Available: <https://jurnal.buddhidharma.ac.id/index.php/algor/index>
12. M. D. Pramita, B. Kurniawan, and N. P. Utama, "Mask Wearing Classification using CNN," in *2020 7th International Conference on Advanced Informatics: Concepts, Theory and Applications, ICAICTA 2020*, Institute of Electrical and Electronics Engineers Inc., Sep. 2020. doi: 10.1109/ICAICTA49861.2020.9429029.
13. A. F. Anavyanto, M. Maimunah, M. R. A. Yudianto, and P. Sukmasetya, "EfficientNetV2M for Image Classification of Tomato Leaf Diseases," *PIKSEL : Penelitian Ilmu Komputer Sistem Embedded and Logic*, vol. 11, no. 1, pp. 55–76, Mar. 2023, doi: 10.33558/piksel.v11i1.5925.
14. G. Xie, A. Shanguan, R. Fei, W. Ji, W. Ma, and X. Hei, "Motion trajectory prediction based on a CNN-LSTM sequential model," *Science China Information Sciences*, vol. 63, no. 11, Nov. 2020, doi: 10.1007/s11432-019-2761-y.
15. IEEE Communications Society. Indonesia Chapter., Universitas Telkom., and Institute of Electrical and Electronics Engineers, *Proceedings, the 2020 IEEE International Conference on Industry 4.0, Artificial Intelligence, and Communications Technology : July 7-8, 2020, Bali, Indonesia*.
16. T. Purwaningsih, T. Nurhikmat, and P. B. Utami, "Image Classification of Golek Puppet Images using Convolutional Neural Networks Algorithm," *Int. J. Advance Soft Compu. Appl*, vol. 11, no. 1, 2019.
17. STMIK AKAKOM Yogyakarta, Institute of Electrical and Electronics Engineers. Indonesia Section, and Institute of Electrical and Electronics Engineers, *2nd ISRITI 2019 proceeding : the 2nd International Seminar on Research of Information Technology and Intelligent Systems 2019 : "The future & challenges of extended intelligence" : Yogyakarta, Indonesia, 05-06 December 2019*.
18. S. Bhattacharyya, RCC Institute of Information Technology. Department of Information Technology, and Institute of Electrical and Electronics Engineers, *Proceedings, 2018 Fourth IEEE International Conference on*

Research in Computational Intelligence and Communication Networks (ICRCICN) : 22nd and 23rd November, 2018, Kolkata, India.

19. S. Kanungo, "ANALYSIS OF IMAGE CLASSIFICATION DEEP LEARNING ALGORITHM."
20. M. H. Saleem, J. Potgieter, and K. M. Arif, "Plant disease classification: A comparative evaluation of convolutional neural networks and deep learning optimizers," *Plants*, vol. 9, no. 10, pp. 1–17, Oct. 2020, doi: 10.3390/plants9101319.



© 2019 oleh penulis. Dikirim untuk kemungkinan publikasi akses terbuka di bawah syarat dan ketentuan lisensi Creative Commons Attribution (CC BY) (<http://creativecommons.org/licenses/by/4.0/>).

Perbandingan Performa Algoritma VGG16 dan VGG19 Melalui Metode CNN Untuk Klasifikasi Varietas Beras

Adisaputra Zidha Noorizki ¹ dan Weny Indah Kusumawati ^{2*}

^{1, 2*} Teknik Komputer, Fakultas Teknologi dan Informatika, Universitas Dinamika, Surabaya, Indonesia

* Korespondensi: weny@dinamika.ac.id

Abstrak: Beras merupakan jenis pangan utama yang paling populer di konsumsi di seluruh dunia. Berbagai negara pun melakukan proses impor untuk memenuhi kebutuhan penduduk di negaranya, salah satunya adalah Indonesia. Indonesia merupakan negara yang tergolong memiliki permintaan pasar pangan yang tinggi di ruang lingkup global. Masuknya berbagai varietas beras akibat adanya kegiatan impor di Indonesia membuat para importir mengalami kesulitan selama proses penyortiran produknya. Seiring berjalannya kemajuan teknologi, para importir juga harus terus beradaptasi, salah satunya dengan memanfaatkan salah satu teknologi yang akhir-akhir ini sedang marak dikembangkan oleh para peneliti. CNN atau *Convolutional Neural Network* merupakan salah satu teknologi yang memiliki kemampuan yang baik dalam mengklasifikasikan citra gambar. Penelitian ini secara khusus memfokuskan pada perbandingan kinerja dua algoritma CNN, yaitu VGG16 dan VGG19. Hasil penelitian ini menyatakan bahwa algoritma VGG16 mampu mencapai akurasi sebesar 98% dengan waktu *training* 73,405 detik, sementara algoritma VGG19 mencapai akurasi 97% dengan waktu *training* 78,098 detik. Hal ini menunjukkan bahwa algoritma VGG16 secara signifikan lebih unggul dibandingkan dengan algoritma VGG19 dalam mengklasifikasikan varietas beras menggunakan dataset yang terdiri atas lima kelas.

Kata kunci: CNN; Machine Learning; Varietas Beras.

1. Pendahuluan

Beras merupakan salah satu jenis bahan pangan yang secara luas dikonsumsi dan populer di seluruh dunia. Bijinya dihasilkan oleh tanaman padi yang dikenal dengan nama ilmiah *Oryza sativa*. Kandungan karbohidrat dalam beras, terutama dalam bentuk amilosa dan amilopektin, menjadikannya sebagai sumber utama energi bagi manusia dalam menjalani aktivitas sehari-hari [1]. Namun, keberagaman beras tidak hanya berhenti pada aspek nutrisi. Berbagai varietas beras yang ada memiliki peran krusial dalam pertanian dan pangan. Perbedaan-perbedaan ini memengaruhi produktivitas, kualitas, dan adaptasi tanaman terhadap lingkungan di berbagai wilayah. Dalam konteks pertanian, keberagaman varietas beras ini dapat memberikan hasil yang berbeda dalam hal rasa, tekstur, nilai gizi, serta kemampuan tahan terhadap kondisi lingkungan yang beragam pula [2]. Semua variasi ini memberikan konsumen dan pelaku industri pangan pilihan yang beragam sesuai dengan kebutuhan dan preferensi mereka. Dengan keadaan pasar beras yang kompleks, pentingnya pengenalan varietas yang akurat menjadi kunci dalam memenuhi permintaan konsumen dan menjaga kelangsungan pasokan dalam keadaan yang stabil.

Indonesia, sebagai salah satu kontributor penting dalam permintaan pasar pangan global, sangat bergantung pada beras karena mayoritas penduduknya menjadikan beras sebagai makanan pokok [3]. Penduduk asli Indonesia mengandalkan beras sebagai pilar utama dalam pola makan

mereka. Tingginya ketergantungan ini telah menghasilkan peningkatan permintaan yang konsisten, namun juga berdampak pada kesenjangan antara jumlah produsen beras dan konsumen di Indonesia. Hal ini mendorong negara untuk melakukan impor beras dari berbagai negara guna menjaga keseimbangan antara pasokan dan permintaan.

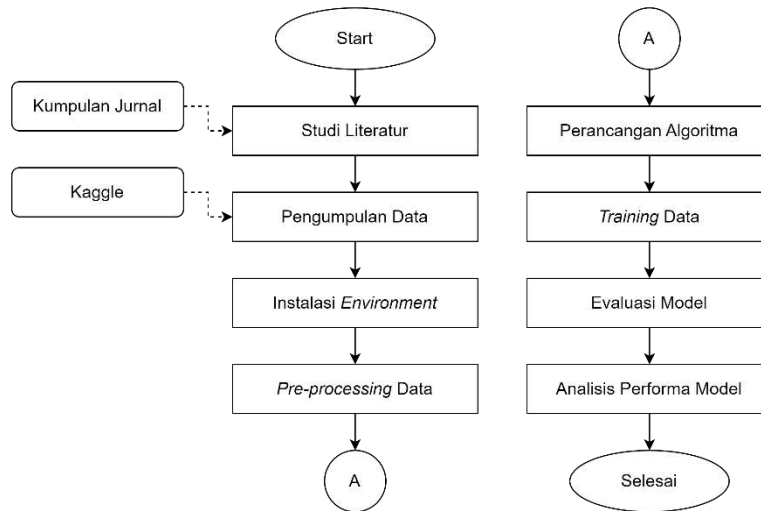
Masuknya berbagai macam varietas beras yang berasal dari berbagai negara pula yang mengakibatkan importir di Indonesia kewalahan saat melakukan kegiatan sortir produk. Berdasarkan data Badan Pusat Statistik (BPS) pada tahun 2022, negara India, Pakistan, Vietnam, Thailand, Myanmar dan Jepang menjadi negara-negara dengan angka kegiatan impor beras tertinggi ke Indonesia, ada sekitar 429,207 ton beras yang telah masuk ke negara Indonesia [4]. Penelitian ini akan difokuskan pada lima jenis beras, yaitu Arburio, Bastami, Ipsala, Jasmine, dan Karacadag. Namun, hingga saat ini belum ada data yang valid mengenai apakah jenis-jenis beras ini beredar di Indonesia. Butuh waktu yang cukup lama bagi beberapa importir baru untuk melakukan kegiatan sortir produk serta memastikan termasuk ke varietas mana beras yang saat itu sedang disortir. Sedangkan beras yang dijual di pasaran diberi harga sesuai dengan beberapa patokan yang umumnya digunakan antara lain tekstur, bentuk, warna, dan masih banyak lagi. Dengan demikian, proses sortir memang sudah semestinya dilakukan guna menentukan harga dari beras tersebut.

Seiring kemajuan teknologi saat ini, sudah semestinya para pelaku usaha terutama dibidang pertanian dan perdagangan terkhusus beras harus terus mengikuti kemajuan teknologi dan beradaptasi. Beberapa tahun terakhir, para ahli telah bersaing untuk melakukan penelitian tentang pengolahan dan klasifikasi citra menggunakan salah satu metode dalam *deep learning* yang dikenal sebagai *Convolutional Neural Network*, atau yang lebih umum dikenal sebagai *CNN*. *Convolutional Neural Network* (*CNN*) merupakan salah satu metode klasifikasi yang menggunakan lapisan konvolusi untuk memproses input dengan filter tertentu [5]. *CNN* merupakan variasi pengembangan dari metode sebelumnya yaitu jaringan syaraf tiruan yang memiliki beberapa lapisan tersembunyi dan memiliki bobot saja. *CNN* umumnya digunakan untuk melakukan tugas terkait dengan pengolahan data gambar yang diklasifikasikan dengan metode *supervised learning*. Pembelajaran yang diawasi/*supervised learning* merupakan jenis algoritma yang menghasilkan fungsi untuk memetakan input menuju output yang diinginkan [6]. Terdapat beberapa lapisan pada *model* dasar *CNN* antara lain lapisan *convolution*/konvolusi, fungsi aktivasi, lapisan *pooling*, lapisan *flatten*, dan yang terakhir adalah lapisan *fully connected* [7].

CNN sendiri memiliki berbagai algoritma yang telah ditemukan oleh peneliti-peneliti terdahulu. Dalam penelitian ini, peneliti membandingkan kinerja antara algoritma *VGG16* dan *VGG19* dalam mengerjakan tugas untuk mengklasifikasikan varietas beras. Nama algoritma ini diambil dari singkatan dari *Visual Geometry Group*, yang disingkat menjadi *VGG* dan kemudian angka 16 dan 19 merupakan jumlah lapisan/*layer* dari algoritma tersebut. *VGG* dibuat oleh sekelompok peneliti Universitas Oxford, Inggris. Kelompok penelitian *VGG* dipimpin oleh Profesor Andrew Zisserman dan Profesor Karen Simonyan [8]. Algoritma *VGG* merupakan hasil perbaikan dan pengembangan dari *AlexNet* yang telah dipublikasikan pada tahun 2012. Arsitektur dari *VGG* berfokus pada meningkatkan proses ekstraksi fitur dengan menggunakan lapisan konvolusi yang lebih banyak [9].

2. Bahan dan Metode

Pada penelitian ini, urutan kegiatan yang dijalani oleh peneliti dapat dilihat dalam diagram alir yang terdapat pada Gambar 1. Penelitian dimulai dengan studi literatur melalui referensi berbagai jurnal, kemudian dilanjutkan dengan pengumpulan data yang digunakan dalam proses pelatihan menggunakan algoritma *VGG16* dan *VGG19*. Proses pelatihan dilakukan dengan beberapa jumlah iterasi. Setelah mendapatkan hasil dari proses pembelajaran, peneliti melakukan perbandingan untuk menentukan algoritma yang memiliki performa lebih baik dalam implementasinya pada klasifikasi citra varietas beras.



Gambar 1. Diagram alir penelitian

2.1. Studi Literatur

Penelitian ini dimulai dengan melakukan studi literatur yang mencakup berbagai jurnal yang relevan dengan penggunaan algoritma VGG16 dan VGG19 melalui metode *Convolutional Neural Network* (CNN) dalam konteks klasifikasi citra gambar dari berbagai objek. Serta dapat menggunakan hasil dari *model* yang telah dilatih pada penelitian-penelitian sebelumnya sebagai referensi untuk *model* yang digunakan dalam penelitian ini.

2.2. Pengumpulan Data

Dalam tahap ini, peneliti menghimpun data yang diperlukan untuk penelitian, memanfaatkan dataset yang tersedia secara terbuka dari sumber Kaggle. Dataset ini terdiri dari total 75,000 citra, dengan fokus pada gambar satu butir beras yang dibagi menjadi lima kelas, yaitu Arburio, Bastami, Ipsala, Jasmine, dan Karacadag. Jumlah data dalam setiap kelas adalah sekitar 15,000 citra [10]. Untuk memberikan gambaran visual tentang jenis citra yang ada dalam dataset ini, peneliti telah menyediakan Tabel 1 yang menggambarkan contoh-contoh dari setiap kelas. Serta setiap representasi citra dalam dataset ini memiliki ekstensi '.jpg'.

Tabel 1. Dataset varietas beras

Gambar Citra	Nama Kelas
	Arburio
	Bastami
	Ipsala
	Jasmine
	Karacadag

2.3. Instalasi Environment

Pada tahap ini, sebelum melanjutkan proses pengolahan data untuk meningkatkan efisiensi pelatihan data dalam *model Convolutional Neural Network (CNN)*, peneliti harus melakukan instalasi lingkungan yang sesuai. Dikarenakan penelitian ini menggunakan bahasa pemrograman *Python* dan berfokus pada klasifikasi gambar, beberapa *library* yang harus diinstal antara lain adalah *Keras*, *NumPy*, *Matplotlib*, *scikit-learn*, dan *OpenCV*. Perintah *install* dapat dilakukan dengan beberapa cara [11], salah satunya dengan menggunakan perintah '*pip*' yang mana dapat dilakukan pada *command prompt*/terminal, serta dapat dilihat pada Gambar 2.

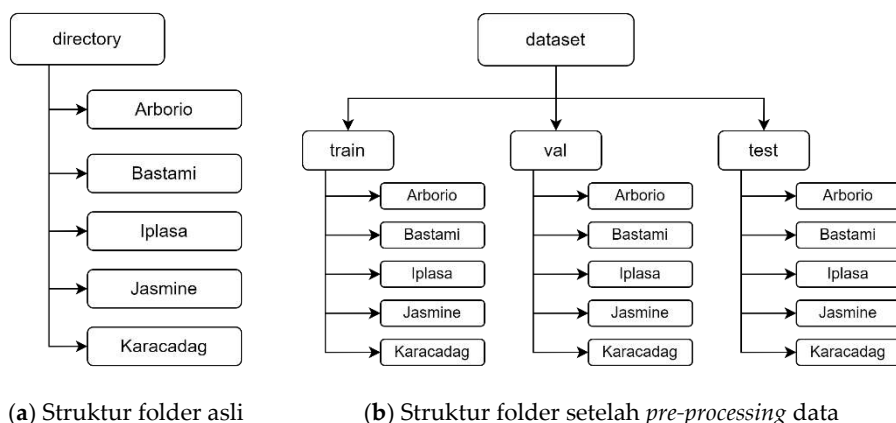
```
pip install keras
pip install numpy
pip install matplotlib
pip install scikit-learn
```

Gambar 2. Instalasi *library* menggunakan perintah '*pip*'

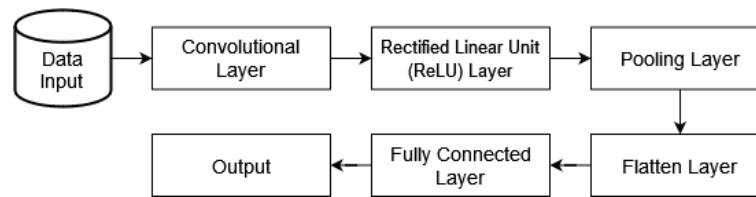
2.4. Pre-processing Data

Pada tahap ini, peneliti bertujuan untuk mengoptimalkan data agar lebih siap untuk proses pembelajaran menggunakan algoritma VGG16 dan VGG19. Dataset yang tersedia di Kaggle terdiri dari 15,000 citra gambar untuk setiap kelas, dan citra-citra tersebut telah dikelompokkan dalam folder sesuai dengan kelasnya. Langkah pertama yang dilakukan adalah memberikan label berupa angka dari 0 hingga 4 kepada setiap kelas. Dalam penelitian ini, peneliti tidak menggunakan seluruh data yang disediakan, namun peneliti menggunakan beberapa data saja, dengan menggunakan metode pengambilan sampel secara acak. Penelitian ini menggunakan data dengan jumlah 700 citra gambar dari setiap kelas.

Selanjutnya, citra-citra ini dibagi kembali menjadi tiga bagian yang berbeda. Antara lain data latih (*training*), data validasi (*validation*), dan data uji (*testing*) seperti yang dapat dilihat dari Gambar 3 (a) dan (b). Pembagian ini akan dilakukan dengan proporsi 8:1.5:0.5 seperti yang telah diuji pada beberapa penelitian sebelumnya [12], atau dalam jumlahnya yaitu 560 citra gambar untuk data *training*, 105 citra gambar untuk data *validation*, dan 35 citra gambar untuk data *testing* setiap kelas varietas berasnya. Dengan total keseluruhan data yang digunakan pada penelitian kali ini sejumlah 3,500 citra gambar yang terbagi menjadi 5 kelas berbeda.



Gambar 3. Struktur folder untuk data citra gambar varietas beras



Gambar 4. Diagram alir algoritma *Convolutional Neural Network* (CNN)

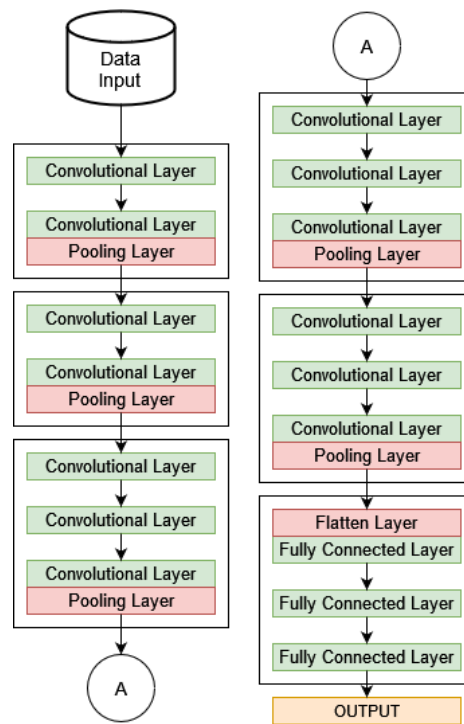
2.5. Training Data dengan metode CNN

Pada tahap ini, peneliti menentukan jumlah dari setiap lapisan/*layer* pada masing-masing algoritma yang digunakan. Dalam pengambilan keputusan ini, peneliti mempertimbangkan beberapa *model* yang telah dilakukan pengujian pada penelitian-penelitian sebelumnya. Hal ini memungkinkan peneliti untuk memanfaatkan *model* yang sudah terbukti berhasil dalam penelitian terdahulu sebagai acuan dalam merancang model pada penelitian kali ini. Dengan demikian, peneliti dapat memastikan bahwa pendekatan yang diambil sudah teruji dan memiliki potensi untuk memberikan hasil yang baik dalam klasifikasi citra gambar yang dilakukan. Demi memberikan gambaran yang lebih komprehensif terkait proses algoritma yang digunakan, peneliti juga menyertakan blok diagram yang mengilustrasikan struktur dan urutan operasi dari proses algoritma *Convolutional Neural Network* (CNN).

