

Jurnal Ilmiah

Informatika Komputer

- IMPACT OF WEB QUALITY ON EMPLOYEES' SATISFACTION IN INDONESIA STATE
EMPLOYMENT DEPARTMENT 161
Adhitiro Novanto, Tety Elida Siregar, Taufik Hidayat
- ALAT PENDETEKSI PELANGGARAN GARIS HENTI KENDARAAN PADA PERSIMPANGAN
LALU LINTAS SATU ARAH MENGGUNAKAN SENSOR LASER BERBASIS ARDUINO
MEGA 2560 170
Rudi Kurniawan, Veronica Ernita Kristianti, Alona Situmeang
- PEMBUATAN APLIKASI KLASIFIKASI CITRA DAUN MENGGUNAKAN WARNA RGB
DAN HSV 180
Haura Sanusi, Suryadi H.S., Diana Tri Susetianingtias
- IMPLEMENTASI ALGORITMA K-MEANS CLUSTERING PADA ANALISIS SENTIMEN
KELUHAN PENGGUNA INDOSAT 191
Tri Iryanto Saputra, Rini Arianty
- PENGENALAN OBJEK MAKANAN CEPAT SAJI PADA VIDEO DAN REAL TIME WEBCAM
MENGGUNAKAN METODE YOU LOOK ONLY ONCE (YOLO) 199
Oktaviani Ella Karlina, Dina Indarti
- DETEKSI SIMILARITAS DOKUMEN ILMIAH MENGGUNAKAN ALGORITMA RABIN-
KARP 209
Hermas Yuda Pamungkas, Fitrianiingsih
- KLASIFIKASI CITRA GENUS PANTHERA MENGGUNAKAN METODE CONVOLUTIONAL
NEURAL NETWORK (CNN) 220
Gusti Alfahmi Anwar, Desti Rimirasih

DEWAN REDAKSI JURNAL ILMIAH INFORMATIKA KOMPUTER

Penanggung Jawab

Prof. Dr. E.S. Margianti, S.E., M.M.
Prof. Suryadi Harmanto, SSI., M.M.S.I.
Drs. Agus Sumin, M.M.S.I.

Dewan Editor

Dr. Fitriarningsih, Universitas Gunadarma
Dr. Rodiah, Universitas Gunadarma
Dr. Ias Sri Wahyuni, S.Si, M.Si, Universitas Gunadarma

Mitra Bebestari

Prof. Dr. Rer. Nat. A. Benny Mutiara, Universitas Gunadarma
Prof. Dr. Achmad Nizar Hidayanto, Universitas Indonesia
Prof. Dr.-Ing. Adang Suhendra, S.Si, S.Kom, M.Sc, Universitas Gunadarma
Prof. Dr. Sarifuddin Madenda, Universitas Gunadarma
Surya Sumpeno, PhD, Institut Teknologi Sepuluh Nopember (ITS)
Noor Akhmad Setiawan, Ph.D, Universitas Gadjah Mada
Dr. Paulus Insap Santosa, Universitas Gadjah Mada
Dr. Agus Harjoko, M.Sc, Ph.D, Universitas Gadjah Mada
Dr. Ernastuti, Universitas Gunadarma
Dewi Agushinta R., Universitas Gunadarma
Dr. Eri Prasetyo, Universitas Gunadarma
Dr. Sunny Arief Sudiro, STMIK Jakarta STI&K
Dr. Tubagus Maulana Kusuma, S.Kom.,Mengsc, Universitas Gunadarma
Dr. Lussiana ETP, S.si., M.T, STMIK Jakarta STI&K
Harry Budi Santoso, Ph.D, Universitas Indonesia

Sekretariat Redaksi

Universitas Gunadarma
Jalan Margonda Raya No. 100 Depok 16424
Phone : (021) 78881112 ext 516.

JURNAL ILMIAH INFORMATIKA KOMPUTER

NOMOR 3, VOLUME 24, DESEMBER 2019

DAFTAR ISI

IMPACT OF WEB QUALITY ON EMPLOYEES' SATISFACTION IN INDONESIA STATE EMPLOYMENT DEPARTMENT Adhitiro Novanto, Tety Elida, Taufik Hidayat	161
ALAT PENDETEKSI PELANGGARAN GARIS HENTI KENDARAAN PADA PERSIMPANGAN LALU LINTAS SATU ARAH MENGGUNAKAN SENSOR LASER BERBASIS ARDUINO MEGA 2560 Rudi Kurniawan, Veronica Ermita Kristianti, Alona Situmeang	170
PEMBUATAN APLIKASI KLASIFIKASI CITRA DAUN MENGGUNAKAN RUANG WARNA RGB DAN HSV Haura Sanusi, Suryadi H.S. Diana Tri Suseningtias	180
IMPLEMENTASI ALGORITMA K-MEANS CLUSTERING PADA ANALISIS SENTIMEN KELUHAN PENGGUNA INDOSAT Try Iryanto Saputra, Rini Arianty	191
PENGENALAN OBJEK MAKANAN CEPAT SAJI PADA VIDEO DAN <i>REAL TIME WEBCAM</i> MENGGUNAKAN METODE <i>YOU LOOK ONLY ONCE</i> (YOLO) Oktaviani Ella Karlina Dina Indarti	199
DETEKSI SIMILARITAS DOKUMEN ILMIAH MENGGUNAKAN ALGORITMA RABIN-KARP Hermas Yuda Pamungkas, Fitrianingih	209
KLASIFIKASI CITRA GENUS PANTHERA MENGGUNAKAN METODE CONVOLUTIONAL NEURAL NETWORK (CNN) Gusti Alfahmi Anwar, Desti Rimirasih	220

IMPACT OF WEB QUALITY ON EMPLOYEES' SATISFACTION IN INDONESIA STATE EMPLOYMENT DEPARTMENT

¹Adhitiro Novanto, ²Tety Elida, ³Taufik Hidayat

¹Business Information System, ²Management Department, ³Information System, Gunadarma University

^{1,2,3}Margonda Raya 100, Depok, West Java, 16435, Indonesia

²tety@staff.gunadarma.ac.id

Abstract

The State Employment Department (BKN) had applied information technology in employee data management since 2011. However, flaws were found in the system, such as occurrence of inaccurate data and complicated navigation. This study was aimed at analyzing the satisfaction of BKN staffs in using DMS web. Web quality was assessed using 3 dimensions: usability, information quality and service information quality. These 3 dimensions were then associated with the satisfaction of BKN staffs as the users. Data was collected through questionnaires to 206 staffs with authorized accesses to the web. The results showed that the quality of three dimensions of the web influenced users' satisfaction.

Keywords: *E-government, employee satisfaction, user satisfaction, web quality, technology administration*

INTRODUCTION

The state employment department (BKN) is an Indonesian government institution managing documents of government employees, including in electronic forms. Employee performance is managed via web <https://dms.bkn.go.id>. Problems encountered in the system were occurrence of inaccurate data and complicated navigation. Accurate and updated employee data was of absolute necessity as it concerned with the management of human resources performance within government institutions.

Information technology in government institutions created efficient management system [1][2][3], and elevated the effectiveness of human resources management [4]. [5] figured out that the failure of e-

government in developing countries was caused by lacking sustainability of the application system and insufficient performance of the application system to meet the expected results. Another issue was inadequate infrastructure provision that possibly hindered the application use [6] [7].

The utilization of the web <https://dms.bkn.go.id> had never been evaluated despite having implemented since 2011. Evaluation was urgent as flaws triggered serious issues. It could be done through analyzing the web quality. The concept of measurement quality such as SERVQUAL was applicable to accomplish evaluation [8] [9]. Yoo and Donthu stated that the quality of a web could be measured through 4 dimensions; ease of use, aesthetic design, processing speed, and interactive respon-

siveness [10]. Huizingh focused on content and design [11]. Lociacono, Watson & Goodhue, developed web quality measurement through measurement of interaction quality between users and the web [12].

Barnes and Vigen formulated 5 dimensions to measure web quality: usability, design, information, trust, and empathy [13]. Yang et al also stated that web service quality should be assessed by using 5 factors, such as usability, usefulness of content, adequacy of information, accessibility and interaction [14]. Meanwhile Parasuraman, Zeithaml & Malhotra stated that e-service quality included efficiency, fulfillment, system availability and privacy [15].

Barnes & Vidgen and Al-Manasra, Zaid, & TaherQutaishaf stated that 3 main factors used to measure web quality were information quality, interaction and service quality, and information content [13] [16]. This study was to analyze the quality of DMS web associated with BKN employee satisfaction as web users.

RESEARCH METHODOLOGY

The study was intended for BKN employees with access authorities to

<https://dms.bkn.go.id> web, in BKN centre. There were 206 employees having web accesses. Data was collected through questionnaires made in *Likert* scale 1 to 5. Score 1 indicated most negative response, meanwhile score 5 indicated most positive response.

Web quality was measured using 3 variables, namely usability, information quality and service interaction quality [17]. The model of the study was showed in Figure 1, showing DMS web user satisfaction: usability, information quality and service interaction quality.

Usability was quality concerning site layouts, such as interface, user-friendly, navigation and overview for users [17]. *Information Quality* was quality of web content; information accuracy, format and relevance [17]. *Service Interaction Quality* was quality of service interaction utilized by users when exploring deeper into the web, such as issue of transaction security and information safety, personalization and communication [17]. The correlation between web quality variable and user satisfaction was analyzed using linear regression.

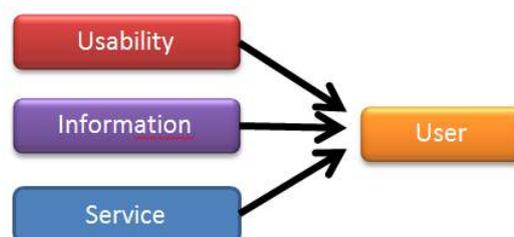


Figure 1. Model of the Study

RESULTS AND DISCUSSION

Respondent Description

From 206 questionnaires distributed in BKN center, 200 respondents submitted the answers. Distribution of educational background, age and gender was showed in Figure 2.

The majority of employees in BKN center that manage employment data came from senior high school background (54%), and most of them were beyond 40 years old (86%) and 70% of employees were male.

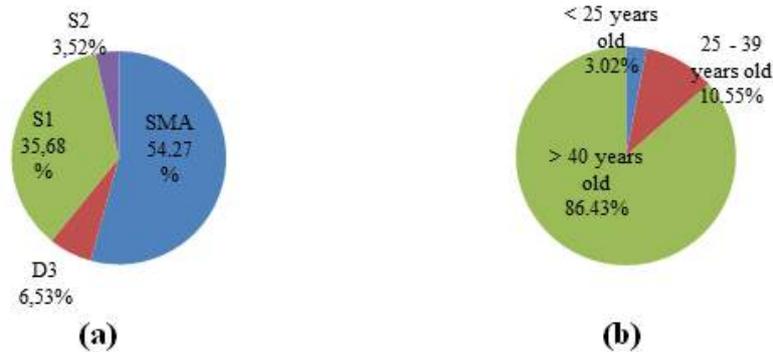


Figure 2. Respondent Distribution Based on Educational Background and Age

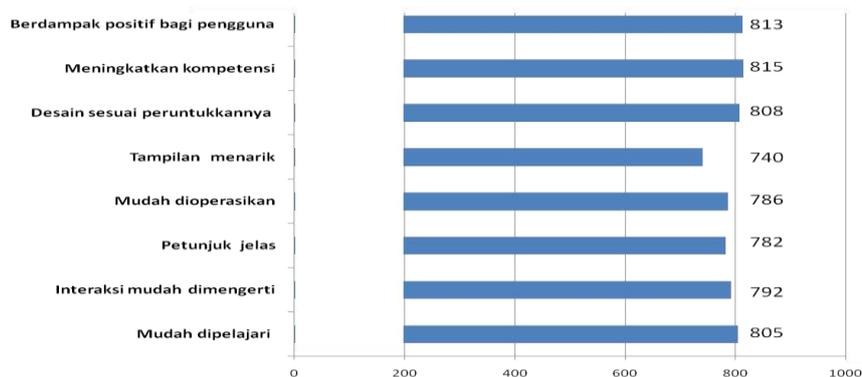


Figure 3. Respondent Assessment on Usability

In reference to Figure 3, there were several aspects respondents considered inadequate, such as unattractive web interface, easy-to-use and clarity of instructions. Web interface was considered basic, probably as DMS web was not meant for commercial web but application web. In DMS web, instruction and navigation were not enhanced with support menu so users encountered difficulties in understanding the navigation functions.

In general, respondent assessment on information quality indicated dissatisfaction

(Figure 4). Several aspects to improve were details of data, data accuracy and punctuality of data presentation. In several occurrences, inaccurate data emerged as showed in Figure 5. Search result showed inaccuracy between employee ID number and letter of staffing decision.

The role and precision of scanning staffs should be emphasized as to diminish negligence during document uploading to guarantee accurate, relevant, and trust-worthy data provision.

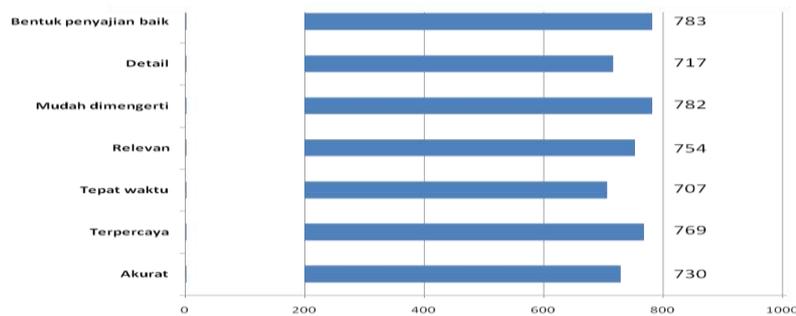


Figure 4. Respondent Assessment Scores on Information Quality

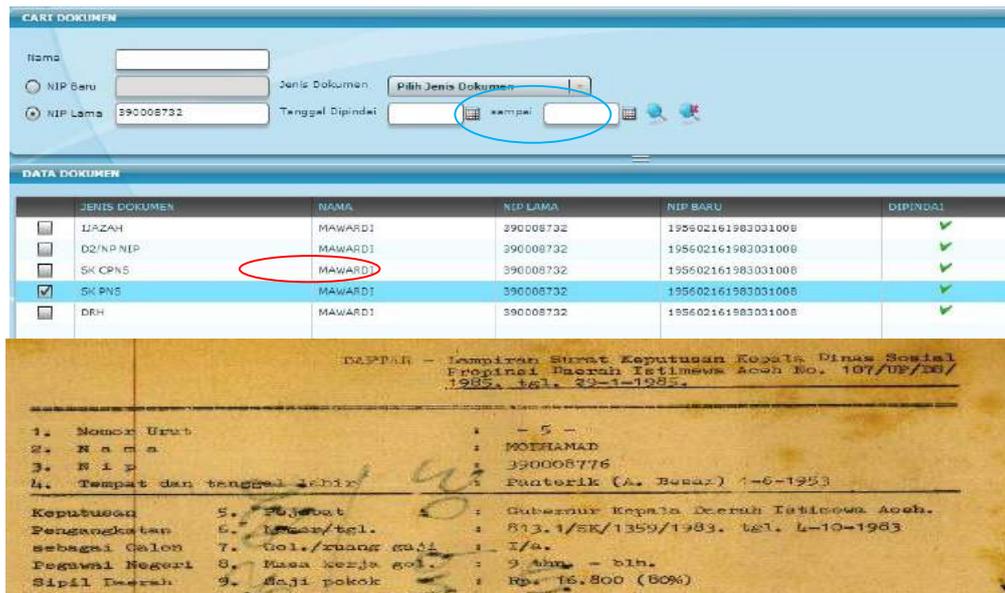


Figure 5. Sample of Inaccurate Staffing Decree

Figure 6 showed respondent assessment scoring towards service interaction quality. The results showed that data security should be enhanced. The validation team in employment management should be empowered well to ensure valid information.

The Influence of Web Quality Towards DMS Web User Satisfaction

Table 2 (a) showed that usability, influenced BKN staffs satisfaction as the users. To prove so, the coefficient regression equation on this variable showed significance score (a) lower than 0.05. Similar outcomes

were also resulted from other variables; information quality and service information quality. The significant value (b and c) on regression equation coefficient showed scores lower than 0.05. To conclude, the satisfaction of DMS web users was influenced by usability, information quality, and service information quality of the web.

The three variables: usability, information quality, service information quality also simultaneously influenced satisfaction of web users which was indicated by significant value of 0.000 in Table 2 (b).

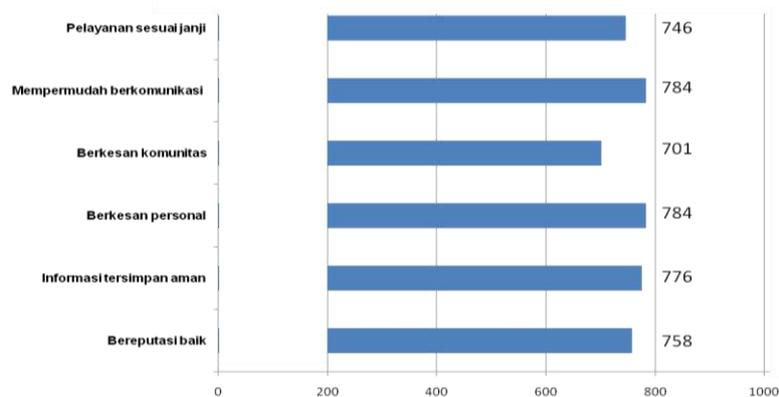


Figure 6. Respondent Assessment Scores on Service Interaction Quality

Table 2 (a). Coefficients of Regression Equation

Model	Unstandardized Coefficients		Standardized Coefficients	Sig.
	B	Std. Error	Beta	
(Constant)	6.037	1.632		.000
Usability	.217	.063	.170	.001 (a)
Information Quality	.435	.070	.357	.000 (b)
Service information Quality	.625	.075	.438	.000 (c)

Dependent variable: user satisfaction

Tabel 2 (b). Anava

Model	Sum of Squares	df	Mean Square	F	Sig.
Regression	2792.966	3	930.989	163.444	.000 ^a
Residual	1116.429	196	5.696		
Total	3909.395	199			

a. Predictors: (Constant), Usability, Information Quality, Service Information Quality

b. Dependent Variable: User Satisfaction

Table 2 (c). Determinant Coefficient

Model	R	R Square	Adjusted R Square	Std. Error of the Estimate
1	.845 ^a	.714	.710	2.387

a. Predictors: (Constant), Usability, Information Quality, Service Information Quality

The influence of the three variables towards web users' satisfaction was indicated by determinant coefficient value of 0.710 (Table 2(c)), meaning that 71% DMS web user satisfaction was influenced by usability, information quality, service information quality. It implied that the influence of the three variables on user satisfaction was significantly high.

The findings showed that age and educational background distribution did not affect on satisfaction of DMS web users. Davis, Bagozzi, and Warshaw; Venkatesh and Davis; Chang and Yang stated that technology acceptance was determined by attitude and behaviour [18][19][20]. Behaviour would be influenced by age [21]. Age distribution indeed was homogenous, but not educational background. Although 54 % employees were graduated from senior high school but it did not influence the users satisfaction of technology. It implied that there should have

been other factors determining users satisfaction.

As DMS web users' satisfaction was influenced by usability, information quality, and service information quality, several aspects of the three dimensions should be improved. Janita, M. S., and Miranda, F. J.; AL-Zoubi, M and AL-Zawaideh F. stated that web with high usability could contribute significantly to user satisfaction [22][23]. Despite already having sufficient web design highlighted in descriptive analysis on usability, it should still be enhanced as it was considered unattractive. Thus, DMS web should not be less attractive than commercial web, such as in web page background and layout colors.

The next aspect needed more attention was the information produced. Al-Manasra stated that the personalization on the process of interaction and transaction security could improve user interest on the web [16]. Data accuracy and security should be ensured. In

other words, every BKN staff should ensure valid and secure personal data that no other parties could manipulate.

The utilization of information technology to increase the efficiency of management system demanded consequences for the authority. Government as DMS web provider should consider several issues, one of which is infrastructure provision to support the web application. Sufficient infrastructure supporting web could ensure satisfaction as it guarantees sustainable communication between user and the web. As stated by Siau and Long; Srivastava and Teo; and Krishnan and Teo that infrastructure investment on communication could enhance coverage and quality of service [4][6][24]. Similar notion was also delivered by Karatepe, Yavas and Babakus; and Rosenberg that good communication network provision affected user satisfaction and competitive advantage of an organization [25][26]. Sufficient infrastructure support is urgent due to the fact that service information quality influenced significantly to user satisfaction in utilizing the web.

CONCLUSION AND SUGGESTION

DMS web as one of the media utilized to manage government employee performance should sustain user interest. The sustainable utilization should be supported with user competence in form of hard skills as well as trust towards the web. Improving data accuracy and web interface were proven necessary.

BIBLIOGRAPHY

- [1] C. W. Chen, S. L. Pan, and W. Huang, "Managing E-Government Implementation in China: A Process Perspective", *Information & Management*, vol. 46, no. 4, pp. 203-212, 2009.
- [2] A. T. K. Ho, "Reinventing Local Government and The E-Government Initiative", *Public Administration Review*, vol. 62, no. 4, pp. 434-444, 2002.
- [3] C. W. Tan and S. L. Pan, "Managing E-Transformation in The Public Sector: an E-Government Study of The Inland Revenue Authority of Singapore (IRAS)", *European Journal of Information Systems*, vol. 12, no. 4, pp. 269-281, 2003.
- [4] K. Siau and Y. Long, "Factors Impacting E-Government Development", *Journal of Computer Information Systems*, vol. 50, no. 1, pp. 98-107, 2009
- [5] R. Heeks, "Success and Failure in E-Government Project", 2008. [Online]. Available: <http://www.egov4dev.org/success/sfrates.shtml>. [Accessed: October 19, 2008].
- [6] S. C. Srivastava and T. S. H. Teo, "E-Government, E-Business, And National Economic Performance", *Communications of the Association for*

- Information Systems*, vol. 26, no. 1, pp. 267-286, 2010.
- [7] H. Singh, A. Das, and D. Joseph, "Country-Level Determinants of E-Government Maturity", *Communications of the Association for Information Systems*, vol. 20, no. 1, pp. 632-648, 2007.
- [8] S. Negash, T. Ryan, and M. Igbariab, "Quality and Effectiveness in Web-Based Customer Support Systems", *Information & Management*, vol. 40, pp. 757-768, 2003
- [9] J. V. Iwaarden, T. V. Wiele, L. Ball, and R. Millen, "Perceptions About The Quality of Web Sites: A survey amongst Student at Nort Eastern University", *Information & Management*, vol. 41, no. 8, pp. 947-959, 2004
- [10] B. Yoo and N. Donthu, "Developing a Scale to Measure The Perceived Quality of an Internet Shopping Site (SITEQUAL)", *Quarterly Journal of Electronic Commerce*, vol. 2, no. 1, pp. 31-45, 2001.
- [11] E. K. Huizingh, "The Content and Design of Web Site: an Empirical Study", *Information & Management*, vol. 37, no. 3, pp. 123-134, 2000.
- [12] E. T. Loiacono, R. T. Watson, and D. L. Goodhue, "WebQual: a Measure of Web Site Quality, Marketing Educators", in *Conference: Marketing Theory and Applications*, vol. 13, pp. 432-437, 2002.
- [13] S. Barnes and R. Vidgen, "Interactive E-Government: Evaluating The Web Site of the UK Inland Revenue", *Journal of Electronic Commerce in Organizations*, vol. 2, no. 1, pp. 42-63, 2003.
- [14] Z. Yang, S. Cai, Z. Zhou, and N. Zhou, "Development and Validation of an Instrument to Measure User Perceived Service Quality of Information Presenting Web Portals", *Information & Management*, vol. 42, pp. 575-589, 2005.
- [15] A. Parasuraman, V. A. Zeithaml, and A. Malhotra, "ESQUAL: A multiple Itemscale for Assessing Electronic Service Quality", *Journal of Service Research*, vol. 7, no. 3, pp. 213-234, 2005.
- [16] E. A. Al-Manasra, M.K.S.A. Zaid, and F. TaherQutaishaf, "Investigating The Impact of Website Quality on Consumers Satisfaction in Jordanian Telecommunication Sector", *Arab Economic and Business Journal*, vol. 8, pp. 31-37, 2013.
- [17] S. Barnes and R. Vidgen, "An Integrative Approach to The Assessment of E-Commerce Quality", *Journal of Electronic Commerce Research*, vol. 3, no. 3, pp. 114-127, 2002.

