

ULTIMA Computing

Jurnal Sistem Komputer

SOFIAN WIRAHADI, SUPARNI, ACHMAD BAROQAH POHAN

Sistem Pendukung Keputusan Penilaian Kinerja Karyawan PT. Injep Inti Cemerlang Dengan Metode AHP dan TOPSIS

I MADE ASTAWA, SISWAHYUDI AZHARI, I PUTU ANANTA YOGISWARA, WIDRIANTO SIH PINASTIKO

Rancang Bangun Prototipe Sistem Multilateration 2-Dimensi Menggunakan Data ADS-B

DWAYNE JEREMY EUAGELLINO PRIHANTO, MAGDALENA A. INEKE PAKERENG

Perancangan Teknik Kriptografi Block Cipher Berbasis Pola Tarian Sajojo Papua

DIAN PUSPITA CHANDRA, ALETHEA SURYADIBRATA

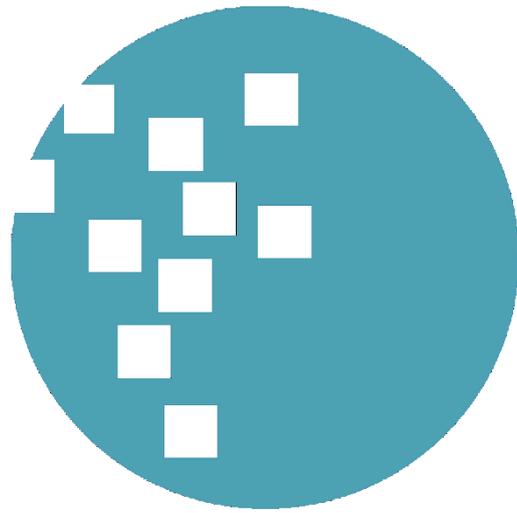
Implementasi Jaringan Saraf Tiruan Backpropagation Untuk Pengenalan Karakter Pada Dokumen Tercetak

FENINA ADLINE TWINCE TOBING, MUHAMMAD IQBAL DZULHAQ, RAMZI FAJAR SIDIQ

Penerapan Metode Fuzzy AHP Untuk Sistem Pendukung Keputusan Pemilihan Pemasok Terbaik

I NYOMAN ADITYA YUDISWARA, ABBA SUGANDA

Analisis Kinerja Algoritma Quick Double Merge Sort Paralel Menggunakan openMP



UMN

**UNIVERSITAS
MULTIMEDIA
NUSANTARA**

SUSUNAN REDAKSI

Pelindung

Dr. Ninok Leksono

Penanggungjawab

Dr. Ir. P.M. Winarno, M.Kom.

Pemimpin Umum

Samuel Hutagalung, M.T.I.

Mitra Bestari

(UMN) Hira Meidia, B.Eng., Ph.D.

(UMN) Dr. Rangga Winantyo, Ph.D.

(Universitas Indonesia) Filbert Hilman

Juwono, S.T., M.T.

(Tanri Abeng University) Nur Afny Catur

Andryani, M.Sc.

(UMN) Kanisius Karyono, S.T., M.T.

(UMN) Ahmad Syahril Muharom, S.Pd., M.T.

(UMN) Adhi Kusnadi, S.T., M.Kom.

(UMN) Dennis Gunawan S.Kom., M.Sc.,

CEH, CEI, CND

Ketua Dewan Redaksi

Ni Made Satvika Iswari, S.T., M.T.

Dewan Redaksi

Wella, S.Kom., M.S.I., COBIT5

M.B.Nugraha, S.T., M.T.

Andre Rusli, S.Kom., M.Sc

Desainer dan Layouter

M.B.Nugraha, S.T., M.T.

Dyah Ayu Anggreini Tuasikal, S.T., M.T.

Sirkulasi dan Distribusi

Sularmin

Keuangan

I Made Gede Suteja, S.E.

ALAMAT REDAKSI

Universitas Multimedia Nusantara (UMN)

Jl. Scientia Boulevard

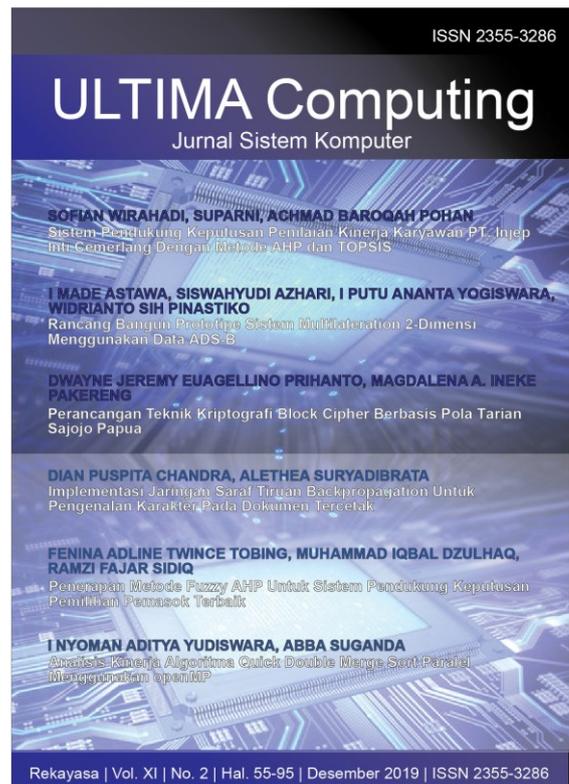
Gading Serpong

Tangerang, Banten - 15811

Telp. (021) 5422 0808

Faks. (021) 5422 0800

Surel. ultimacomputing@umn.ac.id



Jurnal ULTIMA Computing merupakan Jurnal Program Studi Sistem Komputer Universitas Multimedia Nusantara yang menyajikan artikel-artikel penelitian ilmiah dalam bidang Sistem Komputer serta isu-isu teoritis dan praktis yang terkini, mencakup komputasi, organisasi dan arsitektur komputer, programming, embedded system, sistem operasi, jaringan dan internet, integrasi sistem, serta topik lainnya di bidang Sistem Komputer. Jurnal ULTIMA Computing terbit secara berkala dua kali dalam setahun (Juni dan Desember) dan dikelola oleh Program Studi Sistem Komputer Universitas Multimedia Nusantara bekerjasama dengan UMN Press.

Call for Papers



International Journal of New Media Technology (IJNMT) is a scholarly open access, peer-reviewed, and interdisciplinary journal focusing on theories, methods and implementations of new media technology. Topics include, but not limited to digital technology for creative industry, infrastructure technology, computing communication and networking, signal and image processing, intelligent system, control and embedded system, mobile and web based system, and robotics. IJNMT is published annually by Information and Communication Technology Faculty of Universitas Multimedia Nusantara in cooperation with UMN Press.



Jurnal ULTIMATICS merupakan Jurnal Program Studi Teknik Informatika Universitas Multimedia Nusantara yang menyajikan artikel-artikel penelitian ilmiah dalam bidang analisis dan desain sistem, *programming*, algoritma, rekayasa perangkat lunak, serta isu-isu teoritis dan praktis yang terkini, mencakup komputasi, kecerdasan buatan, pemrograman sistem *mobile*, serta topik lainnya di bidang Teknik Informatika.



Jurnal ULTIMA Computing merupakan Jurnal Program Studi Sistem Komputer Universitas Multimedia Nusantara yang menyajikan artikel-artikel penelitian ilmiah dalam bidang Sistem Komputer serta isu-isu teoritis dan praktis yang terkini, mencakup komputasi, organisasi dan arsitektur komputer, *programming*, *embedded system*, sistem operasi, jaringan dan internet, integrasi sistem, serta topik lainnya di bidang Sistem Komputer.



Jurnal ULTIMA InfoSys merupakan Jurnal Program Studi Sistem Informasi Universitas Multimedia Nusantara yang menyajikan artikel-artikel penelitian ilmiah dalam bidang Sistem Informasi, serta isu-isu teoritis dan praktis yang terkini, mencakup sistem basis data, sistem informasi manajemen, analisis dan pengembangan sistem, manajemen proyek sistem informasi, *programming*, mobile information system, dan topik lainnya terkait Sistem Informasi.

DAFTAR ISI

Sistem Pendukung Keputusan Penilaian Kinerja Karyawan PT. Injep Inti Cemerlang Dengan Metode AHP dan TOPSIS Sofian Wirahadi, Suparni, Achmad Baroqah Pohan	55-63
Rancang Bangun Prototipe Sistem Multilateration 2-Dimensi Menggunakan Data ADS-B I Made Astawa, Siswahyudi Azhari, I Putu Ananta Yogiswara, Widrianto Sih Pinastiko	64-70
Perancangan Teknik Kriptografi Block Cipher Berbasis Pola Tarian Sajojo Papua Dwayne Jeremy Euagellino Prihanto, Magdalena A. Inneke Pakereng	71-80
Implementasi Jaringan Saraf Tiruan Backpropagation Untuk Pengenalan Karakter Pada Dokumen Tercetak Dian Puspita Chandra, Alethea Suryadibrata	81-89
Penerapan Metode Fuzzy AHP Untuk Sistem Pendukung Keputusan Pemilihan Pemasok Terbaik Fenina Adline Twince Tobing, Muhammad Iqbal Dzulhaq, Ramzi Fajar Sidiq	90-94
Analisis Kinerja Algoritma Quick Double Merge Sort Paralel Menggunakan openMP I Nyoman Aditya Yudiswara, Abba Suganda	95-102

KATA PENGANTAR

Salam ULTIMA!