Blok diagram di atas memperlihatkan bagaimana data masukan melalui berbagai lapisan yang memiliki fungsi dan tugasnya masing-masing. *Layer*/lapisan yang pertama kali dilalui oleh data input adalah lapisan konvolusi yang memiliki fungsi untuk mengekstrak fitur-fitur dari gambar input melalui operasi konvolusi. Bertujuan untuk mendeteksi fitur-fitur seperti sudut, tepi, dan tekstur pada gambar input. Hasil proses konvolusi akan melewati lapisan ReLU yang merupakan salah satu jenis fungsi aktivasi di mana berperan untuk mengubah nilai-nilai negatif dari hasil konvolusi menjadi nol, sementara nilai-nilai positif akan tetap atau tidak berubah. Selanjutnya, keluaran dari lapisan ReLU akan mengalir ke lapisan *Pooling* yang memiliki tanggung jawab untuk mengurangi dimensi dari gambar. Dengan beberapa cara, diantaranya yaitu mengambil nilai rata-rata atau nilai maksimum dari setiap data pada gambar. Lapisan *Pooling* sangat membantu selama proses algoritma CNN berjalan, sebab lapisan ini dalam mempertahankan fitur-fitur penting sambil mengurangi kompleksitas dan sumber daya komputasi yang dibutuhkan. Setelah melewati lapisan konvolusi, ReLU, dan *Pooling*, hasilnya akan diarahkan ke lapisan yang bernama *Flatten* yang memiliki fungsi untuk merubah representasi spasial dari gambar menjadi vektor satu dimensi. Selanjutnya, akan melewati lapisan bernama *Fully Connected*, disetiap neuron di lapisan ini terhubung dengan setiap neuron di lapisan sebelumnya. Lapisan ini memiliki fungsi untuk mengintegrasikan informasi dari fitur-fitur yang telah dikenali dalam tahapan sebelumnya. Selain itu, lapisan ini juga melakukan proses pembelajaran (*learning*) untuk mengenali pola-pola kompleks dalam data dan menyesuaikan bobot-bobot antar-neuron. Setelah melalui lapisan *Fully Connected*, hasil akhirnya akan diteruskan ke lapisan *Output*. Pada lapisan ini, jumlah neuron akan disesuaikan dengan jumlah kelas yang ingin diprediksi menggunakan algoritma CNN. Dengan demikian, diagram pada Gambar 4 akan memberikan gambaran secara umum bagaimana tahapan atau proses yang terjadi pada algoritma *Convolutional Neural Network* (CNN).

2.5.1. VGG16 Model

Proses yang terjadi dalam CNN-VGG16 pada dasarnya tidak berbeda jauh dengan uraian yang telah dijelaskan sebelumnya oleh peneliti. Namun, ada sedikit perbedaan dalam cara menghitung jumlah lapisan serta menamai setiap lapisan tersebut. Pada Gambar 4, peneliti memberikan penjelasan global mengenai lapisan-lapisan mana yang akan dilewati oleh data gambar hingga menghasilkan nilai pada lapisan *output*.



Gambar 5. Diagram alir CNN-VGG16

Sementara pada Gambar 5, terlihat bahwa beberapa nama lapisan tidak tercantum. Hal ini bukan berarti lapisan-lapisan tersebut tidak ada, namun proses-proses yang sebelumnya telah dijelaskan akan tergabung pada satu lapisan. Sehingga satu lapisan dapat memuat lebih dari satu proses yang berlangsung. Dengan demikian, CNN-VGG16 memiliki jumlah lapisan sebanyak 16, di mana terdiri atas 13 lapisan konvolusi dan 3 lapisan *Fully Connected* yang dapat dilihat pula pada Gambar 5 secara visualnya. Lapisan Rectified Linear Unit (ReLU) pada Gambar 5 nampak tidak ada, sebab lapisan ini sudah sekaligus dijalankan prosesnya menjadi satu pada setiap lapisan konvolusi sebelum dilanjutkan ke tahap berikutnya.

Perancangan *model CNN-VGG16* dalam penelitian ini akan menggunakan input berukuran $224 \times 224 \times 3$, yang berarti memiliki panjang 224 piksel, lebar 224 piksel, dan *depth* 3, yang mewakili warna RGB. Selanjutnya, untuk jumlah filter, digunakan 64, 128, 256, dan 512 filter. *Kernel* yang digunakan berukuran 3×3 , dan lapisan *max pooling* memiliki ukuran 2×2 . Secara lengkap, perancangan algoritma CNN-VGG16 dalam penelitian ini dapat ditemukan dalam Tabel 2.

Model CNN-VGG16 pada penelitian ini menggunakan *Stochastic Gradient Descent* (SGD) sebagai *optimizer* dan untuk *loss function* menggunakan *Categorical Crossentropy*. Total parameter dalam *model* ini adalah sebanyak 35.626.053. Sedangkan lapisan/*layer Batch Normalization* berguna untuk membantu menjaga stabilitas dan kecepatan konvergensi selama pelatihan data oleh model terkait. Dan untuk *layer Dropout* berguna untuk mengurangi *overfitting*, yaitu kondisi di mana model terlalu menghafal data pelatihan dan tidak dapat menggeneralisasi dengan baik pada data baru. Nilai *Dropout* yang peneliti gunakan pada penelitian ini adalah sebesar 0.25.

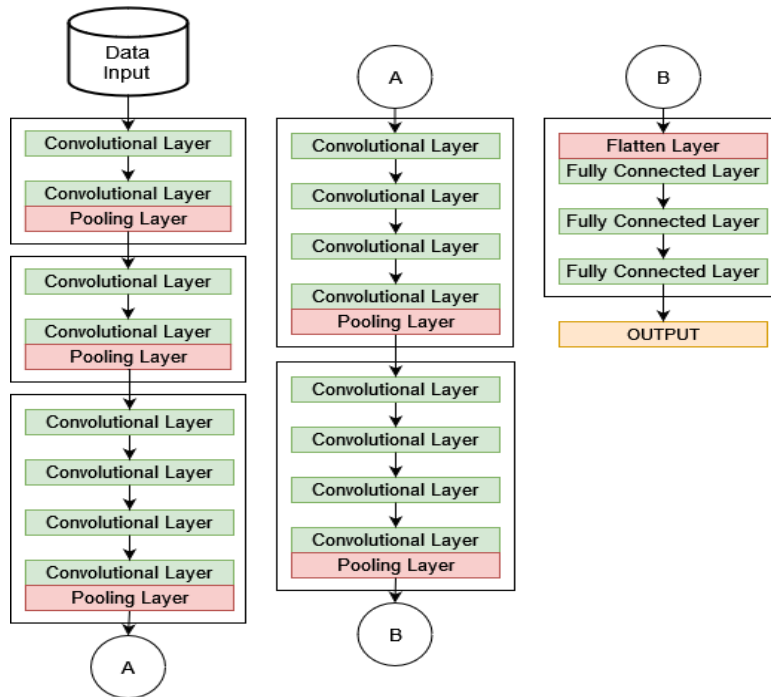
Tabel 2. Algoritma CNN-VGG16

<i>Layer</i>	<i>Output</i>
Input	224 x 224 x 3
64 Conv Filter dan Kernel (3x3)	224 x 244 x 64
64 Conv Filter dan Kernel (3x3)	224 x 244 x 64
Max Pooling (2x2)	224 x 244 x 64
128 Conv Filter dan Kernel (3x3)	224 x 244 x 128
128 Conv Filter dan Kernel (3x3)	224 x 244 x 128
Max Pooling (2x2)	224 x 244 x 128
256 Conv Filter dan Kernel (3x3)	224 x 244 x 256
256 Conv Filter dan Kernel (3x3)	224 x 244 x 256
256 Conv Filter dan Kernel (3x3)	224 x 244 x 256
Max Pooling (2x2)	224 x 244 x 256
256 Conv Filter dan Kernel (3x3)	224 x 244 x 256
256 Conv Filter dan Kernel (3x3)	224 x 244 x 256
256 Conv Filter dan Kernel (3x3)	224 x 244 x 256
Max Pooling (2x2)	224 x 244 x 256
512 Conv Filter dan Kernel (3x3)	224 x 244 x 512
512 Conv Filter dan Kernel (3x3)	224 x 244 x 512
512 Conv Filter dan Kernel (3x3)	224 x 244 x 512
Max Pooling (2x2)	224 x 244 x 512
Flatten	25088
Dense	1024
Dense	512
Batch Normalization	512
Dropout (rate=0.25)	512
Output	5

2.5.2. VGG19 Model

Proses yang terjadi dalam CNN-VGG19 pada dasarnya mengikuti prinsip yang serupa dengan CNN-VGG16 sebagaimana dijelaskan sebelumnya oleh peneliti. Namun, terdapat perbedaan signifikan pada jumlah lapisan yang dimiliki. Dalam struktur CNN-VGG19, terdapat tambahan lapisan konvolusi yang memungkinkan model untuk mengekstraksi fitur-fitur dengan tingkat kompleksitas yang lebih tinggi. CNN-VGG19 terdiri dari 19 lapisan, yang terdiri atas 16 lapisan konvolusi dan 3 lapisan *Fully Connected*. Lapisan konvolusi di dalam VGG19 berperan penting dalam mendeteksi pola-pola kompleks pada berbagai tingkat abstraksi. Selain itu, lapisan-lapisan konvolusi ini juga telah dilengkapi juga dengan *Rectified Linear Unit* (ReLU) sama seperti CNN-VGG16.

Tidak ada perbedaan yang signifikan antara perancangan model CNN-VGG16 dan CNN-VGG19 dalam penelitian ini. Perbedaan utama terletak pada jumlah lapisan yang digunakan. Model CNN-VGG19 memiliki total 19 lapisan, yang terdiri dari lapisan-lapisan yang sama seperti yang telah dijelaskan pada model CNN-VGG16 sebelumnya. Pada Tabel 3, peneliti telah paparkan secara lengkap algoritma CNN-VGG19 yang digunakan pada penelitian ini. Akibat dari perbedaan jumlah lapisan yang dimiliki, maka jumlah dari parameter model ini adalah 39.166.021.



Gambar 6. Diagram alir CNN-VGG19

Tabel 3. Algoritma CNN-VGG19

Layer	Output
Input	224 x 224 x 3
64 Conv Filter dan Kernel (3x3)	224 x 244 x 64
64 Conv Filter dan Kernel (3x3)	224 x 244 x 64
Max Pooling (2x2)	224 x 244 x 64
128 Conv Filter dan Kernel (3x3)	224 x 244 x 128
128 Conv Filter dan Kernel (3x3)	224 x 244 x 128
Max Pooling (2x2)	224 x 244 x 128
256 Conv Filter dan Kernel (3x3)	224 x 244 x 256
256 Conv Filter dan Kernel (3x3)	224 x 244 x 256
256 Conv Filter dan Kernel (3x3)	224 x 244 x 256
256 Conv Filter dan Kernel (3x3)	224 x 244 x 256
Max Pooling (2x2)	224 x 244 x 256
256 Conv Filter dan Kernel (3x3)	224 x 244 x 256
256 Conv Filter dan Kernel (3x3)	224 x 244 x 256
256 Conv Filter dan Kernel (3x3)	224 x 244 x 256
256 Conv Filter dan Kernel (3x3)	224 x 244 x 256
Max Pooling (2x2)	224 x 244 x 256
512 Conv Filter dan Kernel (3x3)	224 x 244 x 512
512 Conv Filter dan Kernel (3x3)	224 x 244 x 512
512 Conv Filter dan Kernel (3x3)	224 x 244 x 512
512 Conv Filter dan Kernel (3x3)	224 x 244 x 512
Max Pooling (2x2)	224 x 244 x 512
Flatten	25088
Dense	1024
Dense	512
Batch Normalization	512
Dropout (rate=0.25)	512
Output	5

2.6. Evaluasi

Pada tahap ini, peneliti memilih untuk menggunakan metode *Confusion Matrix* juga mencakup empat metrik evaluasi kinerja, yaitu *Accuracy*, *Precision*, *Recall*, dan *F1-Score*, untuk mengevaluasi kinerja klasifikasi pada 5 kelas berbeda yang mewakili varietas beras. Selain itu, terdapat istilah-istilah penting lainnya, yaitu *True Positive* (TP), *True Negative* (TN), *False Positive* (FP), dan *False Negative* (FN) [13], yang digunakan untuk merepresentasikan hasil dari proses pembelajaran klasifikasi. Dengan istilah-istilah ini, diharapkan dapat mengidentifikasi berapa banyak hasil prediksi yang benar (positif dan negatif) dan berapa banyak yang salah dari model yang telah melalui proses pembelajaran/*training*.

Accuracy merupakan nilai yang mengukur sejauh mana *model* klasifikasi dapat memprediksi dengan benar label kelas dari data uji. Dalam konteks klasifikasi, *accuracy* mengukur jumlah prediksi yang benar baik positif maupun negatif (*True Positive* (TP)/*True Negative* (TN)) dibagi dengan jumlah total data uji. Jika dituliskan secara matematis dapat dilihat pada persamaan 1.

$$accuracy = \frac{TP+TN}{TP+TN+FP+FN} \quad (1)$$

Precision merupakan nilai untuk mengukur sejauh mana model dapat mengidentifikasi dengan benar kasus positif dari semua kasus yang diprediksi sebagai positif. Jika dituliskan secara matematis dapat dilihat pada persamaan 2.

$$precision = \frac{TP}{TP+FP} \quad (2)$$

Recall (sensitivitas) merupakan nilai untuk mengukur sejauh mana model dapat mengidentifikasi dengan benar semua kasus positif. Jika dituliskan secara matematis dapat dilihat pada persamaan 3.

$$recall = \frac{TP}{TP+FN} \quad (3)$$

F1-Score merupakan metrik yang menggabungkan *Precision* dan *Recall* menjadi satu angka tunggal untuk memberikan gambaran keseluruhan tentang kinerja model. Jika dituliskan secara matematis dapat dilihat pada persamaan 4.

$$f1 - score = 2 \times \frac{precision \times recall}{precision + recall} \quad (4)$$

2.7. Analisis Performa

Pada tahap ini, peneliti melakukan perbandingan antara nilai-nilai metrik yang telah ditemukan seperti *accuracy*, *precision*, *recall*, dan *f1-score*. Di mana sebelumnya telah dilakukan pada tahap evaluasi selama analisis performa setiap *model*. Peneliti memulai dengan menggambarkan hasil perbandingan ini untuk memberikan pemahaman lebih lanjut tentang kinerja masing-masing *model*, baik untuk algoritma *CNN-VGG16* maupun *CNN-VGG19*.

3. Hasil dan Pembahasan

Hasil yang telah ada, diperoleh setelah melakukan beberapa langkah yang telah dipaparkan pada subbab sebelumnya. Beberapa proses yang melibatkan penggunaan bahasa pemrograman *Python* pada penelitian ini, dilakukan dengan memanfaatkan bantuan perangkat lunak/*software* bernama Jupyter-Lab. Peneliti mencoba membandingkan nilai hasil data *training* dengan jumlah *epoch* sebanyak 5, 10, 15, 20, 25, dan 30. Dengan tujuan untuk melihat pada jumlah *epoch* beberapa, *model* mencapai performa terbaiknya. Setelah mengetahui *model* terbaik dari setiap algoritma, selanjutnya peneliti melakukan perbandingan antara nilai-nilai dalam metrik *Confusion Matrix* dari masing-masing algoritma. Untuk mengetahui performa algoritma mana yang lebih unggul dalam mengklasifikasikan varietas beras.

3.1. Hasil Proses *Training Model***Tabel 4.** Hasil proses *training model CNN-VGG16*

<i>Epoch</i>	<i>Average duration per Step (s)</i>	<i>Accuracy</i>	<i>Loss</i>	<i>Validation Accuracy</i>	<i>Validation Loss</i>
5	2608	0,9089	0,2900	0,5867	1,1597
10	2341	0,9350	0,1907	0,7810	0,6993
15	2341	0,9557	0,1210	0,8743	0,4647
20	2341	0,9654	0,1069	0,9200	0,2279
25	2341	0,9714	0,0796	0,9276	0,1809
30	2341	0,9757	0,0699	0,9581	0,1013

Tabel 5. Hasil proses *training model CNN-VGG19*

<i>Epoch</i>	<i>Average duration per Step (s)</i>	<i>Accuracy</i>	<i>Loss</i>	<i>Validation Accuracy</i>	<i>Validation Loss</i>
5	2562	0,8886	0,3253	0,5962	1,2456
10	2802	0,9307	0,2046	0,7905	0,8318
15	2698	0,9525	0,1242	0,7429	0,5536
20	2668	0,9604	0,0642	0,8648	0,3102
25	2628	0,9682	0,0414	0,8705	0,3409
30	2603	0,9707	0,0317	0,9486	0,1066

Hasil proses *training* dari setiap model, baik *CNN-VGG16* dan *CNN-VGG19* pada *epoch* 5, 10, 15, 20, 25, dan 30 dapat dilihat melalui Tabel 4 dan 5. Di mana pada kedua tabel yang sebelumnya telah disebutkan, memiliki istilah yaitu *epoch*. *Epoch* merupakan satu kali proses lengkap di mana seluruh dataset pelatihan digunakan untuk melatih suatu model jaringan saraf [14]. Proses pelatihan pada umumnya melibatkan beberapa *epoch* untuk memungkinkan model mempelajari pola-pola dalam data dengan lebih baik. Sedangkan untuk *loss* mengacu pada nilai seberapa baik *model* dalam memprediksi *output* yang benar untuk *input* tertentu pada *dataset* pelatihan. *Neural network* pada dasarnya melakukan proses belajar untuk meminimalkan jarak antara label yang diprediksi dan label sebenarnya. Untuk mengurangi jarak ini, *neural network* melakukan proses belajar secara berkali-kali dengan menyesuaikan nilai bobot dan bias agar mengurangi nilai *loss* [15].

Melalui Tabel 4 di atas, nilai rata-rata lama proses *training* per-*epoch* hanya pada *epoch* 5 saja yang berbeda, yaitu sekitar 2608 detik, lebih lama dibandingkan dengan rata-rata pada jumlah *epoch* lainnya. Sementara untuk *epoch* 10, 15, 20, 25, dan 30 memiliki rata-rata yang sama, yaitu sekitar 2341 detik untuk setiap *epoch*. Tingkat akurasi *train* dan *validation* antar *epoch* tidak memiliki perbedaan yang signifikan. Namun, perlu diperhatikan bahwa tingkat *loss validation* pada *epoch* 5 memiliki kesenjangan yang cukup besar dibandingkan dengan *epoch* yang lain. Secara keseluruhan, *model CNN-VGG16* telah menunjukkan performa yang baik dalam mengklasifikasikan varietas beras. Tingkat akurasi yang tinggi pada data latih dan validasi serta penurunan *validation loss* seiring berjalannya waktu menunjukkan bahwa model memiliki potensi yang kuat untuk digunakan dalam tugas pengklasifikasian varietas beras.

Melalui Tabel 5 di atas menunjukkan hasil pelatihan *model CNN-VGG19*, yang merupakan kelanjutan dari percobaan yang telah dilakukan pada *model CNN-VGG16* sebagaimana yang telah dijelaskan dalam Tabel 4 sebelumnya. Terdapat beberapa perbedaan yang menarik antara kedua *model* ini. Dalam hal durasi per-*epoch*, *model CNN-VGG19* tidak menunjukkan konsistensi yang lebih baik daripada model sebelumnya. *CNN-VGG19* memiliki durasi yang lebih beragam selama proses pelatihan. Hal ini dapat mengindikasikan bahwa model ini tidak lebih baik pada *model CNN-VGG16*.

Pada akurasi, *model CNN-VGG19* menunjukkan kinerja yang kuat. Pada akhir pelatihan pada *epoch* ke-30, model ini mencapai akurasi yang sangat tinggi sekitar 99.07% pada data latih. Dan untuk nilai *validation loss* pada *epoch* ke-30, menunjukkan angka yang cukup rendah yaitu sekitar 0.1066.

Data yang tertulis pada Tabel 4 dan 5 di atas merupakan nilai akurasi dan *loss* yang muncul saat pertama kali *model neural network* dijalankan. Nilai-nilai tersebut terus berubah apabila proses *training* dijalankan, sebab data *training* dan *validation* yang digunakan setiap proses *training* selalu dipilih secara acak. Sebagai contoh, pada data latih pertama, data yang digunakan mungkin nomor 1, 3, 6, 9, 12, dan seterusnya. Dan untuk data tesnya mungkin nomor 21, 24, 27, 28, dan seterusnya. Pada percobaan kedua, data latih yang diambil adalah nomor 2, 5, 8, 11, dan seterusnya. Sedangkan untuk data tesnya 8, 11, 15, 18, dan seterusnya. Hal ini yang membuat data latih tidak beraturan atau acak. Nilai akurasi dan *loss* dapat lebih tinggi dan bahkan tidak menutup kemungkinan dapat lebih rendah dibandingkan dengan proses sebelumnya [16].