- [18] F. D. Davis, R. P. Bagozzi, and P. R. Warshaw, "Extrinsic and Intrinsic Motivation to Use Computers in The Workplace", *Journal of Applied Social Psychology*, vol. 22, no. 14, pp. 1111–1132, 1992.
- [19] V. Venkatesh and F. D. Davis, "A Model of The Antecedents of Perceived Ease of Use: Development and Test", *Decision Sciences*, vol. 27, no. 3, pp. 451–481, 1996.
- [20] Y. S. Chang and C. Yang, "Why Do We Blog? From The Perspectives of Technology Acceptance and Media Choice Factors", *Behavior & Information Technology*, vol. 32, no. 4, pp. 371–386, 2013
- [21] V. Venkatesh, M. G. Morris, G. B. Davis, and F. D. Davis, "User Acceptance of Information Technology: Toward a Unified View", *MIS Quarterly*, vol. 27, no. 3, pp. 425–478, 2003.
- [22] M. S. Janita and F. J. Miranda, "Quality in E-Government Services: A Proposal of Dimensions from The Perspective of Public Sector Employees", *Telematics and Informatics*, vol. 35, no. 2, pp. 457–469, 2018.
- [23] M. AL-Zoubi and F. AL-Zawaideh, "Web-based for Successful E-Government Adoption: The Jordan National E-Government Portal", *International Review of Management and Business Research*, vol. 6, no. 1, pp. 320-330, 2017.
- [24] S. Krishnan and T. S. H. Teo, "Moderating Effects of Governance on Information Infrastructure and E-Government Development", *Journal of The American Society for Information Science and Technology*, vol. 63, no. 10, pp. 1929-1946, 2012.
- [25] O. M. Karatepe, U. Yavas, and E. Babakus, "Measuring Service Quality of Bank: Scale Development and Validation", *Journal of Retailing and Consumer Service*, vol. 12, pp. 373-383, 2005.
- [26] D. Rosenberg, "Use of e-Government Services in A Deeply Divided Society: A test and An Extension of The Social Inequality Hypotheses", *New Media & Society*, vol. 21, no. 2, pp. 1-19, 2018.

ALAT PENDETEKSI PELANGGARAN GARIS HENTI KENDARAAN PADA PERSIMPANGAN LALU LINTAS SATU ARAH MENGGUNAKAN SENSOR LASER BERBASIS ARDUINO MEGA 2560

¹Rudi Kurniawan, ²Veronica Ernita Kristianti, ³Alona Situmeang

^{1,2,3}Fakultas Teknologi Industri Universitas Gunadarma,

Jl. Margonda Raya No. 100, Depok 16424, Jawa Barat

¹roedikoerniawan@student.gunadarma.ac.id, ^{2,3}{veronica,alona}@staff.gunadarma.ac.id,

Abstrak

Perancangan alat pendeteksi pelanggaran garis henti kendaraan pada persimpangan lalu lintas satu arah menggunakan sensor laser berbasis Arduino Mega 2560. Software yang digunakan untuk memberi perintah Arduino Mega 2560 terhadap seluruh kerja alat adalah Arduino IDE. Sensor laser digunakan sebagai pendeteksi kendaraan yang melewati garis henti kendaraan dan dihubungkan ke input Arduino Mega 2560 untuk mengontrol 2 kondisi lampu lalu lintas, disetiap kondisi lampu lalu lintas terdapat beberapa output yang terdiri dari LCD 16 x 2 yang berfungsi menampilkan "Pelanggaran", buzzer sebagai bunyi peringatan pelanggaran, dan sepasang solenoid valve yang berfungsi menyembrotkan air ke arah jalan saat kendaraan melewati garis henti kendaraan di area lampu lalu lintas. Pengujian alat dilakukan terhadap kinerja komponen dan reaksi terhadap kondisi yang telah diberikan, antara lain tegangan input Arduino Mega 2560, tegangan pada sensor laser, lampu lalu lintas LCD, buzzer dan solenoid valve, mengukur waktu aktif buzzer dan solenoid valve saat terjadi pelanggaran lalu lintas, serta menguji delay respon sensor laser ketika mendeteksi kendaraan yang melewati garis henti kendaraan, serta menguji tegangan pada komponen output yang terdiri dari sensor laser, buzzer, dan solenoid valve ketika terjadi pelanggaran dengan kondisi lampu merah menyala di kedua area lampu lalu lintas.

Kata Kunci: *Arduino Mega 2560, Buzzer, LCD 16 x 2, pelanggaran lalu lintas, sensor laser, Solenoid Valve*

Abstract

The design of a vehicle stop line violation detection device at a one-way traffic junction uses a laser sensor and based on Arduino Mega 2560. The software used to give the Arduino Mega 2560 command to all work tools is the Arduino IDE. The laser sensor is used as a vehicle detector that crosses the vehicle stop line and is connected to the Arduino Mega 2560 input to control 2 traffic light conditions, in every traffic light condition, there are outputs consisting of a LCD 16 x 2 that functions to display "Violation", buzzer as the sound of the violation warning, and a pair of solenoid valves that function to spray water towards the road when the vehicle crosses the vehicle stop line in the traffic light area. Tool testing is carried out on component performance and reaction off the given conditions, some of the tests are input voltage of Arduino Mega 2560, voltage output of the laser sensors, LCD traffic lights, buzzers and solenoid valves, measuring the active time of the buzzer and solenoid valve during traffic violations, and testing the delay of the laser sensor responses when it detects a vehicle that crosses the vehicle stop line, also testing the output voltage off the output components consisting of the laser sensor, buzzer, and solenoid valve when a violation occurs with the red lights on both area of the traffic lights.

Keywords: *Arduino Mega 2560, Buzzer, laser sensor, LCD 16x2, Solenoid Valve, traffic violation*

PENDAHULUAN

Pelanggaran lalu lintas hingga saat ini sudah menjadi kebiasaan masyarakat di sebagian besar kota-kota di Indonesia. Jenis-jenis pelanggaran yang dilakukan yaitu tidak memakai sabuk pengaman saat berada di mobil, melawan arus, tidak membawa kelengkapan surat kendaraan, menerobos lampu lalu lintas di persimpangan lampu lalu lintas, dan lain-lain. Diantara pelanggaran-pelanggaran tersebut, yang paling sering dilakukan adalah menerobos lampu lalu lintas di persimpangan lampu lalu lintas. Akibatnya membahayakan para pengguna jalan lain yang sedang melintas yang bisa mengakibatkan kecelakaan lalu lintas.

Pemanfaatan teknologi di persimpangan lampu lalu lintas hingga kini sudah cukup baik di Indonesia, terutama di Jabodetabek. Elektronik Tilang atau E-Tilang yang direncanakan oleh Direktorat Lalu lintas (Ditlantas) Polda Metro Jaya sudah diterapkan. Namun masalah yang timbul setelah penerapan E-Tilang adalah masih rendahnya kesadaran masyarakat untuk tertib berlalu lintas di Jabodetabek, selain itu tidak semua persimpangan lampu lalu lintas terpasang alat E-Tilang.

Berdasarkan masalah tersebut maka perlu dibuat alat pendeteksi pelanggaran lampu lalu lintas di persimpangan lampu lalu lintas yang berjudul “Rancang Bangun Alat Pendeteksi Pelanggaran Garis Henti Kendaraan Pada Pesimpangan Lalu Lintas Satu

Arah Menggunakan Sensor Laser Berbasis Arduino Mega 2560”. Sensor laser digunakan sebagai *input* untuk mendeteksi kendaraan yang melewati garis henti kendaraan pada saat lampu merah menyala di area lampu lalu lintas [1] [2][3].

Hasil pendeteksian sensor dikirim dan dikontrol menggunakan Arduino Mega 2560 yang menghasilkan beberapa *ouput* [3][1][4], diantaranya *buzzer* yang berbunyi [6], *Liquid Crystal Display* (LCD) untuk menampilkan informasi pelanggaran [3], dan sepasang *solenoid valve* yang dialiri air di dalamnya dan berfungsi membuka dan menutup secara otomatis. Pendeteksian pelanggaran lalu lintas bertujuan untuk mengurangi pelanggaran kendaraan di area lampu lalu lintas untuk tidak menerobos garis henti kendaraan supaya lebih tertib [5].

METODE PENELITIAN

Perancangan alat dalam penelitian ini menggunakan beberapa komponen elektronika yaitu sensor laser, arduino 2560, indikator lampu lalu lintas, solenoid valve, buzzer dan LCD. Komponen-komponen tersebut terintegrasi menjadi satu rangkaian sehingga dapat menghasilkan alat pendeteksi pelanggaran kendaraan pada persimpangan lalu lintas satu arah. Keterkaitan antar komponen dijelaskan pada Gambar 1.

Arduino yang digunakan pada rancangan ini hanya satu untuk mengoperasikan 2 lampu lalu lintas. Sensor laser 1 dan sensor

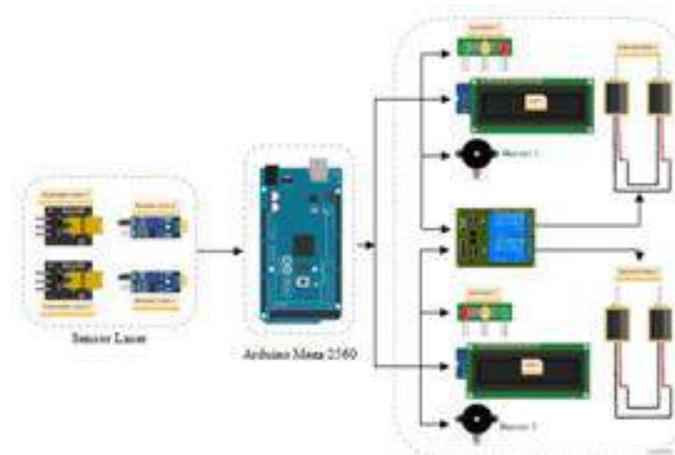
laser 2 dioperasikan dalam 1 arduino untuk memberikan perintah pada lampu lalu lintas 1 dan lampu lalu lintas 2. Sensor akan membaca objek jika salah satu lampu lalu lintas dalam keadaan merah dan terdapat kendaraan yang melewati batas aman, maka solenoid valve akan memutar tuas air, buzzer berbunyi, dan LCD akan menampilkan tulisan Pelanggaran. Lampu lalu lintas, LCD, sensor laser, *buzzer*, dan *solenoid valve* menyala lampu lalu lintas dirancang terus menyala yang digunakan untuk mengontrol arus lalu lintas pada persimpangan sesuai dengan durasi yang ditentukan yaitu 42 detik. Komponen-

komponen elektronika da-lam rancangan penelitian ini di integrasikan menjadi satu rangkaian utuh yang ditunjukkan pada Gambar 2.

Rancangan alat pada penelitian ini terdiri dari 2 sensor laser, 1 arduino 2560, 2 solenoid valve, 2 lampu lalu lintas, 2 LCD, dan 2 buzzer. Penelitian ini memaksimalkan fungsi arduino dalam menjalankan tugas untuk mengoperasikan kerja 2 sensor dan komponen lainnya. Penggunaan 1 arduino untuk mengoperasikan komponen lain dengan tugas yang berbeda, sebagai langkah efisiensi yang dilakukan dalam penelitian ini.



Gambar 1. Diagram Blok Rancangan Alat Pendeteksi Pelanggaran Kendaraan pada Persimpangan Lalu Lintas



Gambar 2. Rangkaian antar Komponen Elektronika Pendeteksi Pelanggaran Lampu Lalu Lintas

HASIL DAN PEMBAHASAN

Pengujian Sensor Laser sebagai Pendeteksi Pelanggaran Lalu Lintas terhadap Kendaraan

Pada pengujian ini, dilakukan untuk mengetahui seluruh perubahan kondisi *output* perancangan alat saat terjadi pelanggaran di area lampu lalu lintas. Kendaraan yang digunakan pada pengujian ini adalah sebuah kendaraan. Kondisi *output* sebelum kendaraan melewati garis henti kendaraan adalah lampu

hijau ON, tampilan LCD bertuliskan “Mohon Tertib”, *buzzer* OFF, dan sepa-sang *solenoid valve* OFF, seperti yang terdapat pada Gambar 3.

Ketika kendaraan dimajukan dengan posisi melewati garis henti kendaraan pada saat lampu hijau menyala, kondisi LCD, *buzzer*, dan sepa-sang *solenoid valve* tidak mengalami perubahan atau tidak terjadi pelanggaran lalu lintas seperti yang terdapat pada Gambar 4.



Gambar 3. Kondisi Alat Sebelum Terjadi Pelanggaran



Gambar 4. Kondisi Alat saat Lampu Hijau dengan Mobil Terdeteksi Sensor Laser

Kondisi yang terjadi pada Gambar 4 dengan posisi kendaraan terdeteksi oleh sensor laser namun tidak terjadi pelanggaran, karena pelanggaran lalu lintas pada area tersebut hanya terjadi pada saat lampu merah menyala saja. Selanjutnya, dengan posisi kendaraan yang tidak berubah dan lampu merah menyala, maka terjadi pelanggaran lalu lintas seperti pada Gambar 5.

Selama terjadi pelanggaran seperti pada Gambar 5, terjadi perubahan pada *output* yaitu LCD bertuliskan “Pelang-garan”, *buzzer* ON, sepasang *solenoid valve* ON dengan menyembrotkan air ke arah jalan. Hal tersebut disebabkan cahaya merah yang dipancarkan laser *transmitter* ke laser *receiver* terhalang, fotodioda yang terdapat di bagian laser *receiver* tidak menerima cahaya sehingga tegangan saat itu menjadi sangat kecil mendekati 0 Volt DC, karena sensor laser bersifat aktif *low*. Apabila kendaraan tidak menghalangi sensor laser, alat kembali ke perintah yang telah dibuat secara berulang sampai terjadi pelanggaran berikutnya.

Pengujian Waktu Aktif Buzzer dan Solenoid Valve saat Terjadi Pelanggaran

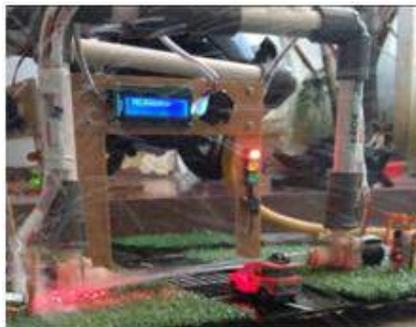
Pada pengujian data ini, dilakukan pengukuran terhadap komponen *output* yang ketika kendaraan melanggar dengan melewati garis henti kendaraan. Komponen-komponen tersebut yaitu bagian dari area lalu lintas 1

dan area lampu lalu lintas 2, diantaranya *buzzer* dan *solenoid valve* yang dilakukan pengujian sebanyak 20 kali percobaan untuk membandingkan antara set waktu terprogram *buzzer* dan *solenoid valve* ketika terjadi pelanggaran pada Arduino Mega 2560 yaitu 3 detik dengan rata-rata waktu aktif pada *buzzer* dan *solenoid valve* yang terukur pada Tabel 1.

Setelah didapat hasil pengukuran pada Tabel 1, selanjutnya dihitung rata-rata waktu aktif yang terukur, maka didapat hasil:

- Rata-rata waktu aktif terukur pada *buzzer* 1 dan *solenoid valve* 1 = 3.40 detik
 $X_{min} = 3.11 \text{ detik}$
 $X_{max} = 3.81 \text{ detik}$
- Rata-rata waktu aktif terukur pada *buzzer* 2 dan *solenoid valve* 2 = 3.35 detik
 $X_{min} = 3.03 \text{ detik}$
 $X_{max} = 3.56 \text{ detik}$

Berdasarkan Tabel 1, pengujian dilakukan sebanyak 20 kali percobaan dan didapat hasil perhitungan di atas, rata-rata waktu aktif terukur pada *buzzer* dan *solenoid* yang terdapat di area lampu lalu lintas 1 saat terjadi pelanggaran adalah 3.40 detik.. Sedangkan rata-rata waktu aktif terukur *buzzer* dan *solenoid* yang terdapat di area lampu lalu lintas 2 saat terjadi pelanggaran adalah 3.35 detik.



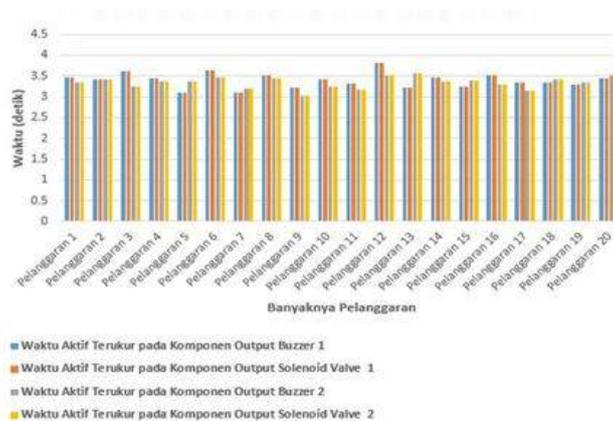
Gambar 5. Kondisi Alat saat Lampu Merah dengan Mobil Terdeteksi Sensor Laser

Tabel 1. Waktu Aktif Buzzer dan Solenoid Valve saat Terjadi Pelanggaran

No.	Banyaknya Pelanggaran	Waktu Aktif Terukur pada Komponen Output (Detik)			
		Buzzer 1	Solenoid Valve 1	Buzzer 2	Solenoid Valve 2
1	Pelanggaran 1	3.46	3.46	3.35	3.35
2	Pelanggaran 2	3.41	3.41	3.41	3.41
3	Pelanggaran 3	3.63	3.63	3.24	3.24
4	Pelanggaran 4	3.44	3.44	3.37	3.37
5	Pelanggaran 5	3.11	3.11	3.38	3.38
6	Pelanggaran 6	3.64	3.64	3.48	3.48
7	Pelanggaran 7	3.11	3.11	3.21	3.21
8	Pelanggaran 8	3.51	3.51	3.45	3.45
9	Pelanggaran 9	3.23	3.23	3.03	3.03
10	Pelanggaran 10	3.41	3.41	3.26	3.26
11	Pelanggaran 11	3.33	3.33	3.18	3.18
12	Pelanggaran 12	3.81	3.81	3.53	3.53
13	Pelanggaran 13	3.23	3.23	3.56	3.56
14	Pelanggaran 14	3.46	3.46	3.36	3.36
15	Pelanggaran 15	3.24	3.24	3.39	3.39
16	Pelanggaran 16	3.53	3.53	3.31	3.31
17	Pelanggaran 17	3.34	3.34	3.15	3.15
18	Pelanggaran 18	3.35	3.35	3.41	3.41
19	Pelanggaran 19	3.31	3.31	3.34	3.34
20	Pelanggaran 20	3.45	3.45	3.52	3.52
Rata-rata Tegangan		3.40	3.40	3.35	3.35

Rata-rata waktu aktif terukur pada *buzzer* dan *solenoidvalve* yang terdapat di area lampu lalu lintas 1 memiliki selisih waktu 0.40 detik lebih lama dari set waktu aktif terprogram pada *buzzer* dan *solenoidvalve*, sedangkan rata-rata waktu aktif terukur pada *buzzer* dan *solenoidvalve* yang terdapat di area lampu lalu lintas 2 memiliki selisih

waktu 0.35 detik lebih lama dari set waktu aktif terprogram pada *buzzer* dan *solenoidvalve*. Perbandingan waktu aktif terukur pada komponen *output* di kedua area lampu lalu lintas saat terjadi pelanggaran lalu lintas disetiap masing-masing pelanggaran dapat dilihat pada Gambar 7.



Gambar 7. Grafik Perbandingan Waktu Aktif Terukur *Buzzer* dan *Solenoid Valve*

Gambar 7 merupakan grafik perbandingan waktu aktif terukur pada komponen *output* lampu lalu lintas 1 dan 2 yang terdiri dari *buzzer* dan *solenoid valve* saat terjadi pelanggaran lalu lintas. Waktu aktif yang terukur pada komponen *output* di area lampu lalu lintas 1 memiliki waktu aktif terukur paling singkat 3.11 detik dan waktu aktif terukur paling lama 3.81 detik, sedangkan waktu aktif yang terukur pada komponen *output* di area lampu lalu lintas 2 memiliki waktu aktif terukur paling singkat 3.03 detik dan waktu aktif terukur paling lama 3.56 detik. Setelah dilakukan pengujian sebanyak 20 kali percobaan pelanggaran, waktu aktif terukur pada komponen *output* di kedua area lampu lalu lintas, terlihat stabil.

Pengujian Delay Respon Sensor Laser sebagai Pendeteksi Pelanggaran Lalu Lintas

Pada pengujian ini, dilakukan pengukuran *delay* respon sensor laser sebagai pendeteksi pelanggaran lalu lintas yang terdapat di area lampu lalu lintas 1 dan di area

lampu lalu lintas 2 ketika kendaraan melewati garis henti kendaraan. Pada bagian laser *receiver* dilakukan pengukuran waktu menggunakan *stopwatch* untuk mengetahui waktu yang diperlukan sensor laser untuk mendeteksi satu kali pelanggaran di garis henti kendaraan. Pengujian ini dilakukan pengukuran sebanyak 20 kali percobaan pelanggaran seperti yang terdapat pada Tabel 2.

Perbandingan waktu aktif terukur pada komponen *output* lampu lalu lintas 1 dan 2 yang terdiri dari *buzzer* dan *solenoid valve* saat terjadi pelanggaran lalu lintas. Waktu aktif yang terukur pada komponen *output* di area lampu lalu lintas 1 memiliki waktu aktif terukur paling singkat 3.11 detik dan waktu aktif terukur paling lama 3.81 detik, sedangkan waktu aktif yang terukur pada komponen *output* di area lampu lalu lintas 2 memiliki waktu aktif terukur paling singkat 3.03 detik dan waktu aktif terukur paling lama 3.56 detik. Setelah dilakukan pengujian sebanyak 20 kali percobaan pelanggaran, waktu aktif terukur pada komponen *output* di kedua area lampu lalu lintas, terlihat stabil.