Jurnal ULTIMA Computing merupakan Jurnal Program Studi Sistem Komputer Universitas Multimedia Nusantara yang menyajikan artikel-artikel penelitian ilmiah dalam bidang Sistem Komputer serta isu-isu teoritis dan praktis yang terkini, mencakup komputasi, organisasi dan arsitektur komputer, programming, *embedded system*, sistem operasi, jaringan dan internet, integrasi sistem, serta topik lainnya di bidang Sistem Komputer.

Pada edisi Desember 2019 ini, ULTIMA Computing telah memasuki Edisi ke-2 dari Volume XI. Dalam jurnal ini terdapat enam buah karya tulis ilmiah yang berasal dari para peneliti, akademisi dan praktisi di bidang Sistem Komputer. Beberapa topik yang diangkat dalam jurnal ini, antara lain Sistem Pendukung Keputusan Penilaian Kinerja Karyawan PT. Injep Inti Cemerlang Dengan Metode AHP dan TOPSIS; Rancang Bangun Prototipe Sistem Multilateration 2-Dimensi Menggunakan Data ADS-B, Perancangan Teknik Kriptografi Block Cipher Berbasis Pola Tarian Sajojo Papua, Implementasi Jaringan Saraf Tiruan Backpropagation untuk Pengenalan Karakter Pada Dokumen Tercetak, Penerapan Metode Fuzzy AHP untuk Sistem Pendukung Keputusan Pemilihan Pemasok Terbaik, Analisis Kinerja Algoritma Quick Double Merge Sort Paralel Menggunakan openMP.

Pada kesempatan kali ini juga kami ingin mengundang partisipasi para pembaca yang budiman, para peneliti, akademisi, maupun praktisi, di bidang Teknik dan Informatika, untuk mengirimkan karya ilmiah yang berkualitas pada: International Journal of New Media Technology (IJNMT), ULTIMATICS, ULTIMA InfoSys dan ULTIMA Computing. Informasi mengenai pedoman dan template penulisan, serta informasi terkait lainnya dapat diperoleh melalui alamat surel ultimacomputing@umn.ac.id.

Akhir kata, kami mengucapkan terima kasih kepada seluruh kontributor dalam ULTIMA Computing Edisi Desember 2019 ini. Kami berharap artikel-artikel ilmiah hasil penelitian dalam jurnal ini dapat bermanfaat dan memberikan sumbangsih terhadap perkembangan penelitian dan keilmuan di Indonesia.

Desember 2019,

Ni Made Satvika Iswari, S.T., M.T.
Ketua Dewan Redaksi

Sistem Pendukung Keputusan Penilaian Kinerja Karyawan PT. Injep Inti Cemerlang Dengan Metode AHP dan TOPSIS

Sofian Wira Hadi¹, Suparni², Achmad Baroqah Pohan³

¹Sistem Informasi, STMIK Nusa Mandiri, Jakarta, Indonesia

^{2,3}Fakultas Ekonomi dan Bisnis, Universitas Bina Sarana Informatika, Jakarta, Indonesia

¹sofianwirahadi1@gmail.com, ²suparni.spn@bsi.ac.id, ³achmad.abq@bsi.ac.id

Diterima 24 September 2019

Disetujui 20 Desember 2019

Abstract—During this time the performance appraisal of PT. Injep Inti Cemerlang has not been implemented optimally, especially in assessing employee performance. So far, performance appraisal is only determined from the results of its work, there are no clear assessment criteria. For this reason, a decision support system is needed to help find the best alternative for the selection of employees. In this study a decision support system for employee performance appraisal will be developed based on Attitude, Responsibility, Attendance, Discipline and Cooperation. This study aims to design a decision support system for employee performance appraisal using data collection methods by observation, interviews and giving questionnaires to employees of PT. Injep Inti Cemerlang. The data collected is carried out the process of analyzing data and looking for weighting values using the AHP method and for ranking using the TOPSIS method, where each criterion is in this case. assessment factors and alternatives in this case employees are compared with criteria that have been weighted through the process of calculating the ahp and topsis method starting from giving the weighting of criteria by calculating with Ms.Excel and calculating with Expert Choice software. The results have been obtained from weighting the next ranking by the TOPSIS method. thus providing a value output that results in a system that assesses employees. This decision support system helps assess the performance of each employee of PT. Injep Inti Cemerlang in determining the employee who has the best performance

Keywords: Decision Support System, Analytical Hierarchy Process (AHP), TOPSIS, Employee Performance

I. PENDAHULUAN

Menurut (Yunus, 2016) Berjalannya suatu manajemen dalam perusahaan tentunya tidak terlepas dari proses evaluasi agar tujuan organisasi dapat dicapai secara efektif dan efisien. Salah satu fungsi evaluasi yang sangat penting dilakukan adalah dengan melakukan penilaian kinerja terhadap karyawan. Kegiatan penilaian kinerja karyawan ini sudah menjadi hal umum khususnya di perusahaan. Kegiatan ini dilaksanakan untuk mengevaluasi kinerja yang ada sehingga dapat segera mengambil tindakan

bila terdapat hal yang menyimpang dari penilaian kinerja tersebut. Selain itu penilaian kinerja karyawan juga mendorong para karyawan untuk meningkatkan kuantitas dan kualitas kerjanya demi perusahaan. Untuk itu perusahaan perlu melakukan penilaian atas kinerja para karyawannya, mengingat karyawan merupakan bagian yang sangat penting dalam berkembangnya sebuah perusahaan.

Menurut (Frieyadi, 2018) Karyawan terbaik menjadi salah satu penunjang utama demi terciptanya prestasi dari suatu instansi. Menurut Undang-Undang RI (No.13, 2003) Tenaga kerja adalah setiap orang yang mampu melakukan pekerjaan guna menghasilkan barang dan/atau jasa baik untuk memenuhi kebutuhan sendiri maupun untuk masyarakat. PT. Injep Inti Cemerlang Merupakan perusahaan yang bergerak di bidang penyedia sarana dan prasarana pelengkap jalan di Indonesia, PT. Injep Inti Cemerlang memiliki karyawan yang berkompeten dibidangnya serta didukung tenaga teknis yang terampil dan berpengalaman, untuk menjamin kualitas hasil pekerjaan yang baik sesuai dengan harapan. PT. Injep Inti Cemerlang Menerapkan *standart operational procedur* (SOP) dengan dukungan data riset yang memadai serta pencapaian standar kendali mutu pada masing-masing bidang dan pihak manajemenpun sudah melakukan suatu proses penilaian. Namun dalam hal ini belum dilaksanakan secara maksimal terutama dalam menilai kinerja karyawan. Selama ini penilaian karyawan hanya ditentukan dari hasil kerjanya, belum ada kriteria penilaian yang jelas. Menurut Rivai di dalam (Rizky, 2018) mengemukakan penilaian kinerja mengacu pada suatu sistem formal dan terstruktur yang digunakan untuk mengukur, menilai dan mempengaruhi sifat-sifat yang berkaitan dengan pekerjaan, perilaku, dan hasil, termasuk tingkat ketidakhadiran.

Sistem pendukung keputusan merupakan solusi yang tepat dalam proses pengambilan keputusan penilaian kinerja karyawan. Penelitian-penelitian yang sudah berkembang sampai saat ini pada umumnya menggunakan Metode SAW dan TOPSIS paling

banyak digunakan dalam proses pengambilan keputusan (Mude, 2016). Pada penelitian sebelumnya dengan judul Sistem Pendukung Keputusan Penilaian Kinerja Karyawan penelitian ini menggunakan metode SAW (*Simple Additive Weighting*) di Universitas Muhammadiyah Purwokerto (Anto, Mustafidah, & Suyadi, 2015). Penelitian membahas lima kriteria untuk melakukan penilaian kinerja karyawan, antara lain, Penilaian umum, Tingkat kehadiran, Tingkat pendidikan, Pengembangan diri, Unsur penunjang. Penelitian selanjutnya dengan judul sistem pendukung keputusan penilaian kinerja karyawan menggunakan metode tophis (*Technique For Others Reference By Similarity To Ideal Solution*) pada PT. Rio Utama Samarinda menggunakan metode TOPSIS (Yusnita, Salmon, & Helmi, 2015) pada penelitian ini membahas tujuh kriteria untuk melakukan penilaian kinerja karyawan, antara lain, Kualitas Kerja, Kedisiplinan, Kompetensi, Kerja Sama, Perilaku, Inisiatif, Penampilan. Alternatif yang dipilih memiliki kedekatan dengan solusi ideal positif dan memiliki jarak yang jauh dari solusi ideal negatif. Komposit dari nilai kerja terbaik yang ditampilkan oleh setiap alternatif untuk setiap atribut akan membentuk solusi ideal. Kekurangan metode ini adalah belum adanya penentuan bobot prioritas yang menjadi prioritas hitungan terhadap kriteria yang terpilih. Penelitian lain dengan judul Sistem Pendukung Keputusan Untuk Penilaian Kinerja Pegawai Menggunakan Metode *Analytical Hierarchy Process* (AHP) Pada RSUD Serang penelitian ini untuk membangun sistem pendukung keputusan penilaian kinerja karyawan menggunakan metode AHP (Saefudin & Wahyuningsih, 2014). Menurut (Candra, Farmadi, & Khairunnisa, 2015) Metode AHP-TOPSIS memiliki peranan masing-masing sehingga mempercepat proses dan menghasilkan nilai keputusan yang optimal. AHP digunakan untuk pembobotan kriteria yang berguna meningkatkan validitas nilai bobot perhitungan kriteria dan TOPSIS berperan dalam menentukan perbandingan alternatif.