3.2. Hasil *Confusion Matrix* dan *Classification Report Model*

3.2.1. *Confusion Matrix Model*

Berikut merupakan nilai dari *confusion matrix* kedua model yang sebelumnya telah dilakukan proses *training* yang disajikan dengan menggunakan tabel, yang dapat dilihat pula pada Tabel 6. Pada Tabel 6, disajikan nilai dari *confusion matrix* untuk dua model yang diuji, yaitu VGG16 dan VGG19. Matriks kebingungannya menggambarkan kinerja model dalam mengklasifikasikan sampel-sampel ke dalam kelas yang benar. Kedua model, baik VGG16 maupun VGG19, diperoleh hasil yang menunjukkan performa yang sangat baik. Dari matriks kebingungannya, terlihat bahwa kedua model tidak menghasilkan *False Positive* (FP) atau *False Negative* (FN) untuk kelas apapun. Hal ini dapat diartikan bahwa tidak ada kasus di mana model memprediksi kelas yang sebenarnya salah. Selain itu, keduanya memiliki nilai *True Positive* (TP) yang tinggi, menandakan bahwa kedua model sangat mampu mengidentifikasi dan mengklasifikasikan sampel yang sebenarnya termasuk dalam kelas tertentu. Demikian pula, *True Negative* (TN) juga menunjukkan bahwa kedua model dapat membedakan dengan benar sampel yang tidak termasuk dalam kelas tertentu. Secara keseluruhan, hasil dari *confusion matrix* menunjukkan bahwa baik model VGG16 maupun VGG19 memberikan performa yang baik dalam tugas klasifikasi yang diuji. Kedua model ini memiliki tingkat akurasi yang tinggi dan mampu membedakan dengan baik antara berbagai kelas yang diuji.

Tabel 6. Nilai *confusion matrix model*

	VGG16	VGG19
True Positive (TP)	34	35
False Positive (FP)	0	0
False Negative (FN)	0	0
True Negative (TN)	34	35

Tabel 7. *Classification report model CNN-VGG16*

	<i>precision</i>	<i>recall</i>	<i>f1-score</i>	<i>support</i>
'Arborio' : 0	1.00	0.97	0.99	35
'Basmati' : 1	0.94	0.97	0.96	35
'Ipsala' : 2	1.00	1.00	1.00	35
'Jasmine' : 3	0.97	0.94	0.96	35
'Karacadag' : 4	0.97	1.00	0.99	35
accuracy			0.98	175
macro avg	0.98	0.98	0.98	175
weighted avg	0.98	0.98	0.98	175

Tabel 8. *Classification report model CNN-VGG19*

	<i>precision</i>	<i>recall</i>	<i>f1-score</i>	<i>support</i>
'Arborio' : 0	0.97	1.00	0.99	35
'Basmati' : 1	0.90	1.00	0.95	35
'Ipsala' : 2	1.00	1.00	1.00	35
'Jasmine' : 3	1.00	0.89	0.94	35
'Karacadag' : 4	1.00	0.97	0.99	35
accuracy			0.97	175
macro avg	0.97	0.97	0.97	175
weighted avg	0.97	0.97	0.97	175

3.2.2. Classification Report Model

Setelah melakukan proses *training* pada setiap *model CNN* dan mengamati hasil pada Tabel 4 dan 5, serta melihat nilai-nilai *confusion matrix* pada Tabel 6 di atas, nilai yang akan digunakan adalah hasil dari *training* pada *epoch* ke-30, baik pada *model CNN-VGG16* maupun *CNN-VGG19*. Berikut ini adalah *classification report* dari setiap *model* menggunakan bantuan *library 'sklearn.metrics'* yang terdapat pada Tabel 7 dan 8 di bawah ini.

Tabel 7 menunjukkan nilai-nilai dari variabel *precision*, *recall*, *f1-score*, dan *support* untuk *model VGG16*. Beberapa kelas menunjukkan performa yang baik karena nilainya mendekati angka 1.00 dan bahkan ada juga pada nilai-nilai di kelas tertentu yang memiliki nilai 1.00. Dapat dilihat bahwa beberapa kelas memiliki performa yang sangat baik, dengan nilai *precision* dan *recall* mendekati 1.00. Sebagai contoh, kelas 'Ipsala' memiliki *precision*, *recall*, dan *f1-score* sebesar 1.00, menunjukkan bahwa model dengan sempurna mengenali dan mengklasifikasikan semua sampel dari kelas tersebut. Hasil yang baik juga terlihat pada kelas 'Arborio' dengan nilai *precision* sekitar 1.00 dan *recall* sekitar 0,97. Secara keseluruhan, *model CNN-VGG16* menunjukkan performa yang sangat baik dengan akurasi sekitar 0.98, yang berarti sekitar 98% dari prediksi *model* benar. Hasil ini menunjukkan bahwa *model* ini efektif dalam mengklasifikasikan berbagai varietas beras.

Tabel 8 menunjukkan hasil dari evaluasi klasifikasi untuk *model CNN-VGG19*. Pada keseluruhan performa *model*, dapat dilihat bahwa *model* ini juga cukup memberikan hasil yang baik. Beberapa kelas menunjukkan performa yang hampir sempurna, terutama 'Ipsala' yang memiliki nilai *precision*, *recall*, dan *f1-score* sebesar 1.00. Dapat diartikan bahwa, *model* dengan sempurna mengenali dan mengklasifikasikan semua sampel dari kelas tersebut. Kelas 'Arborio' dan 'Basmati' juga memiliki hasil yang sangat baik dengan nilai *recall* dan *f1-score* mendekati 1.00. Namun, ada beberapa kelas seperti 'Jasmine' dan 'Karacadag' yang memiliki perbedaan antara *precision* dan *recall*. Ini menandakan bahwa model cenderung lebih baik dalam mengklasifikasikan kelas-kelas lain dibandingkan dengan kelas 'Jasmine' dan 'Karacadag'. Kelas 'Jasmine' memiliki *recall* sedikit lebih rendah, sehingga *model* mungkin membutuhkan peningkatan dalam mengidentifikasi semua kasus 'Jasmine'. Secara keseluruhan, akurasi *model* ini adalah sekitar 0.97, yang berarti sekitar 97% dari prediksi *model* benar. Hasil ini mengindikasikan bahwa *model CNN-VGG19* cukup efektif dalam mengklasifikasikan berbagai varietas beras.

3.3. Analisis Performa Model

Tabel 9. Perbandingan nilai *precision* model VGG16 dan VGG19

	VGG16	VGG19	Gap
'Arborio' : 0	1.00	0.97	0.3
'Basmati' : 1	0.94	0.90	0.4
'Ipsala' : 2	1.00	1.00	0
'Jasmine' : 3	0.97	1.00	0.3
'Karacadag' : 4	0.97	1.00	0.3

Tabel 10. Perbandingan nilai *recall* model VGG16 dan VGG19

	VGG16	VGG19	Gap
'Arborio' : 0	0.97	1.00	0.3
'Basmati' : 1	0.97	1.00	0.3
'Ipsala' : 2	1.00	1.00	0
'Jasmine' : 3	0.94	0.89	0.5
'Karacadag' : 4	1.00	0.97	0.3

Setelah memeriksa performa dari kedua model, seperti yang dapat diamati pada Tabel 7 dan 8 di atas, peneliti akan melakukan analisis lebih lanjut dengan membandingkan nilai-nilai dari *confusion matrix*. *Confusion matrix* meliputi metrik-metrik penting seperti *accuracy*, *precision*, *recall*, dan *f1-score*. Dengan membandingkan nilai-nilai dari *confusion matrix* ini, peneliti dapat membuat keputusan yang lebih terarah mengenai pilihan *model* terbaik untuk tugas klasifikasi jenis padi ini. Hal ini penting untuk memastikan bahwa *model* yang dipilih mampu memberikan hasil yang konsisten dan akurat dalam aplikasi praktis.

Dalam tabel 9, terdapat perbandingan nilai *precision* antara *model* VGG16 dan VGG19 untuk lima kelas beras yang berbeda. Dalam konteks ini, nilai *precision* berkisar antara 0 hingga 1, di mana apabila nilai semakin mendekati nilai 1.00 menunjukkan kinerja yang lebih baik. Berdasarkan hasil yang tercatat dalam tabel, dapat diamati bahwa pada semua kelas beras, *model* VGG19 menunjukkan nilai *precision* yang lebih tinggi dibandingkan dengan VGG16. Sebagai contoh, untuk kelas 'Arborio', VGG16 memiliki nilai *precision* sebesar 0.97, sedangkan VGG19 memiliki nilai *precision* 1.00. Hal serupa juga terlihat pada kelas 'Basmati', di mana VGG16 memiliki *precision* sekitar 0.94 sedangkan VGG19 memiliki nilai yang lebih tinggi yaitu 0.90. Fenomena ini juga teramati pada kelas-kelas lainnya.

Tabel 11. Perbandingan nilai *f1-score* model VGG16 dan VGG19

	VGG16	VGG19	Gap
'Arborio' : 0	0.99	0.99	0
'Basmati' : 1	0.96	0.95	0.1
'Ipsala' : 2	1.00	1.00	0
'Jasmine' : 3	0.96	0.94	0.2
'Karacadag' : 4	0.99	0.99	0

Tabel 12. Perbandingan nilai *accuracy* model VGG16 dan VGG19

	VGG16	VGG19
<i>accuracy</i>	0.98	0.97

Hal ini mengindikasikan bahwa *model VGG19* cenderung memberikan hasil yang lebih akurat dalam mengklasifikasikan varietas beras yang diuji dalam eksperimen ini jika dibandingkan dengan *VGG16*. Oleh karena itu, berdasarkan nilai *precision*, dapat disimpulkan bahwa *model VGG19* menunjukkan performa yang lebih baik dibandingkan dengan *VGG16* dalam tugas klasifikasi varietas beras ini.

Dalam Tabel 10, terdapat perbandingan nilai *recall* antara *model VGG16* dan *VGG19* untuk lima kelas beras yang berbeda. Seperti *precision*, nilai *recall* juga berkisar antara 0 hingga 1, dan semakin mendekati 1 menunjukkan kinerja yang lebih baik. Berdasarkan hasil yang tercatat dalam tabel, dapat diamati bahwa kinerja *VGG16* dan *VGG19* berbeda tergantung pada kelas beras yang diuji. Sebagai contoh, pada kelas 'Jasmine', *VGG16* memiliki nilai *recall* sebesar 0.94, sementara *VGG19* memiliki nilai *recall* 0.89. Hal ini menunjukkan bahwa *VGG16* mampu mengenali lebih banyak kasus positif 'Jasmine' dibandingkan *VGG19*. Namun, pada kelas 'Basmati' dan 'Ipsala', kedua model menunjukkan hasil *recall* yang serupa, yaitu 1.00, menunjukkan kemampuan yang sangat baik dalam mengenali kasus positif. Khusus untuk kelas 'Karacadag', *VGG16* memiliki *recall* sekitar 1.00 sedangkan *VGG19* memiliki *recall* 0.97. Meskipun keduanya memiliki kinerja yang tinggi, *VGG16* mungkin sedikit lebih unggul dalam mengenali kasus positif untuk kelas ini.

Secara keseluruhan, berdasarkan nilai *recall*, tidak ada *model* yang secara konsisten lebih unggul dibandingkan yang lain. Performa keduanya cenderung serupa dan tergantung pada kelas beras yang diuji. Tetapi apabila dilihat pada *gap* dari nilai *recall* kedua *model*, *VGG16* dapat dikatakan memiliki performa yang lebih baik, walaupun hanya sedikit *gap* nilainya.

Dalam Tabel 11, terdapat perbandingan nilai *f1-score* antara *model VGG16* dan *VGG19* untuk lima kelas padi yang berbeda. Berdasarkan hasil yang tercatat dalam tabel, dapat diamati bahwa performa *VGG16* dan *VGG19* tampak sangat serupa dalam klasifikasi jenis padi ini. Secara keseluruhan, tidak terdapat perbedaan yang signifikan antara keduanya dalam hal *f1-score* untuk setiap kelas. Semua *gap* nilai *f1-score* antara keduanya adalah kurang dari 0.2, menunjukkan bahwa keduanya memiliki kinerja yang serupa. Walau dengan kondisi seperti ini, *gap* yang dimiliki kedua model juga dapat terlihat jelas bahwa nilai-nilai *f1-score* yang lebih mendekati angka satu pada setiap kelasnya adalah *model VGG16*. Maka dari itu, dapat disimpulkan bahwa *model VGG16* pada nilai *f1-score* lebih baik dibandingkan dengan *model VGG19*.

Dalam Tabel 12, terdapat perbandingan nilai akurasi (*accuracy*) antara *model VGG16* dan *VGG19*. Akurasi mengukur sejauh mana *model* dapat melakukan klasifikasi dengan benar secara keseluruhan. Berdasarkan hasil yang tercatat dalam tabel, dapat diamati bahwa *VGG16* memiliki tingkat akurasi sebesar 98%, sementara *VGG19* memiliki tingkat akurasi 97%. Hal ini menunjukkan bahwa *VGG16* memiliki sedikit keunggulan dalam hal akurasi dibandingkan dengan *VGG19*.

4. Diskusi

Berdasarkan hasil dan pembahasan terkait perbandingan performa antara algoritma *VGG16* dan *VGG19* dalam proses mengklasifikasikan varietas beras dengan lima kelas, dapat disimpulkan bahwa keduanya menghasilkan hasil yang memuaskan. Proses *training model Convolutional Neural Network* (CNN) dilakukan menggunakan *dataset* dengan perbandingan 8:1.5:0.5, atau dengan total 3500 citra gambar. Hasil evaluasi menunjukkan bahwa algoritma *VGG16* mencapai tingkat akurasi sebesar 98% dengan waktu *training* selama 73,405 detik. Sementara itu, algoritma *VGG19* mencapai tingkat akurasi sebesar 97% dengan waktu *training* 78,098 detik. Selain akurasi, nilai-nilai metrik lain seperti *precision*, *recall*, dan *f1-score* juga dievaluasi. Algoritma *VGG19* menonjol pada nilai *precision*, yang mengukur akurasi dari hasil klasifikasi positif terhadap seluruh prediksi positif yang sebenarnya. Namun, dalam hal *recall* dan *f1-score*, algoritma *VGG16* menunjukkan performa lebih baik. Di mana *recall* merupakan nilai yang mengukur kemampuan model untuk menemukan kembali seluruh kasus positif yang sebenarnya dan untuk *f1-score* merupakan metrik yang menilai keseimbangan antara *precision* dan *recall* dari suatu *model*.

Hasil penelitian menunjukkan bahwa tidak selalu arsitektur yang memiliki lapisan yang lebih kompleks atau jumlah lapisan yang lebih banyak akan memberikan performa yang lebih baik. Dalam

penelitian ini, VGG16 terbukti memiliki performa yang lebih unggul dibandingkan dengan VGG19. Faktor-faktor beragam dapat mempengaruhi hasil akhir perbandingan dua algoritma ini. Salah satu faktor yang paling berpengaruh adalah ketersediaan data atau dataset seperti yang disebutkan oleh peneliti sebelumnya. Jumlah data memiliki dampak yang signifikan terhadap pilihan algoritma yang tepat. Oleh karena itu, dalam kasus ini VGG16 dapat dianggap lebih dapat diandalkan dengan jumlah data sekitar 3500 citra gambar, karena memiliki kecenderungan *overfitting* yang lebih rendah dibandingkan dengan VGG19.

5. Kesimpulan

Hasil penelitian ini menunjukkan bahwa dalam kasus mengklasifikasikan varietas beras dengan lima kelas, algoritma VGG16 telah terbukti lebih unggul daripada algoritma VGG19. Meskipun model VGG16 memiliki jumlah lapisan yang lebih sedikit dibandingkan VGG19, namun hal ini tidak selalu menjadi indikator mutlak. Pada kasus ini, VGG16 menonjol dengan akurasi mencapai 98%, sedangkan VGG19 mencapai akurasi sebesar 97%. Selain *accuracy*, kinerja model juga diukur dengan metrik-metrik lain seperti *precision*, *recall*, dan *f1-score*. Meskipun VGG19 menunjukkan keunggulan pada nilai *precision*, VGG16 tampil lebih baik dalam hal *recall* dan *f1-score*. Keunggulan performa VGG16 mungkin disebabkan oleh kompleksitas model yang lebih rendah dan kemampuannya untuk menghindari *overfitting* pada dataset yang tersedia. Jumlah lapisan yang lebih sedikit mungkin memungkinkan VGG16 untuk lebih efisien dalam mempelajari fitur-fitur penting dari gambar beras. Oleh karena itu, dalam konteks penelitian ini, VGG16 terbukti menjadi pilihan arsitektur yang lebih optimal dibandingkan dengan VGG19.

Referensi

- [1] A. R. Sari, Y. Martono, and F. S. Rondonuwu, "Identifikasi Kualitas Beras Putih (*Oryza sativa* L.) Berdasarkan Kandungan Amilosa dan Amilopektin di Pasar Tradisional dan 'Selepan' Kota Salatiga," *Titian Ilmu J. Ilm. Multi Sci.*, vol. 12, no. 1, pp. 24–30, 2020, doi: 10.30599/jti.v12i1.599.
- [2] H. Windiyani, "Keragaman Varietas Unggul Baru Padi Fungsional Mendukung Ketahanan Pangan dalam Menghadapi Pandemi COVID-19," in *Komoditas Sumber Pangan untuk Meningkatkan Kualitas Kesehatan di Era Pandemi Covid-19*, 2020, pp. 449–456. [Online]. Available: <http://conference.unsri.ac.id/index.php/lahansuboptimal/article/view/1964/0>
- [3] G. Prayitno, *Ruang Berketahanan Pangan: Menjawab Tantangan Produksi Pangan Berkelanjutan dengan Optimasi Keruangan Menuju Indonesia Berdaulat*, 1st ed. Malang: UB Press, 2022. [Online]. Available: <https://edeposit.perpusnas.go.id/collection/ruang-berketahanan-pangan-sumber-elektronis-menjawab-tantangan-produksi-pangan-berkelanjutan-dengan-optimasi-keruangan-menuju-indonesia-berdaulat/112492#>
- [4] A. Ahdiat, "Indonesia Impor Beras 429 Ribu Ton Pada 2022, Ini Negara Asalnya," 2023. <https://databoks.katadata.co.id/datapublish/2023/04/14/indonesia-impor-beras-429-ribu-ton-pada-2022-ini-negara-asalnya#:~:text=Menurut laporan Badan Pusat Statistik,Myanmar seperti terlihat pada grafik.>
- [5] D. Iswantoro and D. Handayani UN, "Klasifikasi Penyakit Tanaman Jagung Menggunakan Metode Convolutional Neural Network (CNN)," *J. Ilm. Univ. Batanghari Jambi*, vol. 22, no. 2, p. 900, 2022, doi: 10.33087/jiubj.v22i2.2065.
- [6] F. S. Pamungkas, B. D. Prasetya, and I. Kharisudin, "Perbandingan Metode Klasifikasi Supervised Learning pada Data Bank Customers Menggunakan Python," *Prism. Pros. Semin. Nas. Mat.*, vol. 3, pp. 689–694, 2020, [Online]. Available: <https://journal.unnes.ac.id/sju/index.php/prisma/article/view/37875>
- [7] A. Mulyanto, E. Susanti, F. Rossi, W. Wajiran, and R. I. Borman, "Penerapan Convolutional Neural

- Network (CNN) pada Pengenalan Aksara Lampung Berbasis Optical Character Recognition (OCR)," *J. Edukasi dan Penelit. Inform.*, vol. 7, no. 1, p. 52, 2021, doi: 10.26418/jp.v7i1.44133.
- [8] K. Simonyan and A. Zisserman, "Very Deep Convolutional Networks for Large-Scale Image Recognition," *3rd Int. Conf. Learn. Represent. ICLR 2015 - Conf. Track Proc.*, no. 6, pp. 1–14, Sep. 2014, [Online]. Available: <https://arxiv.org/abs/1409.1556>
- [9] E. Tanuwijaya and A. Roseanne, "Modifikasi Arsitektur VGG16 untuk Klasifikasi Citra Digital Rempah-Rempah Indonesia," *MATRIK J. Manajemen, Tek. Inform. dan Rekayasa Komput.*, vol. 21, no. 1, pp. 189–196, 2021, doi: 10.30812/matrik.v21i1.1492.
- [10] M. Koklu, I. Cinar, and Y. S. Taspinar, "Classification of rice varieties with deep learning methods," *Comput. Electron. Agric.*, vol. 187, p. 106285, Aug. 2021, doi: 10.1016/j.compag.2021.106285.
- [11] A. Huda, *DASAR-DASAR PEMROGRAMANBERBASIS PYTHON*, 1st ed. Padang: Tim editor UNP Press, 2020.
- [12] M. Ezar Al Rivan and A. Giovri Riyadi, "Perbandingan Arsitektur LeNet dan AlexNet Pada Metode Convolutional Neural Network Untuk Pengenalan American Sign Language," *J. Komput. Terap.*, vol. 7, no. Vol. 7 No. 1 (2021), pp. 53–61, 2021, doi: 10.35143/jkt.v7i1.4489.
- [13] T. Dwi Antoko, M. Azhar Ridani, and A. Eko Minarno, "Klasifikasi Buah Zaitun Menggunakan Convolution Neural Network," *Komputika J. Sist. Komput.*, vol. 10, no. 2, pp. 119–126, 2021, doi: 10.34010/komputika.v10i2.4475.
- [14] F. Masykur, M. B. Setyawan, and K. Winangun, "Epoch Optimization on Rice Leaf Image Classification Using Convolutional Neural Network (CNN) MobileNet," *CESS (Journal Comput. Eng. Syst. Sci.)*, vol. 7, no. 2, p. 581, 2022, doi: 10.24114/cess.v7i2.37336.
- [15] M. Tang, A. Djelouah, F. Perazzi, Y. Boykov, and C. Schroers, "Normalized Cut Loss for Weakly-Supervised CNN Segmentation," *Proc. IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recognit.*, pp. 1818–1827, 2018, doi: 10.1109/CVPR.2018.00195.
- [16] A. Khamparia, D. Gupta, V. H. C. de Albuquerque, A. K. Sangaiah, and R. H. Jhaveri, "Internet of health things-driven deep learning system for detection and classification of cervical cells using transfer learning," *J. Supercomput.*, vol. 76, no. 11, pp. 8590–8608, Nov. 2020, doi: 10.1007/s11227-020-03159-4.