Tabel 2. Delay Respon Sensor Laser untuk Mendeteksi Pelanggaran

No	Banyaknya Pelanggaran	Delay Respon Sensor Laser 1 (Detik)	Delay Respon Sensor Laser 2 (Detik)
1	Pelanggaran 1	0.49	0.46
2	Pelanggaran 2	0.20	0.21
3	Pelanggaran 3	0.36	0.44
4	Pelanggaran 4	0.46	0.43
5	Pelanggaran 5	0.43	0.18
6	Pelanggaran 6	0.35	0.28
7	Pelanggaran 7	0.28	0.38
8	Pelanggaran 8	0.59	0.46
9	Pelanggaran 9	0.13	0.39
10	Pelanggaran 10	0.35	0.30
11	Pelanggaran 11	0.28	0.34
12	Pelanggaran 12	0.15	0.24
13	Pelanggaran 13	0.20	0.26
14	Pelanggaran 14	0.11	0.26
15	Pelanggaran 15	0.18	0.31
16	Pelanggaran 16	0.34	0.33
17	Pelanggaran 17	0.16	0.26
18	Pelanggaran 18	0.36	0.26
19	Pelanggaran 19	0.33	0.26
20	Pelanggaran 20	0.26	0.33
Rata-rata Delay		0.305	0.319

Data pada Tabel 2, digunakan untuk menghitung rata-rata *delay* respon yang diperlukan sensor laser 1 dan 2 untuk mendeteksi kendaraan yang melanggar garis henti kendaraan dapat dihitung dengan persamaan 2, maka didapat hasil:

a. Rata-rata delay respon sensor laser 1 =0.305 detik

$$X_{min} = 0.11 \text{ detik}$$

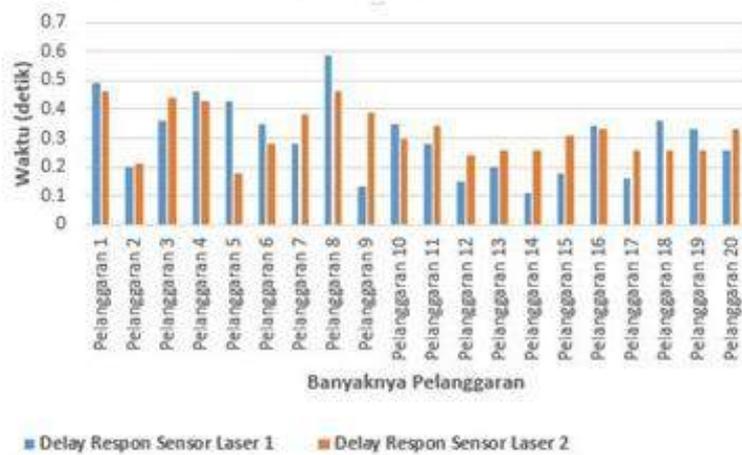
$$X_{max} = 0.59 \text{ detik}$$

b. Rata-rata delay respon sensor laser 2 =0.319 detik

$$X_{min} = 0.18 \text{ detik}$$

$$X_{max} = 0.46 \text{ detik}$$

Berdasarkan pengujian yang dilakukan sebanyak 20 kali percobaan dan hasil perhitungan di atas, rata-rata *delay* respon sensor laser yang terdapat di area lampu lalu lintas 1 saat mendeteksi pelanggaran adalah 0.305 detik, Sedangkan rata-rata *delay* respon sensor laser yang terdapat di area lampu lalu lintas 2 saat mendeteksi pelanggaran adalah 0.319 detik. Untuk melihat perbandingan antara *delay* respon sensor laser 1 dengan *delay* respon sensor laser 2 disetiap masing-masing pelanggaran dapat dilihat pada Gambar 8.



Gambar 8. Grafik Perbandingan *Delay Respon* Sensor Laser

Gambar 8 merupakan grafik perbandingan *delay respon* sensor laser saat terjadi pelanggaran lalu lintas di area lampu lalu lintas 1 dan di area lampu lalu lintas 2, warna biru pada grafik mewakili *delay respon* sensor laser 1 yang memiliki *delay respon* paling singkat 0.11 detik dan *delay respon* paling lama 0.59 detik, sedangkan warna orangepada grafik mewakili *delay respon* sensor laser 2 yang memiliki *delay respon* paling singkat 0.18 detik dan *delay respon* paling lama 0.46 detik. Setelah dilakukan pengujian sebanyak 20 kali percobaan pelanggaran, *delay respon* sensor yang terdapat di area lampu lalu lintas 1 dan di area lampu lalu lintas 2 terlihat tidak stabil.

KESIMPULAN

Hasil perancangan dan analisa alat pendeteksi pelanggaran garis henti kendaraan pada persimpangan lalu lintas satu arah berbasis Arduino Mega 2560 telah berhasil

dengan menggunakan sensor laser sebagai pendeteksi pelanggaran lalu lintas ketika kendaraan melewati garis henti kendaraan di area lampu lalu lintas. Hasil pengujian saat terjadi pelanggaran di area lampu lalu lintas 1 dan di area lampu lalu lintas 2, diperoleh bahwa rata-rata *delay respon* sensor laser yang diperlukan untuk mendeteksi kendaraan yang melewati garis henti kendaraan adalah masing-masing 0.305 detik dan 0.319 detik, rata-rata waktu aktif *buzzer* dan *solenoid valve* adalah masing-masing 3.40 detik dan 3.35 detik. Lampu lalu lintas digunakan sebagai pengatur arus lalu lintas di persimpangan, LCD digunakan sebagai penampil tulisan “Pelanggaran”, *solenoid valve* digunakan sebagai penyemprot air ke arah jalan, dan *buzzer* digunakan sebagai bunyi penanda saat terjadi pelanggaran di area lalu lintas. Besar kecilnya volume air yang dialirkan dari kran ke pipa sambung *solenoid valve* menentukan derasny semprotan air yang dikeluarkan dari *solenoid valve*, semakin

besar volume air yang dialirkan, semakin jauh semprotan air yang dikeluarkan saat terjadi pelanggaran di area lampu lalu lintas.

DAFTAR PUSTAKA

- [1] D. Pancawati dan A. Yulianto, "Implementasi *fuzzy logic controller* untuk mengatur pH nutrisi pada sistem hidroponik *Nutrient Film Technique* (NFT)", *Jurnal Nasional Teknik Elektro*, vol. 5, no. 2, hal. 281-282, 2016.
- [2] D. P. Githa dan W. E. Swastawan, "Sistem pengaman parkir dengan visualisasi jarak menggunakan sensor ping dan LCD", *Jurnal Nasional Pendidikan Teknik Informatika*, vol. 3, no. 1, hal. 11-12, 2014.
- [3] A. Pranata, S. N. Arif dan Yusniah, "Perancangan prototipe sistem parkir cerdas berbasis mikrokontroler Atmega8535", *Jurnal Saintikom*, vol. 14, no. 2, hal. 132-133, 2015.
- [4] E. Mulyanah dan C. M. Hellyana, "Perancangan dan pembuatan alat pengering kerupuk otomatis menggunakan mikrokontroler Atmega16", *Jurnal Evolusi*, vol. 3, no. 2, hal. 44, 2015.
- [5] Mariyam, Asparizal dan A. Azkiya, "Pengembangan simulasi pengendalian lampu lalu lintas dan pendeteksian kepadatan berbasis Arduino Mega 2560 menggunakan LDR dan laser", *Lentera Dumai*, vol. 9, no. 2, hal. 48-49, 2018.
- [6] Siswanto, G. P. Utama dan W. Gata, "Pengamanan ruangan dengan Arduino no R3, Sensor Mc-38, Pir, Notifikasi SMS, Twitter", *Jurnal Rekayasa Sistem dan Teknologi Informasi*, vol. 2, hal. 699, 2018.

PEMBUATAN APLIKASI KLASIFIKASI CITRA DAUN MENGUNAKAN RUANG WARNA RGB DAN HSV

¹Haura Sanusi, ²Suryadi H. S., ³ Diana Tri Susetianingtias
^{1,2,3} Fakultas Teknologi Industri Universitas Gunadarma
Jl. Margonda Raya No. 100, Depok 16424, Jawa Barat
¹haurasanusi@gmail.com, ^{2,3}{suryadi_hs,diantis}@staff.gunadarma.ac.id

Abstrak

Bagi orang awam, mengetahui tumbuhan berdasarkan warna daun tentu tidak mudah, mengingat semua warna daun relatif sama yaitu warna hijau, sehingga akan sulit juga untuk mengetahui manfaat dari tumbuhan tersebut. Oleh karena itu dibutuhkan sebuah sistem klasifikasi berdasarkan warna daun untuk mengetahui apa nama dan manfaat tumbuhan tersebut. Penelitian ini bertujuan untuk menghasilkan ekstraksi nilai red, green, blue, hue, saturation, dan value pada citra daun, menghasilkan klasifikasi citra daun berdasarkan hasil ekstraksi nilai RGB dan HSV, serta menghasilkan nilai akurasi hasil klasifikasi citra daun. Pada penelitian ini, peneliti akan melakukan klasifikasi terhadap daun berdasarkan warna daun. Peneliti menggunakan 200 buah citra dari 10 jenis daun. Klasifikasi daun berdasarkan warna dilakukan peneliti menggunakan ruang warna RGB dan HSV. Hasil klasifikasi citra daun memiliki rata-rata akurasi yang tinggi yaitu 90,08%.

Kata Kunci: Citra Daun, HSV, matlab, RGB

Abstract

Recognizing various types of plants based on their leaves to those who do not know plants well, it is difficult to remember all the leaves based on leaves colour, which is relatively in the same colour, lead them to hardly know the benefits of these plants. Hence, we need a classification system based on leaves to find out what the names and benefits of these plants. This study aims to produce extractions of red, green, blue, hue, saturation, and value values in leaf images, produce classification of leaf images based on the results of the extraction of RGB and HSV values, and accuracy values of the results of classification of leaf images. In this study, researchers will classify leaves based on leaf color. Researchers used 200 pieces of image from 10 types of leaves. Classification of leaves based on color is done by researchers using RGB and HSV color spaces. The results of the classification of leaf images have a high average accuracy of 90.08%.

Keywords: Leaf Image, HSV, matlab, RGB

PENDAHULUAN

Daun merupakan bagian dari tumbuhan. Bagi orang yang mengenal berbagai jenis daun, mengetahui tumbuhan berdasarkan daunnya merupakan hal yang mudah. Namun bagi orang awam tentu sulit,

mengingat semua warna daun relatif sama yaitu warna hijau, sehingga akan sulit juga untuk mengetahui manfaat dari tumbuhan tersebut. Oleh karena itu dibutuhkan sebuah sistem klasifikasi berdasarkan daun untuk mengetahui apa nama dan manfaat tumbuhan tersebut.

Salah satu metode yang dapat digunakan untuk mengetahui sebuah jenis tumbuhan yaitu dengan melakukan klasifikasi berdasarkan warna daun. Dari daun tersebut akan dilakukan ekstraksi untuk mendapatkan nilai *red*, *green*, *blue*, *hue*, *saturation*, dan *value* sehingga daun dapat diklasifikasikan berdasarkan nilai-nilai yang sudah didapatkan.

Beberapa penelitian terkait citra dalam mengetahui informasi dari suatu tumbuhan telah dilakukan peneliti terdahulu. Penelitian yang dilakukan Septiaji dan Firdausy menggunakan metode rentang nilai RGB untuk mendeteksi kematangan tanaman selada berbasis *android* [1]. Prabowo melakukan penelitian menggunakan metode ruang warna RGB berdasar nilai jarak antara histogram citra buah untuk mendeteksi kematangan buah jeruk dengan memiliki nilai acuan yang dicari dari frekuensi atau keanggotaan dalam kombinasi piksel dari sebuah citra gambar [2]. Penelitian yang dilakukan Sari, Hidayat dan Atmaja menggunakan metode *color moments*, GLCM, dan *k-Nearest Neighbor* untuk mendeteksi penyakit tanaman jagung dengan mengklasifikasikan jagung menjadi 4 kelas yaitu netral (tidak berpenyakit), penyakit hawar daun, penyakit bercak daun, dan penyakit karat daun [3]. Lestari, Sari, Sutardi, Purwanti dan Purnama menggunakan metode segmentasi warna RGB-HSV untuk memutihkan area pada daun jeruk yang terkena penyakit, kemudian mengklasifikasikan jenis penyakit pada daun jeruk menggunakan

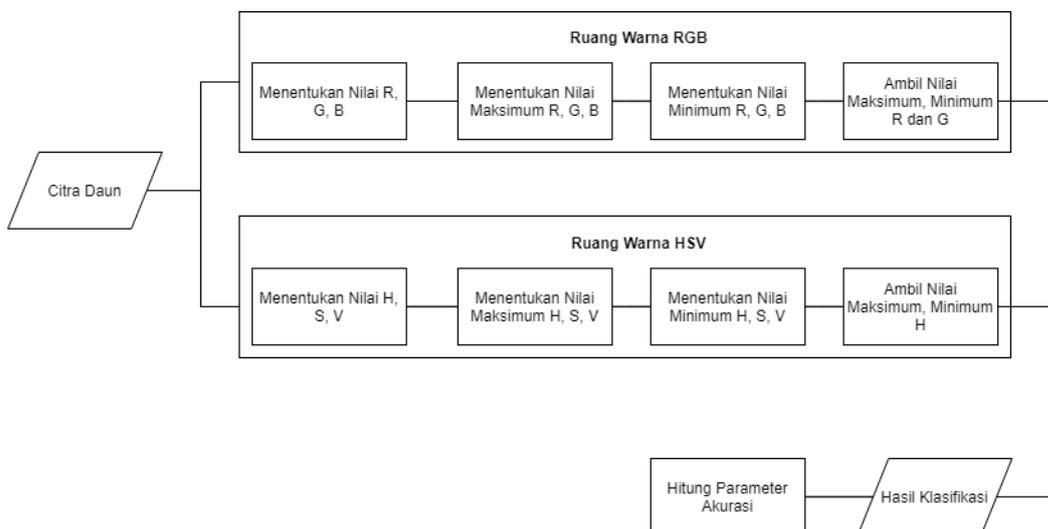
fuzzy k-nearest neighbor [4]. Penelitian Aianto dan Harjoko menggunakan metode segmentasi warna RGB dan *gray*, serta *k-Nearest Neighbor* dimana setiap piksel buah salak pondoh dihitung besar ketidakmiripannya (*Euclidean Distance*) terhadap nilai data fitur RGB dan *gray* pembandingan kemudian klasifikasi, dengan mengetahui nilai *mean R* dan *mean V* [5].

Daun adalah salah satu organ tanaman yang memiliki beberapa sistem jaringan, yaitu jaringan epidermis, mesofil, dan jaringan pengangkut [6]. Pengolahan citra digital merupakan disiplin ilmu yang mempelajari teknik-teknik untuk mengolah sebuah citra menggunakan komputer. Citra yang dimaksud dapat berupa gambar diam atau gambar bergerak [7]. Ruang Warna *Red, Green, Blue* (RGB) adalah ruang warna standar yang didasarkan pada akuisisi frekuensi warna oleh sensor elektronik. Keluaran dari sensor ini berupa sinyal analog. RGB adalah ruang warna aditif, yang berarti bahwa seluruh warna dimulai dengan warna hitam dan dibentuk dengan menambahkan warna hijau, merah, dan biru. Melalui gabungan warna merah, hijau dan biru akan terbentuk warna-warna baru. Ruang Warna *Hue, Saturation, Value* (HSV). Parameter *value* identik dengan parameter *luminance* pada ruang warna HSL. Ruang warna HSV sering disebut *hexcome model*, sedangkan HSL disebut dengan *bi-hexcome model* karena memiliki 2 *cone* [8].

METODE PENELITIAN

Metode penelitian yang digunakan pada penelitian ini terdiri atas empat tahapan utama, ekstraksi ruang warna RGB, ekstraksi ruang warna HSV, proses klasifikasi, dan perhitungan parameter akurasi. Tahapan-tahapan penelitian pada pembuatan aplikasi klasifikasi citra daun menggunakan ruang warna RGB dan HSV dapat dilihat pada Gambar 1. Tahapan ekstraksi ruang warna RGB bertujuan untuk mendapatkan nilai minimum dan maksimum *red* dan *green* dari citra daun. Tahapan ekstraksi ruang warna HSV bertujuan untuk mendapatkan nilai minimum dan maksimum *hue* dari citra daun. Tahapan klasifikasi adalah tahapan dimana dilakukan pengujian model klasifikasi menggunakan data gambar daun. Tahapan perhitungan parameter akurasi adalah tahapan untuk mengetahui apakah nama tumbuhan sesuai dengan gambar daun yang dimasukkan.

Tahapan ekstraksi ruang warna RGB dimulai dengan memasukkan data *input* berupa citra daun, dilanjutkan dengan proses menentukan nilai *red*, *green*, dan *blue*, menentukan nilai maksimum *red*, *green*, dan *blue*, menentukan nilai minimum *red*, *green*, dan *blue*, dan pengambilan nilai maksimum dan minimum *red* dan *green*. Tahapan ekstraksi ruang warna HSV dimulai dengan memasukkan data input berupa citra daun, dilanjutkan dengan proses menentukan nilai *hue*, *saturation*, dan *value*, menentukan nilai maksimum *hue*, *saturation*, dan *value*, menentukan nilai minimum *hue*, *saturation*, dan *value*, dan pengambilan nilai maksimum dan minimum *hue*. Tahapan klasifikasi adalah tahapan menentukan nama daun sesuai dengan gambar yang dimasukkan. Tahap terakhir perhitungan parameter akurasi menggunakan *confusion matrix*.



Gambar 1. Tahapan Penelitian

Pembuatan sistem klasifikasi daun bertujuan untuk mengekstraksi nilai RGB dan HSV dari citra daun. Kemudian menggunakannya untuk mengklasifikasikan daun. Serta mengetahui akurasi yang dihasilkan oleh klasifikasi tersebut. *Confusion matrix* digunakan untuk mengevaluasi model klasifikasi yang dihasilkan. Bahasa pemrograman matlab digunakan untuk mengembangkan aplikasi ini, dengan menggunakan *software* Matlab R2018b.

HASIL DAN PEMBAHASAN

Citra Daun

Citra daun merupakan citra berwarna dari daun yang dimasukkan ke dalam sistem

sebagai data masukan. Citra daun yang digunakan dalam penelitian ini berjumlah 200 buah citra, yang didapat dari *Flavia Dataset*. Detail jumlah citra daun yang digunakan pada penelitian tersebut terdiri dari 10 jenis daun yang dapat dilihat pada Tabel 1.

Ekstraksi Ruang Warna RGB

Langkah pertama yang dilakukan yaitu menentukan nilai *red*, *green*, dan *blue* dari citra tiap daun menggunakan Matlab. Setelah mendapatkan angka-angka pada *workspace* Matlab maka angka tersebut dipindahkan ke *Microsoft Excel*. Gambar 2 memperlihatkan bagaimana proses penentuan nilai *red*, *green*, dan *blue* dari citra daun.

Tabel 1. Kelompok Jenis Daun yang Akan Diklasifikasikan

No	Kelompok Jenis Bentuk Daun	Spesies Tanaman Anggota
1	<i>Bitter Orange</i>	20
2	<i>Caricature Plant</i>	20
3	<i>Chocolate Tree</i>	20
4	<i>Croton</i>	20
5	<i>Duranta Gold</i>	20
6	<i>Ficus</i>	20
7	<i>Geranium</i>	20
8	<i>Sweet Potato</i>	20
9	<i>Thevetia</i>	20
10	<i>Vieux</i>	20
	Total	200



Gambar 2. Proses Penentuan Nilai *Red*, *Green*, dan *Blue* pada Daun *Bitter Orange*

Untuk mendapatkan nilai maksimum *red*, *green*, dan *blue* pada tiap citra daun, digunakan persamaan 1.

$$=MAX(B2:U2) \quad (1)$$

Sedangkan untuk mencari nilai minimum *red*, *green*, dan *blue* pada setiap citra daun menggunakan persamaan 2.

$$=MIN(B2:U2) \quad (2)$$

Persamaan 1 merupakan persamaan pada *microsoft excel* yang berfungsi untuk mencari nilai maksimum dari nilai awal hingga nilai akhir di setiap baris. Persamaan 1 digunakan untuk mencari nilai maksimum *red* dari daun *bitter orange*. Persamaan 2 merupakan persamaan pada *microsoft excel* yang berfungsi untuk mencari nilai minimum dari nilai awal hingga nilai akhir di setiap baris. Persamaan ini digunakan untuk mencari nilai minimum *red* dari daun *bitter orange*. Karena B2 adalah nilai awal pada baris nilai *red*. Sedangkan U2 adalah nilai akhir pada baris nilai *red*.

Setelah mendapatkan nilai maksimum dan minimum dari *red*, *green*, dan *blue* pada

proses sebelumnya, proses pengambilan nilai maksimum dan minimum *red* dan *green* dilakukan pada tahap selanjutnya. Nilai maksimum dan minimum *red* diambil untuk membedakan warna daun yang hijau dengan yang kemerah-merahan. Karena pada *dataset* terdapat daun yang berwarna kemerah-merahan. Sedangkan nilai maksimum dan minimum *green* diambil untuk membedakan warna hijau pada setiap jenis daun.

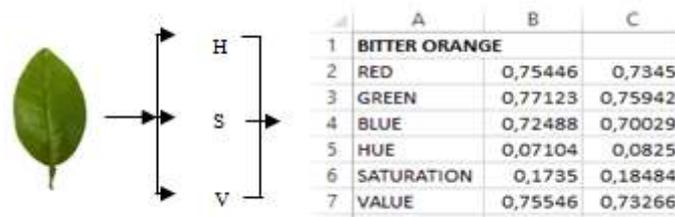
Berikut adalah sebagian hasil penentuan nilai *red*, *green*, dan *blue* pada daun *bitter orange* yang dapat dilihat pada Tabel 2. Kolom pertama berisi nilai *red*, *green*, dan *blue* pada gambar pertama dari daun *bitter orange*. Kolom kedua berisi nilai *red*, *green*, dan *blue* pada gambar kedua dari daun *bitter orange*. Begitu seterusnya hingga kolom ke-20 yang merupakan nilai *red*, *green*, dan *blue* pada gambar ke-20 dari daun *bitter orange*.

Ekstraksi Ruang Warna HSV

Tahap pertama yaitu melakukan proses pencarian nilai *hue*, *saturation*, dan *value* dari tiap citra daun menggunakan Matlab. Kemudian memindahkan angka-angka yang telah didapat pada *workspace* Matlab ke *Microsoft Excel* seperti pada Gambar 3.

Tabel 2. Sebagian Hasil Nilai *Red*, *Green*, dan *Blue* pada Daun *Bitter Orange*

	Bitter Orange						MAX	MIN
<i>RED</i>	0,754459	0,734505	0,703979	0,653152	0,676435	...	0,824409	0,653152
<i>GREEN</i>	0,771234	0,759421	0,73508	0,707021	0,717261	...	0,843749	0,707021
<i>BLUE</i>	0,724876	0,700291	0,654006	0,573735	0,626537	...	0,807401	0,573735



Gambar 3. Proses Penentuan Nilai Hue, Saturation, dan Value pada Daun Bitter Orange

Untuk mendapatkan nilai maksimum hue, saturation, dan value pada tiap citra daun, digunakan Persamaan 3.

$$=MAX(B5:U5) \quad (3)$$

Sedangkan pencarian nilai minimum hue, saturation, dan value pada setiap citra daun menggunakan Persamaan 4.

$$=MIN(B5:U5) \quad (4)$$

Persamaan 3 merupakan rumus pada Microsoft Excel yang berfungsi untuk mencari nilai maksimum dari nilai awal hingga nilai akhir di setiap baris. Persamaan di atas digunakan untuk mencari nilai maksimum hue dari daun bitter orange. Persamaan 4 juga merupakan rumus pada Microsoft Excel. Persamaan ini berfungsi untuk mencari nilai minimum dari nilai awal hingga nilai akhir di setiap baris. Persamaan ini digunakan untuk mencari nilai minimum hue dari daun bitter orange. Karena B5

adalah nilai awal pada baris nilai hue. Sedangkan U5 adalah nilai akhir pada baris nilai hue.