Dari latar belakang pengetahuan yang telah di analisa maka diharapkan mendapatkan hasil yang lebih efektif dan efisien. Hal ini yang membuat peneliti termotivasi untuk membuat penelitian di PT. Injep Inti Cemerlang dengan membuat sistem pendukung keputusan kinerja karyawan dengan menggunakan metode *Analytical Hierarchy Proses* (AHP) dan *Technique For Order Preference by Similarity to Ideal Solution* (TOPSIS).

II. METODE PENELITIAN

A. Observasi

Pada tahap observasi, peneliti melaksanakan pengamatan langsung terhadap proses penilaian kinerja untuk mencari informasi dan mengumpulkan data di PT. Injep Inti Cemerlang Jl. 1 Siliwangi No 12 RT/RW 05/03 Desa, Cibeuteung Udik, Ciseeng, Bogor, Jawa Barat 16120.

B. Wawancara

Dalam tahapan wawancara ini peneliti mewawancarai Ibu Lia Apriyanti, S.Kom bagian staff administrasi PT. Injep Inti Cemerlang dengan menyiapkan beberapa pertanyaan mengenai permasalahan yang terjadi di perusahaan tersebut.

C. Studi Pustaka

Tahapan ini peneliti lakukan dengan mencari referensi dari e-journal, e-book, media online dan buku-buku referensi lainnya yang kemudian peneliti input kedalam aplikasi mendeley untuk pengelolaan referensi secara efektif

D. Kuisisioner

Kuisisioner peneliti berikan kepada staff PT. Injep Inti Cemerlang guna melengkapi data penelitian dalam menentukan ranking penilaian kinerja karyawan

E. Sistem Pendukung Keputusan

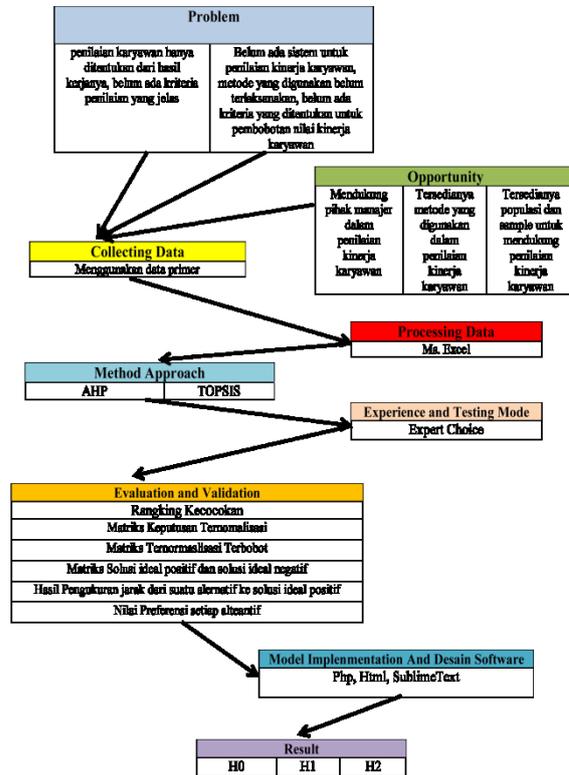
Sistem pendukung keputusan adalah sistem informasi berbasis komputer yang interaktif, dengan cara mengolah data dengan berbagai model untuk memecahkan masalah-masalah yang tidak terstruktur sehingga dapat memberikan informasi yang bisa digunakan oleh para pengambil keputusan dalam membuat sebuah keputusan (Chamid & Murti, 2017). Dari definisi diatas dapat disimpulkan bahwa sistem pendukung keputusan adalah suatu sistem informasi spesifik yang ditujukan untuk membantu manajemen dalam mengambil keputusan yang berkaitan dengan persoalan semi terstruktur. Adapun dukungan DSS sebagai berikut:

- Analisis Jika-Maka (*What-if-analysis*)
Analisa yang dititik beratkan pada pengobservasian perubahan terhadap variabel-variabel tertentu berpengaruh terhadap variabel lain.
- Analisis Sensitivitas (*Sensitivity Analysis*)
Analisa yang dititik beratkan pada pengobservasian pengaruh naik turunnya suatu variabel terhadap variabel-variabel lainnya.
- Analisis Pencarian Sasaran (*Goal Seeking Analysis*)
Analisa yang dititik beratkan pada perubahan nilai-nilai beberapa variabel hingga nilai variabel yang diinginkan mencapai nilai tertentu.
- Analisis Optimisasi (*Optimization Analysis*)
Analisa yang dititik beratkan pada pencarian nilai-nilai optimum dari setiap variabel yang ada berdasarkan pada kendala-kendala yang ada.

Dari beberapa dukungan DSS. Peneliti menggunakan Analisa Jika-maka (*What-if-analysis*) dan Analisa

Pencarian Sasaran (*Goal Seeking Analysis*) dalam tahapan penelitian kinerja karyawan PT. Injep Inti Cemerlang.

F. Kerangka Pemikiran



Gambar 1. Kerangka Pemikiran

III. HASIL DAN PEMBAHASAN

A. Analisa Menggunakan Metode AHP

Untuk menganalisa hasil akhir dari penelitian pemilihan kinerja karyawan terbaik harus sesuai dengan kriteria yang sudah ditentukan mulai dari sikap, tanggung jawab, absensi, kedisiplinan dan kerjasama. Untuk itu diperlukan pembobotan kriteria dengan menggunakan metode AHP dan mencari nilai konsistensinya untuk dipergunakan dalam metode TOPSIS. Data dan nilai yang diambil dari hasil keseluruhan kuesioner untuk menentukan hasil akhir penilaian kinerja karyawan terbaik. Berikut kriteria responden yang ditentukan :

- Karyawan kantor adalah responden
- Syarat untuk menjadi responden adalah jabatan yang paling tinggi di PT Injep Inti Cemerlang dan mengetahui kinerja karyawan lapangan.
- 1 Responden menilai 16 Karyawan lapangan
- Responden berjumlah 4 terdiri dari kepala bagian
- Perhitungan Responden menggunakan *Geometric Mean Theory* Atau disebut

perataan jawaban.untuk mendapatkan nilai tertentu dari nilai tersebut, kemudian hasil harus dikalikan satu sama lain, dan hasil perkalian dipangkatkan dengan 1/n dimana n adalah jumlah responden. Secara sistematis persamaan tersebut adalah sebagai berikut :

$$\sqrt[n]{a_1x_1a_2x_2a_3x_3a_4 \dots a_nx_n} \quad (1)$$

- Kuesioner yang diberikan terdapat dua metode yaitu Kuesioner untuk mengetahui tingkat kesesuaian persepsi (TOPSIS) dan tingkat kepentingan harapan dari setiap kriteria penilaian pada setiap alternatif yang sudah ditentukan (AHP)

Adapun itu analisa untuk menentukan prioritas kriteria dengan menggunakan metode AHP sebagai berikut :

Tabel 1. Matriks Perbandingan Berpasangan

TUJUAN	C1	C2	C3	C4	C5
C1	1,00	5,00	3,00	5,00	2,00
C2	0,20	1,00	0,25	4,00	0,50
C3	0,33	4,00	1,00	5,00	1,00
C4	0,20	0,25	0,20	1,00	0,50
C5	0,50	2,00	1,00	2,00	1,00

Pada Tabel 1 menjelaskan bahwa rekapitulasi hasil matriks perbandingan berpasangan gabungan dari 4 (empat) responden yang terdiri dari.

Tabel 2. Matriks Nilai Kriteria

Tujua n	C1	C2	C3	C4	C5	Jumla h	Prorit y Vector
C1	0,4 5	0,41	0,5 5	0,29	0,40	2,10	0,42
C2	0,0 9	0,08	0,0 5	0,24	0,10	0,55	0,11
C3	0,1 5	0,33	0,1 8	0,29	0,20	1,15	0,23
C4	0,0 9	0,02	0,0 4	0,06	0,10	0,31	0,06
C5	0,2 2	0,16	0,1 8	0,12	0,20	0,89	0,18
Jumlah	1,0 0	1,00	1,0 0	1,00	1,00	5,00	

Pada tabel 2 Menjelaskan bahwa unsur-unsur tiap kolom dibagi dengan jumlah kolom yang berangkutan dan akan diperoleh bobot relatif yang dinormalkan

Tabel 3. Perhitungan Rasio Konsistensi

Tujuan	Jumlah Per Baris	Prioritas	Hasil Kali	Hasil Kali/ Prioritas
C1	16	0,42	2,32	5,52
C2	5,95	0,11	0,58	5,29
C3	11,33	0,23	1,29	5,61
C4	2,15	0,06	0,31	5,13
C5	6,5	0,18	0,96	5,33
Jumlah				5,38

Jumlah = 5,38
 N= 5 (IR = 1.19)
 Maks = 5,38
 $CI = (5,38-5)/5-1=0,09$
 $CR = 0,09/1.19 = 0,08$ Oleh karena $CR < 0,1$, maka rasio konsistensi dari perhitungan tersebut dapat diterima.