© 2019 by the authors. Submitted for possible open access publication under the terms and conditions of the Creative Commons Attribution (CC BY) license (<http://creativecommons.org/licenses/by/4.0/>).

Enhancing Heart Disease Detection Using Convolutional Neural Networks and Classic Machine Learning Methods

Sri Hasta Mulyani^{1*}, Nurhadi Wijaya², and Fike Trinidya³

¹⁻³ Department of Informatics, Universitas Respati Yogyakarta, Yogyakarta, Indonesia

* Correspondence: hasta@respati.ac.id

Abstract: This study addresses the problem of heart disease detection, a critical concern in public health. The research aims to compare the performance of Convolutional Neural Networks (CNN) with conventional machine learning algorithms in diagnosing heart disease using a dataset comprising 14 features. The primary objective is to determine whether CNNs can provide more accurate and reliable results than traditional techniques. The research employs rigorous preprocessing, normalizing relevant features, and splits the dataset into an 80-20 training-testing split. The model is trained for 300 epochs with a batch size of 64, and performance evaluation is conducted using confusion matrices and classification reports. The results reveal that the CNN model achieved a remarkable accuracy of 100%, demonstrating its potential to outperform conventional machine learning algorithms. These findings emphasize the significance of deep learning techniques in improving heart disease diagnostics, although further research is needed to optimize CNN models and address interpretability concerns for practical implementation in healthcare settings.

Keywords: Convolutional Neural Networks; Diagnostic Accuracy; Healthcare Applications; Heart Disease Detection; Machine Learning Algorithms.

1. Introduction

Early detection of heart disease can contribute significantly to public health by enabling timely treatment and reducing mortality. Machine learning techniques, such as Logistic Regression (LR), K-Nearest Neighbor, Decision Tree, Naive Bayes, Random Forest, and Support Vector Machine, have been applied to develop heart disease diagnosis models [1]. These models use various patient characteristics, such as age, chest pain, blood pressure, gender, cholesterol, and heart rate, to identify high-risk individuals and distinguish between patients with and without heart disease [2]. In addition, computer vision and machine learning methods have been used to analyze irises for early detection of heart disease [3]. The proposed system achieved high accuracy in detecting potential heart disease using iridology [4]. By accurately detecting heart disease and supporting doctors in decision-making, this machine learning approach can improve patient outcomes and contribute to public health [5].

Early detection of heart disease is an essential concern in the medical field as it can prevent fatal outcomes and save lives [6]. Heart disease is the leading cause of death worldwide, and detecting it early allows for timely intervention and treatment [7]. By analyzing risk factors and using machine learning techniques, medical professionals can improve decision-making and enhance early detection of heart disease. Deep learning networks and remote diagnostic systems can also support doctors in diagnosing heart disease more effectively. Early detection allows medical practitioners to implement appropriate interventions, such as lifestyle changes, medication, or surgery, to prevent further

complications and improve patient outcomes. Therefore, early detection of heart disease is critical to reducing mortality rates and improving the overall health of individuals.

Heart disease detection methods include analyzing electrocardiogram (ECG) signals, listening to heart sounds, and using medical tests and imaging techniques such as CT scans. Machine learning (ML) and deep learning (DL) models have been used for heart disease classification from ECG signals, with the Generative Adversarial Network (GAN) model and GAN-LSTM ensemble model showing promising results in handling imbalanced data and achieving higher accuracy [8]. Deep learning techniques have also been applied to analyze heartbeat audio recordings, combining convolutional and recurrent neural network models to improve accuracy [9]. Feature extraction methods using ECG signals, such as discrete wavelet transform (DWT) and SVM classifiers, are applicable in early heart disease detection [10]. Machine learning algorithms such as Random Forest (RF), SVM, LR, and KNN have been used with feature selection methods to diagnose heart disease, with RF and RFFS yielding the highest accuracy [11]. Data mining methods, including decision trees, KNN classifiers, naive Bayes, random forests, and SVM, have also been used for heart disease diagnosis, with SVM showing the best performance [12].

Convolutional Neural Network (CNN) was chosen as the primary approach in this study due to its potential for early detection of heart disease and its ability to analyze medical images [13]. CNN architecture can assist in interpreting medical images related to heart disease by utilizing deep learning techniques to process large data sets and accurately predict heart disease [14]. CNN models have been successfully used for left ventricle segmentation in cardiac MRI, providing clinical measurements [15]. The study proposed a CNN-based Fully Connected layer architecture for automatic classification of ECG signals into specific classes, achieving high accuracy results [16]. CNNs have also been widely used in medical image analysis, enabling significant advances in computer-aided diagnosis in image classification, segmentation, detection, and more.

CNNs have several advantages over conventional machine learning techniques in the context of heart disease detection. First, CNNs can automatically learn relevant features from raw data, such as ECG signals, without manual feature engineering [17]. This allows CNNs to capture complex patterns and relationships in the data, improving disease classification accuracy. In addition, CNNs are highly effective in handling high-dimensional data, such as ECG signals, due to their ability to learn hierarchical representations [18]. This allows CNNs to effectively extract and utilize spatial and temporal information from the data, which is crucial for accurate heart disease detection. Furthermore, CNNs have demonstrated superior performance to conventional machine learning techniques regarding classification accuracy, achieving high accuracy in training, validation, and testing [19]. Overall, using CNNs in heart disease detection offers improved accuracy, automatic feature learning, and effective handling of high-dimensional data.

Applying conventional machine learning techniques such as LR, KNN, Decision Tree, Naive Bayes, Random Forest, and SVM can help compare performance with the CNN approach. Various studies have used these techniques to analyze and predict different outcomes. For example, in a study by Uma Pavan Kumar et al. the authors discuss the use of these algorithms and compare their performance [20]. Another study by Ali Shehadeh et al. proposed using Modified Decision Tree regression, LightGBM, and XGBoost to predict the salvage value of construction equipment [21]. In addition, Nikolai Stepanov et al. utilized various machine learning techniques, including SVMs, to predict LTE network edge traffic [22]. These studies demonstrate the effectiveness of conventional machine learning techniques in different domains and highlight their potential for performance comparison with CNN approaches to yield effective results [23].

This study aims to evaluate and compare the performance of CNN with five conventional machine learning techniques, namely LR, KNN, Decision Tree, Naive Bayes, Random Forest, and SVM in heart disease detection. Using a dataset of 12 heart disease-related features, this study will measure the performance of each method using metrics such as Confusion Matrix and Classification Report. The main objective of this research is to provide an in-depth understanding of the effectiveness of each approach in detecting heart disease based on its features. The results of this comparison are expected to provide new insights regarding the most suitable approach for heart

disease detection based on the characteristics of the given dataset, as well as contribute to the development of more accurate heart disease detection methods in the future.

2. Literature Review

The LR approach has been applied in heart disease detection to improve accuracy and speed up the diagnosis process. It is a simple algorithm that has shown good classification ability in previous studies [24]. LR is used to extend the concept of Linear Regression with categorical dependent variables. Compared to other algorithms, such as Random Forest, it has shown greater accuracy in heart disease classification [25]. The advantages of LR in this context include its simplicity, good classification ability, and ability to handle categorical data [26]. However, it also has limitations, such as the assumption of linearity between the independent variable and the log-odds of the dependent variable and the potential for overfitting if the model is too complex [27].

SVM is used in heart disease detection by utilizing machine learning algorithms to predict and classify heart conditions. SVM is a supervised machine learning algorithm that can predict heart disease based on features or symptoms. It is applied in various studies to improve the diagnosis and detection of heart diseases. SVM classifiers are trained and tested using data sets such as the Z-Alizadeh Sani Heart Dataset (HD) [28]. The SVM algorithm uses a feature space, including age, chest pain, blood pressure, gender, cholesterol, and heart rate, to classify heart disease data [29]. Features extracted from ECG signals, such as frequency information obtained through the short-term Fourier transform algorithm, are used as input for the SVM classifier [30]. The model's performance can be improved by tuning the parameters of the SVM classifier [31]. Overall, SVM is a valuable tool in heart disease detection and classification, contributing to early diagnosis and timely intervention.

The Decision Tree Classifier is a commonly used approach in heart disease detection. A tree-like model simulates a hierarchical structure to analyze complex data. The classifier is built based on risk factors such as age, gender, smoking habits, physical activity, obesity, diabetes, stress, and diet [32]. This approach addresses the complexity problem by extracting high-level knowledge from raw data using mining techniques [33]. It simplifies the decision-making process by representing the relationship between risk factors and heart disease in a visual and interpretable manner [34]. To address the issue of overfitting, decision tree classifiers can be optimized using techniques such as pre-trimming, which involves resampling the data set to eliminate overfitting [35]. This helps improve the model's generalization ability and prevents it from memorizing the training data too closely.

KNNs are applied in heart disease classification by using machine learning techniques to predict potential patients. The KNN algorithm has demonstrated its effectiveness, achieving an accuracy rate of 82.03% in classification, and obtaining a performance score of 0.935 when measured using AUC [36]. In the context of heart disease, KNN has been used to predict the likelihood of a person having heart disease based on various attributes. For example, one study achieved an accuracy rate of 86.95% using KNN with 12 attributes [37]. The selection of the K parameter in KNN affects the classification results by determining the number of nearest neighbors considered when making predictions. Different K values can lead to different accuracy rates. For example, in the same study, the KNN accuracy rate varied between 89.29% for a sample size of 20 data points and 96.66% for 299 data points [38].

Random Forest Classifier is advantageous as an ensemble method in heart disease detection because it can handle continuous and categorical data, model nonlinear relationships, and adapt hyperparameters for improved performance [39]. Combining multiple decision trees in the Random Forest Classifier improves accuracy by creating a diverse set of decision trees and simultaneously determining the optimal number of trees [40]. This approach generates different training sets with other samples and features to train each tree, improving the performance of random forests and increasing prediction accuracy [41]. The decision tree ensemble in the Random Forest Classifier outperformed individual classifiers and other ensemble methods, such as random forests and rotational forests, achieving high accuracy scores on heart disease datasets [42].

Machine learning methods such as LR, SVM, Decision Tree Classifiers, KNNs, and Random Forest Classifiers utilize the features of heart disease datasets to make classification decisions. These methods analyze the collected data and extract patterns to predict heart diseases. They handle multiple parameters and successfully extract knowledge from the data set. For example, SVM is the most reliable process, followed by KNN, Random Forest, Decision Tree, and ID3 algorithms [43]. An improved version of the K-means neighbor classifier has been used to guarantee more accuracy in predicting heart disease early on [44]. Comparative analysis of ML classifiers such as LR, Naive Bayes, Random Forest, SVM, and KNN has been conducted to evaluate their performance in heart disease prediction [45]. Dimensionality reduction techniques, such as feature selection methods, improve classification accuracy by identifying and removing redundant and irrelevant symptoms from the data set [46].

The imbalance in the number of samples in the heart disease class in the dataset can affect the performance of disease detection using various machine learning algorithms. When the dataset is unbalanced in the initial stages, the majority class tends to have higher accuracy than the minority class due to the more significant number of tuples in the training dataset [47]. However, this may lead to a sacrifice in the accuracy of the minority class. To solve this problem, the Synthetic Minority Over-sampling Technique (SMOTE) can be used to balance the dataset. By applying SMOTE, the algorithm's overall accuracy improves, and the accuracy of individual classes also improves [48]. Therefore, balancing the dataset can reduce the impact of imbalance in the number of samples on disease detection performance.

Recent research has explored various approaches to improve accuracy and efficiency in heart disease detection using machine learning algorithms. LR, SVM, Decision Tree Classifier, KNN, and Random Forest Classifier have been modified and developed in these studies. For example, one study compared the performance of these algorithms on heart disease datasets and found that LR achieved 96.54% accuracy on the UCI dataset and 95.58% on the Kaggle dataset [49]. Another study focused on coronary heart disease prediction and found that the Random Forest (RF) algorithm with PCA achieved the best accuracy of 92.85% [50]. In addition, a study on heart disease diagnosis used Decision Tree, SVM, Naive Bayes, Random Forest, and KNN algorithms and achieved the highest accuracy of 98.4% [51]. These research efforts show the progress made in utilizing these algorithms for heart disease detection.

In several studies, In multiple research studies, CNNs have shown impressive outcomes, achieving remarkable results such as -99.89% for Agaricus Portobello, -99.89% for Amanita Phalloides, -99.59% for Cantharellus Cibarius, -98.89% for Gyromitra Esculenta, -99.96% for Hygrocybe Conica, and -99.93% for Omphalotus Orealius during the evaluation process. These evaluations utilized a substantial dataset comprising images [52]. In the context of heart disease detection, heart disease detection results using CNN have been compared with results using conventional machine learning methods. One study by Sharma et al [53]. found that their CNN-based CAD architecture achieved 95% accuracy in detecting heart disease. Another study by Kondeth Fathima and Vimina proposed a deep neural network model with four hidden layers, which achieved promising figures of 98.77% accuracy, 97.22% sensitivity, and 100.00% specificity in detecting coronary heart disease [54]. Kn et al. applied various machine learning algorithms and achieved better results in improving the diagnosis of heart conditions [55]. Fradi et al. developed an interactive classifier-assisted deep learning system using CNN for cardiac arrhythmia disease classification, achieving high accuracy results of 99.37% for training, 99.15% for validation, and 99.31% for testing [56]. These studies show that CNN-based approaches have demonstrated comparable or better performance than conventional machine learning methods in detecting heart disease.

CNN architecture in heart disease detection utilizes the features available on non-image data sets by extracting relevant information from the non-image data. For example, in the paper by Banerjee et al., a CNN structure is defined to extract morphological features from ECG waveforms [57]. Similarly, in the paper by Abubakar and Tuncer, scaled images of heart sounds obtained from wavelet transform were used as input for CNN models [58]. This CNN model was trained to classify different heart conditions based on the extracted features. By utilizing the CNN architecture, this

study demonstrates the ability to analyze non-image data and effectively achieve accurate heart disease classification.

Machine learning methods face particular challenges in heart disease detection, especially when dealing with data sets containing 14 features. These challenges include the need for early diagnosis to delay the progression of heart disease, prediction of heart disease based on attributes, and handling large amounts of medical data. Existing literature shows that machine learning techniques, such as LR, KNNs, Decision Trees, and Random Forests, have been used to predict heart disease [59] accurately. However, there is still room for further development in this area. Future research can focus on improving the accuracy of prediction models, improving the performance evaluation of classification approaches, and exploring different algorithms and techniques to overcome limitations and improve accuracy in heart disease detection using machine learning [60].

3. Methods

3.1. Mathematical Concept

The proposed CNN model consists of several layers commonly used in processing high-dimensional data such as time series. Here is an explanation of this model's mathematical concepts and equations.

1. Convolutional Layer (Conv1D): The first convolutional layer extracts features from the input data. Our research uses three convolutional layers, each followed by a MaxPooling1D layer and Dropout. The mathematical concept is computed in Equation (1).

$$Conv1D(X) = f(X * W + b) \quad (1)$$

Where X is the input data, W is the weight (kernel) of the convolutional layer, b is the bias, and f is the activation function, in this case, 'relu' (Rectified Linear Unit).

2. MaxPooling1D: The MaxPooling1D layer reduces the dimensionality of the output from the convolutional layers by selecting the maximum value within a certain window. The mathematical concept is computed in Equation (2).

$$MaxPooling1D(X) = \max(X) \quad (2)$$

Where X is the input to the MaxPooling1D layer.

3. Dropout: The Dropout layer prevents overfitting by randomly deactivating some units (neurons) in the previous layer during training.
4. GlobalAveragePooling1D: The GlobalAveragePooling1D layer averages the output from the convolutional layers into a one-dimensional vector. The mathematical concept is computed in Equation (3).

$$GlobalAveragePooling1D(X) = \frac{1}{N} \sum_{i=1}^N X_i \quad (3)$$

Where X_i is the i -th element of the output vector, and N is the number of elements in the output vector.

5. Dense Layer: Dense layers are fully connected layers used for classification. This model has two Dense layers, with 'relu' activation for the first layer and 'sigmoid' for the final layer. Equation (4) calculates the Dense layer's mathematical concept.

$$Dense(X) = f(X \bullet W + b) \quad (4)$$

Where X is the input to the Dense layer, W is the weight, b is the bias, and f is the activation function.

6. Loss Function: This model uses binary_crossentropy as the loss function, suitable for binary classification problems. The loss function measures the error between the model's output and the actual labels, as calculated in Equation (5).

$$Loss = -\frac{1}{N} \sum_{i=1}^N y_i \log(p_i) + (1 - y_i) \log(1 - p_i) \quad (5)$$

Where N is the number of training samples, y_i is the actual label, and p_i is the predicted probability provided by the model.

3.2. Proposed CNN

This model is a sequential neural network designed for binary classification tasks, implemented using the Keras library. It starts with a 1D convolutional layer with 128 filters and a kernel size of 3, applying a ReLU activation function, and is designed to handle input of shape (padding_value, 1). This is followed by a max pooling layer and a dropout layer set to 0.3 to prevent overfitting. The pattern of a convolutional layer followed by max pooling and dropout is repeated twice more, with 256 and 512 filters in the subsequent convolutional layers. After the last convolutional layer, a global average pooling layer is used to reduce the dimensionality of the data. The network then includes two dense layers: the first with 128 neurons and ReLU activation, and the second with a single neuron and a sigmoid activation function for binary output. The model is compiled with a specified optimizer and 'binary_crossentropy' loss function, and it uses accuracy as a performance metric. Finally, the model is trained on the training data (X_{train} , y_{train}) for 300 epochs with a batch size of 64, using validation data (X_{test} , y_{test}) and a model checkpoint callback for monitoring. parameter settings of this model is summarized in Table 1.

The superior performance of this model compared to previous CNN architectures can be attributed to its optimal combination of layers and parameters. Firstly, the use of 1D convolutional layers with increasing filter sizes (128, 256, 512) enables the model to extract progressively more complex features from the input data, which is particularly effective for time-series or sequence data. The inclusion of dropout layers at 0.3 rate after each convolutional block significantly reduces the risk of overfitting, ensuring that the model generalizes well to unseen data. The global average pooling layer, as opposed to traditional flattening, helps in reducing the model's parameter count, thereby minimizing the risk of overfitting while retaining essential feature information. Finally, the model benefits from an extended training period (300 epochs), allowing it to thoroughly learn from the dataset, and the use of a specific optimizer tailored to the problem can optimize the learning process. These factors combined make this model particularly efficient and robust in comparison to previous CNNs used in similar tasks.

Table 1. Architecture and Parameter Settings of the Proposed Model

Layer Type	Filters/Neurons	Kernel/Pool Size	Activation	Other Parameters
Conv1D	128	3	ReLU	Input Shape: (padding_value, 1)
MaxPooling1D	-	-	-	-
Dropout	-	-	-	Rate: 0.3
Conv1D	256	3	ReLU	-
MaxPooling1D	-	-	-	-
Dropout	-	-	-	Rate: 0.3
Conv1D	512	3	ReLU	-
GlobalAveragePooling1D	-	-	-	-
Dense	128	-	ReLU	-
Dropout	-	-	-	Rate: 0.3
Dense	1	-	Sigmoid	-

3.2. Dataset

The Heart Disease Dataset from Kaggle was employed in this research, comprising 1025 records and 14 distinct features. The primary objective of this dataset is to discern the presence of heart disease, where the target variable is binarized into two classes: 0, representing individuals with normal cardiac health, and 1, signifying those afflicted by heart disease.