Setelah mendapatkan nilai maksimum dan minimum dari hue, saturation, dan value pada proses sebelumnya, proses pengambilan nilai maksimum dan minimum hue dilakukan pada tahap selanjutnya. Nilai maksimum dan minimum hue diambil sebagai penentuan warna hijau dan warna kuning. Karena hue merepresentasikan warna dari cahaya tampak.

Berikut adalah sebagian hasil penentuan nilai hue, saturation, dan value pada daun bitter orange yang dapat dilihat pada Tabel 3. Kolom pertama berisi nilai hue, saturation, dan value pada gambar pertama dari daun bitter orange. Kolom kedua berisi nilai hue, saturation, dan value pada gambar kedua dari daun bitter orange. Begitu seterusnya hingga kolom ke-20 yang merupakan nilai hue, saturation, dan value pada gambar ke-20 dari daun bitter orange.

Tabel 3. Sebagian Hasil Nilai Hue, Saturation, dan Value pada Daun Bitter Orange

	Bitter Orange						MAX	MIN
HUE	0,071042	0,082499	0,081549	0,096827	0,100003	...	0,110697	0,053313
SATURATION	0,173497	0,184841	0,248831	0,299998	0,238156	...	0,299998	0,135163
VALUE	0,755461	0,73266	0,66962	0,603174	0,66184	...	0,811524	0,603174

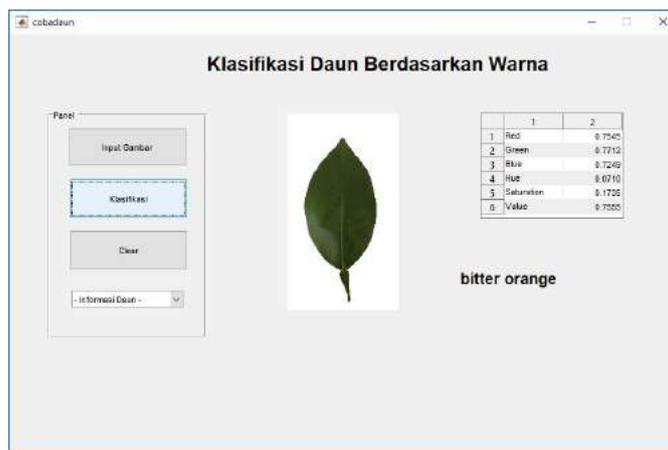
Hasil Klasifikasi Citra Daun

Pada tahap ini, dilakukan pengujian menggunakan data gambar daun yang dimasukkan. Citra daun tersebut tentunya melewati tahapan-tahapan yang sudah dijelaskan sebelumnya. Klasifikasi dilakukan dengan mencocokkan hasil ekstraksi citra daun yang dimasukkan dengan parameter-parameter yang telah dituliskan pada kode program.

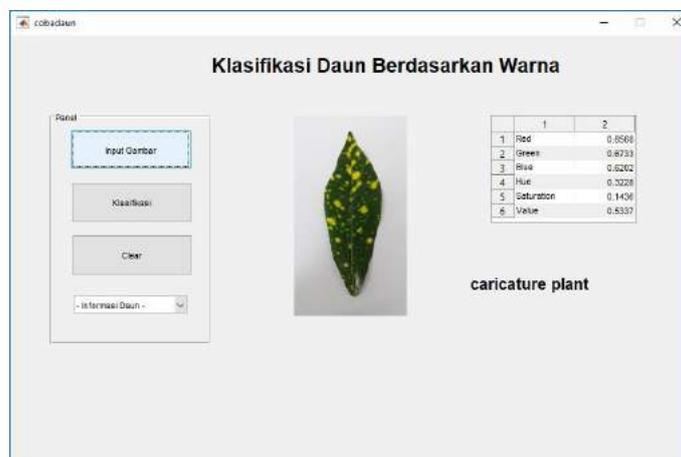
Berikut ini adalah gambar hasil klasifikasi pada salah satu citra daun *bitter orange* yang dapat dilihat pada Gambar 4. Dari gambar tersebut dapat diketahui bahwa gam-

bar yang dimasukkan sesuai dengan hasil klasifikasi. Citra daun yang dimasukkan merupakan daun *bitter orange*. Hasil klasifikasi juga menampilkan nama daun *bitter orange*.

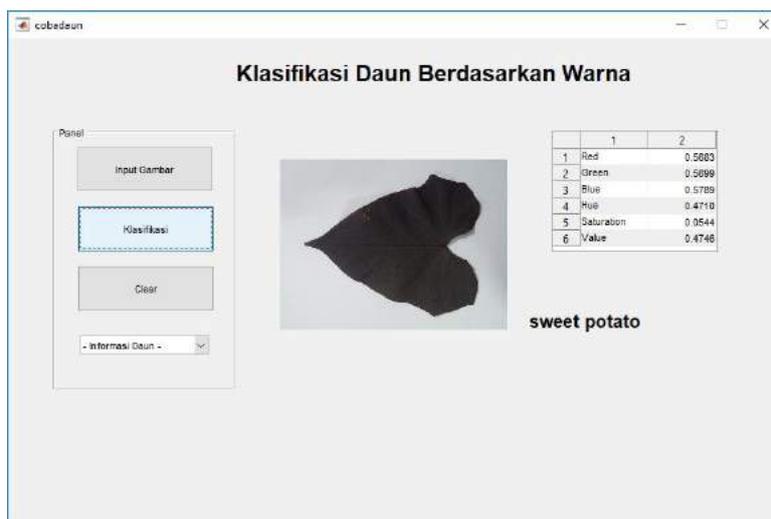
Gambar 5 merupakan gambar hasil klasifikasi pada salah satu citra daun *croton*. Dari gambar tersebut dapat diketahui bahwa gambar yang dimasukkan tidak sesuai dengan hasil klasifikasi. Citra daun yang dimasukkan merupakan daun *croton* sedangkan hasil klasifikasi menampilkan nama daun *caricature plant*.



Gambar 4. Hasil Klasifikasi pada Salah Satu Citra Daun *Bitter Orange*



Gambar 5. Hasil Klasifikasi pada Salah Satu Citra Daun *Croton*



Gambar 6. Hasil Klasifikasi pada Salah Satu Citra Daun *Sweet Potato*

Gambar 6 merupakan gambar hasil klasifikasi dari sebuah citra daun *sweet potato*. Dari gambar tersebut dapat diketahui bahwa gambar yang dimasukkan sesuai dengan hasil klasifikasi. Citra daun yang dimasukkan merupakan daun *sweet potato*. Hasil klasifikasi juga menampilkan nama daun *sweet potato*.

Hasil Parameter Akurasi

Perhitungan parameter akurasi menggunakan *confusion matrix* untuk menentukan *precision*, *recall*, *f1 score*, dan *accuracy*. *Confusion matrix* adalah alat evaluasi atau analisa seberapa baik model klasifikasi yang digunakan untuk mengenali *tuple* dari kelas yang berbeda [9].

Perhitungan akurasi aplikasi ini menghasilkan sebuah *confusion chart* seperti pada Gambar 7.

Dari gambar 7 diketahui bahwa daun *bitter orange* terdeteksi sebagai daun *bitter orange* sebanyak 19 gambar dan terdeteksi

sebagai daun *duranta gold* sebanyak satu gambar. Daun *caricature plant* terdeteksi sebagai daun *caricature plant* sebanyak 20 gambar. Daun *chocolate tree* terdeteksi sebagai *unknown* sebanyak satu gambar, terdeteksi sebagai daun *bitter orange* sebanyak 4 gambar, terdeteksi sebagai daun *caricature plant* sebanyak satu gambar, terdeteksi sebagai daun *chocolate tree* sebanyak 13 gambar, dan terdeteksi sebagai daun *duranta gold* sebanyak satu gambar. Daun *croton* terdeteksi sebagai daun *caricature plant* sebanyak 14 gambar, terdeteksi sebagai daun *croton* sebanyak 5 gambar, dan terdeteksi sebagai daun *sweet potato* sebanyak satu gambar. Daun *duranta gold* terdeteksi sebagai *unknown* sebanyak dua gambar, terdeteksi sebagai daun *chocolate tree* sebanyak 6 gambar, dan terdeteksi sebagai daun *duranta gold* sebanyak 12 gambar. Daun *ficus* terdeteksi sebagai daun *caricature plant* sebanyak 11 gambar dan terdeteksi sebagai daun *ficus* sebanyak 9

gambar. Daun *geranium* terdeteksi sebagai daun *caricature plant* sebanyak 16 gambar, terdeteksi sebagai daun *croton* sebanyak satu gambar, terdeteksi sebagai daun *geranium* sebanyak dua gambar, dan terdeteksi sebagai daun *sweet potato* sebanyak satu gambar. Daun *sweet potato* terdeteksi sebagai *unknown* sebanyak satu gambar, terdeteksi sebagai daun *chocolate tree* sebanyak satu daun, dan terdeteksi sebagai daun *sweet potato* sebanyak 18 gambar. Daun *thevetia* terdeteksi sebagai daun *caricature plant* sebanyak tiga gambar, terdeteksi sebagai daun *croton* sebanyak 6 gambar, terdeteksi sebagai daun *geranium* sebanyak satu gambar, dan terdeteksi sebagai daun *sweet potato* sebanyak 10 gambar. Daun *vieux* terdeteksi

sebagai *unknown* sebanyak tiga gambar, terdeteksi sebagai daun *caricature plant* sebanyak 13 gambar, terdeteksi sebagai daun *duranta gold* sebanyak satu gambar, terdeteksi sebagai daun *geranium* sebanyak satu gambar, dan terdeteksi sebagai daun *sweet potato* sebanyak dua gambar.

Setelah mendapatkan hasil *confusion chart*, dilakukan perhitungan untuk mendapatkan *precision*, *recall*, *f1 score*, dan *accuracy*. Perhitungan *precision*, *recall*, *f1 score*, dan *accuracy* untuk seluruh jenis daun secara lengkap ditunjukkan pada Tabel 3.

Dari tabel 4 dapat diambil kesimpulan bahwa rata-rata untuk *precision*, *recall*, *f1 score*, dan *accuracy* berturut-turut yaitu 49,3%, 49%, 44,24%, dan 90,08%.



Gambar 7. Hasil *Confusion Chart*

Tabel 4. Hasil Perhitungan pada Tiap Jenis Daun

No	Citra Daun	Precision	Recall	F1 Score	Accuracy
1	<i>Bitter Orange</i>	82,6 %	95 %	88,36 %	97,5 %
2	<i>Caricature Plant</i>	25,64 %	100 %	40,81 %	71 %
3	<i>Chocolate Tree</i>	65 %	65 %	65 %	93 %
4	<i>Croton</i>	41,66 %	25 %	31,24 %	87,3 %
5	<i>Duranta Gold</i>	80 %	60 %	68,57 %	94,55 %
6	<i>Ficus</i>	100 %	45 %	62,06 %	94,5 %
7	<i>Geranium</i>	40 %	10 %	16 %	89,5 %
8	<i>Sweet Potato</i>	58,06 %	90 %	70,32 %	92,5 %
9	<i>Thevetia</i>	0 %	0 %	0 %	90,9 %
10	<i>Vieux</i>	0 %	0 %	0 %	90 %
Rata-rata		49,3 %	49 %	44,24 %	90,08 %

KESIMPULAN DAN SARAN

Beberapa kesimpulan yang dapat diambil pada penelitian ini, antara lain proses ekstraksi nilai *red*, *green*, *blue*, *hue*, *saturation*, dan *value* pada citra daun berhasil didapatkan dari setiap citra daun. Klasifikasi citra daun berdasarkan hasil ekstraksi nilai RGB dan HSV berhasil dilakukan dan menampilkan nama daun di setiap proses implementasi. Hasil klasifikasi citra daun memiliki rata-rata akurasi yang tinggi yaitu 90,08%.

Pengembangan lebih lanjut dapat dilakukan untuk penelitian selanjutnya, antara lain dengan menggunakan faktor lain tidak hanya berdasarkan pada citra daun, tetapi juga berdasarkan tekstur, bentuk daun, maupun tepi daun. Selain itu, dapat pula melakukan pengembangan dengan proses machine learning agar mempermudah proses pelatihan dan iden-

tifikasi yang sudah tersedia menggunakan library khusus.

DAFTAR PUSTAKA

- [1] K. D. Septiaji dan K. Firdausy, "Deteksi Kematangan Daun Selada (*Lactuca Sativa L*) Berbasis Android Menggunakan Nilai RGB Citra", *Jurnal Ilmu Teknik Elektro Komputer dan Informatika (JITEKI)*, vol. 4, no. 1, hal. 20-27, 2018.
- [2] H. Prabowo, "Deteksi Kondisi Kematangan Buah Jeruk Berdasarkan Kemiripan Warna pada Ruang Warna RGB Berbasis Android", *Jurnal Elektronik Sistem Informasi dan Komputer*, vol.3, no.2, hal. 9-19, 2017.
- [3] I. P. Sari, B. Hidayat, dan R. D. Atmaja, "Perancangan dan Simulasi

- Deteksi Penyakit Tanaman Jagung Berbasis Pengolahan Citra Digital Menggunakan Metode *Color Moments* dan GLCM”, dalam Seminar Nasional Inovasi dan Aplikasi Teknologi di Industri (SENIATI), 2016, hal. 215-220.
- [4] F. R. Lestari, J. Y. Sari, Sutardi, I. Purwanti, dan N. Purnama, ”Deteksi Penyakit Tanaman Jeruk Siam Berdasarkan Citra Daun Menggunakan Segmentasi Warna *RGB-HSV*”, dalam Seminar Nasional Teknologi Terapan Berbasis Kearifan Lokal (SNT2BKL), 2018, hal. 276-283.
- [5] P. Rianto dan A. Harjoko, ”Penentuan Kematangan Buah Salak Pondoh di Pohon Berbasis Pengolahan Citra Digital”, *International Journal of Computing and Cybernetics Systems*, vol. 11, no.2, hal. 143-154, 2017.
- [6] I. Y. Ningsih, ”Anatomi dan Morfologi Daun”, Modul Botani Farmasi, Universitas Jember, 2015.
- [7] R. Kusumanto dan A. N. Tomponu, ”Pengolahan Citra Digital untuk Mendeteksi Obyek Menggunakan Pengolahan Warna Model Normalisasi *RGB*“, dalam Seminar Nasional Teknologi Informasi & Komunikasi Terapan, 2011.
- [8] S. Madenda, *Pengolahan Citra & Video Digital*. Jakarta: Erlangga, 2015.
- [9] J. Han, M. Kamber, dan J. Pei, *Data Mining Concept and Techniques*, USA: Elsevier, 2012. [E-book]

IMPLEMENTASI ALGORITMA K-MEANS CLUSTERING PADA ANALISIS SENTIMEN KELUHAN PENGGUNA INDOSAT

¹Try Iryanto Saputra, ²Rini Arianty

^{1,2}Fakultas Ilmu Komputer Universitas Gunadarma
Jl. Margonda Raya No. 100, Depok 16424, Jawa Barat
² rinia @staff.gunadarma.ac.id,

Abstrak

Penyampaian keluhan konsumen lewat akun media sosial seperti Twitter dimaksudkan agar masalah yang dihadapi konsumen dapat diselesaikan dengan cepat. Pada penelitian ini, akan dilakukan analisis sentimen terhadap konsumen pengguna provider Indosat, menggunakan data tweet sejumlah 300 data acak yang di kumpulkan dari bulan desember 2018 hingga bulan april 2019. Data yang dianalisis adalah kalimat berbahasa Indonesia. Preprocessing pada penelitian ini terdiri dari beberapa tahapan proses antara lain tokenizing, filtering, stop word, dan stemming. Analisis dilakukan menggunakan metode K-Means Clustering. Penelitian ini berhasil menampilkan kelompok dari anggota masing-masing cluster yang berbentuk wordcloud ke dalam 3 buah wordcloud berbeda, pada wordcloud cluster 0 anggotanya berbicara tentang jaringan Indosat yang parah, pada wordcloud cluster 1 anggotanya berbicara tentang permintaan perbaikan jaringan sinyal Indosat, dan pada wordcloud cluster 2 anggotanya berbicara tentang jaringan sinyal parah Indosat pada daerah Bogor. Hasil penelitian ini diharapkan dapat menjadi masukan untuk provider dalam melihat keluhan yang masuk dari para konsumen mereka sehingga pihak provider dapat meningkatkan pelayanannya.

Kata Kunci: Filter, K-Means Clustering, stemming, token, Tweet.

Abstract

Submitting a consumer complaint through this social media such as Twitter account responds so that the problems the consumer asks can be resolved quickly. In this study, sentiment analysis will be carried out by Indosat user opinions, using tweet data based on 300 random data collected from December 2018 to April 2019. The successful data is the Indonesian dialogue sentence. Preprocessing in this study consists of several processes including tokenizing, filtering, stop word, and stemming. The analysis was performed using the K-Means Clustering method. This research succeeded in displaying the groups of each group consisting of wordcloud into 3 different wordcloud, in wordcloud cluster 0 the members talked about bad Indosat networks, in wordcloud cluster 1 the members needed the help of Indosat communication networks, and in wordcloud cluster 2 its members talked about Indosat's bad signal network in Bogor area. The results of this study are expected to be input for providers in seeing complaints coming from their customers so that providers can improve their services.

Key Words : Filter, K-Means Clustering, stemming, token, Tweet

PENDAHULUAN

Banyaknya pengguna media sosial aktif dan banyaknya manfaat yang diberikan oleh

media sosial, membuat perusahaan ikut membuat sebuah sarana yang dapat menampung masalah yang dihadapi kliennya melalui sebuah akun media sosial. Salah

satunya adalah perusahaan *provider* di Indonesia seperti XL, Indosat, Smartfren dan lainnya, membuat akun layanan untuk menampung masalah konsumen mereka. Tujuan penyampaian keluhan konsumen lewat akun media sosial ini dimaksudkan agar masalah yang dihadapi konsumen dapat diselesaikan dengan cepat. Sayangnya dibalik manfaatnya tersebut para konsumen tersebut menggunakan kata-kata kasar untuk mengekspresikan kekecewaan atas pelayanan perusahaan tersebut. Media sosial yang digunakan perusahaan untuk akun keluhan konsumen salah satunya adalah twitter. Postingan *tweet* yang dikirim untuk perusahaan tidak hanya sebuah *tweet* yang berisi masalah, ada juga sebuah bentuk *tweet* yang mendukung dan mengapresiasi hasil dari pelayanan perusahaan tersebut. Dari masalah tersebut peneliti tertarik membuat penelitian Tentang analisis sentimen data *tweet* opini konsumen terhadap layanan *provider* Indosat. Analisis sentimen atau *opinion mining* adalah studi komputasional dari opini-opini orang, sentimen [1] dan emosi melalui entitas dan atribut yang dimiliki yang diekspresikan dalam bentuk teks [2]. Tugas dasar dalam analisis sentimen adalah mengelompokkan teks yang ada dalam sebuah kalimat atau dokumen [3] kemudian menentukan pendapat yang dikemukakan dalam kalimat atau dokumen tersebut apakah bersifat positif, negatif atau netral [4]. Analisis sentimen adalah proses yang digunakan untuk menentukan opini, emosi dan sikap yang

dicerminkan melalui teks, dan biasanya diklasifikasikan menjadi opini positif dan negatif[5].

Beberapa penelitian dilakukan terkait sentimen analisis khususnya terhadap data sosial media. Penelitian dilakukan [6] menggunakan metode K-Means *Clustering*. Percobaan dilakukan pada 1000 data yang dikelompokkan menjadi lima *cluster* yaitu cinta, marah, sedih, senang dan takut. Implementasi metode K-Means pada penelitian tersebut menghasilkan akurasi sebesar 76,3%. Selanjutnya penelitian dilakukan [7] menggunakan metode K-Means untuk optimasi klasifikasi tema tugas akhir mahasiswa menggunakan *Support Vector Machine* (SVM). Hasil yang diperoleh memiliki tingkat akurasi yang lebih baik yaitu 86,21%. Penelitian yang dilakukan [8] menggunakan dataset positif dan dataset negatif, untuk analisis sentimen *review* film menggunakan algoritma K-Means. Dari setiap uji coba menggunakan data dengan jumlah berbeda menunjukkan bahwa akurasi mencapai 57,83% didapatkan menggunakan 300 dataset positif dan 300 dataset negatif, sedangkan 700 dataset positif dan 700 dataset negatif menunjukkan akurasi mencapai 56,71%, jika menggunakan 1000 dataset positif dan 1000 dataset negatif akurasinya mencapai 50,40%. Penelitian [9] melakukan penelitian analisis sentimen kurikulum 2013 pada twitter menggunakan *Ensemble Feature* dan metode K-Nearest Neighbor. Penelitian ini digunakan untuk mengetahui opini yang berkembang

mengenai kurikulum 2013 yang dibagi kedalam opini positif atau opini negatif. Berdasarkan hasil pengujian dengan menggabungkan kedua fitur tersebut mendapatkan hasil akhir akurasi mencapai 96%. Penelitian analisis sentimen [10] data komentar sosial media facebook dengan K-Nearest Neighbor dengan studi kasus pada akun jasa ekspedisi barang J&T. dalam tahapan penelitiannya dilalui melalui beberapa tahap yaitu *preprocessing* yang terdiri dari *case folding*, *tokenizing*, *stopword removal*, dan *stemming*. Hasil yang didapatkan dari implementasi metode KNN dengan akurasi tertinggi adalah 79,21% sedangkan akurasi terendahnya adalah 70,3%.

METODE PENELITIAN

Analisis sentimen terhadap data keluhan pengguna Indosat lewat media sosial twitter terdiri atas tahapan preprocessing kalimat berupa case folding, tokenization,

filterisasi, stopword, stemming dan labelisasi menggunakan TF-IDF yang kemudian akan dilakukan clustering menggunakan K-means *clustering* untuk mendapatkan *wordcloud*.

Data Tweet

Pengumpulan data dilakukan secara manual dengan mengumpulkan komentar pada website twitter dengan alamat <https://twitter.com/IndosatCare>. Data komentar yang dikumpulkan dari @IndosatCare diambil dari bulan desember 2018 hingga bulan April 2019. Komentar yang diambil sebagai data berdasarkan komentar opini dari konsumen @IndosatCare terhadap layanan yang diberikan oleh @IndosatCare terhadap konsumen dengan jumlah data yang dikumpulkan ialah 300 data komentar yang terdiri dari komentar positif dan negatif seperti dapat dilihat pada Gambar 1. Data kemudian simpan data dengan ekstensi CSV(*Comma Separated Values File*).