Dari hasil perhitungan pada tabel diatas menunjukkan bahwa kriteria Sikap merupakan kriteria yang paling penting dalam penilaian kinerja karyawan menjadi prioritas ke-1 dengan nilai bobot 42%, berikutnya absensi menjadi menjadi prioritas ke-2 dengan nilai bobot 23%, kemudian kerjasama menjadi prioritas ke-3 dengan nilai bobot 18%, selanjutnya tanggung jawab menjadi prioritas ke-4 dengan nilai bobot 11% dan yang terakhir kedisiplinan menjadi prioritas ke-5 dengan nilai bobot 6%.

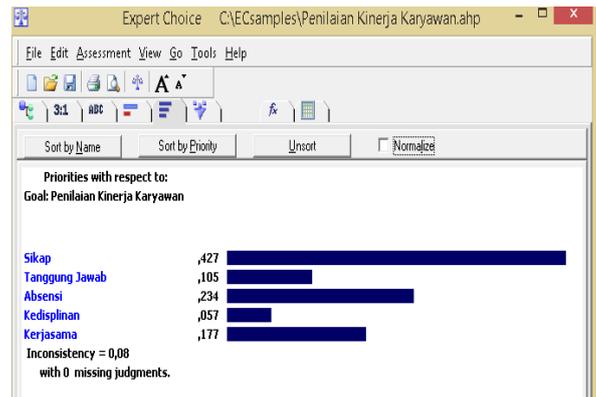
B. Pengujian Metode AHP dengan Expert Choice

Untuk menentukan hasil yang lebih maksimal diperlukan pengujian metode AHP dengan Expert Choice. Hasil yang diperoleh dalam Expert Choice sebagai berikut :

Sikap	Tanggung Jawab	Absensi	Kedisiplinan	Kerjasama
	5,0	3,0	5,0	2,0
		4,0	4,0	2,0
			5,0	1,0
				2,0
Incon: 0,08				

Gambar 2. Matriks Perbandingan Berpasangan

Pada gambar IV.1 menjelaskan bahwa rekapitulasi hasil matriks perbandingan berpasangan gabungan dari 4 (empat) responden yang di input kedalam Expert Choice.



Gambar 3. Rasio Konsistensi

Overall inconsistency 0,08 Oleh karena $CR < 0,1$, maka rasio konsistensi dari perhitungan tersebut dapat diterima.

Dari hasil perhitungan pada gambar diatas (Expert Choice) menunjukan bahwa kriteria Sikap merupakan kriteria yang paling penting dalam penilaian kinerja karyawan menjadi prioritas ke-1 dengan nilai bobot 0,427 berikutnya absensi menjadi menjadi prioritas ke-2 dengan nilai bobot 0,234%, kemudian kerjasama menjadi prioritas ke-3 dengan nilai bobot 0,177, selanjutnya tanggung jawab menjadi prioritas ke-4 dengan nilai bobot 0,105 dan yang terakhir kedisiplinan menjadi prioritas ke-5 dengan nilai bobot 0,057.

Maka dari itu penghitungan manual sama dengan penghitungan didalam Expert Choice dengan memiliki nilai rasio konsistensi yaitu 0,08 karena kurang dari 0,1 perhitungan tersebut dapat diterima dan bisa di implementasikan

C. Analisa Dengan Metode TOPSIS

Metode TOPSIS berperan dalam menentukan perangkaan alternatif. Untuk memperoleh ranking kecocokan dari alternatif penilaian kinerja karyawan maka harus menentukan nilai kecocokan antara alternatif dan kriteria dengan interval 1 sampai 5, dengan ketentuan 1= Tidak Baik, 2=Kurang Baik, 3=Cukup Baik, 4=Baik, 5= Sangat Baik.

Tabel 4.Interval Penilaian

No	Atribut	Himpunan	Nilai
C1	Sikap	Tidak Baik	1
		Kurang Baik	2
		Cukup Baik	3
		Baik	4

		Sangat Baik	5
C2	Tanggung Jawab	Tidak Baik	1
		Kurang Baik	2
		Cukup Baik	3
		Baik	4
		Sangat Baik	5
C3	Absensi	Tidak Baik	1
		Kurang Baik	2
		Cukup Baik	3
		Baik	4
		Sangat Baik	5
C4	Kedisiplinan	Tidak Baik	1
		Kurang Baik	2
		Cukup Baik	3
		Baik	4
		Sangat Baik	5
C5	Kerjasama	Tidak Baik	1
		Kurang Baik	2
		Cukup Baik	3
		Baik	4

		Sangat Baik	5
--	--	-------------	---

Tabel 5. Rangkings Kecocokan

Alternatif	(C1)	(C2)	(C3)	(C4)	(C5)
K1	4	3	4	4	3
K2	4	4	4	3	3
K3	4	4	4	3	4
K4	4	4	3	4	3
K5	3	4	4	3	4
K6	3	3	4	3	3
K7	4	4	4	4	3
K8	4	4	4	4	3
K9	4	4	4	3	4
K10	4	3	3	3	3
K11	3	2	5	4	3
K12	4	5	3	2	4
K13	3	2	2	3	4
K14	4	5	3	4	3
K15	3	4	3	2	2
K16	5	4	3	3	2

Setiap rangking kecocokan dari setiap alternatif pada setiap kriteria. Setiap kriteria dinilai dengan 1 sampai 5, dengan ketentuan 1= tidak baik, 2= kurang baik, 3 = cukup baik, 4 = baik, 5 = sangat baik

Tabel 6. Matriks Keputusan Ternormalisasi

Alternatif	R1	R2	R3	R4	R5
K1	0,2638	0,1982	0,2754	0,3015	0,2308
K2	0,2638	0,2643	0,2754	0,2261	0,2308
K3	0,2638	0,2643	0,2754	0,2261	0,3077
K4	0,2638	0,2643	0,2065	0,3015	0,2308
K5	0,1978	0,2643	0,2754	0,2261	0,3077
K6	0,1978	0,1982	0,2754	0,2261	0,2308
K7	0,2638	0,2643	0,2754	0,3015	0,2308

K8	0,2638	0,2643	0,2754	0,3015	0,2308
K9	0,2638	0,2643	0,2754	0,2261	0,3077
K10	0,2638	0,1982	0,2065	0,2261	0,2308
K11	0,1978	0,1322	0,3442	0,3015	0,2308
K12	0,2638	0,3304	0,2065	0,1508	0,3077
K13	0,1978	0,1322	0,1377	0,2261	0,3077
K14	0,2638	0,3304	0,2065	0,3015	0,2308
K15	0,1978	0,2643	0,2065	0,1508	0,1538
K16	0,3297	0,2643	0,2065	0,2261	0,1538

Setiap elemen matriks D dinormalisasikan untuk mendapatkan matriks normalisasi R. Setiap normalisasi dari nilai rij

Tabel 7. Matriks Ternormalisasi Terbobot

Alternatif	Y1	Y2	Y3	Y4	Y5
K1	0,1108	0,0218	0,0633	0,0181	0,0415
K2	0,1108	0,0291	0,0633	0,0136	0,0415
K3	0,1108	0,0291	0,0633	0,0136	0,0554
K4	0,1108	0,0291	0,0475	0,0181	0,0415
K5	0,0831	0,0291	0,0633	0,0136	0,0554
K6	0,0831	0,0218	0,0633	0,0136	0,0415
K7	0,1108	0,0291	0,0633	0,0181	0,0415
K8	0,1108	0,0291	0,0633	0,0181	0,0415
K9	0,1108	0,0291	0,0633	0,0136	0,0554
K10	0,1108	0,0218	0,0475	0,0136	0,0415
K11	0,0831	0,0145	0,0792	0,0181	0,0415
K12	0,1108	0,0363	0,0475	0,009	0,0554
K13	0,0831	0,0145	0,0317	0,0136	0,0554
K14	0,1108	0,0363	0,0475	0,0181	0,0415
K15	0,0831	0,0291	0,0475	0,009	0,0277
K16	0,1385	0,0291	0,0475	0,0136	0,0277

Perhitungan matrik keputusan normalisasi terbobot, mengambil bobot masing-masing kriteria dari metode AHP. Bobot prioritas kriteria 0,42 untuk

sikap, 0,11 untuk tanggung jawab, 0,23 untuk absensi, 0,06 untuk kedisiplinan, dan 0,18 untuk kerjasama. Perhitungan matrik keputusan normalisasi terbobot, mengambil bobot masing-masing kriteria dari metode AHP. Bobot prioritas kriteria 0,42 untuk sikap, 0,11 untuk tanggung jawab, 0,23 untuk absensi, 0,06 untuk kedisiplinan, dan 0,18 untuk kerjasama.