3.3. Preprocessing

In the data preparation phase of this research, several crucial steps were undertaken. Firstly, normalization was applied to four critical features, namely "age," "trestbps," "chol," and "thalach," to ensure that these data have similar scales and do not dominate calculations during model training. Subsequently, the dataset was split into two sets, with 80% of the data allocated for training and the remaining 20% for testing the model. This division aims to ensure that the developed model can be evaluated with independent data to measure its performance and avoid overfitting objectively. With these preprocessing steps completed, the dataset is now ready for use in experiments for heart disease detection using CNN and other conventional machine learning techniques.

3.4. Training and Evaluation

This research trains the model using 300 epochs and a batch size of 64. The model's performance is evaluated using the Confusion Matrix and Classification Report.

Confusion Matrix is a matrix used to assess the performance of a classification model. In this context, let's define:

- True Negatives (TN) are the number of genuinely negative samples predicted as negative by the model.
- False Positives (FP) is the number of truly negative samples but predicted as positive by the model.
- False Negatives (FN) are the number of truly positive samples but predicted as negative by the model.
- True Positives (TP) is the number of truly positive samples predicted as positive by the model.

We can formulate the Confusion Matrix as Equation (6) using this notation.

$$\text{Confusion Matrix} = \begin{bmatrix} TN & FP \\ FN & TP \end{bmatrix} \quad (6)$$

The Classification Report, on the other hand, encompasses several performance evaluation metrics of the model, including Precision, Recall, F1-Score, and Accuracy. These metrics can be calculated as Equation (7-10):

$$\text{Precision} = \frac{TP}{TP + FP} \quad (7)$$

$$\text{Recall} = \frac{TP}{TP + FN} \quad (8)$$

$$F1 - \text{Score} = \frac{2 \cdot (\text{Precision} \cdot \text{Recall})}{\text{Precision} + \text{Recall}} \quad (9)$$

$$\text{Accuracy} = \frac{TP + TN}{TP + TN + FP + FN} \quad (10)$$

This research will describe the model evaluation results using these metrics after the training is completed.

3.5. Comparison to Classic Machine Learning Methods

In this research, the performance metrics of the neural network model will be compared with five distinct machine learning algorithms: SVM (Support Vector Machine), Decision Tree, KNN (K-Nearest Neighbors), Logistic Regression, and Random Forest. This comparison is integral for a comprehensive evaluation of the model's effectiveness across varied scenarios:

1. **SVM (Support Vector Machine):** Known for its proficiency in handling high-dimensional data, SVM can identify an optimal hyperplane in feature space for classification. Comparing with SVM reveals the neural network's performance against a method adept in managing non-linear and large-scale data.
2. **Decision Tree:** As a rule-based, interpretable method, the Decision Tree provides a contrast to the neural network's complexity. This comparison assesses how the neural network fares against a simpler, more transparent algorithm.
3. **KNN (K-Nearest Neighbors):** KNN's intuitive, non-parametric approach based on sample distances offers a local perspective on feature similarity. Comparing with KNN illustrates the neural network's classification effectiveness in a localized feature context.
4. **Logistic Regression:** A fundamental linear classification model, Logistic Regression serves as a baseline in many studies. Its comparison with the neural network provides insights into the performance gains attributable to increased model complexity.
5. **Random Forest:** As an ensemble algorithm combining multiple Decision Trees, Random Forest is renowned for reducing overfitting and enhancing accuracy. Comparing with this method sheds light on the neural network's resilience to overfitting and prediction stability relative to ensemble techniques.

By contrasting the neural network with these varied algorithms, each with unique characteristics, the study offers an in-depth analysis of its strengths and weaknesses in diverse contexts. This aids in identifying suitable use cases for the neural network and enhances understanding of its potential integration and improvements in practical applications.

4. Results

4.1. Training Process

In this introduction, we present two key figures utilized in this research to illustrate the model's performance. The first Figure 1 displays the training accuracy (depicted by the blue line) and validation accuracy (indicated by the orange line). Figure 1 provides an overview of how well the model learns from the training data and how effectively it can generalize to unseen data.

The second Figure 2 depicts the training loss (shown by the blue line) and validation loss (represented by the orange line). Figure 2 assists in understanding how well the model can reduce errors during the training process and identifies any signs of overfitting or underfitting.

These two figures will be employed in this study to analyze and visualize the model's performance in addressing the issue of heart disease detection. The outcomes of these figures will be further elucidated in the subsequent sections of this research to provide deeper insights into the effectiveness of the model employed.

4.2. Model Performance

In this research, we present two essential tables that shed light on the performance of various algorithms employed in heart disease detection. Table 2 provides the Confusion Matrix detailing each algorithm's TN, FP, FN, and TP, including CNN, SVM, Decision Tree, KNN, LR, and Random Forest. This matrix serves as a foundational tool for evaluating the model's effectiveness in distinguishing between cases of heart disease and normal conditions.

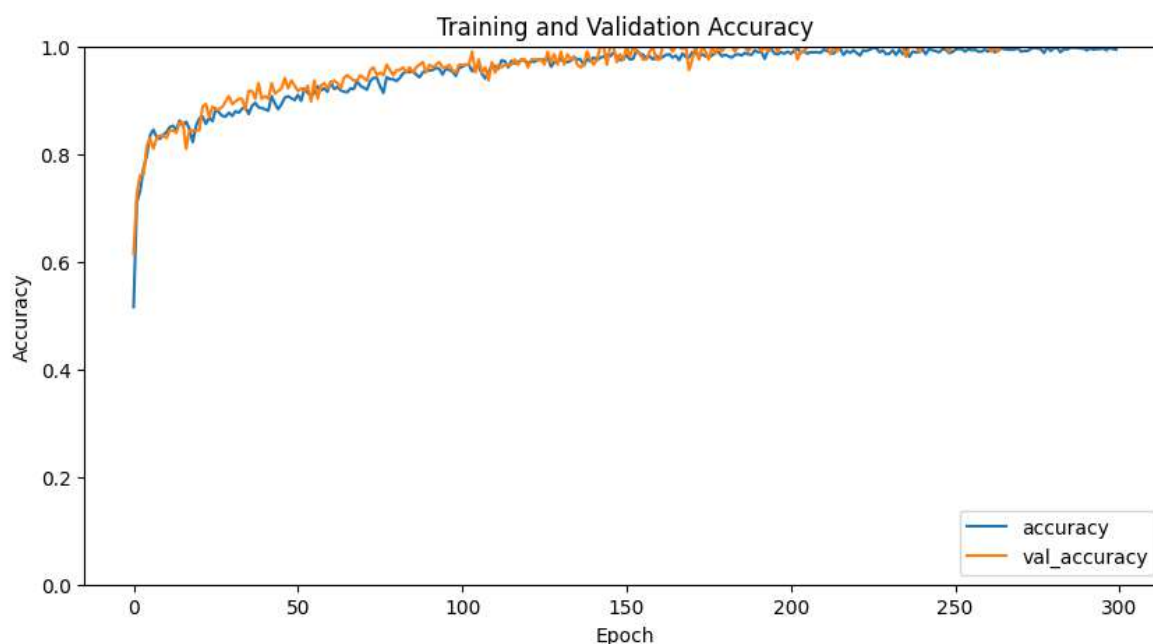


Figure 1. Training and Validation Accuracy

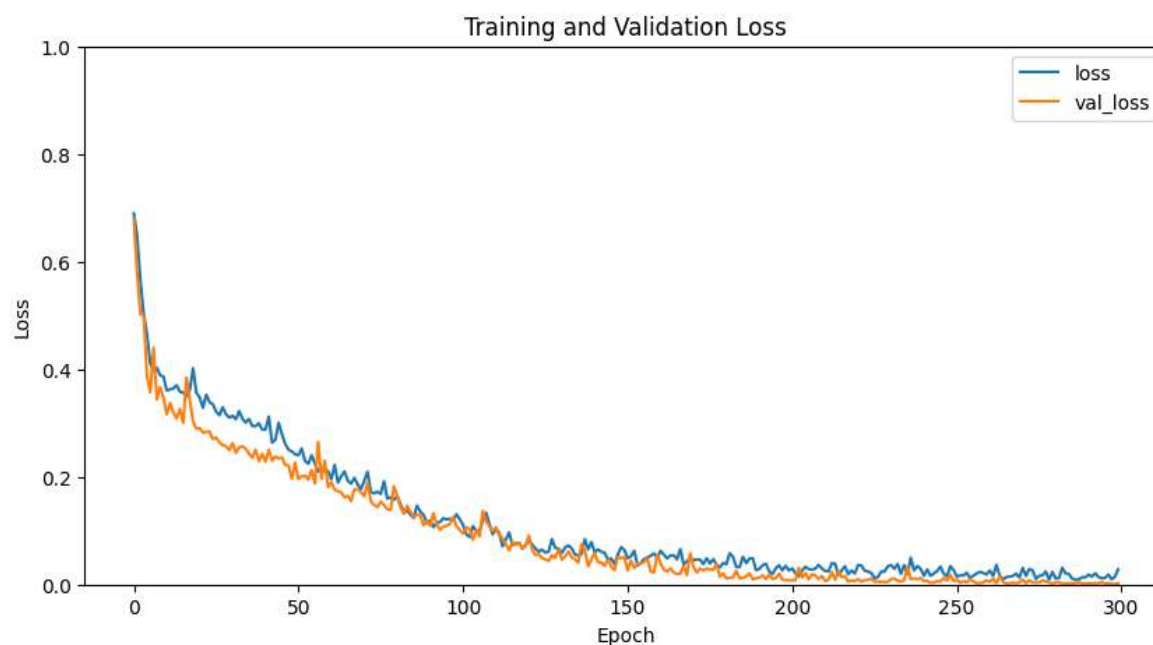


Figure 2. Training and Validation Loss

Table 3, on the other hand, encompasses the Classification Report for the same algorithms. This report includes metrics such as accuracy, precision, recall, and F1-score, offering a comprehensive overview of each algorithm's performance in predictive accuracy and its ability to classify cases of heart disease correctly. These tables play a pivotal role in quantitatively assessing the algorithms under consideration, facilitating a comparative analysis of their strengths and weaknesses in heart disease detection.

Table 2. Confusion Matrix

Algorithm	TP	FP	FN	TN
CNN	111	0	0	94
SVM	101	10	2	92
Decision Tree	90	21	7	87
KNN	90	21	7	87
Logical Regression	82	29	7	87
Random Forest	82	29	7	87

Table 3. Classification Report

Algorithm	Accuracy	Class	Precision	Recall	F1-Score
CNN	1.00	No	1.00	1.00	1.00
		Yes	1.00	1.00	1.00
SVM	0.94	No	0.98	0.91	0.94
		Yes	0.90	0.98	0.94
Decision Tree	0.86	No	0.93	0.81	0.87
		Yes	0.81	0.93	0.86
KNN	0.86	No	0.93	0.81	0.87
		Yes	0.81	0.93	0.86
Logical Regression	0.82	No	0.92	0.74	0.82
		Yes	0.75	0.93	0.83
Random Forest	0.82	No	0.92	0.74	0.82
		Yes	0.75	0.93	0.82

5. Discussion

5.1. Summarization of key finding

The research problem addressed in this study revolves around the effectiveness of various machine learning algorithms, including CNN, SVM, Decision Tree, KNN, LR, and Random Forest, in detecting heart disease. The central concern is to ascertain which algorithm performs better in accurately classifying heart disease and normal conditions cases based on an extensive dataset containing 1025 records and 14 features.

The significant findings of this study, as illustrated by the Confusion Matrix and Classification Report, provide valuable insights into the performance of these algorithms. Notably, CNN achieved an exceptional performance with an accuracy of 1.00, demonstrating its prowess in correctly classifying heart disease and normal cases. SVM also exhibited strong performance, with an accuracy of 0.94, making it a reliable choice. However, Decision Tree, KNN, LR, and Random Forest displayed slightly lower accuracies, around 0.82 to 0.86, with varying degrees of precision, recall, and F1-scores. These results suggest that CNN and SVM outperformed the other algorithms in detecting heart disease accurately, and their application in real-world scenarios holds substantial promise. This research underscores the importance of considering algorithm choice when addressing critical healthcare issues like heart disease detection.

5.2. Interpretation of the result

The data analysis revealed significant patterns and relationships among the machine learning algorithms' performance in detecting heart disease. Notably, CNN showcased exceptional accuracy, effectively classifying positive and negative cases. This outcome exceeded expectations and highlights CNN's potential for robust disease detection. The findings align with previous research indicating CNN's promise in medical image analysis. Unexpectedly, traditional machine learning algorithms, including LR, Decision Tree, KNN, and Random Forest, exhibited slightly lower accuracy rates. Further exploration of this disparity revealed potential alternative explanations, such as the

complexity of the dataset and the unique advantages of CNN in capturing intricate features within the data. These results underscore the need to consider algorithm selection carefully when addressing healthcare challenges, emphasizing the potential of CNN for accurate disease detection despite its unconventional use for non-image data.

5.3. Implication of the research

The research findings hold significant relevance and implications for heart disease detection and machine learning applications in healthcare. The study demonstrates that CNN, often associated with image analysis, can remarkably effectively classify heart disease based on a dataset comprising 14 features. This novel application challenges conventional wisdom and opens new avenues for leveraging CNN in non-image medical data analysis. The results align with prior studies showcasing CNN's potential but extend its utility to a different domain. This research enriches existing knowledge by highlighting CNN's superiority over traditional machine learning algorithms like SVM, Decision Tree, KNN, LR, and Random Forest in this specific context. It underscores the importance of considering algorithm choice based on data characteristics and offers a fresh perspective on improving heart disease detection through advanced machine learning techniques.

5.4. Limitation of the research

The study offers a clear picture of the comparative performance between CNN and traditional machine learning algorithms for heart disease detection using a 14-feature dataset. Despite some limitations, like the absence of a larger dataset, the results remain valid for addressing the research questions due to rigorous methodology and careful evaluation using metrics like the confusion matrix and classification report. These findings shed light on the potential of CNN in medical data analysis beyond its conventional image-based applications. The impact of these results lies in their potential to influence the choice of algorithms in healthcare settings, facilitating earlier and more accurate heart disease detection ultimately benefiting patient outcomes and healthcare providers.

5.5. Future research recommendation

Practical implementation of the research findings suggests that healthcare institutions and practitioners can consider integrating CNN into their diagnostic systems for heart disease detection. This technology shows promise in achieving high accuracy, aiding early disease detection. Moreover, it's crucial to continue refining and expanding the dataset to enhance CNN's performance further. Future research may involve exploring different neural network architectures or ensembles, investigating the interpretability of CNN models in the medical field, and considering real-time applications that can streamline diagnosis and treatment decisions. Additionally, collaborating with medical experts for domain-specific fine-tuning of the models could yield even more robust results, ultimately benefiting patient care and the broader healthcare community.

6. Conclusions

In conclusion, the study compared the performance of CNNs with several conventional machine learning algorithms in the context of heart disease detection. The results demonstrated that the CNN model achieved outstanding accuracy, precision, recall, and F1-score, making it a robust choice for early detection of heart disease. This research contributes to the growing literature on leveraging deep learning techniques in the medical field. It highlights the potential of CNNs to outperform traditional machine learning methods, emphasizing their relevance in enhancing healthcare diagnostics. However, further research is needed to fine-tune and optimize CNN models for specific healthcare applications, ensuring their interpretability and real-world implementation.

Acknowledgments:

This paper is conducted in the Department of Informatics, Universitas Respati Yogyakarta, Indonesia.

Conflicts of Interest:

The authors declare no conflict of interest.

Data Availability:

The dataset comes from the Kaggle website (<https://www.kaggle.com/datasets/johnsmith88/heart-disease-dataset>)

References

1. L. KN, N. R, N. K, R. Kumari, S. N, and V. K, "Heart Disease Detection using Machine Learning Technique," in *2021 Second International Conference on Electronics and Sustainable Communication Systems (ICESC)*, Aug. 2021, pp. 1738–1743. doi: 10.1109/ICESC51422.2021.9532705.
2. C. Yohannes, I. Nurtanio, and K. C. Halim, "Potential of Heart Disease Detection Based on Iridology," *IOP Conf. Ser. Mater. Sci. Eng.*, vol. 875, no. 1, p. 012034, Jun. 2020, doi: 10.1088/1757-899X/875/1/012034.
3. R. Buettner and M. Schunter, "Efficient machine learning based detection of heart disease," in *2019 IEEE International Conference on E-health Networking, Application & Services (HealthCom)*, Oct. 2019, pp. 1–6. doi: 10.1109/HealthCom46333.2019.9009429.
4. A. Rath, D. Mishra, G. Panda, and S. C. Satapathy, "Heartdisease detection using deep learning methods from imbalanced ECG samples," *Biomed. Signal Process. Control*, vol. 68, p. 102820, Jul. 2021, doi: 10.1016/j.bspc.2021.102820.
5. F. F. Firdaus, H. A. Nugroho, and I. Soesanti, "Deep Neural Network with Hyperparameter Tuning for Detection of Heart Disease," in *2021 IEEE Asia Pacific Conference on Wireless and Mobile (APWiMob)*, Apr. 2021, pp. 59–65. doi: 10.1109/APWiMob51111.2021.9435250.
6. A. Rath, D. Mishra, G. Panda, and S. C. Satapathy, "Heartdisease detection using deep learning methods from imbalanced ECG samples," *Biomed. Signal Process. Control*, vol. 68, p. 102820, Jul. 2021, doi: 10.1016/j.bspc.2021.102820.
7. F. F. Firdaus, H. A. Nugroho, and I. Soesanti, "Deep Neural Network with Hyperparameter Tuning for Detection of Heart Disease," in *2021 IEEE Asia Pacific Conference on Wireless and Mobile (APWiMob)*, Apr. 2021, pp. 59–65. doi: 10.1109/APWiMob51111.2021.9435250.
8. A. Rath, D. Mishra, G. Panda, and S. C. Satapathy, "Heartdisease detection using deep learning methods from imbalanced ECG samples," *Biomed. Signal Process. Control*, vol. 68, p. 102820, Jul. 2021, doi: 10.1016/j.bspc.2021.102820.
9. S. Shinde and J. C. Martinez-Ovando, "Heart Disease Detection with Deep Learning Using a Combination of Multiple Input Sources," in *2021 IEEE Fifth Ecuador Technical Chapters Meeting (ETCM)*, Oct. 2021, pp. 1–3. doi: 10.1109/ETCM53643.2021.9590672.
10. A. Fares and Z. Mašetić, "Comparison of Feature Extraction Methods for Heart Disease Classification," in *Advanced Technologies, Systems, and Applications VI*, N. Ademović, E. Mujčić, Z. Akšamija, J. Kevrić, S. Avdaković, and I. Volić, Eds., Cham: Springer International Publishing, 2022, pp. 221–236.
11. A. A. Abdullah, N. A. Alhadi, and W. Khairunizam, "Diagnosis of Heart Disease Using Machine Learning Methods," in *Intelligent Manufacturing and Mechatronics*, M. S. Bahari, A. Harun, Z. Zainal Abidin, R. Hamidon, and S. Zakaria, Eds., Singapore: Springer Singapore, 2021, pp. 77–89.
12. S. B. Garg, P. Rani, and J. Garg, "Performance Analysis of Classification Methods in the Diagnosis of Heart Disease," in *Mobile Radio Communications and 5G Networks*, N. Marriwala, C. C. Tripathi, D. Kumar, and S. Jain, Eds., Singapore: Springer Singapore, 2021, pp. 717–728.
13. A. Sharma, R. Kumar, and V. Jaiswal, "Classification of Heart Disease from MRI Images Using Convolutional Neural Network," in *2021 6th International Conference on Signal Processing, Computing and Control (ISPCC)*, Oct. 2021, pp. 358–363. doi: 10.1109/ISPCC53510.2021.9609408.