Gambar 1. Pengambilan Data KomentarPP

Preprocessing

Preprocessing pada penelitian ini terdiri atas tahapan sebagai berikut :

1. *Case Folding*. Data komentar akan di proses untuk merubah semua bentuk huruf besar yang ada pada data komentar dirubah menjadi huruf kecil.
2. *Tokenizing*. Data komentar yang telah masuk ke proses *tokenizing*, akan berubah menjadi potongan kata yang masing-masing katanya memiliki nilai yang berbeda.
3. *Filtering* dan *Stopword*. Pada tahap *filtering* merupakan tahapan ketiga dari proses *preprocessing*, pada tahap ini dilakukan penghapusan tanda baca dan hashtag yang tidak memiliki arti. Pada proses ini kata-kata yang memiliki nilai kecil pada data komentar akan dihilangkan dan akan disesuaikan dengan daftar *stopword*.
4. *Stemming*. Pada tahap *stemming* merupakan lanjutan dari tahap *filtering* dan *Stopword*, data komentar yang telah masuk tahap *filtering* dan *Stopword* yang berupa penggalan kata dasar dari suatu komentar akan dirubah menjadi bentuk kata baku bahasa Indonesia yang baik dan benar.

Pembobotan *Term Weighting* TF-IDF

Setelah melakukan proses *pre-processing*, selanjutnya masuk ke proses pembobotan kata. Pembobotan kata dilakukan untuk memberikan sebuah nilai terhadap suatu kata berdasarkan tingkat kepentingan kata pada kumpulan dokumen yang telah dimasukan ke dalam sistem. Sebagai contoh digunakan lima kalimat komentar sebagai salah satu contoh perhitungan, dengan uraian satu kalimat digunakan sebagai data uji dan empat kalimat digunakan sebagai data latih.

Tahap pembobotan kata terdapat empat langkah yang harus dilakukan, diantaranya sebagai berikut:

1. Menghitung jumlah *term frequency* (*tf*) tiap kata. Setelah melakukan tahap *pre-processing*, langkah selanjutnya ialah menghitung jumlah *term frequency*(*tf*), penghitungan dilakukan dengan memecah kalimat menjadi kata perkata seperti dapat dilihat pada Gambar 2.
2. Menghitung jumlah *document frequency* (*df*) tiap kata.
3. Menghitung *inverse document frequency* (*idf*) menggunakan persamaan 1.

$$idf = \log \frac{N}{df} \quad (1)$$

Signal Lot Bgt Min	Tf Signal = 1 Lot = 1 Bgt = 1 Min = 1
-----------------------------	--

Gambar 2. Contoh Perhitungan *Term Frequency* pada Kalimat

Contoh:

Pada kata “jam” hanya terdapat didalam satu dokumen saja, maka perhitungannya adalah sebagai berikut :

$$idf = \log \frac{5}{1} = \log 5 = 0.6989700043$$

Pada kata “gue” ditemukan pada dua dokumen yang sama, maka perhitungannya adalah sebagai berikut :

$$idf = \log \frac{5}{2} = \log 2.5 = 0.3979400087$$

Kata “min” ditemukan pada tiga dokumen yang berbeda, maka perhitungannya adalah sebagai berikut:

$$idf = \log \frac{5}{3} = \log 1.66666667 = 0.2218487505$$

- Menghitung bobot atau *weight* (w) dari hasil perkalian *term frequency* (tf) dengan *inverse document frequency* (idf). Langkah pertama dilakukan dengan mengkalikan nilai tf dengan idf , hasil dari perkalian seluruh kata dijumlahkan untuk mendapatkan bobot atau *weight* (w) dari setiap kalimat dapat dilihat pada Persamaan 2.

$$tfidf = tf(t, d) * idf \quad (2)$$

Persamaan 3 digunakan untuk mendapatkan nilai bobot atau *weight* dari keseluruhan kalimat menggunakan rumus sebagai berikut :

$$w = (tfidf_1 + tfidf_2 + \dots + tfidf_n) \quad (3)$$

Contoh hasil perhitungan $tf-idf$ kalimat negatif pada masing-masing kata dan nilai bobot dalam setiap kalimat disajikan pada Tabel 1.

Analisis Metode K-Means Clustering

Langkah-langkah yang dilakukan untuk menganalisis menggunakan metode K-Means yaitu:

- Menentukan jumlah k cluster. Jumlah cluster yang digunakan ialah 2 cluster.
- Menentukan *centroid* awal (C_i). Setelah melakukan *pre-processing* dan pembobotan (w), maka setiap bobot kalimat akan digunakan untuk perhitungan K-Means clustering. Berikut data yang didapat dari perhitungan bobot (w):

- Kalimat 1 (k_0) = 9.08661006
- Kalimat 2 (k_1) = 2.79588002
- Kalimat 3 (k_2) = 8.38764005
- Kalimat 4 (k_3) = 3.49485002
- Kalimat 5 (k_4) = 2.09691001

Tabel 1. Contoh Perhitungan Bobot Kalimat

No	Kata	Tf	Idf	Tf-idf
1	signal	1	0.6989700043	0.6989700043
2	lot	1	0.6989700043	0.6989700043
3	Bgt	1	0.6989700043	0.6989700043
4	Min	1	0.6989700043	0.6989700043
Bobot atau Weight (w)				2.79588002

Berdasarkan data hasil pembobotan kalimat di atas, diambil nilai terkecil (kalimat 5 (k_4) = 2.09691001) dan nilai terbesar (kalimat 1 (k_0) = 9.08661006) untuk menentukan *centroid*.

3. Mengelompokkan setiap titik data ke *cluster* terdekat untuk menemukan *centroid* baru (C_i). Setelah menentukan titik pusat atau *centroid*, selanjutnya adalah mengelompokkan data ke dalam masing-masing *cluster*. Setiap *cluster* akan mendapatkan anggota yang ditentukan dari nilai bobot (w) kalimat, hasil penentuan akan menentukan anggota tersebut masuk ke dalam *cluster* C1 atau *cluster* C2. Masing-masing *cluster* akan memiliki anggota seperti berikut ini :

$$1) C1 = k_1, k_3, k_4$$

$$2) C2 = k_0, k_2$$

4. Menghitung jarak masing-masing anggota *cluster*, untuk mendapatkan *centroid* baru dengan menggunakan persamaan 4.

$$C_{(i)} = \frac{(x_1 + x_2 + \dots + x_n)}{\sum x} \quad (4)$$

Keterangan

$C_{(i)}$ = *centroid* baru

x_1, x_2, x_n = merupakan nilai pada masing-masing anggota *cluster*

$\sum x$ = jumlah anggota *cluster*

Mengulangi kembali langkah 3 dan 4 sampai pusat *cluster* dan anggota *cluster* tidak berubah atau tetap sama.

HASIL DAN PEMBAHASAN

Hasil perhitungan jarak masing-masing anggota *cluster*, untuk mendapatkan *centroid* baru dengan menggunakan persamaan 4 didapatkan nilai *cluster* C1 sebesar 2.79588002 dan *cluster* C2 sebesar 8.73712505.

Gambar 3 merupakan hasil Tabel *content* yang mengambil data dari tahap *stemming* untuk diproses kedalam perhitungan K-Means *Clustering* untuk menentukan kumpulan kalimat dari data yang diolah masuk ke *cluster* mana dengan menampilkan kumpulan data *testing* dengan menampilkan masing-masing kalimat tersebut memasuki *cluster* yang sesuai dengan hasil dari algoritma K-Means *Clustering*.

Implementasi Gambar 3 kemudian dibuat sebuah *wordcloud* untuk menampilkan sebaran kata dari data komentar sesuai masing-masing *clusternya* dengan menampilkan *wordcloud* dengan judul *cluster 0*, *cluster 1* dan *cluster 2* seperti dapat dilihat pada Gambar 4.

Testing	Cluster
0 [halah, komhof]	2
1 [sinyal, nya, ga, stable, nih, banget, darurat, udah, udah, normal, darurat, bom, ment, pag, gnu]	0
2 [balikin, pulsa, dulu, kang, sedot, layan, sms, gak]	1
3 [indosat, ga, tanggung, pulsa, sedot, paket, ga, udah, jaring, pulsa, sedot]	1
4 [si, im, gim, kemarin, jaring, bunuk, beli, kartu, kuota, eh, gak, kuota, cepet, gak, asik]	2
5 [bawak, jaring, jek, indosat, bay, bgt, telkomsel]	2
6 [tolong, jaring, daerah, bogor, bogor, benger, jek, udah, kaya, kartu, ajr, sinyal]	0
7 [iya, nih, udah, bingung, ment, reconnect, paket, data, buan, ga, nepake]	1
8 [jresok, pake, indosat, faka]	1
9 [pabokk, gue, mokein, paket, yellow, udah, tulis, akrip, bohevi, tuk, gue, tek, bahas, stfhar, paketnya, nih, gimana, pulsa, hilang, gak, pakettanakin, moles, pake, indosat, lot, curang, jaring, lot, kembali, pulsa, kampereeeeeeet]	1
10 [mtr, kayer, lagi, sinyal, jekki, parah, mo, pake, kartu, indosat, pindah, kartu]	1
11 [sinyal, tp, ga, konektivitas, bngst]	1

Gambar 3. Implementasi K-Means Clustering



Gambar 4. Implementasi Tampilan Cluster K-Means

KESIMPULAN DAN SARAN

Implementasi metode K-Means Clustering menggunakan data komentar tweet @Indosatcare telah berhasil dibuat dan menghasilkan 3 buah cluster dengan anggota berbeda di tiap cluster-nya. Penelitian ini berhasil menampilkan kelompok dari anggota masing-masing cluster yang berbentuk word-cloud ke dalam 3 buah wordcloud berbeda, pada wordcloud cluster 0 anggotanya berbicara tentang jaringan Indosat yang parah, pada wordcloud cluster 1 anggotanya berbicara tentang permintaan perbaikan jaringan

sinyal Indosat, dan pada wordcloud cluster 2 anggotanya berbicara tentang jaringan sinyal parah Indosat pada daerah Bogor.

Hasil penelitian ini diharapkan dapat menjadi masukan untuk provider dalam melihat keluhan yang masuk dari para konsumen mereka sehingga pihak provider dapat meningkatkan pelayanannya.

Pengembangan lebih lanjut dapat dilakukan untuk menambahkan metode klasifikasi lainnya kedalam metode K-Means Clustering, dengan menggabungkan 2 metode yang berbeda untuk mendapatkan hasil yang berbeda.

DAFTAR PUSTAKA

- [1] B. Pang dan L. Lee, "Opinion Mining and Sentiment Analysis", *Foundations and Trends in Information Retrieval*, vol. 2, no. 1-2, hal. 1–135, 2008.
- [2] A. Chaudhuri dan J. Chant, "Protein-interaction mapping in search of effective drug target", *BioEssays* 27(9): 958–969, 2005
- [3] B. Liu, *Sentiment Analysis and Opinion Mining*. Morgan & Claypool Publisher, 2012
- [4] C. D. Manning, P. Raghavan, dan H. Schutze, *An introduction to information retrieval*. Cambridge: Cambridge University Press, 2008.
- [5] Y. Y. Luhulima, "Sentimen Analysis Pada Review Barang Berbahasa Indonesia dengan Metode K-Nearest Neighbor", Skripsi Sarjana, Universitas Brawijaya, Malang, Indonesia, 2013.
- [6] B. Nugroho, *Membuat Website Sendiri Dengan PHP & MySql*. Jakarta: MediaKita, 2009.
- [7] O. Somantri, S. Wiyono dan Dairoh, "Metode *K-Means* untuk optimasi klasifikasi tema tugas akhir mahasiswa menggunakan *Support Vector Machine (SVM)*", *Scientific Journal of Informatics*, vol. 3, no. 1, 2016.
- [8] S. Budi, "*Text mining* untuk analisis sentimen review film menggunakan algoritma *K-Means*", *TehcnoCOM*, vol. 16, no. 1, 2017
- [9] N. D. Mentari, M. A. Fauzi, dan L. Muflikhah, L, "Analisis sentimen kurikulum 2013 pada sosial media Twitter menggunakan metode *K-Nearest Neighbor* dan *Feature Selection Query Expansion Ranking*", *Jurnal Pengembangan Teknologi Informasi dan Ilmu Komputer (J-PTIIK)*, vol. 2, no. 8, hal. 2739–2743, 2018.
- [10] A. Salam, J. Zeniarja, dan R. S. U. Khasanah, "Analisis sentimen data komentar sosial media Facebook dengan *K-Nearest Neighbor* (Studi kasus pada akun jasa barang J&T Ekspres Indonesia)", dalam *Prosiding SINTAK*, 2018, hal.480–486.

PENGENALAN OBJEK MAKANAN CEPAT SAJI PADA VIDEO DAN REAL TIME WEBCAM MENGGUNAKAN METODE YOU LOOK ONLY ONCE (YOLO)

¹Oktaviani Ella Karlina, ²Dina Indarti

^{1,2}Fakultas Teknologi Industri Universitas Gunadarma
Jl. Margonda Raya No. 100, Depok 16424, Jawa Barat
¹oek40506@gmail.com, ²dina_indarti@staff.gunadarma.ac.id

Abstrak

Makanan cepat saji banyak diminati masyarakat di Indonesia saat ini. Makanan cepat saji mengandung lemak dan kalori yang tinggi tetapi kurang mengandung gizi yang sangat dibutuhkan tubuh sehingga dapat menimbulkan berbagai penyakit. Salah satu cara untuk mengontrol konsumsi makanan cepat saji yaitu penggunaan metode pengenalan objek makanan cepat saji dari video dan real time webcam. Metode yang dapat digunakan untuk pengenalan objek pada citra makanan cepat saji adalah deep learning. You Look Only Once (YOLO) merupakan salah satu model deep learning yang dapat digunakan untuk pengenalan objek. Penelitian ini bertujuan untuk pengenalan objek pada citra makanan cepat saji menggunakan YOLO. Penelitian ini terdiri dari beberapa tahap yaitu pengumpulan data, pra-proses data, konfigurasi jaringan YOLO, pelatihan model YOLO dan pengujian. Jumlah data citra yang digunakan dalam pelatihan yaitu 468 citra yang terdiri dari tiga jenis makanan cepat saji. Nilai avg loss pada model akhir yang dibangun dengan YOLO yaitu 4.6% dan nilai validasi mAP 100%. Berdasarkan hasil pengujian yang dilakukan menggunakan video dan real time webcam, objek pada citra makanan cepat saji berhasil dikenali dengan akurasi 63% sampai 100%.

Kata Kunci: Deep learning, makanan cepat saji, pengenalan objek, You Only Look Once

Abstract

Fast food is much in demand by people in Indonesia today. Fast food contains high fat and calories but lacks nutrients that are needed by the body so that it can cause various diseases. One way to control fast food consumption is the use of fast food object recognition methods from video and real time webcam. The method that can be used for object recognition in fast food images is deep learning. You Look Only Once (YOLO) is a deep learning model that can be used for object recognition. This study aims to recognize objects in the image of fast food using YOLO. This research consists of several stages, namely data collection, data preprocessing, YOLO network configuration, YOLO model training and testing. The amount of image data used in the training is 468 images consisting of three types of fast food. The avg loss value in the final model built with YOLO is 4.6% and the mAP validation value is 100%. Based on the results of tests conducted using video and real time webcam, objects in the image of fast food have been detected properly.

Keywords: Deep learning, fast food, object recognition, You Only Look Once

PENDAHULUAN

Manusia memiliki kebutuhan paling mendasar yang merupakan prioritas diban-

dingkan kebutuhan lainnya yaitu kebutuhan fisiologis. Kebutuhan fisiologis merupakan kebutuhan yang harus terpenuhi sehingga manusia dapat bertahan hidup. Salah satu

kebutuhan fisiologis yaitu makanan atau pangan [1]. Definisi pangan menurut Badan Pengawas Obat dan Makanan Republik Indonesia (BPOM RI) adalah segala sesuatu yang berasal dari sumber hayati baik yang diolah maupun tidak diolah yang digunakan untuk konsumsi manusia [2].

Indonesia dengan jumlah penduduk mencapai 261,9 juta pada tahun 2017 menjadi negara tujuan pemasaran produk pangan atau makanan dari berbagai negara [3]. Berbagai produk makanan masuk ke Indonesia sehingga terdapat banyak alternatif pilihan makanan mulai dari jenis, harga, maupun kualitas makanan yang beraneka ragam. Pengaruh masuknya berbagai produk makanan dari berbagai negara berdampak pada pergeseran konsep makanan masyarakat di Indonesia. Salah satu bentuk pergeseran konsep makanan di Indonesia yaitu adanya makanan cepat saji (*fast food*) dalam pilihan konsumsi masyarakat di Indonesia [4].

Makanan cepat saji merupakan makanan yang dapat dikonsumsi dalam waktu seminimal mungkin [5]. Makanan cepat saji sangat mudah ditemui dan hampir semua masyarakat pernah mengkonsumsi. Para ahli gizi dan dokter mengatakan bahwa makanan cepat saji mengandung lemak dan kalori yang tinggi tetapi kurang mengandung gizi yang sangat dibutuhkan tubuh. Oleh karena itu, jika terlalu sering mengkonsumsi makanan cepat saji dapat membahayakan kesehatan tubuh [6]. Banyak jenis makanan cepat saji yang diminati masyarakat di Indonesia antara lain

pizza, *hamburger*, dan *hotdog*. Salah satu cara untuk mengontrol konsumsi makanan cepat saji diperlukan suatu metode pengenalan objek makanan cepat saji dari video dan *realtime webcam*.

Pengenalan objek merupakan salah satu permasalahan dalam *computer vision* yang berhubungan dengan pengenalan suatu objek dalam citra dan video digital [7]. *Computer vision* merupakan salah satu cabang dari kecerdasan buatan (*artificial intelligence*) yang difokuskan pada pengembangan algoritma untuk menganalisis informasi dari suatu citra ke dalam bentuk informasi yang sebenarnya di dunia nyata. Saat ini *deep learning* banyak digunakan pada penelitian mengenai *computer vision*. Terdapat dua pendekatan dalam pengenalan objek menggunakan *deep learning* yaitu pendekatan berdasarkan *region proposal* atau metode dua tahap dan pendekatan berdasarkan metode regresi atau metode satu tahap. Metode pengenalan objek yang menggunakan dua tahap antara lain *region-based convolutional neural networks* (R-CNN) [8], *Spatial Pyramid Pooling Network* (SPP-Net) [9], *Fast R-CNN* [10], and *Faster R-CNN* [11]. Metode tersebut menggunakan dua tahap karena mengklasifikasikan ulang dengan menerapkan model ke citra di beberapa lokasi dan skala lalu memberi nilai pada citra sebagai bahan evaluasi untuk pengenalan objek. Metode pengenalan objek dalam satu tahap tanpa melakukan klasifikasi ulang sehingga menjadi lebih cepat yaitu *You Only*

Look Once (YOLO) [12], modifikasi YOLO (YOLOv2) [13], dan YOLO-LITE [14]. YOLO menerapkan *neural network* pada sebuah citra, kemudian membagi citra menjadi daerah dan memprediksi *bounding box* serta probabilitas untuk masing-masing daerah. Probabilitas untuk setiap *bounding box* kemudian dihitung untuk mengklasifikasikan sebagai objek atau bukan. YOLO dapat melakukan pengenalan objek secara *real-time* dengan kecepatan 45 *frame per second* [12]. YOLO-LITE merupakan pengembangan dari YOLO yang dapat dijalankan pada komputer *portable* seperti laptop atau *cellphone* dengan keterbatasan *Graphics Processing Unit* (GPU) [14].

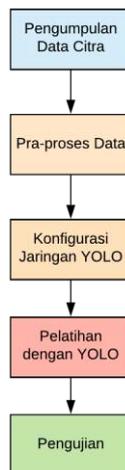
Beberapa penelitian pengenalan objek menggunakan YOLO telah dilakukan. Penelitian yang dilakukan oleh Zhang, Huang, Jin, dan Li membahas mengenai algoritma pengenalan rambu lalu lintas secara *real-time* menggunakan modifikasi YOLO. Hasil penelitian tersebut menunjukkan bahwa metode modifikasi YOLOv2 dapat mendeteksi rambu lalu lintas dengan lebih cepat dan lebih baik [15]. Liu, Chen, Li, dan Hu melakukan penelitian mengenai metode pendeteksian citra pejalan kaki berdasarkan pada YOLOv2. Hasil penelitian tersebut menunjukkan bahwa metode YOLOv2 memiliki akurasi dan kecepatan yang baik dalam mendeteksi objek

pada citra pejalan kaki [16]. Hasil penelitian yang dilakukan oleh Jupiyandi, Saniputra, Pratama, Dharmawan, dan Cholissodin menunjukkan bahwa algoritma modifikasi YOLO dapat mendeteksi jumlah mobil dengan tepat sehingga dapat mengetahui jumlah- tempat parkir mobil yang masih tersedia [17].

Oleh karena itu, pada penelitian ini membahas mengenai pengenalan objek pada citra makanan cepat saji menggunakan *You Only Look Once* (YOLO). Model YOLO yang diperoleh selanjutnya digunakan untuk mengenali objek dari video dan *real-time webcam*.

METODE PENELITIAN

Tahapan-tahapan penelitian secara umum dapat dilihat pada Gambar 1. Berdasarkan Gambar 1, penelitian dimulai dengan pengumpulan data citra lalu pra-proses data citra yang terdiri dari pelabelan dan perubahan ukuran citra. Kemudian dilakukan konfigurasi jaringan YOLO sesuai dengan data citra dan dilatih sehingga model baru YOLO terbentuk. Tahapan terakhir penelitian yaitu melakukan pengujian model menggunakan data video dan *real-time webcam*.



Gambar 1. Skema Penelitian

Pengumpulan Data Citra

Citra makanan cepat saji yang digunakan dalam penelitian ini diambil dari *dataset* pada penelitian yang dilakukan oleh Matsuda, Hoashi, dan Yanai [18]. Jumlah citra sebagai *dataset* adalah 468 citra dengan *format* jpg, yang terdiri dari tiga jenis makanan yaitu *hamburger* berjumlah 233 citra, *pizza* berjumlah 133 citra dan *hot dog* berjumlah 102 citra.

Pra-proses Data Citra

Selanjutnya dilakukan Pra-proses data yang terdiri dari pelabelan dan perubahan ukuran citra. Pelabelan citra merupakan tahap awal dimana setiap citra pada *dataset* diberikan label dengan tujuan untuk menyimpan informasi citra. Proses label dilakukan dengan cara memberikan *bounding box* beserta nama kelas pada setiap objek citra. Selanjutnya dilakukan perubahan ukuran citra untuk meningkatkan performa model YOLO dalam pengenalan objek.

Konfigurasi Jaringan YOLO

Konfigurasi jaringan diperlukan sebagai jaringan model untuk memuat data yang akan dilatih. *Batch* atau iterasi yang digunakan yaitu 64 yang berarti setiap iterasi memuat 64 citra. *Subdivision* yaitu 8 sehingga proses pemodelan membagi *batch* menjadi 8 bagian. Oleh karena itu, terdapat $64/8 = 8$ citra per *minibatch* yang dikirim ke GPU untuk diproses dan diulang sebanyak 8 kali hingga selesai. Pada iterasi selanjutnya dimulai dengan 64 citra baru.