Tabel 8. Matriks Solusi Ideal Positif

S1+	0,03769
S2+	0,03581
S3+	0,03303
S4+	0,04488
S5+	0,05824
S6+	0,06117
S7+	0,03553
S8+	0,03553
S9+	0,03303
S10+	0,04683
S11+	0,06112
S12+	0,04303
S13+	0,07629
S14+	0,04429
S15+	0,07051
S16+	0,04293

Menentukan matrik solusi ideal positif dan matrik solusi ideal negatif berdasarkan persamaan 3 dan 4.

Tabel 9. Hasil Pengukuran Jarak dari Suatu Alternatif ke Solusi Ideal Positif

Y	Max	Min
Y1	0,1385	0,0831
Y2	0,0363	0,0145
Y3	0,0792	0,0317
Y4	0,0181	0,009
Y5	0,0554	0,0415

Tabel 10. Solusi Ideal Negatif

S1-	0,04364
S2-	0,04474
S3-	0,04683
S4-	0,036206
S5-	0,037767
S6-	0,032805
S7-	0,04542
S8-	0,04542
S9-	0,046833
S10-	0,03303
S11-	0,048355
S12-	0,041048
S13-	0,014566
S14-	0,039687
S15-	0,025569
S16-	0,061172

Menghitung separation measure. Separation measure ini merupakan pengukuran jarak dari suatu alternatif ke solusi ideal positif dan solusi ideal negative

Tabel 11. Nilai Preferensi Setiap Alternatif untuk prangkingan setiap alternatif

V1	0,53656
V2	0,5554
V3	0,58642
V4	0,4465
V5	0,39338
V6	0,34907
V7	0,5611
V8	0,5611
V9	0,58642
V10	0,41358
V11	0,44172
V12	0,48821
V13	0,16032
V14	0,4726
V15	0,26612
V16	0,58762

DSS Dari nilai hasil kriteria yang ditentukan Dery memiliki nilai 4,73 (sikap) Sangat Baik, 3,46 (Tanggung Jawab) Cukup Baik, 3,46 (Absensi) Cukup Baik, 4,47 (Kedisiplinan) Cukup Baik, 4,00 (Kerja Sama) Baik. Dari perhitungan TOPSIS dapat

diperoleh urutan rekomendasi penilaian kinerja terbaik karyawan PT Injep Inti Cemerlang yang memenuhi kriteria adalah V16 yaitu Dery, Usia 24 Tahun dengan nilai preferensi 0,58762. Hal ini dapat dijadikan bahan evaluasi untuk kinerja karyawan PT. Injep Inti Cemerlang

IV. IMPLEMENTASI

Implementasi sistem dilakukan berdasarkan spesifikasi rancangan yang telah ditentukan. Hasil implementasi sistem pendukung keputusan keputusan penilaian kinerja karyawan sebagai berikut:

A. Halaman Kriteria

Halaman Kriteria berfungsi untuk menambahkan atau menghapus kriteria dan pemberian bobot nilai yang sudah di cari dengan menggunakan software expert choice.

Id Kriteria	Nama Kriteria	Bobot	Poin 1	Poin 2	Poin 3	Poin 4	Poin 5	Sifat Kriteria	Pilihan
kr001	Sikap	42	1	2	3	4	5	benefit	Ubah Hapus
kr002	Tanggung Jawab	11	1	2	3	4	5	benefit	Ubah Hapus
kr003	Absensi	23	1	2	3	4	5	benefit	Ubah Hapus
kr004	Kedisiplinan	6	1	2	3	4	5	benefit	Ubah Hapus
kr005	Kerjasama	18	1	2	3	4	5	benefit	Ubah Hapus

Gambar 4. Halaman Kriteria

B. Halaman Alternatif

Halaman Alternatif berfungsi untuk menambahkan atau menghapus kandidat karyawan yang ingin diberi penilaian.

Id Alternatif	Nama Alternatif	Pilihan
al001	Ridwan	Ubah Hapus
al002	Abeng	Ubah Hapus
al003	Achmad	Ubah Hapus
al004	Awang	Ubah Hapus
al005	eeng	Ubah Hapus
al006	Aji	Ubah Hapus
al007	Itang	Ubah Hapus

Gambar 5. Halaman Alternatif

C. Nilai Matriks

Halaman nilai matriks adalah halaman untuk menilai per alternatif dengan kriteria yang sudah ditentukan

Gambar 6. Nilai Matriks

D. Halaman Hasil TOPSIS

Halaman hasil TOPSIS adalah halaman untuk melihat hasil perhitungan, nilai matrik ternormalisasi nilai bobot ternormalisasi, matriks ideal positif/negatif, jarak solusi ideal positif/negatif dan nilai preferensi

Nomor	Nama	V_i
1	Ridwan	0.8739
2	Abeng	0.6305
3	Azhmad	0.5052
4	Awang	0.2092

Gambar 7. Halaman Hasil TOPSIS

V. KESIMPULAN

Setelah melalui tahapan penilaian dan melakukan analisa data, peneliti dapat menyimpulkan mengenai penilaian kinerja karyawan dengan 16 kandidat karyawan lapangan yang berada di PT Injep Inti Cemerlang, kesimpulan yang telah didapatkan peneliti adalah sebagai berikut :

- Kriteria yang diberikan peneliti sangat berpengaruh terhadap penilaian kinerja karyawan, hal ini dapat dilihat dari perhitungan data yang diperoleh, yaitu kriteria Sikap merupakan kriteria yang

paling penting dalam penilaian kinerja karyawan menjadi prioritas ke-1 dengan nilai bobot 42%, berikutnya absensi menjadi prioritas ke-2 dengan nilai bobot 23%, kemudian kerjasama menjadi prioritas ke-3 dengan nilai bobot 18%, selanjutnya tanggung jawab menjadi prioritas ke-4 dengan nilai bobot 11% dan yang terakhir kedisiplinan menjadi prioritas ke-5 dengan nilai bobot 6%

- Dari hasil pembobotan kriteria, dapat digunakan untuk penentu prangkingan dengan menggunakan metode TOPSIS. Hal ini cukup membantu untuk penilaian kinerja karyawan dan hasil yang didapatkan dari metode AHP dan TOPSIS, yaitu Derry, Usia 24 Tahun dengan nilai preferensi 0,58762.
- Metode AHP dan TOPSIS merupakan metode yang cocok digunakan untuk mengambil sebuah keputusan, bukan hanya penilaian kinerja karyawan saja namun juga untuk pemilihan lainnya. Hipotesa yang diterima adalah H_0

DAFTAR PUSTAKA

- [1] Anto, A. G., Mustafidah, H., & Suyadi, A. (2015). Sistem Pendukung Keputusan Penilaian Kinerja Karyawan Menggunakan Metode SAW (Simple Additive Weighting) di Universitas Muhammadiyah Purwokerto (Decision Support System of Human Resources Performance Assessment Using SAW (Simple Additive Weighting) M, III(November), 193–200.
- [2] Busro, M. (2018). TEORI-TEORI MANAJEMEN SUMBER DAYA MANUSIA (pertama). Jakarta.
- [3] Candra, H. K., Farmadi, A., & Khairunnisa. (2015). PENERAPAN METODE AHP TOPSIS PADA SISTEM PENDUKUNG KEPUTUSAN PENENTUAN TAMAN, 02(01), 1–10.
- [4] Chamid, A. A., & Murti, A. C. (2017). Fakultas Teknik – Universitas Muria Kudus 115, 115–119.
- [5] Desyanti. (2016). MENGGUNAKAN METODE ANALITICITY HIERARCY PROCESS (AHP), 1(1).
- [6] Djunaedi, A., Subiyakto, A., & Fetrina, E. (2017). SISTEM PENDUKUNG KEPUTUSAN PENILAIAN KINERJA PEGAWAI (Studi Kasus : PT . PLN (Persero Distribusi Jakarta Raya Area Pondok Gede)), 10(1), 37–44].
- [7] Friyadie. (2018). Metode ahp sebagai penunjang keputusan untuk penilaian kinerja kerja karyawan spbu, 15(1), 63–68.
- [8] Mardiana, T. (2018). SISTEM PENDUKUNG KEPUTUSAN PENERIMAAN ASISTEN LABORATORIUM KOMPUTER MENGGUNAKAN METODE AHP-TOPSIS, 3(2), Bandung 159–166.
- [9] Mude, M. A. (2016). PADA KASUS UMKM, 8(Agustus), 76–81.
- [10] Noermijati. (2013). Kajian Tentang AKTUALISASI TEORI HERZBERG, KEPUASAN KERJA DAN KINERJA SPRITUAL MANAJER OPERASIONAL (Pertama). Malang.
- [11] Nofriansyah, D., & Defit, S. (2017). MULTI CRITERIA DECISION MAKING (MCDM) pada Sistem Pendukung Keputusan (pertama). yogyakarta.
- [12] Nurdianto, H., & Meilia, H. (2016). SISTEM PENDUKUNG KEPUTUSAN PENENTUAN PRIORITAS PENGEMBANGAN INDUSTRI KECIL DAN MENENGAH DI LAMPUNG TENGAH MENGGUNAKAN