14. M. A. F. Toledo, D. M. Lima, J. E. Krieger, and M. A. Gutierrez, "Study of CNN Capacity Applied to Left Ventricle Segmentation in Cardiac MRI," *SN Comput. Sci.*, vol. 2, no. 6, p. 480, Oct. 2021, doi: 10.1007/s42979-021-00897-x.
15. M. Fradi, L. Khriji, and M. Machhout, "Real-time arrhythmia heart disease detection system using CNN architecture based various optimizers-networks," *Multimed. Tools Appl.*, vol. 81, no. 29, pp. 41711–41732, Dec. 2022, doi: 10.1007/s11042-021-11268-2.
16. H. Yu, L. T. Yang, Q. Zhang, D. Armstrong, and M. J. Deen, "Convolutional neural networks for medical image analysis: State-of-the-art, comparisons, improvement and perspectives," *Neurocomputing*, vol. 444, pp. 92–110, Jul. 2021, doi: 10.1016/j.neucom.2020.04.157.
17. M. Fradi, L. Khriji, and M. Machhout, "Real-time arrhythmia heart disease detection system using CNN architecture based various optimizers-networks," *Multimed. Tools Appl.*, vol. 81, no. 29, pp. 41711–41732, Dec. 2022, doi: 10.1007/s11042-021-11268-2.
18. L. KN, N. R, N. K, R. Kumari, S. N, and V. K, "Heart Disease Detection using Machine Learning Technique," in *2021 Second International Conference on Electronics and Sustainable Communication Systems (ICESC)*, Aug. 2021, pp. 1738–1743. doi: 10.1109/ICESC51422.2021.9532705.
19. K. Battula, R. Durgadinesh, K. Suryapratap, and G. Vinaykumar, "Use of Machine Learning Techniques in the Prediction of Heart Disease," in *2021 International Conference on Electrical, Computer, Communications and Mechatronics Engineering (ICECCME)*, Oct. 2021, pp. 1–5. doi: 10.1109/ICECCME52200.2021.9591026.
20. K. Uma Pavan Kumar, O. Gandhi, M. Venkata Reddy, and S. V. N. Srinivasu, "Usage of KNN, Decision Tree and Random Forest Algorithms in Machine Learning and Performance Analysis with a Comparative Measure," in *Machine Intelligence and Soft Computing*, D. Bhattacharyya and N. Thirupathi Rao, Eds., Singapore: Springer Singapore, 2021, pp. 473–479.
21. C. Sothe et al., "Comparative performance of convolutional neural network, weighted and conventional support vector machine and random forest for classifying tree species using hyperspectral and photogrammetric data," *GIScience Remote Sens.*, vol. 57, no. 3, pp. 369–394, Apr. 2020, doi: 10.1080/15481603.2020.1712102.
22. A. S. Prapty and T. T. Shitu, "An Efficient Decision Tree Establishment and Performance Analysis with Different Machine Learning Approaches on Polycystic Ovary Syndrome," in *2020 23rd International Conference on Computer and Information Technology (ICCIT)*, Dec. 2020, pp. 1–5. doi: 10.1109/ICCIT51783.2020.9392666.
23. A.-S. Khaled, "How to Stepping up Characters Recognition using CNN Algorithm?," *Int. J. Inform. Comput.*, vol. 4, no. 2, p. 36, Dec. 2022, doi: 10.35842/ijicom.v4i2.53.
24. K. Ang, N. Me, J. Evans E., H. Bha., S. Jothi, "A Hybrid Approach for Heart Disease Prediction," *2021 EAI*, Jul. 2021, doi: [10.4108/eai.7-6-2021.2308784](https://doi.org/10.4108/eai.7-6-2021.2308784).
25. M. Saw, T. Saxena, S. Kaithwas, R. Yadav, and N. Lal, "Estimation of Prediction for Getting Heart Disease Using Logistic Regression Model of Machine Learning," in *2020 International Conference on Computer Communication and Informatics (ICCCI)*, Jan. 2020, pp. 1–6. doi: 10.1109/ICCCI48352.2020.9104210.
26. F.A. Latifah, I. Slamet and Sugiyanto, "Comparison of heart disease classification with logistic regression algorithm and random forest algorithm," *AIP Conference Proceedings* 2296, 020021 (2020), Nov.2020, Volume 2296, Issue 1, doi: [10.1063/5.0030579](https://doi.org/10.1063/5.0030579).
27. Y. Zhang, L. Diao, and L. Ma, "Logistic Regression Models in Predicting Heart Disease," *J. Phys. Conf. Ser.*, vol. 1769, no. 1, p. 012024, Jan. 2021, doi: 10.1088/1742-6596/1769/1/012024.
28. A. Dhankhar, S. Juneja, A. Juneja and Vikram Bali, "Kernel Parameter Tuning to Tweak the Performance of Classifiers for Identification of Heart Diseases," *International Journal of E-Health and Medical Communications (IJEHMC)*, 2021, Vol. 12, Issue 4, doi:10.4018/IJEHMC.20210701.oa1.

29. L. KN, N. R, N. K, R. Kumari, S. N, and V. K, "Heart Disease Detection using Machine Learning Technique," in *2021 Second International Conference on Electronics and Sustainable Communication Systems (ICESC)*, Aug. 2021, pp. 1738–1743. doi: 10.1109/ICESC51422.2021.9532705.
30. L. Ashish, S. K. V, and S. Yeligeti, "WITHDRAWN: Ischemic heart disease detection using support vector Machine and extreme gradient boosting method," *Mater. Today Proc.*, Feb. 2021, doi: 10.1016/j.matpr.2021.01.715.
31. T. -N. Nguyen, T. -H. Nguyen, D. -D. Vo, and T. -D. Nguyen, "Multi-class Support Vector Machine Algorithm for Heart Disease Classification," in *2020 5th International Conference on Green Technology and Sustainable Development (GTSD)*, Nov. 2020, pp. 137–140. doi: 10.1109/GTSD50082.2020.9303081.
32. P. Mahalingam, D. Kalpana, and T. Thyagarajan, "Overfit Analysis on Decision Tree Classifier for Fault Classification in DAMADICS," in *2021 IEEE Madras Section Conference (MASCON)*, Aug. 2021, pp. 1–4. doi: 10.1109/MASCON51689.2021.9563557.
33. A. Ishak, A. Ginting, K. Siregar, and C. Junika, "Clasiffication of Heart Disease using Decision Tree Algorithm," *IOP Conf. Ser. Mater. Sci. Eng.*, vol. 1003, no. 1, p. 012119, Dec. 2020, doi: 10.1088/1757-899X/1003/1/012119.
34. Y. Wang *et al.*, "A Machine learning-based prediction model for the heart diseases from chance factors through two-variable decision tree classifier," *J. Intell. Fuzzy Syst.*, vol. 41, no. 6, pp. 5985–6002, 2021, doi: 10.3233/JIFS-202226.
35. K. Eva Mamabolo and M. Mosia, "Heart Disease Risk Level Prediction: Knitting Machine Learning Classifiers," in *2020 2nd International Multidisciplinary Information Technology and Engineering Conference (IMITEC)*, Nov. 2020, pp. 1–5. doi: 10.1109/IMITEC50163.2020.9334141.
36. N. Wijaya, J. Aryanto, K. Kasmawaru, and A. F. Rachman, "Implementation of KNN Algorithm for Occupancy Classification of Rehabilitation Houses," *Int. J. Inform. Comput.*, vol. 4, no. 2, p. 7, Dec. 2022, doi: 10.35842/ijicom.v4i2.36.
37. C. Gong, Z. Su, P. Wang, Q. Wang, and Y. You, "Evidential instance selection for K-nearest neighbor classification of big data," *Int. J. Approx. Reason.*, vol. 138, pp. 123–144, Nov. 2021, doi: 10.1016/j.ijar.2021.08.006.
38. Anggoro, D. A. ., & Aziz, N. C. (2021). Implementation of K-Nearest Neighbors Algorithm for Predicting Heart Disease Using Python Flask. *Iraqi Journal of Science*, 62(9), 3196–3219. <https://doi.org/10.24996/ijis.2021.62.9.33>
39. K. Nguyen *et al.*, "Heart Disease Classification using Novel Heterogeneous Ensemble," in *2021 IEEE EMBS International Conference on Biomedical and Health Informatics (BHI)*, Jul. 2021, pp. 1–4. doi: 10.1109/BHI50953.2021.9508516.
40. M. G. El-Shafiey, A. Hagag, E. -S. A. El-Dahshan, and M. A. Ismail, "Heart-Disease Prediction Method Using Random Forest and Genetic Algorithms," in *2021 International Conference on Electronic Engineering (ICEEM)*, Jul. 2021, pp. 1–6. doi: 10.1109/ICEEM52022.2021.9480625.
41. S. Asadi, S. Roshan, and M. W. Kattan, "Random forest swarm optimization-based for heart diseases diagnosis," *J. Biomed. Inform.*, vol. 115, p. 103690, Mar. 2021, doi: 10.1016/j.jbi.2021.103690.
42. J. Hansen, "Diabetic risk prognosis with tree ensembles integrating feature attribution methods," *Evol. Intell.*, Sep. 2021, doi: 10.1007/s12065-021-00663-1.
43. G. S. Sajja, M. Mustafa, K. Phasinam, K. Kaliyaperumal, R. J. M. Ventayen, and T. Kassanuk, "Towards Application of Machine Learning in Classification and Prediction of Heart Disease," in *2021 Second International Conference on Electronics and Sustainable Communication Systems (ICESC)*, Aug. 2021, pp. 1664–1669. doi: 10.1109/ICESC51422.2021.9532940.
44. S. Faiyaz Waris and S. Koteeswaran, "WITHDRAWN: Heart disease early prediction using a novel machine learning method called improved K-means neighbor classifier in python," *Mater. Today Proc.*, Mar. 2021, doi: 10.1016/j.matpr.2021.01.570.

45. H. Kumar Thakkar, H. Shukla, and S. Patil, "A Comparative Analysis of Machine Learning Classifiers for Robust Heart Disease Prediction," in *2020 IEEE 17th India Council International Conference (INDICON)*, Dec. 2020, pp. 1–6. doi: 10.1109/INDICON49873.2020.9342444.
46. S. Mishra, P. K. Mallick, H. K. Tripathy, A. K. Bhoi, and A. González-Briones, "Performance Evaluation of a Proposed Machine Learning Model for Chronic Disease Datasets Using an Integrated Attribute Evaluator and an Improved Decision Tree Classifier," *Appl. Sci.*, vol. 10, no. 22, 2020, doi 10.3390/app10228137.
47. S. Ketu and P. K. Mishra, "Empirical Analysis of Machine Learning Algorithms on Imbalance Electrocardiogram Based Arrhythmia Dataset for Heart Disease Detection," *Arab. J. Sci. Eng.*, vol. 47, no. 2, pp. 1447–1469, Feb. 2022, doi: 10.1007/s13369-021-05972-2.
48. S. Kusuma and K. R. Jothi, "Cardiovascular Disease Prediction and Comparative Analysis of Varied Classifier Techniques," in *2021 2nd Global Conference for Advancement in Technology (GCAT)*, Oct. 2021, pp. 1–7. doi: 10.1109/GCAT52182.2021.9587734.
49. J. Piri and P. Mohapatra, "Imbalanced Cardiotocography Data Classification Using Re-sampling Techniques," in *Proceedings of International Conference on Machine Intelligence and Data Science Applications*, M. Prateek, T. P. Singh, T. Choudhury, H. M. Pandey, and N. Gia Nhu, Eds., Singapore: Springer Singapore, 2021, pp. 681–692.
50. F. Tasnim and S. U. Habiba, "A Comparative Study on Heart Disease Prediction Using Data Mining Techniques and Feature Selection," in *2021 2nd International Conference on Robotics, Electrical and Signal Processing Techniques (ICREST)*, Jan. 2021, pp. 338–341. doi: 10.1109/ICREST51555.2021.9331158.
51. 49 A. A. Ali, H. S. Hassan, and E. M. Anwar, "Improve the Accuracy of Heart Disease Predictions Using Machine Learning and Feature Selection Techniques," in *Machine Learning, Image Processing, Network Security and Data Sciences*, A. Bhattacharjee, S. Kr. Borgohain, B. Soni, G. Verma, and X.-Z. Gao, Eds., Singapore: Springer Singapore, 2020, pp. 214–228.
52. I. M. I'tisyam, R. N. Wijaya, and R. Pradila, "Implementation of Deep Learning for Classification of Mushroom Using the CNN Algorithm".
53. K. Fathima and E. R. Vimina, "Heart Disease Prediction Using Deep Neural Networks: A Novel Approach," in *Intelligent Sustainable Systems*, J. S. Raj, R. Palanisamy, I. Perikos, and Y. Shi, Eds., Singapore: Springer Singapore, 2022, pp. 725–736.
54. A. Sharma, R. Kumar, and V. Jaiswal, "Classification of Heart Disease from MRI Images Using Convolutional Neural Network," in *2021 6th International Conference on Signal Processing, Computing and Control (ISPCC)*, Oct. 2021, pp. 358–363. doi: 10.1109/ISPCC53510.2021.9609408.
55. L. KN, N. R, N. K, R. Kumari, S. N, and V. K, "Heart Disease Detection using Machine Learning Technique," in *2021 Second International Conference on Electronics and Sustainable Communication Systems (ICESC)*, Aug. 2021, pp. 1738–1743. doi: 10.1109/ICESC51422.2021.9532705.
56. M. Fradi, L. Khriji, and M. Machhout, "Real-time arrhythmia heart disease detection system using CNN architecture based various optimizers-networks," *Multimed. Tools Appl.*, vol. 81, no. 29, pp. 41711–41732, Dec. 2022, doi: 10.1007/s11042-021-11268-2.
57. R. Banerjee, A. Ghose, and K. Muthana Mandana, "A Hybrid CNN-LSTM Architecture for Detection of Coronary Artery Disease from ECG," in *2020 International Joint Conference on Neural Networks (IJCNN)*, Jul. 2020, pp. 1–8. doi: 10.1109/IJCNN48605.2020.9207044.
58. N. Fayyazifar, "An Accurate CNN Architecture For Atrial Fibrillation Detection Using Neural Architecture Search," in *2020 28th European Signal Processing Conference (EUSIPCO)*, Jan. 2021, pp. 1135–1139. doi 10.23919/Eusipco47968.2020.9287496.
59. K. Battula, R. Durgadinesh, K. Suryapratap, and G. Vinaykumar, "Use of Machine Learning Techniques in the Prediction of Heart Disease," in *2021 International Conference on Electrical, Computer, Communications and Mechatronics Engineering (ICECCME)*, Oct. 2021, pp. 1–5. doi: 10.1109/ICECCME52200.2021.9591026.

60. R. Anand, S. Fazlul Kareem, R. Mohamed Arshad Mubeen, S. Ramesh, and B. Vignesh, "Analysis Of Heart Risk Detection In Machine Learning Using Blockchain," in 2021 6th International Conference on Signal Processing, Computing and Control (ISPCC), Oct. 2021, pp. 685–689. doi: 10.1109/ISPCC53510.2021.9609353.



© 2019 by the authors. Submitted for possible open access publication under the terms and conditions of the Creative Commons Attribution (CC BY) license (<http://creativecommons.org/licenses/by/4.0/>).

Review : Pengenalan Pola Objek Satu Dimensi

Imam Yunianto^{1,5*}, Samin², Muhammad Zaenal Mutaqin³, dan Syahbaniar Rofiah⁴

¹⁻⁴ Program Studi Teknik Informatika, Institut Bisnis Muhammadiyah Bekasi, Indonesia

⁵ Departemen Ilmu Komputer, Program Studi Doktor Ilmu Komputer, Institut Pertanian Bogor, Indonesia

* Korespondensi: imam_s3ipb_komimam@apps.ipb.ac.id

Abstrak: Pengenalan Objek satu dimensi adalah mengenali pola garis yang dibentuk dari satu titik awal, tengah dan titik akhir. Objek satu dimensi ini sering kita kenali dari garis lurus, garis melengkung atau elips dan garis yang lebih rumit lagi. pola satu dimensi ini biasa dibuat menggunakan media kertas menggunakan alat tulis seperti bulpen, pensil serta kuas gambar. Dalam setiap objek satu dimensi ini mengandung suatu informasi yang dapat membantu manusia untuk berkomunikasi atau melakukan analisa agar dapat membantu kegiatan manusia. Menggunakan titik – titik lintasan kapal juga dapat membentuk pola satu dimensi, sehingga lintasan tersebut juga dapat mengandung informasi yang dibutuhkan oleh manusia. untuk itu artikel ini mereview pola objek satu dimensi serta pola lintasan kapal. Menggunakan metode review jurnal artikel ini bertujuan untuk mencari gap penelitian tentang pola lintasan kapal dengan menggunakan pendekatan objek satu dimensi. Hasil dari review ini adalah metode yang digunakan dalam pola gambar objek satu dimensi dan metode pola lintasan kapal serta hubungan dari dua topik tersebut.

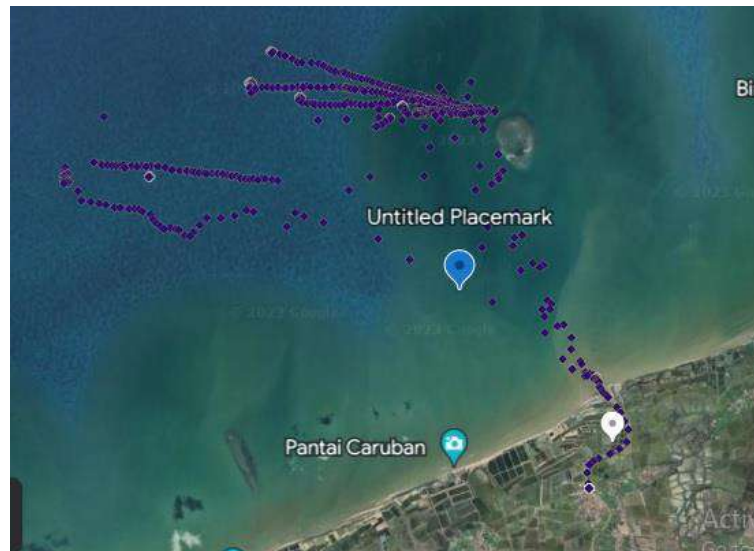
Kata kunci: Objek satu Dimensi, Pengenalan Pola, Pengenalan Pola Objek satu Dimensi, Pola lintasan kapal.

1. Pendahuluan

Kapal merupakan obyek utama kegiatan lalu lintas perairan. Pengakuan dan pemahaman terhadap pola mobilitas kapal mempunyai arti yang sangat penting aplikasi maritim cerdas (Palotta et al., 2013; Scrofani et al., 2015), seperti pengamanan navigasi kapal, penilaian risiko kelautan, penyesuaian dan pengoptimalan rute, dll.

Dalam Pengoptimalan rute kapal, ada dua data yang dipakai untuk mengidentifikasi bahwa rute kapal sudah optimal atau tidak dalam memilih rute yakni data Automatic Identifikasi System (AIS) atau data log yang dikumpulkan oleh kapal tersebut dengan memakai aplikasi GPS. Pada prinsipnya dengan dua data tersebut diatas, untuk mengidentifikasi rute kapal sudah optimal atau tidak, parameternya adalah dengan memulai dengan titik keberangkatan dan diakhiri dengan titik tujuan. Diiringi dengan titik titik antara titik keberangkatan dan titik tujuan hingga membentuk pola garis atau objek satu dimensi. Pola garis ini dapat dilihat pada gambar 1.0 dibawah ini.

Pada gambar 1 kita melihat titik – titik lintasan kapal nelayan yang sedang melakukan kegiatan operasional penangkapan ikan. Dari gambar titik – titik tersebut kita juga melihat bahwa titik tersebut seolah membuat pola garis satu dimensi yang mengindikasikan pola yang sama dan juga pola yang berbeda. Namun demikian gambar diatas belum bisa diidentifikasi mana pola yang sama dan mana pola yang berbeda. Karna data tersebut belum lengkap. Untuk mengidentifikasi pola lintasan kapal tersebut, kita bisa menggunakan dua pendekatan yakni pendekatan bagaimana gambar satu dimensi diidentifikasi seperti yang ditunjukkan pada tabel 1 dan pendekatan kedua dengan menganalisa studi kasus bagaimana mengidentifikasi pola lintasan kapal seperti yang ditunjukkan pada tabel 2 Pengenalan pola lintasan kapal.



Gambar 1. Pola lintasan Operasi Penangkapan Ikan
Sumber: hasil olahan sendiri dari data

Pada tabel 1 Pengenalan Pola pada gambar satu dimensi, direview bagaimana gambar satu dimensi atau gambar pola garis bisa diidentifikasi perbedaannya dengan garis lainnya. Dalam jurnal review ini dibahas mengenai bagaimana pola satu dimensi, baik dari gambar dua dimensi yang direduksi menjadi satu dimensi atau memang awalnya satu dimensi diidentifikasi perbedaan antar garis tersebut. Dalam tabel 2 Pola Pengenalan Lintasan Kapal, mereview penelitian tentang bagaimana Pola lintasan kapal diidentifikasi antara satu lintasan dengan lintasan yang lain. Metode apa yang digunakan untuk mengidentifikasi lintasan lintasan tersebut serta tujuan dari penelitian itu digunakan.