Pelatihan Model YOLO

Model YOLO yang digunakan pada penelitian ini memiliki jaringan saraf yang terdiri dari lapisan konvolusi dengan *kernel* 3×3 dan lapisan *max-pooling* dengan 2×2 *kernel* seperti terlihat pada Gambar 2. Lapisan konvolusional terakhir memiliki 1×1 *kernel* digunakan untuk mengecilkan data ke bentuk $13 \times 13 \times 40$. 13×13 adalah ukuran *grid*, 40 didapat dari penjumlahan rumus *filter*.

layer	filters	size/strd(dil)	input	output
0 conv	16	3 x 3/ 1	416 x 416 x 3 ->	416 x 416 x 16 0.150 BF
1 max		2 x 2/ 2	416 x 416 x 16 ->	208 x 208 x 16 0.003 BF
2 conv	32	3 x 3/ 1	208 x 208 x 16 ->	208 x 208 x 32 0.399 BF
3 max		2 x 2/ 2	208 x 208 x 32 ->	104 x 104 x 32 0.001 BF
4 conv	64	3 x 3/ 1	104 x 104 x 32 ->	104 x 104 x 64 0.399 BF
5 max		2 x 2/ 2	104 x 104 x 64 ->	52 x 52 x 64 0.001 BF
6 conv	128	3 x 3/ 1	52 x 52 x 64 ->	52 x 52 x 128 0.399 BF
7 max		2 x 2/ 2	52 x 52 x 128 ->	26 x 26 x 128 0.000 BF
8 conv	256	3 x 3/ 1	26 x 26 x 128 ->	26 x 26 x 256 0.399 BF
9 max		2 x 2/ 2	26 x 26 x 256 ->	13 x 13 x 256 0.000 BF
10 conv	512	3 x 3/ 1	13 x 13 x 256 ->	13 x 13 x 512 0.399 BF
11 max		2 x 2/ 1	13 x 13 x 512 ->	13 x 13 x 512 0.000 BF
12 conv	1024	3 x 3/ 1	13 x 13 x 512 ->	13 x 13 x1024 1.595 BF
13 conv	512	3 x 3/ 1	13 x 13 x1024 ->	13 x 13 x 512 1.595 BF
14 conv	40	1 x 1/ 1	13 x 13 x 512 ->	13 x 13 x 40 0.007 BF
15 detection				

Gambar 2. Model YOLO

Pengujian Model YOLO

Pengujian dilakukan dengan dua cara yaitu video dan *real-time* menggunakan *webcam* laptop. Video didapat dari situs *youtube* menggunakan kata kunci makanan *hamburger, pizza, dan hot dog*. Pengujian dilakukan untuk mengetahui tingkat akurasi objek dengan model baru yang sudah dilatih sebelumnya.

HASIL DAN PEMBAHASAN

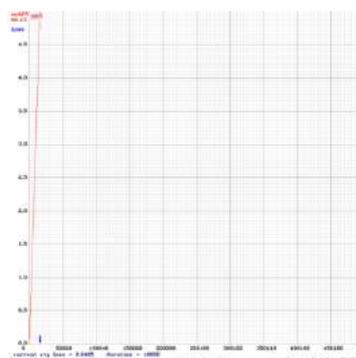
Hasil Pelatihan

Proses pelatihan dilakukan sebanyak 16.600 iterasi yang menghasilkan sebuah model baru. Model akhir yang dibangun memiliki nilai *avg loss* 4.6% dan nilai validasi

mAP (mean Average Precision) 100% seperti terlihat pada Gambar 3. Nilai *mAP* 100% menunjukkan bahwa tingkat akurasi pengenalan objek pada citra makanan cepat saji menggunakan model YOLO yang telah dilatih sebesar 100%.

Hasil Pengujian Pada Video

Salah satu hasil pengujian pada sebuah video dengan objek *pizza* dapat dilihat pada Gambar 4. Berdasarkan Gambar 4, akurasi pengenalan objek *pizza* pada video sebesar 99%. Pengenalan objek pada video dapat memberikan hasil akurasi yang berbeda-beda pada saat objek yang dikenali berpindah posisi.



Gambar 3. Nilai Avg Loss dan mAP Hasil Pelatihan



Gambar 4. Hasil Pengenalan Objek *Pizza* Pada Video

Pengujian model YOLO dalam pengenalan objek *hamburger*, *hotdog* dan *pizza* pada video dilakukan dengan menghitung akurasi *frame per second*. Video secara keseluruhan memiliki durasi 11 menit 7 detik. Hasil pengujian *frame per second* pada video dengan objek *hamburger* dapat dilihat pada Tabel 1.

Berdasarkan Tabel 1, objek *hamburger* dapat dikenali dengan akurasi antara 80% sampai 91%. Hasil pengujian untuk objek *hotdog* pada video dapat dilihat pada Tabel 2.

Objek *hotdog* dapat dikenali dengan akurasi antara 81% sampai 96% seperti yang dapat dilihat pada Tabel 2. Hasil pengujian untuk objek *pizza* pada video dapat dilihat pada Tabel 3.

Berdasarkan Tabel 3, objek *pizza* dapat dikenali dengan akurasi antara 81% sampai 100%. Hasil pengujian menggunakan sebuah video yang terdapat tiga objek yaitu *hamburger*, *hotdog*, dan *pizza* menunjukkan bahwa model YOLO dapat mengenali ketiga objek dengan baik.

Tabel 1. Hasil Pengujian Pengenalan Objek *Hamburger* Pada Video

Waktu (menit:detik)	Akurasi
00:00 – 00:03	83%
00:03 – 00:06	89%
00:07 – 00:08	83%
00:09 – 00:11	88%
00:12 – 00:13	82%
00:14 – 00:16	88%
00:17 – 00:18	82%
00:19 – 00:21	84%
00:22 – 00:23	81%
00:24 – 01:01	83%
01:02 – 01:03	80%
01:04 – 01:05	91%

Tabel 2. Hasil Pengujian Pengenalan Objek *Hotdog* Pada Video

Waktu (menit:detik)	Akurasi	Waktu (menit:detik)	Akurasi
01:06 – 01:08	85%	03:14 – 03:16	93%
01:09 – 01:13	83%	03:17 – 03:18	94%
01:14 – 01:16	81%	03:19 – 03:21	90%
01:17 – 01:18	86%	03:22 – 03:23	92%
01:19 – 01:21	83%	03:24 – 04:01	90%
01:22 – 01:23	80%	04:02 – 04:06	87%
01:24 – 02:01	87%	04:07 – 04:08	91%
02:02 – 02:03	88%	04:09 – 04:11	94%
02:02 – 02:06	87%	04:12 – 04:13	96%
02:07 – 02:08	85%	04:14 – 04:16	95%
02:09 – 02:11	88%	04:17 – 04:18	94%
02:12 – 02:13	87%	04:19 – 04:21	96%
02:14 – 02:16	86%	04:22 – 05:03	95%
02:17 – 02:18	89%	05:04 – 05:08	94%
02:19 – 02:21	87%	05:09 – 05:11	93%
02:22 – 02:23	93%	05:12 – 05:16	94%
02:24 – 03:01	92%	05:17 – 05:18	81%
03:02 – 03:03	94%	05:19 – 05:21	84%
03:04 – 03:06	91%	05:22 – 05:23	91%
03:07 – 03:08	90%	05:24 – 06:00	87%
03:09 – 03:13	92%		

Tabel 3. Hasil Pengujian Pengenalan Objek Hamburger Pada Video

Waktu (menit:detik)	Akurasi	Waktu (menit:detik)	Akurasi
06:01 – 06:02	90%	09:10 – 09:12	96%
06:02 – 06:09	97%	09:13 – 09:14	99%
06:10 – 06:12	98%	09:15 – 09:17	100%
06:13 – 06:14	96%	09:18 – 09:19	99%
06:15 – 06:17	97%	09:20 – 09:22	97%
06:18 – 07:02	98%	09:23 – 10:02	91%
07:03 – 07:12	99%	10:03 – 10:04	92%
07:13 – 07:17	100%	10:05 – 10:07	83%
07:18 – 07:24	99%	10:08 – 10:09	86%
08:00 – 08:12	100%	10:10 – 10:12	82%
08:13 – 08:19	99%	10:13 – 10:14	88%
08:20 – 08:22	89%	10:15 – 10:22	95%
08:23 – 09:04	99%	10:23 – 10:24	85%
09:05 – 09:07	100%	11:00 – 11:02	87%
09:08 – 09:09	84%	11:03 – 11:07	81%

Hasil Pengujian Secara *Real Time*

Menggunakan *Webcam*

Pengujian model YOLO mengenali objek yaitu *hamburger*, *pizza* dan *hotdog*

menggunakan *webcam* dapat dilihat pada Gambar 5. Akurasi pengenalan objek *hamburger* sebesar 87%, 79% untuk objek *hotdog* dan 63% untuk objek *pizza*.



Gambar 5. Hasil Pengenalan Objek *Hamburger, Pizza, dan Hotdog* Menggunakan *Webcam*

KESIMPULAN DAN SARAN

Implementasi algoritma *You Look Only Once* (YOLO) dalam pengenalan objek makanan cepat saji memiliki akurasi validasi mAP sebesar 100% dan *avg loss* sebesar 4.6%. Oleh karena itu dapat disimpulkan bahwa implementasi *deep learning* menggunakan YOLO mampu melakukan pengenalan objek pada citra makanan cepat saji dengan baik. Hasil dari pengujian model dalam pengenalan objek tiga jenis makanan cepat saji menggunakan video dan *real time webcam* memiliki akurasi 63% sampai 100%. Pengenalan objek memberikan hasil akurasi yang berbeda-beda pada saat objek yang dikenali berpindah posisi.

Pada penelitian lebih lanjut dapat menambahkan jumlah citra dalam tahap pelatihan sehingga saat objek berpindah posisi, pengenalan objek bisa lebih cepat dan akurat. Selain itu, spesifikasi komputer diharapkan memiliki kapasitas RAM dan kecepatan GPU yang tinggi sehingga mampu mengurangi waktu dalam pelatihan model.

DAFTAR PUSTAKA

- [1] N. S. Kasiati dan N. W. D. Rosmalawati, *Kebutuhan dasar manusia I*. Jakarta: Kementerian Kesehatan Republik Indonesia, 2016.
- [2] Badan Pengawas Obat dan Makanan Republik Indonesia, *Peraturan Kepala Badan Pengawas Obat dan Makanan Republik Indonesia Nomor 1 Tahun 2017 Tentang Pengawasan Pangan Olahan Organik*. Jakarta: Badan Pengawas Obat dan Makanan Republik Indonesia, 2017.
- [3] Biro Pusat Statistik, *Statistik Indonesia, Statistical Yearbook of Indonesia*. Jakarta: Biro Pusat Statistik, 2018.
- [4] Sudyarto, "Daya saing ketahanan pangan melalui identifikasi sikap kepercayaan konsumen remaja terhadap produk makanan cepat saji (*fast food*) dan makanan lokal (tradisional)", *Jurnal Riset Ekonomi dan Bisnis*, vol. 9, no. 2, hal. 91 – 97, 2009.

- [5] F. Hayati, “Faktor-faktor yang berhubungan dengan konsumsi *fast food* waralaba modern dan tradisional pada remaja siswa SMU Negeri di Jakarta Selatan”, *Skripsi*, Institut Pertanian Bogor, Bogor, 2000.
- [6] J. Young-seon, *Why? Food and nutrition*. Jakarta: PT. Elex Media Komputindo, 2010.
- [7] R. Munir, *Pengolahan citra digital*. Bandung: Informatika, 2007.
- [8] R. Girshick, J. Donahue, T. Darrell, dan J. Malik, “Rich feature hierarchies for accurate object detection and semantic segmentation”, dalam *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2014, hal. 580 – 587.
- [9] K. He, X. Zhang, S. Ren, dan J. Sun, “Spatial pyramid pooling in deep convolutional networks for visual recognition”, *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 37, no. 9, hal. 1904 – 1916, 2015.
- [10] R. Girshick, “Fast R-CNN”, dalam *Proceedings of the IEEE International Conference on Computer Vision*, 2015, hal. 1440 – 1448.
- [11] S. Ren, K. He, R. Girshick, dan J. Sun, “Faster R-CNN: Towards real-time object detection with region proposal networks”, *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 39, no. 6, hal. 1137 – 1149, 2017.
- [12] J. Redmon, S. Divvala, R. Girshick, dan A. Farhadi, “You only look once: Unified, real-time object detection”, dalam *Proceedings of the 2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2016, hal. 779 – 788.
- [13] J. Redmon dan A. Farhadi, “YOLO9000: Better, faster, stronger”, dalam *Proceedings of the 2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2016, hal. 6517 – 6525.
- [14] R. Huang, J. Pedoeem, dan C. Chen, “YOLO-LITE: A real-time object detection algorithm optimized for Non-GPU computers”, dalam *Proceedings of the IEEE International Conference on Big Data*, 2018, hal. 2503 – 2510.
- [15] J. Zhang, M. Huang, X. Jin, dan A. Li, “A real-time Chinese traffic sign detection algorithm based on modified YOLOv2”, *Algorithms*, vol. 10, no. 4, 127, 2017.
- [16] Z. Liu, Z. Chen, Z. Li, dan W. Hu, “An efficient pedestrian detection method based on YOLOv2”, *Mathematical Problems in Engineering*, vol. 2018, 2018.
- [17] S. Jupiyandi, F. R. Saniputra, Y. Pratama, M. R. Dharmawan, dan I. Cholissodin, “Pengembangan deteksi citra mobil untuk mengetahui jumlah

tempat parkir menggunakan CUDA dan modified YOLO”, *Jurnal Teknologi Informasi dan Ilmu Komputer (JTIIK)*, vol. 6, no. 4, hal. 413 – 419, 2019.

[18] Y. Matsuda, H. Hoashi, dan K. Yanai, “Recognition of multiple-food images by detecting candidate regions”, dalam *Proceedings of the IEEE International Conference on Multimedia and Expo (ICME)*, 2012, hal. 25 – 30.

DETEKSI SIMILARITAS DOKUMEN ILMIAH MENGGUNAKAN ALGORITMA RABIN-KARP

¹Hermas Yuda Pamungkas , ²Fitrianingsih
^{1,2}Fakultas Teknologi Industri Universitas Gunadarma
Jl. Margonda Raya No. 100, Depok 16424, Jawa Barat
²fitrianingsih@staff.gunadarma.ac.id,

Abstrak

Karya ilmiah adalah salah satu hak kekayaan atas intelektual seseorang. Dalam menulis karya ilmiah tentu memiliki tantangan tertentu agar tulisan tersebut merupakan tulisan yang orisinal dan tidak menjiplak secara langsung terhadap karya tulisan orang lain. Di jaman yang serba modern ini sumber daya tulisan mudah didapatkan dimana saja melalui internet. Kemudahan tersebut membuat seseorang dapat melakukan plagiarisme dari sumber tulisan lain dengan melakukan copy-paste pada penulisannya tanpa mengubah struktur kalimat dan mencantumkan sumber tulisan tersebut. Pada penelitian ini akan mengimplementasikan algoritma Rabin – Karp dalam melakukan pendeteksian similaritas suatu dokumen artikel ilmiah dengan dokumen lainnya. Tahapan dari penelitian ini terdiri dari pemisahan baris kalimat, querying google search, preprocessing, yang terdiri dari proses tokenizing, filtering, dan stemming, k-gram, hasing, perhitungan kemiripan kalimat, dan perhitungan kemiripan dokumen input. Dokumen yang dibandingkan adalah kalimat per kalimat pada dokumen input dengan hasil pencarian kalimat tersebut pada google search. Perbandingan tersebut dengan cara membandingkan total hash dari kalimat query dengan hash hasil pencarian kalimat tersebut sehingga dihasilkan similaritas kalimat tersebut yang nantinya diakumulasikan menjadi similaritas dokumen input tersebut. Hasil penelitian ini diharapkan dapat mendeteksi plagiarisme pada sebuah dokumen.

Kata Kunci: *Dokumen, pencocokan, string, similaritas, query*

Abstract

Twitter as one of the social media that the company uses to account for consumer complaints. Submitting Scientific work is one of the intellectual property rights of a person. In writing scientific papers certainly have certain challenges so that the writing is original writing and does not copy directly to the writings of others. In this modern era, writing resources are easily available anywhere on the internet. This convenience allows one to plagiarism from other sources of writing by copy-pasting the writing without changing the structure of the sentence and including the source of the writing. This research will implement the Rabin - Karp algorithm in detecting similarity of a scientific article document with other documents. The stages of this research consist of the separation of sentence lines, querying google search, preprocessing, which consists of the process of tokenizing, filtering, and stemming, k-gram, hasing, calculation of sentence similarity, and calculation of similarity of input documents. The document being compared is the sentence per sentence in the input document with the results of the sentence search on google search. The comparison is by comparing the total hash of the query sentence with the hash of the result of the sentence search so that the similarity of the sentence is generated which will later be accumulated into the similarity of the input document. The results of this study are expected to detect plagiarism in a document.

Keywords : *Document, matching, string, similarity, query*

PENDAHULUAN

Penulisan suatu karya ilmiah membutuhkan data atau acuan yang sesuai dengan topik yang dibahas pada penulisan tersebut. Seiring dengan berkembangnya teknologi internet tentu mempermudah seseorang dalam mencari jurnal-jurnal dan tulisan-tulisan yang berkaitan dengan materi yang akan dicari. Adanya mesin pencarian seperti *Google* mempermudah kita dalam mencari tiap hal yang kita ketikkan pada mesin pencari tersebut[1]. Data yang didapatkan menggunakan mesin pencari tersebut dapat berupa sebuah artikel, jurnal, karya ilmiah, dan lain lain[2]. Data digital tersebut dengan mudah diduplikasi kedalam penulisan suatu karya ilmiah. Kemudahan akses terhadap data yang ada di internet tersebut memperbesar kemungkinan terjadinya plagiarisme [3] terhadap tiap tulisan yang tersebar di internet. Penjiplakan atau plagiarisme berarti mencontoh atau meniru atau mencuri tulisan dan karya orang lain [4] yang kemudian diakui sebagai karangannya sendiri dengan ataupun tanpa seizin penulisnya[5].

Beberapa penelitian terkait algoritma dalam menentukan tingkat kemiripan suatu dokumen telah banyak dilakukan. Penelitian [6] Menggunakan Metode *String Matching* dengan menggunakan algoritma *Rabin – Karp*. Pada penelitian tersebut dilakukan pembuatan aplikasi pendeteksi similaritas antara 2 dokumen menggunakan algoritma *Rabin – Karp*. Pada penelitian [7] peneliti

melakukan percobaan dimana dapat disimpulkan bahwa jumlah dari K-gram menentukan similaritas dari dokumen yang sedang diuji. Pada penelitian [8] , peneliti mengimplementasikan tahapan *preprocessing* pada metode *Levensthein Distance* untuk mempengaruhi nilai ke-miripan dokumen dan waktu proses perhitungan tersebut. Pada penelitian [9] dilakukan pengimplementasian algoritma *Rabin–Karp* untuk mendeteksi tingkat ke-miripan dari source code pada pemrograman lisp menggunakan perhitungan *Dice Similarity Coeficient*. Pada jurnal [10] menimplementasikan algoritma *stemming* Nazief– Adriani pada pendeteksian kemiripan dokumen berbahasa indonesia yang didapatkan kesimpulan bahwa *stemming* mempengaruhi hasil kemiripan pendeteksian similaritas dokumen tersebut.

Pada penelitian ini peneliti akan mengimplementasikan penggunaan algoritma *Rabin-Karp* melalui metode pencocokan string (*String Matching*) untuk menentukan indeks similaritas dari sebuah dokumen berbentuk artikel ilmiah berbahasa Indonesia pada hasil pencarian *Google*. Pengambilan data dari hasil pencarian mesin pencari *Google* menggunakan teknik *crawling*. Proses *stemming* menggunakan algoritma Nazief – Adriani untuk dokumen berbahasa Indonesia dan proses *k-gram* dan *hashing* pada sebuah dokumen input dan hasil pencarian kalimat tersebut. Implementasi algoritma ini diharapkan dapat mendeteksi plagiarisme pada sebuah dokumen.

METODE PENELITIAN

Tahapan awal dimulai dari pembacaan dokumen sumber yang selanjutnya akan dipecah menjadi perkalimat pada tahap pemisahan kalimat, lalu dari kalimat tersebut akan dijadikan *query* untuk pencarian *google*. Dari data tersebut akan dilanjutkan proses *preprocessing* yaitu proses *Tokenizing*, *Filtering*, dan *Stemming*. Dari hasil *preprocessing* akan dilanjutkan dengan proses *k-gram* yang selanjutnya akan dilakukan proses *hasing*. Kemudian dari hasil *hasing* tersebut akan dihitung kemiripan kalimat dengan hasil pencarian *google* dan dirata-ratakan sehingga menghasilkan hasil kemiripan keseluruhan dokumen.

Dokumen Input

Dokumen input yang digunakan adalah dokumen yang akan dilakukan analisa kemiripan dokumen. Segala konten yang diinputkan ke dalam sistem akan di analisa perkalimat dari konten tersebut. Kalimat perkalimat akan dipisahkan dari dokumen tersebut pada langkah selanjutnya. Setelah itu, perkalimat tersebut akan dilakukan tahapan selanjutnya yaitu *querying google search*.

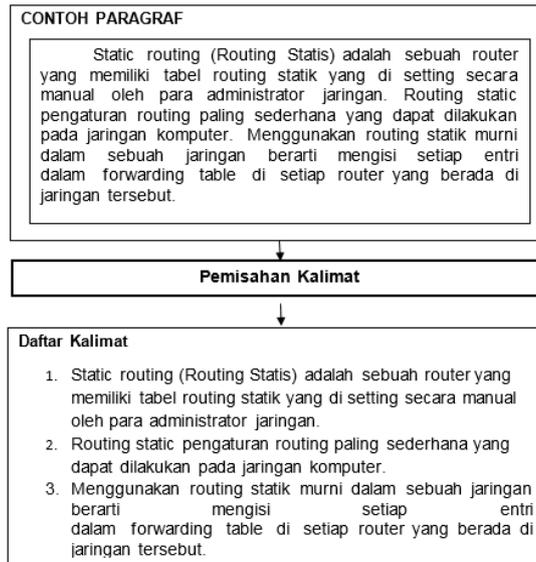
Pemisahan Kalimat

Pada tahap ini akan dilakukan pemisahan kalimat perkalimat dari sebuah

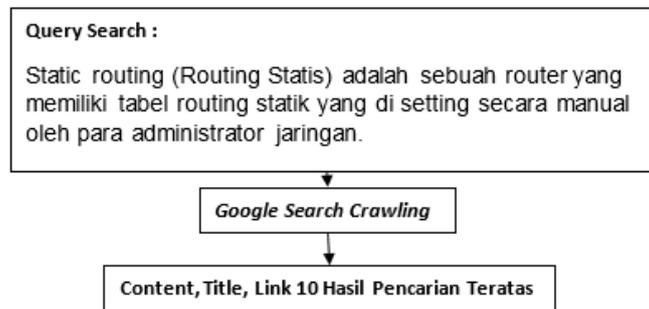
dokumen input. Pemisahan ini dilakukan dari awal kalimat hingga akhir kalimat yang ditandai oleh titik. Teknik pemisahan ini menggunakan *pattern regex* pada java sebagai indikator dari kalimat tersebut. Dari sebuah teks panjang dipotong menggunakan *regex* sehingga dihasilkan kumpulan kalimat dari *string* tersebut yang nantinya digunakan dalam *crawling* untuk mendapatkan hasil yang similar dengan tiap kalimat pada kumpulan kalimat tersebut. Pada method diatas kalimat akan dipisahkan berdasarkan titik, koma, tanda seru, tanda tanya, dan titik dua. Hasil dari *method* tersebut berupa *list* yang berisi kumpulan kalimat dari sebuah dokumen input tersebut. Gambar 1 merupakan tahapan pemisahan kalimat.