- ANALITICAL HIERARCHY PROCESS (AHP), Yogyakarta, 6-7.
- [13] Undang-undang Republik Indonesia Nomor 13 Tahun (2003).
- [14] Rizky, U. D. (2018). PEMBERIAN REWARD BERDASARKAN PENILAIAN KINERJA KARYAWAN DENGAN METODE AHP PADA PT.ANUGERAH PROTECINDO (2), 181-188.
- [15] Saefudin, & Wahyuningsih, S. (2014). Sistem Pendukung Keputusan Untuk Penilaian Kinerja Pegawai Menggunakan Metode Analytical Hierarchy Process (Ahp) Pada RSUD Serang, (1), 33-37.
- [16] Sari, F. (2018). METODE DALAM PENGAMBILAN KEPUTUSAN. Yogyakarta.
- [17] Taufik, Pratomo, P. A., Ipnuwati, S., & Wulandari. (2018). Implementasi Multi Attribute Decision Making Metode AHP dan TOPSIS untuk Menunjang Keputusan dalam Hal Penerimaan Pekerja (Studi Kasus: PT . Bank Rakyat Indonesia (Persero), 418-431.
- [18] Yusnita, A., Salmon, & Helmi, R. (2015). SISTEM PENDUKUNG KEPUTUSAN PENILAIAN KINERJA KARYAWAN MENGGUNAKAN METODE TOPSIS (TECHNIQUE FOR OTHERS REFERENCE BY SIMILARITY TO IDEAL SOLUTION) PADA PT . RIO UTAMA SAMARINDA, 1(Senatkom), 84-89.
- [19] Yunus, E. (2016). Manajemen Strategis. (A. A. Charitsman, Ed.). Yogyakarta.
- [20] Yusnita, A., Salmon, & Helmi, R. (2015). SISTEM PENDUKUNG KEPUTUSAN PENILAIAN KINERJA KARYAWAN MENGGUNAKAN METODE TOPSIS (TECHNIQUE FOR OTHERS REFERENCE BY SIMILARITY TO IDEAL SOLUTION) PADA PT . RIO UTAMA SAMARINDA, 1(Senatkom), 84-89.

Rancang Bangun Prototipe Sistem Multilateration 2-Dimensi Menggunakan Data ADS-B

I Made Astawa¹, Siswayudi Azhari², I Putu Ananta Yogiswara³, Widrianto Sih Pinastiko⁴
 Pusat Teknologi Elektronika, Badan Pengkajian dan Penerapan Teknologi (BPPT)
 Gedung Teknologi 3, Kawasan PUSPIPTEK, Serpong 15314.

¹made.astawa@bppt.go.id, ²siswayudi.azhari@bppt.go.id, ³putu.ananta@bppt.go.id, ⁴widrianto.sih@bppt.go.id

Diterima 3 November 2019

Disetujui 20 Desember 2019

Abstract—Multilateration (MLAT) is a technology used to determine the position of an object by using a Time Difference of Arrival (TDOA). MLAT is used at airports as one of the sensors in the A-SMGCS system to determine the position of an aircraft or vehicle. Developing an MLAT system should ideally be done at the airport because it requires aircraft and vehicle objects that are already equipped with ADS-B transmitters, but developing an MLAT system at the airport requires high costs for equipment installation and licensing because it risks disrupting airport operations and can disrupt existing systems. The Center of Technology for Electronics - BPPT is developing the A-SMGCS system and is based in the serpong puspipstek area, where this area has a large enough area, making it possible to design prototypes of the MLAT system. Three ADS-B receivers are placed in three different buildings that are still connected to a computer network, so that data collection from each receiver can be done through the BPPT local network. The object to be calculated is using a car equipped with an ADS-B transmitter. To get more accurate results, every time the receiver is synchronized using GPS. The experimental results show that there is a difference between the position data sent by the ADS-B transmitter and the results of the MLAT calculation, this is due to the difference in height of the three receivers, accuracy of GPS and accuracy of the ADS-B receiver timestamp used. Although there are differences, the proof of concept design of this system is in accordance with the characteristics of the MLAT system. The benefit gained from the design of this prototype is that it can develop an MLAT system in a place that is not high risk with low cost but the results can be applied at the airport.

Keywords : A-DSB, A-SMGCS, MLAT, Multilateration, TDOA

I. PENDAHULUAN

Pusat Teknologi Elektronika (PTE) – BPPT sedang mengembangkan sistem A-SMGCS (*Advanced-Surface Movement Guidance and Control System*). A-SMGCS adalah sistem di bandara yang digunakan untuk meningkatkan efisiensi penggunaan landasan pacu, apron dan taxiway dengan memaksimalkan kapasitas landasan pacu dalam segala situasi dan cuaca dengan tetap mempertahankan tingkat keamanan yang dipersyaratkan. Pada sistem A-SMGCS terdapat beberapa sensor untuk menentukan lokasi pesawat atau

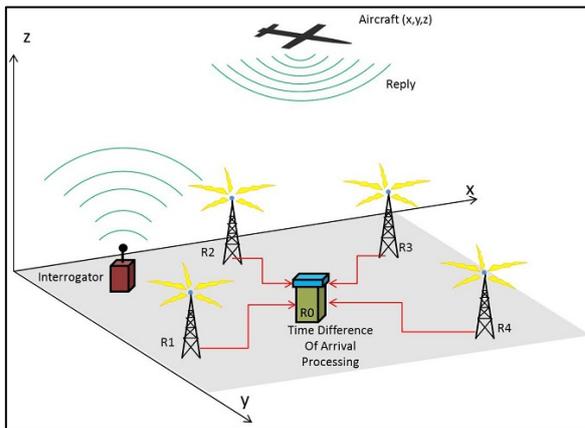
kendaraan dan salah satu diantaranya adalah sensor *Multilateration (MLAT)*.

Multilateration adalah teknologi yang telah digunakan selama beberapa dekade untuk menentukan posisi target di bandara secara akurat dengan menggunakan perbedaan waktu tiba yang dikenal dengan *Time Difference of Arrival (TDOA)*. Target adalah pesawat terbang, kendaraan atau rintangan (*obstacles*), apakah diam atau bergerak, yang berada di area bandara. Multilateration menggunakan sejumlah receiver yang ditempatkan di darat pada lokasi strategis di sekitar bandara agar dapat menerima sebanyak mungkin target yang ada bandara dan wilayah udara sekitarnya. Karena masing-masing target akan berada pada jarak yang berbeda dari ADS-B receiver, maka data yang dikirim oleh target akan diterima oleh setiap receiver pada waktu yang berbeda seperti gambar 1. Dengan menggunakan model matematika dan teknik pemrosesan data di komputer, perbedaan waktu ini memungkinkan posisi target dapat dihitung secara tepat. Estimasi posisi target dilakukan dalam dua tahap, dan tahap pertama menghitung perbedaan waktu kedatangan (TDOA) dan tahap kedua nilai TDOA ditransformasikan menjadi perbedaan jarak antara dua *receiver* yang akan menghasilkan sejumlah persamaan hiperbolik non-linear. Pada tahap kedua ini perlu menggunakan algoritma yang efisien untuk menyelesaikan persamaan hiperbolik tersebut. Banyak metode yang sudah dikembangkan untuk menyelesaikan model matematika sistem MLAT.

Untuk membuat sistem MLAT yang akan digunakan sebagai salah satu sensor pada sistem A-SMGCS sebaiknya langsung dilakukan di bandara agar antara target dan receiver tidak ada yang menghalangi (*line of sight*). Tetapi untuk melakukan percobaan langsung di bandara akan menemui banyak kendala seperti biaya instalasi perangkat yang cukup tinggi, harus memperhatikan faktor keamanan seperti dapat mengganggu operasional bandara, dapat mengganggu sistem yang sudah ada, dan akan memerlukan waktu yang lebih lama karena perlu proses perijinan.

Tujuan dari rancang bangun prototipe sistem multilateration ini adalah untuk mengembangkan

sistem MLAT seperti gambar 1 dengan biaya yang rendah dan tidak mengganggu operasional bandara dan tidak mengganggu sistem yang sudah terpasang di bandara. Rancang bangun prototipe sistem MLAT ini dibuat dikawasan puspipstek serpong, dimana pada kawasan ini terdapat banyak gedung perkantoran, laboratorium dan pepohonan yang bisa mengganggu propagasi sinyal transmitter ke receiver. Walaupun banyak gangguan secara *proof of concept (POC)* rancang bangun sistem MLAT ini akan bermanfaat untuk pengembangan selanjutnya di lokasi yang sebenarnya yaitu bandara.