Berdasarkan pembahasan, untuk menjawab rumusan masalah, maka lingkup pembahasan penelitian ini difokuskan pada:

1. Review artikel dengan tema Pengenalan Pola pada gambar satu dimensi
2. Review artikel dengan tema Pola Pengenalan lintasan Kapal.
3. Hubungan Pengenalan Pola gambar satu dimensi dengan Pengenalan Pola lintasan kapal.

Tujuan utama dari penelitian ini adalah untuk menghasilkan pengetahuan tentang research gap didalam penelitian Pengenalan Pola satu dimensi dan pola pengenalan lintasan kapal.

2. Bahan dan Metode

Metodologi Penelitian dalam artikel ini menggunakan metode Review Jurnal, dimana jurnal dikumpulkan dengan cara mencari di google Scholar, di schimago dan lain – lain dengan kata kunci “Pengenalan Objek Satu Dimensi “dan “Pengenalan Pola Lintasan Kapal”. Dari dua kata kunci jurnal yang telah didownload, di buat review dengan tabel 1 review Pengenalan Pola dengan gambar satu dimensi dan tabel 2 review Pengenalan Pola Lintasan Kapal. Didalam tabel – tabel tersebut dibahas tentang variabel yang digunakan, metode yang digunakan juga kesimpulan yang didapatkan. Setelah direview jurnal yang telah didapatkan serta dianalisa metode yang dipakai juga kesimpulan dari penelitian tersebut, maka akan dibuatkan kesimpulan review untuk membantu peneliti dalam memetakan penelitian apa yang telah dilakukan dan juga peluang penelitian yang dapat dilakukan.

3. Hasil

3.1. Pengenalan Pola pada gambar satu dimensi

Pengenalan pola adalah cabang dari kecerdasan buatan yang berfokus pada identifikasi dan interpretasi pola atau struktur yang terdapat dalam data. Dalam konteks ini, kita akan membahas pengenalan pola pada data satu dimensi, di mana data direpresentasikan dalam satu sumbu saja.

Pemrosesan dan pengolahan citra adalah metode untuk melakukan beberapa operasi pada gambar, untuk mendapatkan gambar yang disempurnakan atau untuk mengekstrak beberapa informasi yang berguna di dalamnya (Marpaung et al., 2022). Informasi dari hasil pemrosesan itu yang menjadi pengenalan pola sehingga didapatkan pengetahuan tentang bagaimana setiap pola gambar dapat diidentifikasi.

Gambar satu dimensi adalah representasi visual dari data yang hanya memiliki satu dimensi, yang berarti data tersebut disusun dalam satu urutan atau satu sumbu. Contohnya adalah deret waktu, di mana nilai-nilai diurutkan secara berurutan pada satu sumbu. Ini bisa juga berupa grafik garis sederhana yang menunjukkan hubungan antara satu variabel dengan variabel lainnya. Dalam konteks pengenalan pola, gambar satu dimensi dapat merujuk pada visualisasi data yang terdiri dari satu sumbu atau satu dimensi saja. Contoh umumnya meliputi deret waktu, plot garis, atau grafik satu dimensi lainnya yang memvisualisasikan data sepanjang satu sumbu.

Seluruh review jurnal ini dapat dilihat pada Tabel 1. Taksonomi Review Jurnal Pengenalan Pola dengan gambar satu dimensi dibawah ini.

Tabel 1. Taksonomi Review Jurnal Pengenalan Pola dengan gambar satu dimensi

No	Judul	Variabel	Metode	Hasil
1	A New Development on ANN in China - Biomimetic Pattern Recognition and Multi Weight Vector Neurons [1]	A New Development on ANN in China -Biomimetic, Pattern Recognition dan Multi Weight Vector Neurons	Pengenalan Pola Biomimetik, model matematika dan metode analisis ANN mendapat terobosan: model matematika serbaguna baru telah dikembangkan, yang dapat mensimulasikan semua jenis arsitektur neuron, termasuk model RBF dan BP.	hasilnya Multi Weight Vector Neurons lebih baik dari SVM.
2	Comparative Analysis Of Statistical Pattern Recognition Methods In High Dimensional Settings[2]	QDA, LDA, RDA, KNN, Classification after reducing the dimensionality, Fisher's discriminant plane, Fisher-Fukunaffa-Koonz transform, Fisher-radius plane transform (FR) dan Fisher-variance plane	metode klasifikasi yang didasarkan pada reduksi awal dari dimensi	Hasilnya menunjukkan bahwa jika RDA digunakan, dimensi tinggi kemungkinan besar akan bermanfaat dalam menentukan seberapa baik kelas dapat dipisahkan. Ditemukan bahwa dalam sebagian besar kasus, pengurangan dimensi masalah tertentu dengan

No	Judul	Variabel	Metode	Hasil
				metode ekstraksi fitur yang diselidiki di sini menghasilkan hasil klasifikasi yang lebih rendah dibandingkan dengan yang dicapai oleh RDA dalam ruang fitur penuh.
3	Pattern Recognition Using Invariants Defined from Higher Order Spectra: 2-D Image Inputs[3]	Pattern Recognition, Invariants Defined from Higher Order Spectra dan 2-D Image Inputs	Bispectral Invariants From 1-D Sequences Dan Feature Extraction From 2-D Images	Algoritme baru untuk pengenalan objek dari gambar 2-D berdasarkan fitur spektral tingkat tinggi yang invarian terhadap terjemahan, rotasi, dan penskalaan terbukti akurat membedakan bentuk 2D yang serupa. Ekstraksi fitur prosedur menggabungkan sebagian besar nonlinier yang diperlukan dalam pemetaan dari input ke kelas, memungkinkan penggunaan linier pengklasifikasi
4	Automatic reduction of NMR spectroscopic data for statistical and pattern recognition classification of samples [4]	Automatic reduction of NMR spectroscopic data, statistical dan pattern recognition classification of samples	Experimental	Pendekatan pembuatan segmen otomatis tampaknya menawarkan metode yang cepat dan kuat reduksi data primer memberikan hasil yang setidaknya sebanding dengan pendekatan manual dan pada pada saat yang sama merangkum lebih banyak yang laten informasi dalam spektrum
5	Image Classification and Retrieval are ONE [5]	Image Classification and Retrieval	A Unified Framework, Online Nearest-neighbor Estimation, Object Proposals, Approximate Nearest-neighbor Search dan GPU Acceleration	Hasil percobaan verifikasi bahwa ONE mencapai akurasi tercanggih secara luas berbagai klasifikasi gambar dan tolok ukur pengambilan.

No	Judul	Variabel	Metode	Hasil
6	A Tutorial on Support Vector Machines for Pattern Recognition [6]	Support Vector Machines dan Pattern Recognition	SVM	Hasil beberapa percobaan memberikan banyak contoh dan bukti dari sebagian besar teorema utama.
7	Pattern Recognition As Information Compression [7]	Pattern Recognition Dan Information Compression	Klasifikasi	Dalam hubungan ini perlu ditekankan bahwa setiap metode pengenalan pola struktural bergantung pada metode non-struktural untuk pengenalan primitif
8	Biomimetic (Topological) Pattern Recognition A new Model of Pattern Recognition Theory and Its Application [8]	Biomimetic (Topological) Pattern Recognition A new Model of Pattern Recognition, Theory dan Application	Prinsip Homologi-Kontinuitas (PHC), alat Matematika Pengenalan Pola dan Neural Network.	Dari hasil percobaan diatas dapat kita peroleh kesimpulan berikut: 1) Objek apa pun yang tidak terlatih tidak akan dikenali secara salah dengan BPR. (Dalam percobaan, meskipun garis besarnya punggung hewan sangat mirip, tingkat pengenalan kesalahannya adalah 0.) Itu hanyalah ciri fungsi pengenalan manusia keberadaan dan titik lemah dari Pengakuan Pola tradisional 2) Di BPR setiap kelas sampel dilatih untuk menjadi "dikenali" satu per satu
9	Hybrid of Rough Neural Networks for Arabic/Farsi Handwriting Recognition [9]	Hybrid of Rough Neural Networks dan Arabic/Farsi Handwriting Recognition	hybrid model of rough neural network	Hasilnya diuji pada data standar dan membuktikan efisiensi metode kami. Pendekatan ini secara efisien memilih a metode segmentasi agar sesuai dengan permintaan kami. Pendekatan kami berhasil merancang dan mengimplementasikan jaringan saraf kasar yang berjalan tanpa tuntutan. Setelah itu RS-RNN mampu

No	Judul	Variabel	Metode	Hasil
				memahami angka Arab/Farsi yang manual ditulis oleh pengguna
10	Pattern Recognition and Neural Networks[10]	Pattern Recognition dan Neural Networks	Neural Network	Jaringan Syaraf Tiruan khususnya NN propagasi balik multilapis menyediakan metode yang sederhana namun kuat dan umum untuk mensintesis pengklasifikasi dengan upaya minimal Namun sebagian besar sistem praktis menggabungkan NN dengan teknik lain untuk pra dan pasca pemrosesan Pada tugas pengenalan karakter yang terisolasi, jaringan multilapis dilatih dengan varian propagasi mundur telah mendekati keakuratan manusia
11	Feature Extraction for Object Recognition and Image Classification [11]	Feature Extraction, Object Recognition dan Image Classification	Fuzzy color moments, Gabor Filter, dan Haar Wavelet Decomposition and Wavelet GLCM bentuk.	Hasil yang didapat menunjukkan bahwa Momen Warna Fuzzy bersifat komparatif teknik yang baik di antara teknik yang dijelaskan di atas untuk pengenalan objek
12	Pattern Recognition: An overview [12]	Pattern Recognition	Statistik, data clustering, fuzzy, Neural Network, Structural Patern recognition, syntatic, SVM dan Approximate reasoning approach	Dalam makalah ini Pengenalan pola adalah diperkenalkan meliputi konsep, metode, aplikasi dan integrasi. Pada saat yang sama, sepuluh definisi dan lebih dari sepuluh metode pengenalan pola dirangkum.
13	Prima: A New Pattern Recognition Method [13]	Pattern Recognition Method	PRIMA	Pengalaman sejauh ini menunjukkan bahwa metode PRIMA dapat diterapkan kurang ketat dibandingkan metode pengenalan pola diawasi lainnya.
14	Pattern recognition of fruit shape based on the	Pattern recognition dan neural networks	neural networks	Hasil penelitian menunjukkan bahwa

No	Judul	Variabel	Metode	Hasil
	concept of chaos and neural networks [14]			kegunaan penarik, dimensi fraktal dan jaringan saraf memungkinkan kompleksitas bentuk buah dievaluasi secara kuantitatif.
15	Spectral–Spatial Feature Extraction for Hyperspectral Image Classification: A Dimension Reduction and Deep Learning Approach[15]	Ekstraksi Fitur Spektral–Spasial untuk Hiperspektral Klasifikasi Gambar: Pengurangan Dimensi dan Pendekatan Pembelajaran Mendalam	metode SSFC	Hasil eksperimen pada kumpulan data hiperspektral yang terkenal menunjukkan bahwa metode SSFC yang diusulkan mampu melakukan metode lain yang umum digunakan untuk gambar hiperspektral. klasifikasi
16	Deformation Models for Image Recognition[16]	Deformation Models dan Image Recognition	the Hungarian algorithm	Kami menunjukkan secara eksperimental bahwa model tersebut berkinerja sangat baik
17	Studi Komparasi Bitmap dan Vektor Terhadap Kualitas Digital Art: Case Study Deviantart Nabhan dan PIXELJEFF1995 [17]	Bitmap Dan Vektor	kuantitatif deskriptif	Menurut (Wu, 2020), diagram vektor, juga dikenal sebagai grafik vektor, umumnya menggambarkan grafik dengangaris dan kurva. Elemen-lernen dari garis grafik ini adalah beberapa titik, garis, persegi panjang, lingkaran, busur, dan lain-lain. Semua diperoleh dengan proses operasi matematika. Grafik vektor sering dibentuk dengan mengisi beberapa baris kontur, yang tidak bergantung pada resolusi, sehingga volume file pada umumnya kecil. Keuntungan yang paling signifikan dari gambar vektor adalah gambar tidak berubah apabila diperbesar, diperkecil, atau diputar. Tetapi kualitas gambar rendah, dan warnanya tidak berlimpah.

No	Judul	Variabel	Metode	Hasil
18	An Efficient Algorithm for Raster-to-Vector Data Conversion [18]	Efficient Algorithm dan Raster-to-Vector Data Conversion	algoritma R2V dan Two-Arm Chains Edge Tracing (TACET),	Program berdasarkan algoritma ini lebih cepat dalam memproses gambar RS berukuran besar dibandingkan dengan perangkat lunak komersial seperti ENVI

3.2. Pengenalan Pola Lintasan Kapal

Pengenalan Pola Lintasan Kapal adalah bidang dalam pengenalan pola yang fokus pada analisis dan identifikasi pola dari lintasan atau jalur pergerakan kapal di laut atau perairan lainnya. Tujuannya adalah untuk memahami dan mengklasifikasikan perilaku kapal berdasarkan pola pergerakan yang diamati.

Beberapa aspek yang dapat dianalisis dalam Pengenalan Pola Lintasan Kapal meliputi Kecepatan dan Arah, Durasi Pergerakan, Rute Kapal, Pola Berulang, Klasifikasi Jenis Kapal dan Deteksi Anomali.

Kecepatan dan Arah Menganalisis pola kecepatan dan arah kapal untuk memahami apakah kapal tersebut sedang berlayar, berlabuh, atau berada dalam kondisi khusus lainnya. Durasi Pergerakan Mengukur berapa lama kapal berada dalam suatu wilayah atau melakukan suatu aktivitas tertentu. Rute Kapal: Menganalisis rute atau jalur yang ditempuh oleh kapal, termasuk posisi awal dan akhir serta titik-titik antara keduanya. Pola Berulang Mencari pola pergerakan yang mungkin menunjukkan aktivitas rutin atau kebiasaan kapal. Klasifikasi Jenis Kapal: Mengklasifikasikan jenis kapal berdasarkan karakteristik lintasannya. Misalnya, apakah itu kapal kargo, kapal penangkap ikan, kapal pesiar, dan sebagainya. Deteksi Anomali: Mengidentifikasi pergerakan yang tidak biasa atau aneh yang mungkin memerlukan investigasi lebih lanjut.

Dalam review jurnal ini aspek yang dibahas adalah rute atau jalur yang ditempuh oleh kapal seperti (Wiharta et al., 2022), (Rauzatul Nazzla, Totok Hestirianoto, 2023), (Zhang et al., 2021) dan (L. Wang et al., 2021).

Tabel 2. Taksonomi Review Jurnal Pengenalan Pola lintasan kapal

No	Judul	Variabel	Metode	Hasil
1	Identifikasi Aktivitas Illegal Transshipment Berbasis Kepadatan Point Lintasan Pada Data Ais [19]	Aktivitas, Illegal Transshipment, Kepadatan Point Lintasan Dan Data Ais	DB SCAN	Berdasarkan pengujian dengan metode Silhouette Coefficient, kualitas klaster yang dihasilkan pada kerangka kerja yang dibangun memiliki hasil yang cukup kuat
2	Uji Coba Purnarupa Sistem Pelacakan Jejak Kapal Tradisional dengan Visualisasi Berbasis Web [20]	Purnarupa sistem pelacakan jejak kapal Tradisional dengan visualisasi berbasis web	Prototype	Jika dibandingkan GPS handheld maka alat ini memiliki akurasi dan presisi yang baik diindikasikan dengan nilai selang kepercayaan 95%.

No	Judul	Variabel	Metode	Hasil
3	Ship AIS Trajectory Clustering: An HDBSCAN-Based Approach[21]	Ship AIS Trajectory Clustering dan HDBSCAN	HDBSCAN	Hasil percobaan membuktikan bahwa metode ini mempunyai hasil pengelompokan yang baik pada lintasan kapal di perairan yang kompleks.
4	Adaptive background mixture models for real-time tracking[22]	Adaptive background mixture models dan real-time tracking	Gaussian	Makalah ini telah menunjukkan metode probabilistik yang baru untuk pengurangan latar belakang. Ini melibatkan pemodelan masing-masing piksel sebagai model campuran terpisah. Kami menerapkan metode perkiraan waktu nyata yang stabil dan kokoh. Metode ini hanya membutuhkan dua parameter, α dan T. Kedua parameter ini kuat terhadap perbedaan kamera dan pemandangan yang berbeda.
5	Data mining approach for automatic ship-route design for coastal seas using AIS trajectory clustering analysis[23]	Data mining approach, automatic ship-route design, dan AIS trajectory clustering analysis	FA-DBSCAN, SRU neural network dan LSTM	Hasilnya menunjukkan bahwa algoritma SRU meningkatkan kecepatan dan akurasi pelatihan dibandingkan dengan LSTM, sedangkan rute otomatis yang dihasilkan memenuhi persyaratan praktik navigasi
6	Ship-handling behavior pattern recognition using AIS subtrajectory clustering analysis based on the T-SNE and spectral clustering algorithms dan clustering algorithms[24]	Ship-handling behavior pattern recognition dan sub-trajectory clustering analysis	T-SNE dan algoritma pengelompokan spektral	Pendekatan ini dapat membantu untuk lebih memahami dan memperjelas pola perilaku penanganan kapal sekaligus meningkatkan mesin secara signifikan
7	Ship trajectory uncertainty prediction based on a Gaussian	Ship trajectory uncertainty prediction dan Gaussian Process model	Gaussian Process model	Hasil yang diperoleh menunjukkan bahwa metode yang diusulkan memiliki akurasi prediksi yang tinggi

No	Judul	Variabel	Metode	Hasil
	Process model[25]			dan memenuhi tuntutan aplikasi real-time.
8	Mobility pattern analysis of ship trajectories based on semantic transformation and topic model[26]	Mobility pattern analysis dan semantic transformation and topic model	Latent Dirichlet Allocation	Hasil percobaan menggambarkan bahwa usulan Metode ini efektif untuk pengenalan pola lintasan dan dapat memberikan pola gerak yang dapat dipahami dengan baik
9	A novel method for restoring the trajectory of the inland waterway ship by using AIS data [27]	metode baru dan memulihkan lintasan kapal perairan pedalaman	novel method	Hasilnya menunjukkan bahwa metode yang diusulkan dapat digunakan untuk memulihkan lintasan penuh secara efektif dengan menggunakan data AIS
10	Comparison study on AIS data of ship traffic behavior[28]	AIS data dan ship traffic behavior	Analisis Studi kasus metode statistik	Hal ini menunjukkan bahwa distribusi statistik langsung bisa digunakan untuk mengkarakterisasi posisi lateral, kecepatan, arah dan waktu interval untuk berbagai jenis dan ukuran kapal
11	Analysis of waterway transportation in Southeast Texas waterway based on AIS data [29]	Analysis of waterway transportation dan Ais Data	Analisa studi kasus	Frekuensi konflik kapal diperkirakan mencerminkan risiko kapal tabrakan
12	Detecting anchored fish aggregating devices (FADs) and estimating use patterns from vessel tracking data in small scale tuna fisheries in Indonesia[30]	Deteksi, Perangkat pengumpul ikan dan Pola penggunaan Vessel Tracking data	Density-based spatial clustering of applications with noise (DBSCAN)	hasil dari model aditif umum (GAM) menunjukkan bahwa lama pelayaran lebih berkorelasi dengan total tangkapan dibandingkan dengan jumlah rumpon yang dikunjungi kapal dalam sekali pelayaran.

4. Diskusi

4.1. Pengenalan Pola dengan Gambar Satu Dimensi

Makalah ini mereview jurnal tentang pengenalan pola gambar satu dimensi. Terdapat delapan belas jurnal yang direview dengan topik pengenalan pola gambar satu dimensi. Dari delapan belas jurnal yang di review terbagi menjadi beberapa bagian yang berbeda metode seperti pengenalan pola menggunakan metode Neural Network, metode reduksi dan extract Fiture, usulan metode baru dan SVM serta jurnal review tentang pengenalan pola.