Querying Google Search

Dari hasil kalimat yang sebelumnya sudah dipisahkan dari tiap paragraf pada dokumen input, dilakukan kueri pencarian pada mesin pencari *google*. Dari hasil pencarian tersebut dilakukan teknik *scrapping* pada halaman html *google* tersebut sehingga hanya didapatkan 10 pencarian teratas terhadap kalimat tersebut. Daftar pencarian tersebut dimasukkan kedalam sebuah *list object* pada java yang terdiri atas judul tulisan, link/url sumber tulisan, dan konten tulisan yang terdapat pada hasil pencarian tersebut seperti dapat dilihat pada ilustrasi Gambar 2.



Gambar 1. Ilustrasi Pemenggalan Kalimat pada Paragraf



Gambar 2. Ilustrasi Pencarian Kalimat *Query*

Pada sebuah pencarian pada *google* data ditampilkan dalam format *html* dengan tag dan *class* tertentu. Untuk mendapatkan data terlebih dahulu harus dilakukan pengambilan konten dari hasil pencarian tersebut menggunakan teknik *crawling*.

Pada Gambar 3 ditampilkan salah satu data hasil pencarian pada *google* dengan

format *html*. Dari hasil pencarian tersebut memungkinkan untuk mendapatkan title, content, dan link sumber data tersebut. Dengan menggunakan library JSoup seluruh file *html* pada halaman *google* hasil pencarian kalimat input dapat diekstrak menjadi komponen tersebut.



Gambar 3. Contoh Hasil Pencarian pada Google

Preprocessing

Preprocessing adalah tahap dimana sebuah kalimat di lakukan pemilihan kata atau *term* yang digunakan dalam proses perhitungan kemiripan sehingga hanya tersisa kata-kata yang memiliki makna dalam kalimat tersebut. Kalimat dari dokumen input dan hasil pencarian kalimat tersebut akan dilakukan proses *preprocessing* sebelum dilanjutkan pada proses perhitungan kemiripan. Proses ini terdiri atas 3 proses yaitu *Tokennize, Filtering,* dan *Stemming.*

1) **Tokenizing.** Melakukan pemecahan kalimat menjadi per-kata yang di tampung kedalam bentuk List pada java. Dalam melakukan proses ini dibantu menggunakan library OpenNLP yang merupakan sebuah library pada java yang digunakan dalam pemrosesan natural language processing berbasis teks. OpenNLP memiliki salah satu fitur

dengan nama tokenization yang menyediakan 3 klasifikasi yaitu Simple Tokenizer, Whitespace Tokenizer, dan Tokenizer ME. Pada proses ini klasifikasi yang digunakan adalah Simpletokenizer yang hanya memisahkan kata per kata berdasarkan spasi dan kelas karakter. Pada tahap ini juga dilakukan pengilangan karakter khusus sehingga yang akan dihasilkan hanyalah alfabet dan angka. Kalimat dari dokumen beserta 10 hasil pencarian kalimat tersebut akan dilakukan tokenisasi untuk dilanjutkan pada tahap filtering.

2) **Filtering.** Setelah dilakukan proses tokenizing, list perkata dari kalimat tersebut akan disaring pada proses ini untuk menghilangkan kata-kata ataupun simbol yang dianggap tidak penting pada kalimat tersebut. Kata-kata yang dihi-

langkah pada proses ini adalah kata – kata *stopwords* atau kumpulan kata– kata yang tidak memiliki makna atau dianggap kurang penting dalam sebuah kalimat sehingga hanya tersisa kata– kata yang dianggap memiliki makna pada kalimat tersebut pada hasil proses *filtering*.

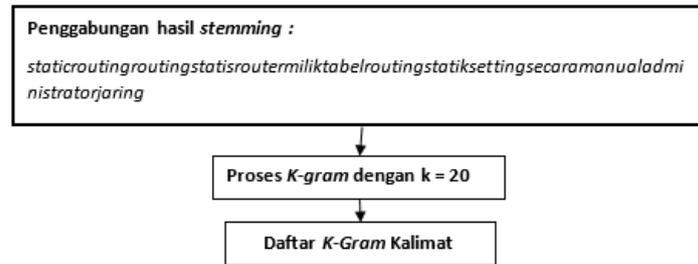
Langkah-langkah dari proses *filtering* pada penelitian ini adalah sebagai berikut:

1. Filterisasi kata akan dilakukan sebanyak jumlah kata pada hasil tokenisasi.
 2. Fungsi filetrisasi akan melakukan pengecekan pada daftar *stopwords* bahasa Indonesia, apabila terdapat kata pada daftar tersebut maka kata tersebut akan dihapus pada daftar hasil filterisasi. Jika tidak maka perulangan akan berlanjut pada kata selanjutnya
 3. Perulangan akan terhenti apabila sudah tidak ada kata lagi yang akan diperiksa.
- 3) **Stemming.** Proses mengubah kata menjadi kata dasar dari kata tersebut. proses ini akan menghilangkan semua imbuhan pada sebuah kata sehingga dihasilkan hanya kata dasarnya saja. Sehingga hasil dari kalimat yang di

lakukan proses perhitungan kemiripan lebih optimal. Pada tahap ini hasil dari tahap *filtering* akan diubah menjadi kata dasarnya sehingga yang diproses pada perhitungan kemiripan kalimat hanyalah daftar kata dasar yang sudah dihilangkan imbuhan dari awal hingga akhir. Algoritma *stemming* yang digunakan pada penelitian ini adalah algoritma *stemming* Nazief–Adriani untuk dokumen berbahasa Indonesia. Algoritma Nazief – Adriani didasarkan pada aturan morfologi Bahasa Indonesia yang dikumpulkan menjadi satu grup dan di enkapsulasi pada imbuhan/*affixes* yang diperbolehkan (*allowed affixes*) dan imbuhan/*affixes* yang tidak diperbolehkan (*disallowed affixes*).

K-Gram

Pada tahap ini hasil dari daftar kata pada proses *stemming* akan dibentuk suatu string berurutan yang nantinya akan dipotong berdasarkan jumlah k-gram yang nanti akan dilanjutkan pada proses *hashing*. Metode *k-gram* sendiri adalah proses pemotongan sejumlah karakter pada sebuah string dengan jumlah tertentu. Berikut adalah ilustrasi dari proses *k-gram* dari sebuah kalimat yang akan diproses pada tahap sebelumnya. Gambar 4 merupakan ilustrasi dari *k-gram*.



Gambar 4. Ilustrasi Proses Stemming

Tabel 1. Contoh Hasil Proses Hashing

No.	Hasil K-Gram	Hasil Hash
1.	staticroutingrouting	2731643090344028
2.	taticroutingroutings	2690960813048380
3.	aticroutingroutingst	2392181994929516
4.	ticroutingroutingsta	2710269105507052
5.	icroutingroutingstat	2488723457222876
6.	croutingroutingstati	2430036963848860

Hashing

Hashing yang digunakan pada penelitian ini adalah *rolling hash* dimana *list* k-gram pada tahap sebelumnya akan di hash sebelum diperiksa kemiripan karakternya dengan hasil pencarian kalimat tersebut. Dari hasil *k-gram* sebelumnya akan dihasilkan hasil *hash* yang unik sehingga dalam perbandingan kemiripan kalimat lebih akurat. Tabel 1 merupakan contoh hasil dari proses *hashing* dari sebuah daftar *string* yang dihasilkan pada tahap *k-gram* sebelumnya.

Perhitungan Kemiripan Kalimat

Sebelum dilakukan perhitungan terlebih dahulu dilakukan pengecekan *hash* antara *hash list* kalimat uji dengan 10 *content* hasil pencarian kalimat tersebut pada *google search*. Pengecekan kemiripan tiap object

hash disini menggunakan kelas java *stream()* dengan menggunakan fungsi *contains*. Penggunaan fungsi tersebut mengurangi beban dalam perhitungan kemiripan dengan kompleksitas $O(n)$ dibandingkan *nested loop* dengan kompleksitas $O(n*n)$. Dari hasil pencarian tersebut akan didapatkan daftar *hash* yang sama dari *list hash* kalimat *query* dengan *hash* hasil pencarian kalimat tersebut yang selanjutnya akan ditampung pada *list match* untuk didapatkan banyaknya *list* yang sama tersebut dengan method *size()*.

```

List<Words> match = searchData
    .stream()
    .filter(queries::contains)
    .collect(Collectors.toList());
  
```

Dari jumlah *hash* yang sama tersebut dapat dihitung persentase kemiripan antara

kalimat *query* dengan hasil pencarian kalimat tersebut pada mesin pencari *google*. Perhitungan yang dilakukan adalah dengan membagi keseluruhan total *hash* yang sama dengan total *hash* pada kalimat dikalikan dengan 100 untuk menghasilkan persentase kemiripan kalimat menggunakan persamaan 1.

$$= \frac{\text{Similaritas Kalimat} \times \sum \text{Total Hash Sama}}{\sum \text{Total Hash Query}} \times 100\% \quad (1)$$

Perhitungan Kemiripan Keseluruhan Dokumen

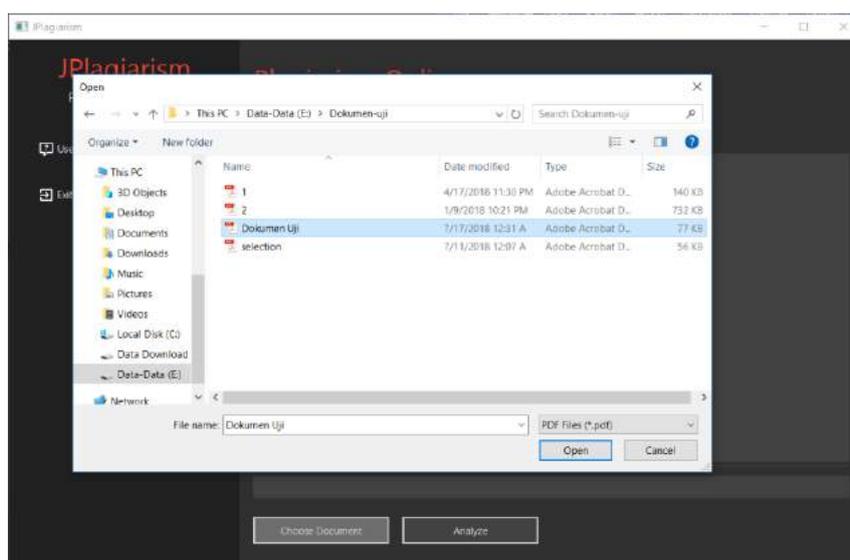
Perhitungan kemiripan keseluruhan dokumen dihitung dari total persentase dari semua kalimat yang telah dilakukan pengecekan kemiripan dibagi total kalimat dari dokumen tersebut. Penentuan apakah dokumen tersebut termasuk plagiarisme berat atau tidak dirujuk pada persentase kemiripan (Tabel 2.3. iThenticate, 2018)

Ttotal similaritas dokumen maka dilakukan penjumlahan total persentase similaritas tiap kalimat pada dokumen tersebut yang selanjutnya akan dibagi dengan jumlah kalimat yang terdapat pada dokumen tersebut seperti pada persamaan 2.

$$\text{Similaritas Dokumen} = \frac{\sum \text{Total Similaritas Kalimat}}{\sum \text{Total Kalimat Pada Dokumen}} \quad (2)$$

HASIL DAN PEMBAHASAN

Langkah awal dilakukan penginputan dokumen input berupa file pdf untuk menganalisa dokumen. Pada kasus ini dokumen input bernama “Dokumen Uji.pdf” akan dilakukan pengujian plagiarisme. Pilih dokumen lalu tekan open untuk membuka file tersebut seperti dapat dilihat pada Gambar 5.

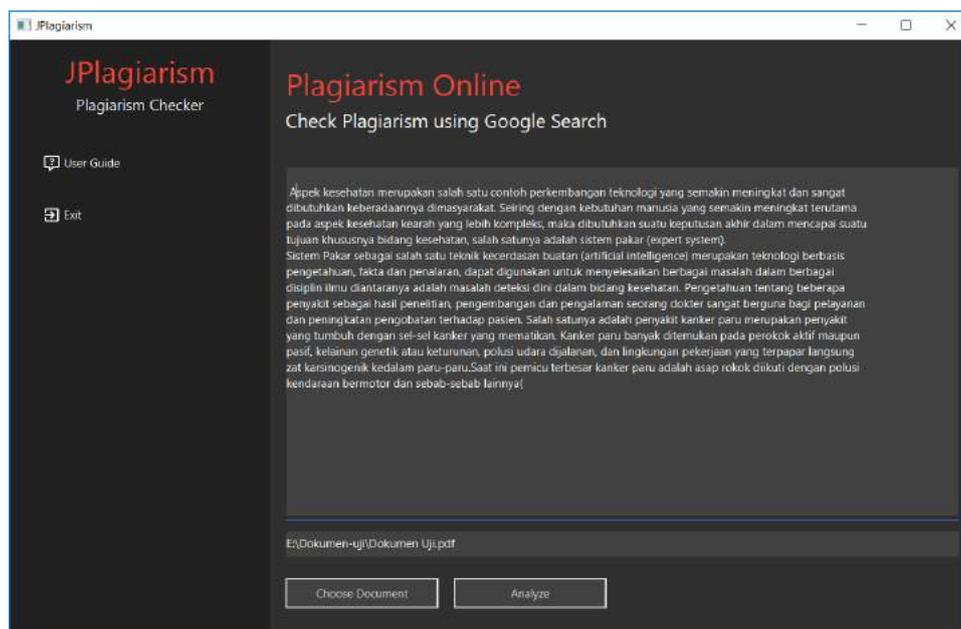


Gambar 5. Pemilihan File Uji

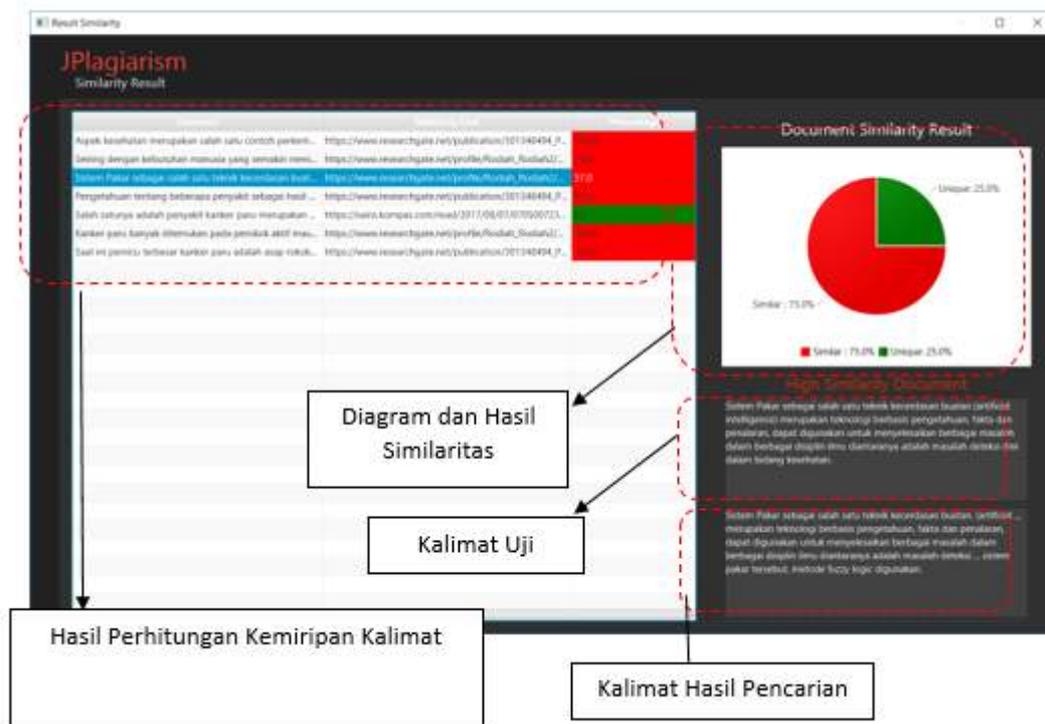
Setelah file terpilih maka akan tampil teks hasil parsing file pdf tersebut menjadi teks dengan library iText. Jika teks sudah terisi maka pencarian similaritas akan dapat dilakukan. Tekan *analyze* untuk melanjutkan pada tahap pencarian similaritas. Pada tahap pencarian similaritas ini waktu yang dibutuhkan dalam pencarian bergantung pada koneksi internet dan banyaknya kalimat pada dokumen input tersebut. Semakin banyak kalimat dalam dokumen tersebut maka akan semakin lama proses perhitungannya. Gambar 6 merupakan tampilan setelah diinputkan file uji.

Setelah proses pencarian kalimat, *preprocessing*, hingga tahap perhitungan selesai maka akan tampil halaman report pada Gambar 7, dimana akan ditampilkan dokumen input yang sudah dipisahkan perkalimat menggunakan *regex* dengan url yang mengan-

dung kalimat tersebut serta nilai similaritas. Seluruh kalimat pada dokumen input dan hasil pencarian similaritasnya akan ditampilkan pada tabel pada halaman tersebut. Untuk dapat melihat *content* dari hasil *crawl* dapat menekan baris pada tabel tersebut lalu akan tampil pada textfield bagian bawah chart dimana textfield pertama menampilkan kalimat dari dokumen input yang digunakan sebagai kalimat *query*, dan *textfield* kedua sebagai *content* dari hasil pencarian dengan indeks similaritas tertinggi. Hasil keseluruhan similaritas kalimat pada dokumen tersebut ditampilkan dalam bentuk *pie chart* dan kesimpulan akhir dari kalkulasi persentase similaritas dokumen tersebut dengan teks dimana kesimpulan tersebut didapatkan berdasarkan penilaian indeks similaritas menurut iThenticate.



Gambar 6. Tampilan Teks Dokumen Uji pada Halaman Utama



Gambar 7. Tampilan Hasil Akhir Perhitungan Kemiripan Dokumen

KESIMPULAN DAN SARAN

Beberapa kesimpulan yang dapat diambil pada penelitian ini antara lain : Pemisahan kalimat pada dokumen input menggunakan *pattern regex* telah berhasil dilakukan dengan baik. Pencarian kalimat pada dokumen input pada mesin pencari *google* dan *crawling* data dari hasil pencarian tersebut telah berhasil dilakukan. Proses *Stemming* untuk mendapatkan kata dasar dari sebuah kata dengan menggunakan algoritma *stemming* untuk dokumen berbahasa Indonesia Nazief-Adriani pada sebuah kalimat input maupun konten hasil pencarian kalimat tersebut telah berhasil dilakukan dengan baik. Penelitian ini berhasil melakukan pembentukan *k-gram* dan *hashing* dari

string output tahap preprocessing telah berhasil dilakukan dengan baik. Proses perhitungan jumlah *hash* yang sama pada kalimat *query* dan 10 konten hasil pencarian pada mesin pencari *google* telah berhasil dilakukan dengan baik. Penelitian ini berhasil membentuk aplikasi pendeteksi similaritas tulisan ilmiah menggunakan sumber hasil pencarian mesin pencari *google*.

Pengembangan lebih lanjut yang dapat dilakukan pada penelitian selanjutnya antara lain dengan mengimplementasikan algoritma yang dapat mempertimbangkan posisi dan urutan kata, sinonim kata dari sebuah kalimat, dan kemiripan semantik dari sebuah dokumen input sehingga menambah nilai akurasi dari similaritas dokumen dengan dokumen pembandingan. Selain itu, dalam teknik *crawling*

dapat dikembangkan dengan melakukan *crawling* dari tiap website yang diketahui memiliki similaritas dengan kalimat input secara utuh, tidak hanya dari hasil pencarian *google search*.

DAFTAR PUSTAKA

- [1] S. S. Rajmohammad, P. P. Arun, dan G. P. Dnyandeo, "Web crawler : Extracting the web data", *International Engineering Research Journal (IERJ)*, vol. 1, no. 8, hal. 629–632, 2015.
- [2] N. Wubbeling, "The South African Journal of Science-iThenticate", Academy of Science of South Africa (ASSAF), 2016.
- [3] R. Janani dan S. Vijayarani, "An efficient text pattern matching algorithm for retrieving information from desktop", *Indian Journal of Science and Technology*, vol. 9, no. 43, 2016.
- [4] C. D. Manning, P. Raghavan, dan H. Schutze, *An introduction to information retrieval*. Cambridge: Cambridge University Press, 2008.
- [5] B. Gipp dan N. Meuschke, "Citation pattern matching algorithms for citation-based plagiarism detection: greedy citation tiling, citation chunking and longest common Citation Sequence", dalam Proceedings of the 11th ACM Symposium on Document Engineering - DocEng '11, 2011, hal. 249–258.
- [6] A. Prastyanti, "Sistem deteksi kemiripan kata pada dua dokumen menggunakan algoritma Rabin-Karp", Skripsi Sarjana, Universitas Diponegoro, Semarang, 2014.
- [7] A. P. U. Siahaan, Mesran, R. Rahim, dan D. Siregar, "K-Gram as a determinant of plagiarism level in Rabin-Karp algorithm", *International Journal of Scientific & Technology Research*, vol. 6, no. 7, 2017.
- [8] N. H. Ariyani, Sutardi, dan R. Ramadhan, "Aplikasi pendeteksi kemiripan isi teks dokumen menggunakan metode Levenshtein Distance", *SemanTIK*, vol. 2, no. 1, hal. 279–286, 2016.
- [9] A. A. Wiguna dan I. Rizqa, "Pemanfaatan algoritma Rabin-Karp untuk mengetahui tingkat kemiripan dari *source code* pada pemrograman Lisp", Skripsi Sarjana, Universitas Dian Nuswantoro, Semarang, 2016.
- [10] A. Firdaus, Ernawati, dan A. Vatesia, "Aplikasi pendeteksi kemiripan pada dokumen teks menggunakan algoritma Nazief & Andriani dan metode *Cosine Similarity*", *Jurnal Teknologi Informasi*, vol. 10, no. 1, hal. 96–109, 2014.

KLASIFIKASI CITRA GENUS PANTHERA MENGGUNAKAN METODE CONVOLUTIONAL NEURAL NETWORK (CNN)

¹Gusti Alfahmi Anwar, ²Desti Rimirasih,
^{1,2}Fakultas Teknologi Industri Universitas Gunadarma
Jl. Margonda Raya No. 100, Depok 16424, Jawa Barat
¹gustialfahmi@gmail.com, ²destimath@staff.gunadarma.ac.id

Abstrak

Panthera merupakan genus dari keluarga kucing yang memiliki empat spesies populer yaitu, harimau, jaguar, macan tutul, singa. Singa memiliki warna keemasan dan tidak memiliki motif, harimau memiliki motif loreng dengan garis-garis panjang, jaguar memiliki tubuh yang lebih besar dari pada macan tutul serta memiliki motif tutul yang lebih lebar, sedangkan macan tutul memiliki tubuh yang sedikit lebih ramping dari pada jaguar dan memiliki tutul yang tidak terlalu lebar. Pada penelitian ini dilakukan klasifikasi genus panther yaitu harimau, jaguar, macan tutul, dan singa menggunakan metode Convolutional Neural Network. Model Convolutional Neural Network yang digunakan memiliki 1 input layer, 5 convolution layer, dan 2 fully connected layer. Dataset yang digunakan berupa citra harimau, jaguar, macan tutul, dan singa. Data training terdiri dari 3840 citra, data validasi sebanyak 960 citra, dan data testing sebanyak 800 citra. Hasil akurasi dari pelatihan model untuk training yaitu 92,31% dan validasi yaitu 81,88%, pengujian model menggunakan dataset testing mendapatkan hasil 68%. Hasil akurasi prediksi didapatkan dari nilai F1-Score pada pengujian didapatkan sebesar 78% untuk harimau, 70% untuk jaguar, 37% untuk macan tutul, 74% untuk singa. Macan tutul mendapatkan akurasi terendah dibandingkan 3 hewan lainnya tetapi lebih baik dibandingkan hasil penelitian sebelumnya.