Gambar 1. Sistem MLAT [1]

II. TINJAUAN PUSTAKA

A. Model Matematika

Pada gambar 2, misalkan x, y adalah koordinat sebenarnya dari posisi target dan xi, yi adalah koordinat receiver yang ke-i. Jarak antara target dengan receiver dapat dihitung dengan rumus sbb :

$$R_i = \sqrt{(x_i - x)^2 + (y_i - y)^2}$$

$$R_i^2 = \sqrt{x_i^2 + y_i^2 - 2x_ix - 2y_iy + x^2 + y^2} \dots\dots\dots (1)$$

Untuk perhitungan posisi 2-dimensi diperlukan 3 buah receiver (i = 1, 2, 3). Jarak receiver ke-i dengan target adalah :

$$R_i = (t_i - t) \cdot c$$

dimana :
 c = kecepatan propagasi signal = 3 x 10⁵ Km/sec
 t = waktu data dikirim oleh target
 t_i = waktu data diterima oleh receiver ke-i
 Perbedaan jarak antara R₁ dengan R_i adalah sbb:
 R_i - R₁ = ((t_i - t) . c) - ((t₁ - t) . c) = (t_i - t₁) . c
 (t_i - t₁) = disebut dengan TDOA (Time Difference Of Arrival). Jika R₁ digunakan sebagai referensi maka dapat dituliskan sbb :

R_i - R₁ ditulis

$$R_{i,1} = \sqrt{(x_i - x)^2 + (y_i - y)^2} - \sqrt{(x_1 - x)^2 + (y_1 - y)^2}$$

..... (2)

$$R_i^2 = (R_i - R_1 + R_1)^2 = (R_{i,1} + R_1)^2 \dots\dots\dots(3)$$

Dari persamaan (1) didapat :

$$R_i^2 = x_i^2 + y_i^2 - 2x_ix - 2y_iy + x^2 + y^2 \dots\dots\dots (4)$$

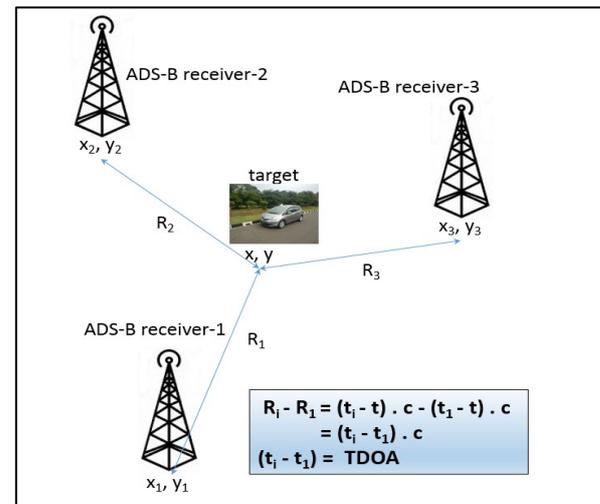
Dari persamaan (3) dan (4) didapat :

$$R_{i,1}^2 + 2R_{i,1}R_1 + R_1^2 = x_i^2 + y_i^2 - 2x_ix - 2y_iy + x^2 + y^2 \dots\dots\dots (5)$$

Dari persamaan (4) untuk i=1, dimasukkan ke persamaan (5) didapat :

$$R_{i,1}^2 + 2R_{i,1}R_1 = x_i^2 + y_i^2 - 2x_{i,1}x - 2y_{i,1}y - x_1^2 - y_1^2 \dots\dots\dots (6)$$

Memecahkan persamaan nonlinier seperti diatas adalah cukup sulit, oleh karena itu perlu dilakukan linearisasi dari sejumlah persamaan. Salah satu cara nya dengan mentransformasikan sejumlah persamaan nonlinier (2) menjadi beberapa persamaan lainnya.



Gambar 2. TDOA

B. Metode Chan

Dari 3 buah receiver akan menghasilkan 2 TDOA, maka dengan metode Chan nilai x dan y dapat diselesaikan sebagai berikut : [2]

Dari persamaan (6) untuk i=2 dan i=3 didapat :

$$R_{2,1}^2 + 2R_{2,1}R_1 = x_2^2 + y_2^2 - 2x_{2,1}x - 2y_{2,1}y - x_1^2 - y_1^2$$

$$R_{3,1}^2 + 2R_{3,1}R_1 = x_3^2 + y_3^2 - 2x_{3,1}x - 2y_{3,1}y - x_1^2 - y_1^2$$

jika :
 K₁ = x₁² + y₁² dan K₂ = x₂² + y₂² dan K₃ = x₃² + y₃²
 Maka :

$$x_{2,1}x + y_{2,1}y = -R_{2,1}R_1 - \frac{1}{2}(R_{2,1}^2 + K_1 - K_2)$$

$$x_{3,1}x + y_{3,1}y = -R_{3,1}R_1 - \frac{1}{2}(R_{3,1}^2 + K_1 - K_3)$$

$$\begin{bmatrix} x_{2,1} & y_{2,1} \\ x_{3,1} & y_{3,1} \end{bmatrix} \begin{bmatrix} x \\ y \end{bmatrix} = \begin{bmatrix} -R_{2,1}R_1 - \frac{1}{2}(R_{2,1}^2 + K_1 - K_2) \\ -R_{3,1}R_1 - \frac{1}{2}(R_{3,1}^2 + K_1 - K_3) \end{bmatrix} \dots\dots(7)$$

Persamaan (7) akan menghasilkan nilai x dan y sebagai fungsi dari R_1 . Kemudian masukkan nilai x, y ke persamaan (1) untuk $i=1$, maka akan di dapat persamaan kwadrat dalam bentuk :

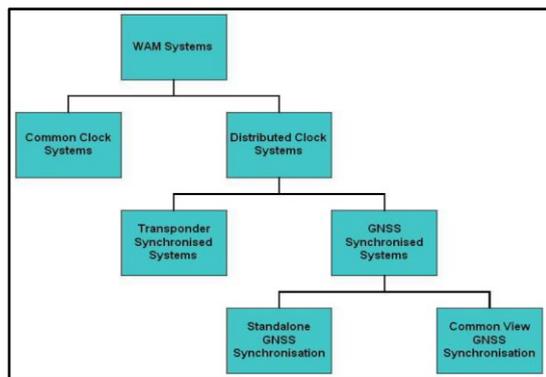
$$aR_1^2 + bR_1 + c = 0 \dots\dots\dots (8)$$

akan diperoleh dua nilai R_1 , karena R_1 adalah jarak antara receiver-1 dengan target maka nilai R_1 yang digunakan adalah yang positif.

C. Sinkronisasi Waktu

Sinkronisasi waktu pada unit receiver sangat penting untuk sistem TDOA, meskipun metode penerapan sinkronisasi umumnya berbeda. Untuk menghitung posisi perlu diketahui perbedaan waktu dari sinyal yang tiba di satu antena receiver dengan antena receiver yang lainnya. Sinyal akan di timestamp selama proses digitalisasi sehingga akan ada penundaan waktu relative terhadap kedatangan di antena. Oleh karena itu untuk menghitung TDOA secara akurat penundaan ini harus diketahui dengan tepat dan diperhitungkan.

Diagram gambar 3 di bawah ini menunjukkan topologi dari berbagai metode sinkronisasi yang digunakan pada sistem WAM (Wide Area MLAT), yang diperlukan untuk menghitung TDOA.



Gambar 3. Metode Sinkronisasi [3]

D. Proyeksi Peta

Pengertian dari proyeksi peta adalah cara memindahkan garis lintang dan garis bujur berbentuk bola atau globe ke bidang datar. Hasil pemindahan dari globe ke bidang datar ini akan menjadi peta. [4]

Proyeksi mercator merupakan proyeksi silinder, dimana seluruh muka bumi dilukiskan pada bidang silinder yang sumbu nya berimpit dengan bola bumi, kemudian silindernya dibuka menjadi bidang datar. Proyeksi mercator sangat baik untuk menggambarkan daerah equator. Kondisi geografi negara Indonesia yang membujur di sekitar garis khatulistiwa atau garis lingkaran equator dari barat sampai ke timur, menjadikan sistem proyeksi mercator yang paling ideal karena memberikan hasil dengan distorsi minimal [5].

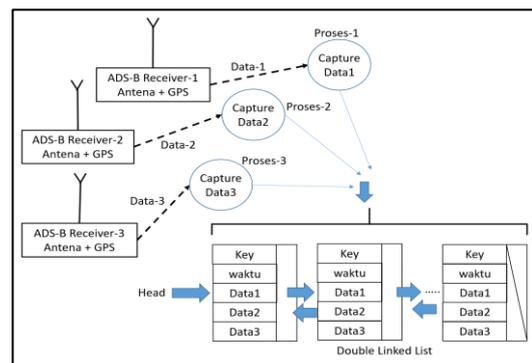
III. METODOLOGI DAN PERANCANGAN

Perancangan sistem MLAT dilakukan dengan menempatkan 3 buah receiver yang membentuk segitiga dengan jarak antara 1 s/d 2 Km. Lokasi setiap receiver (latitude dan longitude) ditentukan dengan GPS di proyeksikan ke koordinat kartesian (x, y). Semua receiver dihubungkan dengan local area network (LAN). Salah satu receiver digunakan sebagai referensi untuk menentukan TDOA (*Time Difference Of Arrival*). Sebuah program aplikasi dikembangkan untuk mengumpulkan data dan memproses data tersebut untuk mendapatkan TDOA dan selanjutnya menggunakan TDOA untuk menghitung posisi target.