Terdapat enam jurnal pengenalan pola menggunakan metode neural network yakni (S. Wang, 2003), (Xu, n.d.), (Radwan, 2013), (Lecun et al., n.d.), (Liu et al., 2006) dan (Morimoto et al., 2000). Membandingkan metode Multi Weight Vector Neurons dengan Suport vector Machine, hasilnya Multi Vector Neurons lebih baik dari SVM (S. Wang, 2003). Menggunakan metode Prinsip Homologi-Kontinuitas (PHC), alat Matematika Pengenalan Pola dan Neural Network, menghasilkan kesimpulan berikut Objek apa pun yang tidak terlatih tidak akan dikenali secara salah dengan BPR. (Dalam percobaan, meskipun garis besarnya punggung hewan sangat mirip, tingkat pengenalan kesalahannya adalah 0.) Itu hanyalah ciri fungsi pengenalan manusia keberadaan dan titik lemah dari Pengakuan Pola tradisional dan Di BPR setiap kelas sampel dilatih untuk menjadi "dikenali" satu per satu. metode hybrid model of rough neural network yang dipakai dalam penelitian ini menghasilkan diuji pada data standar dan membuktikannya efisiensi metode kami. Pendekatan ini secara efisien memilih sebuah metode segmentasi agar sesuai dengan permintaan kami. Pendekatan ini berhasil merancang dan mengimplementasikan jaringan saraf kasar yang berjalan tanpa tuntutan. Setelah itu RS-RNN mampu memahami angka Arab/Farsi yang manual ditulis oleh pengguna (Radwan, 2013). Neural Network metode yang digunakan dalam peneltian ini menghasilkan khususnya NN propagasi balik multilapis menyediakan metode yang sederhana namun kuat dan umum untuk mensintesis pengklasifikasi dengan upaya minimal Namun sebagian besar sistem praktis menggabungkan NN dengan teknik lain untuk pra dan pasca pemrosesan Pada tugas pengenalan karakter yang terisolasi, jaringan multilapis dilatih dengan varian propagasi mundur telah mendekati keakuratan manusia (Lecun et al., n.d.). Penelitian (Liu et al., 2006) menggunakan metode Statistik, data clustering, fuzzy, Neural Network, Structural Patern recognition, syntatic, SVM dan Approximate reasoning approach dan menghasilkan kesimpulan Dalam makalah ini Pengenalan pola adalah diperkenalkan meliputi konsep, metode, aplikasi dan integrasi, Pada saat yang sama, sepuluh definisi dan lebih dari sepuluh metode pengenalan pola dirangkum. Hasil penelitian (Morimoto et al., 2000) yang menggunakan metode Neural Network menunjukkan bahwa kegunaan penarik, dimensi fraktal dan jaringan saraf memungkinkan kompleksitas bentuk buah dievaluasi secara kuantitatif.

Jurnal menggunakan metode reduksi dan ekstrak fiture sebanyak enam jurnal yakni (Aeberhard et al., 1994), (Chandran et al., 1997), (Spraul et al., 1994), (Zhao & Du, 2016), (Tiwari et al., 2013) dan (Watanabe, n.d.). Metode reduksi yang digunakan dengan mereduksi dimensi gambar dua dimensi menjadi satu dimensi dalam penelitian ini (Aeberhard et al., 1994) menghasilkan bahwa jika RDA digunakan, dimensi tinggi kemungkinan besar akan bermanfaat dalam menentukan seberapa baik kelas dapat dipisahkan. Ditemukan bahwa dalam sebagian besar kasus, pengurangan dimensi masalah tertentu dengan metode ekstraksi fitur yang diselidiki di sini menghasilkan hasil klasifikasi yang lebih rendah dibandingkan dengan yang dicapai oleh RDA dalam ruang fitur penuh. Menggunakan metode Bispectral Invariants From 1-D Sequences Dan Feature Extraction From 2-D Images, penelitian ini menghasilkan kesimpulan bahwa Algoritme baru untuk pengenalan objek dari gambar 2-D berdasarkan fitur spektral tingkat tinggi yang invarian terhadap terjemahan, rotasi, dan penskalaan terbukti akurat membedakan bentuk 2D yang serupa, Ekstraksi fitur prosedur menggabungkan sebagian besar nonlinier yang diperlukan dalam pemetaan dari input ke kelas, memungkinkan penggunaan linier pengklasifikasi (Chandran et al., 1997). Makalah ini menggunakan metode experimental dalam penelitiannya sehingga menghasilkan kesimpulan

Algoritme baru untuk pengenalan objek dari gambar 2-D berdasarkan fitur spektral tingkat tinggi yang invarian terhadap terjemahan, rotasi, dan penskalaan terbukti akurat membedakan bentuk 2D yang serupa. Ekstraksi fitur prosedur menggabungkan sebagian besar nonlinier yang diperlukan dalam pemetaan dari input ke kelas, memungkinkan penggunaan linier pengklasifikasi (Spraul et al., 1994). Memakai algoritma SSFC, penelitian ini menghasilkan kesimpulan bahwa Hasil eksperimen pada kumpulan data hiperspektral yang terkenal menunjukkan bahwa metode SSFC yang diusulkan mampu melakukan metode lain yang umum digunakan untuk gambar hiperspektral (Zhao & Du, 2016). Penelitian ini menggunakan algoritma Fuzzy color moments, Gabor Filter, dan Haar Wavelet Decomposition and Wavelet GLCM bentuk, menghasilkan kesimpulan Hasil yang didapat menunjukkan bahwa Momen Warna Fuzzy bersifat komparatif teknik yang baik di antara teknik yang dijelaskan di atas untuk pengenalan objek (Tiwari et al., 2013). Algoritma klasifikasi yang digunakan dalam penelitian ini menyimpulkan bahwa Dalam hubungan ini perlu ditekankan bahwa setiap metode pengenalan pola struktural bergantung pada metode non-struktural untuk pengenalan primitif (Watanabe, n.d.).

Penggunaan metode baru dan SVM sejumlah enam jurnal (Mining et al., 1998), (Juricskay & Veress, 1985), (Xie et al., n.d.), (Keysers et al., 2007), (Pixeljeff, 2020), dan (Teng et al., 2019). Mengusulkan metode PRIMA menghasilkan kesimpulan bahwa metode PRIMA dapat diterapkan kurang ketat dibandingkan metode pengenalan pola diawasi lainnya (Juricskay & Veress, 1985). Menggunakan algoritma Support Vector Machine (SVM) menghasilkan kesimpulan bahwa beberapa percobaan memberikan banyak contoh dan bukti dari sebagian besar teorema utama (Mining et al., 1998). Algoritma A Unified Framework, Online Nearest-neighbor Estimation, Object Proposals, Approximate Nearest-neighbor Search dan GPU Acceleration menghasilkan kesimpulan Hasil percobaan verifikasi bahwa ONE mencapai akurasi tercanggih secara luas berbagai klasifikasi gambar dan tolok ukur pengambilan (Xie et al., n.d.). penggunaan algoritma Hungarian menunjukkan bahwa algoritma tersebut bekirneja baik (Keysers et al., 2007). Menggunakan metode kuantitatif kualitatif menghasilkan kesimpulan Menurut (Wu, 2020), diagram vektor, juga dikenal sebagai grafik vektor, umumnya menggambarkan grafik dengangaris dan kurva. Elemen-lernen dari garis grafik ini adalah beberapa titik, garis, persegi panjang, lingkaran, busur, dan lain-lain. Semua diperoleh dengan proses operasi matematika. Grafik vektor sering dibentuk dengan mengisi beberapa baris kontur, yang tidak bergantung pada resolusi, sehingga volume file pada umumnya kecil. Keuntungan yang paling signifikan dari gambar vektor adalah gambar tidak berubah apabila diperbesar, diperkecil, atau diputar. Tetapi kualitas gambar rendah, dan warnanya tidak berlimpah (Pixeljeff, 2020). algoritma R2V dan Two-Arm Chains Edge Tracing (TACET) menghasilkan kesimpulan Program berdasarkan algoritma ini lebih cepat dalam memproses gambar RS berukuran besar dibandingkan dengan perangkat lunak komersial seperti ENVI (Teng et al., 2019).

4.2. Pengenalan Pola Lintasan Kapal

Makalah ini mereview jurnal tentang Pengenalan Pola Lintasan Kapal. Terdapat dua belas jurnal yang di review dan terbagi kedalam empat kelompok metode yang digunakan yakni metode DBSCAN, Prototype dan Studi kasus, Gaussian serta metode pengelompokan Latent Dirichlet Allocation dan usulan metode baru. Makalah yang direview ini juga dapat dilihat pada tabel 2..

Tabel 2 Taksonomi Review Jurnal Pengenalan Pola lintasan kapal menunjukan terdapat empat jurnal yang menggunakan metode DBSCAN dalam penelitiannya. Tiga jurnal dengan menggunakan metode Prototype dan Studi kasus. Metode Gaussian sebanyak dua jurnal dan metode pengelompokan Latent Dirichlet Allocation dan usulan metode baru sebanyak tiga jurnal.

Metode DBSCAN digunakan dalam penelitian (Wiharta et al., 2022), (L. Wang et al., 2021) (Zhang et al., 2021) dan (Widyatmoko, 2018). Metode DBSCAN yang digunakan dalam penelitian ini (Wiharta et al., 2022) menghasilkan kesimpulan Berdasarkan pengujian dengan metode Silhouette Coefficient, kualitas klaster yang dihasilkan pada kerangka kerja yang dibangun memiliki hasil yang

cukup kuat. Penelitian (L. Wang et al., 2021) menghasilkan kesimpulan Hasil percobaan membuktikan bahwa metode ini mempunyai hasil pengelompokan yang baik pada lintasan kapal di perairan yang kompleks. Jurnal (Zhang et al., 2021) menghasilkan Hasilnya menunjukkan bahwa algoritma SRU meningkatkan kecepatan dan akurasi pelatihan dibandingkan dengan LSTM, sedangkan rute otomatis yang dihasilkan memenuhi persyaratan praktik navigasi. Makalah (Widyatmoko, 2018) menghasilkan kesimpulan dengan hasil dari model aditif umum (GAM) menunjukkan bahwa lama pelayaran lebih berkorelasi dengan total tangkapan dibandingkan dengan jumlah rumpun yang dikunjungi kapal dalam sekali pelayaran.

Algoritma Gaussian digunakan dalam penelitian (Stauffer & Grimson, n.d.), (Rong et al., 2019) dan (Stauffer & Grimson, n.d.). menggunakan algoritma Gaussian yang menghasilkan kesimpulan menunjukkan metode probabilistik yang baru untuk pengurangan latar belakang. Ini melibatkan pemodelan masing-masing piksel sebagai model campuran terpisah. Kami menerapkan metode perkiraan waktu nyata yang stabil dan kokoh. Metode ini hanya membutuhkan dua parameter, α dan T . Kedua parameter ini kuat terhadap perbedaan kamera dan pemandangan yang berbeda. Dan (Rong et al., 2019) mendapatkan kesimpulan bahwa metode yang Gaussian memiliki akurasi prediksi yang tinggi dan memenuhi tuntutan aplikasi real-time.

Prototype dan Studi kasus digunakan dalam jurnal (Rauzatul Nazzla, Totok Hestirianoto, 2023), (Xiao et al., 2015) dan (Wu et al., 2016). Metode Prototype yang digunakan dalam penelitian ini menghasilkan kesimpulan Jika dibandingkan GPS handheld maka alat ini memiliki akurasi dan presisi yang baik diindikasikan dengan nilai selang kepercayaan 95% (Rauzatul Nazzla, Totok Hestirianoto, 2023). Menggunakan metode studi kasus dan statistik menghasilkan kesimpulan bahwa distribusi statistik langsung bisa digunakan untuk mengkarakterisasi posisi lateral, kecepatan, arah dan waktu interval untuk berbagai jenis dan ukuran kapal (Xiao et al., 2015) dan jurnal (Wu et al., 2016) menggunakan metode studi kasus menghasilkan kesimpulan Frekuensi konflik kapal diperkirakan mencerminkan risiko kapal tabrakan.

Metode pengelompokan Latent Dirichlet Allocation dan usulan metode baru dipakai dalam makalah (Gao & Shi, 2020), (Huang et al., 2020) dan (Sang et al., 2015). Pada penelitian (Gao & Shi, 2020) menggunakan metode T-SNE dan algoritma pengelompokan spektral menghasilkan kesimpulan Pendekatan ini dapat membantu untuk lebih memahami dan memperjelas pola perilaku penanganan kapal sekaligus meningkatkan mesin secara signifikan. Metode Latent Dirichlet Allocation yang digunakan dalam penelitian (Huang et al., 2020) menghasilkan kesimpulan Hasil percobaan menggambarkan bahwa usulan Metode ini efektif untuk pengenalan pola lintasan dan dapat memberikan pola gerak yang dapat dipahami dengan baik. Dan jurnal (Sang et al., 2015) mengusulkan metode baru dalam penelitian, menghasilkan kesimpulan metode yang diusulkan dapat digunakan untuk memulihkan lintasan penuh secara efektif dengan menggunakan data AIS.

4.3. Hubungan antara Pengenalan Pola dengan gambar satu dimensi dan Pengenalan Pola lintasan kapal

Melihat Metode Pengenalan Pola dengan gambar satu dimensi dan Pengenalan Pola lintasan kapal yang direview variabel, metode dan hasilnya memang tidak ada hubungan antara kedua topik tersebut. Namun jika melihat Pengenalan Pola lintasan kapal yang menjadikan Parameter lokasi satu ke lokasi yang lain sehingga membentuk garis lintasan sebagai input dari penelitian dan Pengenalan Pola dengan gambar satu dimensi yang berupa garis seperti penelitian Pattern recognition of fruit shape based on the concept of chaos and neural networks [14] yang menjadikan Pola dua dimensi menjadi satu dimensi atau garis, sama seperti lintasan kapal yang dibentuk oleh titik lokasi membentuk satu garis lintasan. Begitupun dengan penelitian lain yang ada di tabel 1 Taksonomi Review Jurnal Pengenalan Pola dengan gambar satu dimensi.

5. Kesimpulan

Dari tiga puluh jurnal yang telah direview terdapat beberapa jurnal yang merujuk dengan topik yang sama. Pertama yaitu pengenalan pola dengan gambar satu dimensi. Dari delapan belas jurnal yang di review terbagi menjadi beberapa bagian yang berbeda metode seperti pengenalan pola menggunakan metode Neural Network, metode reduksi dan extract Fiture, usulan metode baru dan SVM serta jurnal review tentang pengenalan pola. Kedua, pengenalan pola lintasan kapal. Terdapat dua belas jurnal yang di review dan terbagi kedalam empat kelompok metode yang digunakan yakni metode DBSCAN, Prototype dan Studi kasus, Gaussian serta metode pengelompokan Latent Dirichlet Allocation dan usulan metode baru. Ketiga, penelitian pola dengan gambar satu dimensi dan Pola lintasan kapal memiliki persamaan menjadikan Pola dengan gambar satu dimensi yang dimiliki sebagai untuk identifikasi objek.

Referensi

- 1 S. Wang, "A New Development on ANN in China – Biomimetic Pattern Recognition and Multi Weight Vector Neurons," pp. 35–43, 2003.
- 2 S. Aeberhard, D. Coomans, and O. D. E. Velt, "Pergamon COMPARATIVE ANALYSIS OF STATISTICAL PATTERN D I M E N S I O N A L SETTINGS," vol. 27, no. 8, 1994.
- 3 V. Chandran, B. Carswell, B. Boashash, and S. Member, "Pattern Recognition Using Invariants Defined from Higher Order Spectra : 2-D Image Inputs," vol. 6, no. 5, pp. 703–712, 1997.
- 4 M. Spraul, P. Neidig, U. Klauck, P. Kessler, E. Holmes, and J. K. Nicholson, "Automatic reduction of NMR spectroscopic data for statistical and pattern recognition classification of samples," vol. 12, no. 10, pp. 1215–1225, 1994.
- 5 L. Xie, R. Hong, B. Zhang, and Q. Tian, "Image Classification and Retrieval are ONE," pp. 3–10.
- 6 D. Mining, K. Discovery, B. Laboratories, and L. Technologies, "A Tutorial on Support Vector Machines for Pattern Recognition," vol. 167, pp. 121–167, 1998.
- 7 S. Watanabe, PATTERN RECOGNITION AS INFORMATION COMPRESSION, Second Edi. ACADEMIC PRESS, INC.
- 8 W. S. C. Xu, "A new Model of Pattern Recognition Theory and Its Application," pp. 2258–2262.
- 9 E. Radwan, "Hybrid of Rough Neural Networks for Arabic / Farsi Handwriting Recognition," vol. 2, no. 2, pp. 39–47, 2013.
- 10 Y. Lecun, Y. Bengio, and U. De Montreal, "Pattern Recognition and Neural Networks," pp. 1–24.
- 11 A. Tiwari, A. K. Goswami, and M. Saraswat, "Feature Extraction for Object Recognition and Image Classification," vol. 2, no. 10, pp. 1238–1246, 2013.
- 12 J. Liu, J. Sun, and S. Wang, "Pattern Recognition : An overview," vol. 6, no. 6, pp. 57–61, 2006.
- 13 I. Juriskay and G. E. Veress, "Prima: a new pattern recognition method," vol. 171, pp. 61–76, 1985.
- 14 T. Morimoto, T. Takeuchi, H. Miyata, and Y. Hashimoto, "Pattern recognition of fruit shape based on the concept of chaos and neural networks," vol. 26, pp. 171–186, 2000.
- 15 W. Zhao and S. Du, "Spectral – Spatial Feature Extraction for Hyperspectral Image Classification : A Dimension Reduction and Deep Learning Approach," pp. 1–11, 2016.
- 16 D. Keysers, T. Deselaers, S. Member, C. Gollan, and H. Ney, "Deformation Models for Image Recognition," vol. 29, no. 8, pp. 1422–1435, 2007.

- 17 D. A. N. Pixeljeff, "STUDI KOMPARASI BITMAP DAN VEKTOR TERHADAP KUALITAS DIGITAL ART : CASE STUDY DEVIANTART NABHAN," vol. 1, 2020.
- 18 J. Teng, F. Wang, and Y. Liu, "An Efficient Algorithm for Raster-to-Vector Data Conversion An Efficient Algorithm for Raster-to-Vector Data Conversion," no. June 2008, 2019, doi: 10.1080/10824000809480639.
- 19 D. M. Wiharta et al., "IDENTIFIKASI AKTIVITAS ILLEGAL TRANSSHIPMENT BERBASIS," vol. 5, no. 1, pp. 38–46, 2022.
- 20 S. P. Rauzatul Nazzla, Totok Hestirianoto, "Uji coba purnarupa sistem pelacakan jejak kapal tradisional dengan visualisasi berbasis web," vol. 3, pp. 61–75, 2023.
- 21 L. Wang, P. Chen, L. Chen, and J. Mou, "Ship AIS Trajectory Clustering : An HDBSCAN-Based Approach," 2021.
- 22 C. Stauffer and W. E. L. Grimson, "Adaptive background mixture models for real-time tracking."
- 23 D. Zhang, Y. Zhang, and C. Zhang, "Data mining approach for automatic ship-route design for coastal seas using AIS trajectory clustering analysis," *Ocean Eng.*, vol. 236, no. July, p. 109535, 2021, doi: 10.1016/j.oceaneng.2021.109535.
- 24 M. Gao and G. Shi, "Ship-handling behavior pattern recognition using AIS sub-trajectory clustering analysis based on the T-SNE and spectral clustering algorithms," *Ocean Eng.*, vol. 205, no. April, p. 106919, 2020, doi: 10.1016/j.oceaneng.2020.106919.
- 25 H. Rong, A. P. Teixeira, and C. G. Soares, "Ship trajectory uncertainty prediction based on a Gaussian Process model," *Ocean Eng.*, vol. 182, no. December 2018, pp. 499–511, 2019, doi: 10.1016/j.oceaneng.2019.04.024.
- 26 L. Huang, Y. Wen, W. Guo, X. Zhu, C. Zhou, and F. Zhang, "Mobility pattern analysis of ship trajectories based on semantic transformation and topic model," *Ocean Eng.*, vol. 201, no. February, p. 107092, 2020, doi: 10.1016/j.oceaneng.2020.107092.
- 27 L. Sang, A. Wall, Z. Mao, X. Yan, and J. Wang, "A novel method for restoring the trajectory of the inland waterway ship by using AIS data," *Ocean Eng.*, vol. 110, pp. 183–194, 2015, doi: 10.1016/j.oceaneng.2015.10.021.
- 28 F. Xiao, H. Ligteringen, C. Van Gulijk, and B. Ale, "Comparison study on AIS data of ship traffic behavior," *Ocean Eng.*, vol. 95, pp. 84–93, 2015, doi: 10.1016/j.oceaneng.2014.11.020.
- 29 X. Wu, A. L. Mehta, V. A. Zaloom, and B. N. Craig, "Analysis of waterway transportation in Southeast Texas waterway based on AIS data," *Ocean Eng.*, vol. 121, pp. 196–209, 2016, doi: 10.1016/j.oceaneng.2016.05.012.
- 30 A. C. Widyatmoko, "Detecting anchored fish aggregating devices (FADs) and estimating use patterns from vessel tracking data in small scale tuna fisheries in Indonesia," no. September, 2018.



© 2019 oleh penulis. Dikirim untuk kemungkinan publikasi akses terbuka di bawah syarat dan ketentuan lisensi Creative Commons Attribution (CC BY) (<http://creativecommons.org/licenses/by/4.0/>).