Kata Kunci: Citra, Convolutional Neural Network, Genus Panthera, klasifikasi

Abstract

Panthera is a genus of the cat family that has four popular species namely, tiger, jaguar, leopard, lion. Previous research has been conducted using the Naïve Bayes method. Accuracy results obtained by 100% for tigers, jaguars, lions, and 0% for leopards (Suryanto 2014). The research still uses little data, and cannot yet predict leopard. In this study classifying tiger, jaguar, leopard and lion using the Convolutional Neural Network method. This study aims to determine the accuracy of the results obtained from the results of training and testing. The Convolutional Neural Network model used has 1 input layer, 5 convolution layers, and 2 fully connected layers. Using the dataset for training as many as 3840, as many as 960 validations, testing as many as 800. The accuracy of the model training for training is 92.31% and validation is 81.88%, testing the model using dataset testing has 68% results. Prediction accuracy results obtained from the F1-Score in the test obtained by 78% for tigers, 70% for jaguars, 37% for leopards, 74% for lions. Leopards get the lowest accuracy compared to 3 other animals.

Keywords: Classification, Convolutional Neural Network, Genus Panthera, image

PENDAHULUAN

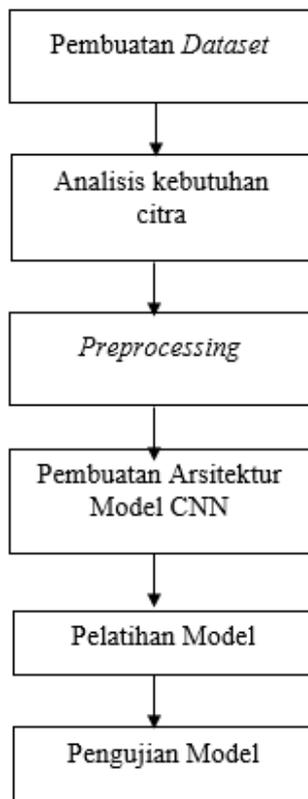
Panthera adalah genus dari keluarga felidae (kucing) yang memiliki empat spesies populer, yaitu harimau, singa, jaguar dan macan tutul. Genus ini terdiri dari sekitar setengah dari subfamili pantherinae atau kucing besar yang memiliki struktur tubuh dan motif kulit yang berbeda. Singa memiliki warna keemasan dan tidak memiliki motif, harimau memiliki motif loreng dengan garis-garis panjang, jaguar memiliki tubuh yang lebih besar dari pada macan tutul serta memiliki motif tutul yang lebih lebar, sedangkan macan tutul memiliki tubuh yang sedikit lebih ramping dari pada jaguar dan memiliki tutul yang tidak terlalu lebar.

Perkembangan *machine learning* pada era sekarang ini memungkinkan klasifikasi otomatis yang dilakukan oleh komputer. Penelitian sebelumnya pengklasifikasian genus panthera menggunakan *machine learning* dengan metode *naive bayes* untuk pengklasifikasian genus panthera yang dilakukan dengan melihat warna kulit dari masing-masing spesies. Hasil penelitiannya menunjukkan bahwa tingkat akurasi pada harimau 100%, jaguar 100%, macan tutul 0%, dan singa 100% [1]. Pada penelitian tersebut digunakan dataset dengan jumlah yang sedikit dan tidak dapat mengenali macan tutul. *Deep learning* adalah sebuah bidang keilmuan baru

dalam bidang *machine learning* sebagai solusi dari klasifikasi citra. Bidang ini memiliki kemampuan yang sangat baik dalam *computer vision*. Beberapa metode yang ada yaitu , MLP, CNN, dan RNN [2]. Salah satu metode yang dapat digunakan untuk klasifikasi citra yaitu *Convolutional Neural Network* (CNN). Hal tersebut dikarenakan CNN berusaha meniru sistem pengenalan citra pada *visual cortex* manusia sehingga memiliki kemampuan mengolah informasi citra. *Convolutional Neural Network* (CNN) mengklasifikasi citra dalam dua tahap yaitu tahap pelatihan dan pengujian. Penelitian sebelumnya mengenai klasifikasi ungags dilakuakn dengan metode CNN dengan dataset yang sedikit dan hasil akurasi yang masih kecil [3]. Pada tahap pelatihan mempelajari ekstrasi fitur dari setiap data agar dapat membedakan label dengan label yang lain. Pada tahap pengujian data yang sudah diuji akan dianalisis dari hasil tahap pelatihan. Tujuan penelitian ini adalah klasifikasi genus panthera menggunakan metode *Convolutional Neural Network*.

METODE PENELITIAN

Metode penelitian yang digunakan pada penelitian ini terdiri atas beberapa tahap proses. Tahapan yang diperlukan dapat dilihat pada Gambar 1.

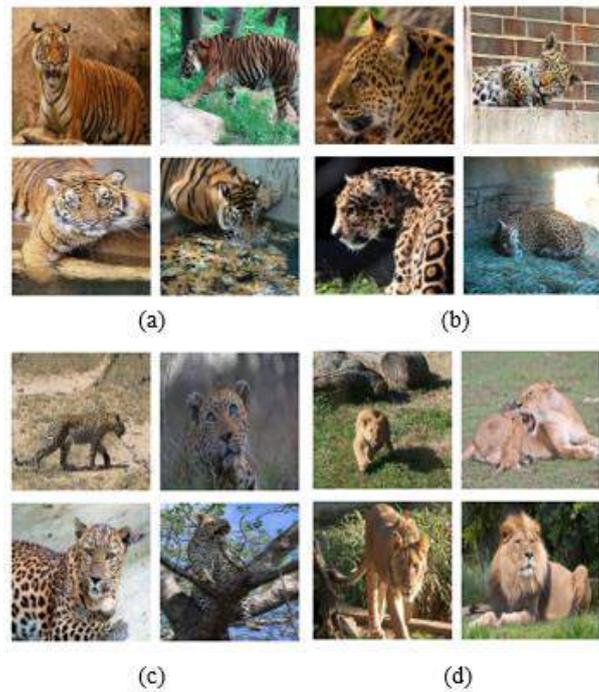


Gambar 1. Tahapan Penelitian

Tahap pertama pada penelitian ini dilakukan pengumpulan *dataset* untuk *training* dan *testing*. Pada tahap ini dilakukan pengunduhan *dataset* melalui situs *image-net.org* dan citra akan dihapus jika yang tidak terbaca, tidak sesuai dengan labelnya, duplikasi, serta objek kurang sesuai. Selanjutnya *dataset* akan dibagi secara acak untuk *training* dan *testing* dan di unggah ke *google drive*.

Citra yang di unduh yaitu harimau, jaguar, macan tutul, dan singa. Jumlah citra tersebut sudah disamakan seluruhnya sebanyak 5600. *Dataset* tersebut dibagi menjadi

2 bagian yaitu *dataset training* dan *dataset testing*. *Dataset training* memiliki citra sebanyak 4800 dan *dataset testing* sebanyak 800 citra. . Citra yang di unduh yaitu harimau, jaguar, macan tutul, dan singa. Jumlah citra tersebut sudah disamakan seluruhnya sebanyak 5600. *Dataset* tersebut dibagi menjadi 2 bagian yaitu *dataset training* dan *dataset testing*. *Dataset training* memiliki citra sebanyak 4800 dan *dataset testing* sebanyak 800 citra. Beberapa contoh citra genus *panthera* yang unduh dapat dilihat pada Gambar 2.



Gambar 2. Citra Genus Panthera
Keterangan : (a) Harimau, (b) Jaguar, (c) Macan Tutul, (d) Singa

Pelatihan klasifikasi citra genus *panthera* menggunakan pendekatan *supervised learning*. Citra yang digunakan seperti yang dapat dilihat pada Gambar 2 sudah dikategorikan berdasarkan nama label masing-masing yaitu harimau, jaguar, macan tutul, singa. Model CNN tidak menginterpretasikan *input* maupun *output* berdasarkan sebuah kata melainkan, dalam bentuk integer sehingga kata dari harimau, jaguar, macan tutul, dan singa direpresentasikan sebagai integer yaitu 0 untuk harimau, 1 untuk jaguar, 2 untuk macan tutul, 3 untuk singa.

Tahap kedua adalah *preprocessing*. Pada tahap ini dilakukan proses *resize* citra agar semua citra memiliki ukuran yang sama. Ukuran *dataset* citra setelah *resize* menjadi 128x128. Ukuran ini sudah cukup untuk

pelatihan karena ukuran ini tidak terlalu besar dan tidak terlalu kecil sehingga cukup untuk mempertahankan *feature* pada citra. Selanjutnya *dataset* disimpan kedalam file berekstensi *.pickle*. Setelah itu, file dimuat dan akan dilakukan normalisasi. Pada tahap normalisasi nilai *pixel* yang sebelumnya 0-255 diubah menjadi 0-1 dengan cara membagi nilai *pixel*-nya dengan 255. Pembagian dengan 255 bertujuan untuk mempercepat waktu komputasi pada saat pelatihan. Label citra yang bertipe data integer selanjutnya diubah menjadi *one-hot encoding*. Hal ini bertujuan untuk menghindari permasalahan pada algoritma *machine learning* dimana angka yang lebih besar adalah angka yang lebih baik. Oleh karena itu, solusi yang tepat adalah menggunakan *one-hot encoding*.

Data *training* selanjutnya dibagi menjadi data *training* dan data *validation* didapatkan sebanyak 20% dari *dataset training*, sedangkan untuk testing dipilih secara acak dengan total 800 citra dari image-net.org dengan masing-masing 200 citra setiap kelas dan disimpan ke dalam pickle. Setelah itu dilakukan pembuatan arsitek model CNN beserta pelatihan arsitek model CNN. Setelah pelatihan arsitek model selesai, lalu model disimpan. Model yang telah disimpan digunakan untuk tahap pengujian terhadap data *testing*. Dari pengujian tersebut akurasi model didapatkan.

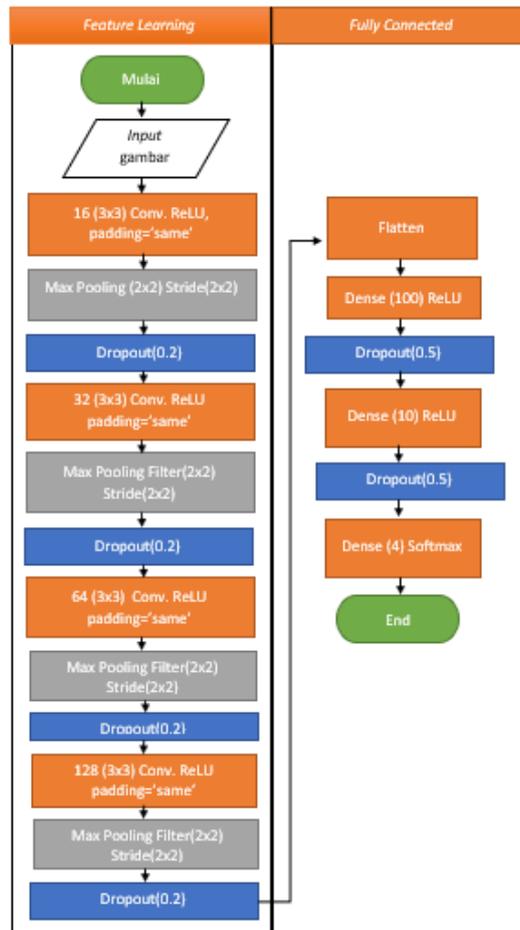
Arsitektur Model CNN

Pada tahap ini arsitektur yang digunakan memiliki 1 *input layer*, 5 *convolution layer*, 5 *pooling layer*, 1 *flatten layer*, 2 *fully-connected layers*, dan 1 *output layer*. *Input layer* menerima citra yang memiliki ukuran 128x128x3. *Convolution layers* memiliki filter sebesar 3x3 yang diakhiri dengan fungsi aktivasi ReLU. *Convolution layer 1 – 5* memiliki filter yang berjumlah 32, 64, 128, 256, 512. *Pooling layers* memiliki ukuran sebesar 2x2 dengan *strides* sebesar 2x2. *Dropout* setelah *convolution layers* adalah 0.2. *Fully-connected layers* memiliki 256 *dense* yang diakhiri dengan fungsi aktivasi ReLU serta nilai *dropout* adalah 0.5. *Output layers* memiliki 4 *dense*. Model CNN secara lengkap dapat dilihat pada Gambar 3.

Pada Gambar 3 ditunjukkan model CNN yang digunakan pada penelitian ini.

Model CNN memiliki 2 tahap yaitu *feature learning* dan *fully connected*. Pada tahap *feature learning* terdapat 3 proses yaitu *convolution*, *pooling* dan *dropout*. Sedangkan tahap *fully connected* memiliki 3 proses yaitu *flatten*, *fully-connected*, dan *dropout*.

Model memiliki *input* citra yang berukuran 128x128x3. 128 adalah ukuran *pixel* dan 3 adalah komponen warna RGB. Citra akan dikonvolusi, kemudian akan dilakukan proses *pooling* dan ditambahkan *dropout*. Arsitektur Model CNN pada *feature learning* untuk konvolusi pertama menggunakan jumlah filter sebanyak 16 dengan ukuran kernel 3x3, lalu dilanjutkan dengan proses *pooling* dengan ukuran 2x2 dengan ukuran *stride* 2x2 dan dilanjutkan dengan *dropout* dengan nilai 0.2. Untuk konvolusi kedua menggunakan jumlah filter sebanyak 32 dengan ukuran kernel 3x3, lalu dilanjutkan dengan proses *pooling* dengan ukuran 2x2 dengan ukuran *stride* 2x2 dan dilanjutkan dengan *dropout* dengan nilai 0.2. Untuk konvolusi ketiga menggunakan jumlah filter sebanyak 64 dengan ukuran kernel 3x3, lalu dilanjutkan dengan proses *pooling* dengan ukuran 2x2 dengan ukuran *stride* 2x2 dan dilanjutkan dengan *dropout* dengan nilai 0.2. Untuk konvolusi keempat menggunakan jumlah filter sebanyak 128 dengan ukuran kernel 3x3, lalu dilanjutkan dengan proses *pooling* dengan ukuran 2x2 dengan ukuran *stride* 2x2 dan dilanjutkan dengan *dropout* yang bernilai 0.2.



Gambar 3. Arsitektur Model CNN

Pada *fully connected* dilanjutkan ke-tahap flatten, yaitu mengubah hasil dari konvolusi menjadi vektor 1 dimensi. Lalu dilanjutkan ke lapisan fully connected pertama yang memiliki *dense* 100 dan dilanjutkan dengan dropout yang bernilai 0.5. Lalu dilanjutkan ke lapisan fully connected kedua yang memiliki *dense* 10 dan dilanjutkan dengan dropout yang bernilai 0.5. lalu dilanjutkan ke lapisan terakhir yang memiliki *dense* 4 yang setiap densenya mewakili 1 jenis citra.

HASIL DAN PEMBAHASAN

Hasil Pelatihan Model

Pelatihan model CNN dilakukan dengan *steps* sebanyak 100, *epoch* sebanyak 100, *batch* sebanyak 64, waktu *training* 46 menit, *dataset training* menghasilkan akurasi sebesar 0.9231 dan loss 0.2062. *Dataset validation* memiliki akurasi sebesar 0.8188 dan loss 0.6469. Hasil dari pelatihan ini ditunjukkan pada Tabel 1.

Tabel 1. Hasil Akurasi pelatihan

Pelatihan	Akurasi	Loss
<i>Training</i>	92,31%	20,62%
<i>Validation</i>	81,88%	64,69%

Tabel 2. Confusion Matrix Validation

		PREDIKSI			
		Harimau	Jaguar	Macan	Singa
SEBENARNYA	Harimau	232	6	9	16
	Jaguar	6	181	39	2
	Macan	2	71	157	16
	Tutul				
	Singa	3	1	6	216

Pada Tabel 2 disajikan hasil prediksi dari *dataset validation* untuk masing-masing kelas pada genus *Panthera*. *Dataset validation* terdiri dari 960 citra atau 20% dari *dataset training*.

Pada Tabel 2 diperoleh informasi keberhasilan model dalam mengklasifikasikan citra sesuai dengan kelasnya. Tingkat keakuratan *validation* diukur untuk melihat ketepatan model dalam mengklasifikasikan *genus panthera*. Dari 960 citra data *validation*, model dapat

memprediksi dengan nilai akurasi sebesar 68% didapatkan dengan perhitungan berikut.

$$\frac{232 + 181 + 157 + 216}{960} * 100\% = 81,88\%$$

Hasil Pengujian Model

Pada Tabel 3 berikut disajikan hasil prediksi dari *dataset training* untuk masing-masing kelas pada genus *Panthera*. *Dataset training* terdiri dari 800 citra dengan banyaknya citra tiap kelas adalah 200 citra.

Tabel 3. Confusion Matrix Testing

		PREDIKSI			
		Harimau	Jaguar	Macan	Singa
SEBENARNYA	Harimau	181	14	2	3
	Jaguar	5	190	4	1
	Macan	18	130	49	3
	Tutul				
	Singa	59	12	7	122

Tingkat keakuratan *testing* diukur untuk melihat ketepatan model dalam mengklasifikasi *genus panthera*. Dari 800 *citra* data *testing*, model dapat memprediksi dengan nilai akurasi sebesar 68% didapatkan dengan perhitungan berikut.

$$\frac{181 + 190 + 49 + 122}{800} * 100\% = 68\%$$

Berdasarkan hasil pengujian maka diperoleh hasil bahwa tingkat akurasi untuk model CNN yang telah dibuat untuk klasifikasi *genus panthera* adalah sebesar 68%. Meskipun tingkat akurasi belum mencapai 90% tetapi sudah berhasil memperbaiki hasil penelitian sebelumnya terutama untuk klasifikasi macan tutul. Pada penelitian sebelumnya [1] macan tutul belum berhasil diklasifikasikan. Pada penelitian ini, dari 200 *citra* macan tutul, sudah 49 *citra* yang berhasil dikenali sebagai macan tutul sedangkan sebagian besar macan tutul dikenali sebagai jaguar.

KESIMPULAN DAN SARAN

Pada penelitian ini telah berhasil dilakukan klasifikasi *genus panthera* dengan menggunakan bahasa pemrograman python dengan metode *Convolutional Neural Network*. Arsitektur model CNN yang digunakan memiliki 1 *input layer*, 5 *convolution layer*, dan 2 *fully connected layer*.

Hasil pelatihan yang menggunakan *dataset training* sebanyak 3840 *citra* memiliki akurasi sebesar 92,31% dan *loss* sebesar 20,62%, sedangkan pada *dataset validation*

yang memiliki *citra* sebanyak 960 memiliki akurasi sebesar 81,88% dan *loss* sebesar 64,69%.

Hasil pelatihan terhadap *dataset validation* yang memiliki 960 *citra* memiliki akurasi sebesar 81,88%. *Precision* sebesar 95% untuk harimau, 70% untuk jaguar, 74% untuk macan tutul, 87% untuk singa. *Recall* sebesar 88% untuk harimau, 79% untuk jaguar, 65% untuk macan tutul, 96% untuk singa. *F1-score* sebesar 92% untuk harimau, 74% untuk jaguar, 69% untuk macan tutul, 91% untuk singa. Macan tutul mendapatkan akurasi terendah dibandingkan 3 yang lainnya.

Hasil pengujian yang menggunakan *dataset testing* yang memiliki 800 *citra* memiliki akurasi sebesar 68%. *Precision* sebesar 69% untuk harimau, 55% untuk jaguar, 79% untuk macan tutul, 95% untuk singa. *Recall* sebesar 91% untuk harimau, 95% untuk jaguar, 24% untuk macan tutul, 61% untuk singa. *F1-score* sebesar 78% untuk harimau, 70% untuk jaguar, 37% untuk macan tutul, 74% untuk singa. Macan tutul mendapatkan akurasi terendah dibandingkan 3 yang lainnya.

Pada penelitian selanjutnya saran yang dapat dilakukan untuk meningkatkan penelitian terkait klasifikasi *genus panthera* dengan menggunakan metode *Convolutional Neural Network* adalah menggunakan ukuran *citra* yang lebih besar dari 128x128. Dengan menambahkan jumlah *citra* untuk meningkatkan performa pelatihan model. Dengan penambahan ekstraksi fitur tekstur pada *citra*.

DAFTAR PUSTAKA

- [1] A. Suryanto, B. Andrianto, F. Alvianda, H. A. Saputor, dan M. S. Siregar, "Pengklasifikasian Genus Panthera (Harimau, Singa, Jaguar dan Macan Tutul dengan metode Naïve Bayes)", 2014. [Daring]. Tersedia: <https://fdokumen.com/document/pengklasifikasi-genus-panthera-harimau-singa-jaguar-dan-program-teknologi.html>, [Diakses 23 Agustus 2019].
- [2] Ensiklopedia Biologi Dunia Hewan Vol. 2 Mamalia. Jakarta: Lentera Abadi, 2017.
- [3] I. W. Suartika E. P., A. Y. Wijaya, dan R. Soelaiman, "Klasifikasi citra menggunakan *Convolutional Neural Network* (CNN) pada Caltech 101", *Jurnal Teknik ITS*, vol. 5, no. 1, 2016.
- [4] K. Chauhan dan S. Ram, "Image classification with deep learning and comparison between different convolutional neural network structures using tensorflow and keras", *International Journal of Advance Engineering and Research Development*, vol. 5, Issue 02, hal. 533-538, 2018.
- [5] S. Samuel, "Pengenalan Deep Learning Part 7 : Convolutional Neural Network", *medium.com*, Nov 13, 2017. [Daring]. Tersedia: <https://medium.com/@samuelsena/pengenalan-deep-learning-part-7-convolutional-neural-network-cnn-b003b477dc94>. [Diakses 23 Agustus 2019].
- [6] J. Brownlee, "How to configure image data augmentation in Keras", April, 2019. [Daring]. Tersedia: <https://machinelearningmastery.com/how-to-configure-image-data-augmentation-when-training-deep-learning-neural-networks/>. [Diakses 5 Juli 2019].
- [7] L. Zmudzinski, "Deep learning guinea pig image classification using Nvidia DIGITS and GoogLeNet", 2016. [Daring]. Tersedia: <http://ceur-ws.org/Vol-2240/paper16.pdf>. [Diakses 26 Agustus 2019].
- [8] N. Srivastava, G. Hinton, A. Krizhevsky, I. Sutskever, dan R. Salakhutdinov, "Dropout: A Simple way to prevent neural networks from overfitting", *Journal Machine Learning Research*, vol. 15, no. 1, hal. 1929-1958, 2014.
- [9] Keras Documentation. "Keras : The Python Deep Learning library". [Daring]. Tersedia: <https://keras.io/>. [Diakses 26 Agustus 2019].