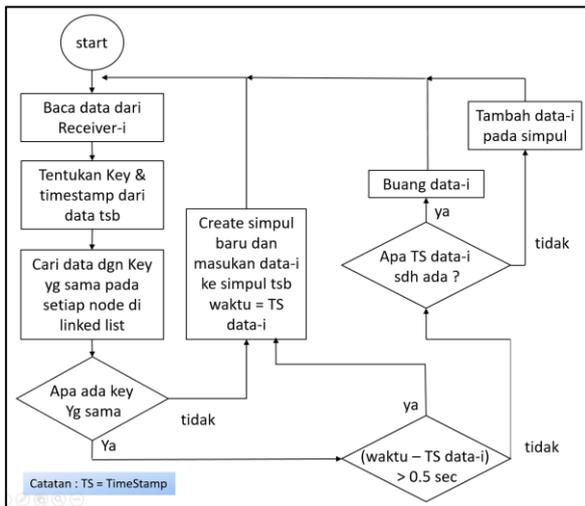
A. Sinkronisasi Waktu dan Pengumpulan Data

Metode sinkronisasi waktu di setiap receiver menggunakan GPS yang dipasang di setiap receiver (standalone GNSS synchronization). Metode pengumpulan data seperti pada gambar 4, dimana data dikumpulkan pada satu buffer dengan struktur data double linked list. Tiga proses (Capture Data1 s/d Capture Data3) akan mengambil data secara real time pada 3 receiver. Setiap data harus ditentukan kuncinya (key) untuk membedakan data yang satu dengan data yang lainnya. Yang dijadikan key disetiap data pada rancangan ini adalah gabungan antara data dan timestamp. Flowchart untuk proses Capture Data dapat dilihat pada gambar 5.

Setiap proses akan membaca data dari ADS-B receiver dan menentukan key-nya, kemudian melakukan pencarian apakah ada data dengan key yang sama pada struktur data double linked list tersebut, jika tidak ada maka data tersebut akan ditambahkan sebagai simpul baru. Jika data ada dengan key yang sama maka bandingkan waktu yang ada pada simpul dengan timestamp. Jika selisih (timestamp - waktu) < 0.5 second maka data tersebut dimasukkan ke dalam simpul, jika tidak tambahkan data pada linked list sebagai simpul baru. Update rate data ADS-B paling cepat 0.5 second, sehingga kalau selisih datanya lebih kecil dari 0.5 second berarti data tersebut adalah berpasangan dan siap untuk dicari TDOA nya.



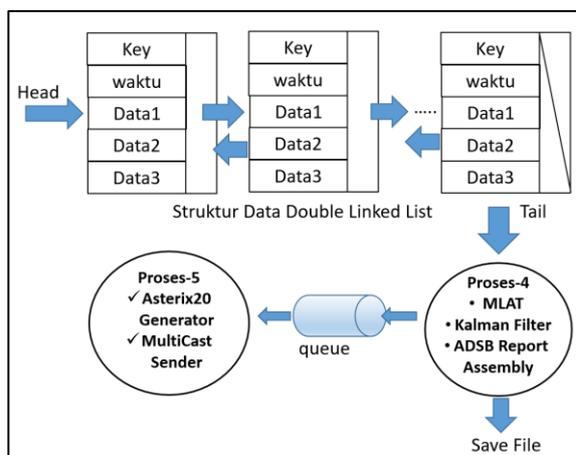
Gambar 4. Rancangan Sistem untuk Pengumpulan Data



Gambar 5. Flowchart pada proses Capture Data

B. Perhitungan Posisi Target

Perancangan perhitungan untuk menentukan posisi target seperti gambar 6, yang terdiri dari 2 buah proses, yaitu Proses-4 dan Proses-5. Proses-4 akan menghitung posisi berdasarkan TDOA menggunakan metode Chan dan hasilnya di filter terlebih dahulu dengan Kalman Filter untuk mendapatkan hasil yang lebih baik. Disamping itu pada Proses-4 ini dilakukan *ADSB Report Assembly* yaitu proses untuk menentukan posisi berdasarkan GPS yang ada pada ADS-B Transmitter. Data posisi berdasarkan TDOA dan data posisi berdasarkan GPS inilah yang akan dibandingkan dan dianalisis pada penelitian ini.



Gambar 6. Rancangan Sistem untuk Proses MLAT

Pada Proses-5 dilakukan pembuatan data dengan format Asterix-20, kemudian data tersebut akan dikirim secara MultiCast ke jaringan komputer. Format data Asterix-20 adalah standard dari keluaran sistem MLAT. Komunikasi ke tiga proses ini menggunakan antrian (Queue). Proses-4 mengambil data dari bagian Tail double linked list, dan jika data1, data2 dan data3

pada setiap simpul lengkap maka perhitungan posisi target dapat diproses dan jika data tidak lengkap maka perhitungan posisi target tidak bisa diproses dan data tersebut segera dibuang dan dilanjutkan untuk memproses data berikutnya. Untuk data target selain posisi seperti ICAO Address, Call Sign (ID), Track Angle, Ground Speed dan lain-lain, diperoleh dari proses *ADSB Report Assembly*.

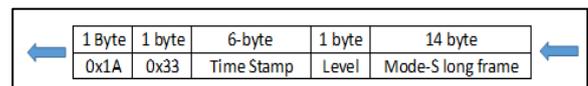
IV. IMPLEMENTASI DAN UJICOBA SISTEM

A. Receiver

Receiver yang digunakan pada penelitian ini adalah ADS-B receiver merk Radarcape seperti gambar 7. Banyak tipe format data yang dikeluarkan oleh ADS-B receiver ini dan yang digunakan pada ujicoba adalah tipe *mode-s long frame* untuk *surface position* seperti gambar 8. Data mode-s long frame ini diawali dengan $(1A)_{16}$ dan $(33)_{16}$ kemudian diikuti dengan timestamp 6 byte. Waktu saat receiver menerima data dari Transmitter disimpan pada timestamp ini yang satuannya dalam nanosecond.



Gambar 7. ADS-B receiver Radarcape



Gambar 8. Format data dari ADS-B receiver Radarcape

B. Transmitter

Transmitter yang digunakan pada penelitian ini adalah ADS-B Transmitter merk Becker seperti gambar 9, unit transmitter ini dipasang diatas kendaraan dan secara periodik akan mengirimkan datanya secara broadcast yang akan diterima oleh ADS-B receiver radarcape.



Gambar 9. ADS-B Transmitter untuk Ground Vehicles

C. Implementasi

Implementasi dilakukan di kawasan Puspiptek Serpong dengan 3 buah ADS-B receiver ditempatkan di 3 tempat seperti gambar 10 dengan koordinat lokasi seperti table-1. Untuk menyelesaikan persamaan MLAT diperlukan koordinat Cartesian, oleh karena itu diperlukan konversi atau proyeksi dari koordinat geodetic ke koordinat Cartesian dan pada penelitian ini menggunakan proyeksi Mercator. Pembuatan program untuk prototipe sistem MLAT ini menggunakan bahasa pemrograman Java. Semua data dari receiver dikumpulkan dan diproses pada komputer server yang berada di gedung teknologi-3 melalui jaringan LAN - BPPT.



Gambar 10. Posisi Receiver dan Target saat ujicoba di kawasan Puspiptek Serpong

D. Implementasi Metode Chan untuk MLAT 2-Dimensi

Agar bisa dibuat programnya maka nilai a, b dan c pada persamaan (8) diatas harus dicari terlebih dahulu. Nilai a, b dan c dapat dihitung sebagai berikut :

$$D = \begin{bmatrix} x_{2,1} & y_{2,1} \\ x_{3,1} & y_{3,1} \end{bmatrix}$$

$$D_x = \begin{bmatrix} (-R_{2,1}R_1 - \frac{1}{2}(R_{2,1}^2 + K_1 - K_2)) & y_{2,1} \\ (-R_{3,1}R_1 - \frac{1}{2}(R_{3,1}^2 + K_1 - K_3)) & y_{3,1} \end{bmatrix}$$

$$D_y = \begin{bmatrix} x_{2,1} & (-R_{2,1}R_1 - \frac{1}{2}(R_{2,1}^2 + K_1 - K_2)) \\ x_{3,1} & (-R_{3,1}R_1 - \frac{1}{2}(R_{3,1}^2 + K_1 - K_3)) \end{bmatrix}$$

Maka nilai x dan y dapat dihitung :

$$x = \frac{|D_x|}{|D|}, \text{ dan } y = \frac{|D_y|}{|D|} \dots\dots\dots (9)$$

Masukkan nilai x, y ke persamaan (1) untuk i=1, maka di dapat :

$$a = 1 - \frac{F^2}{|D|} - \frac{H^2}{|D|}$$

$$b = \frac{2GH}{|D|^2} - \frac{2EF}{|D|^2} + \frac{2y_1H}{|D|} + \frac{2x_1F}{|D|}$$

$$c = -x_1^2 - y_1^2 + \frac{2x_1E}{|D|} + \frac{2y_1G}{|D|} - \frac{E^2}{|D|^2} - \frac{G^2}{|D|^2}$$

Dimana :

$$A = -\frac{1}{2}(R_{2,1}^2 + K_1 - K_2)$$

$$B = -\frac{1}{2}(R_{3,1}^2 + K_1 - K_3)$$

$$E = y_{3,1}A - y_{2,1}B$$

$$F = y_{3,1}R_{2,1} - R_{3,1}y_{2,1}$$

$$G = x_{2,1}B - x_{3,1}A$$

$$H = x_{2,1}R_{3,1} - x_{3,1}R_{2,1}$$

Pada persamaan (8) akan didapat dua nilai R₁, karena nilai ini menyatakan jarak maka dipilih nilai R₁ yang positif. Dengan diketahui nilai R₁ maka nilai x dan y bisa ditentukan dari persamaan (9) dimana :

$$|D_x| = E - F \cdot R_1$$

$$|D_y| = G - H \cdot R_1$$

$$|D| = (x_{2,1} \cdot y_{3,1}) - (x_{3,1} \cdot y_{2,1})$$

Tabel 1. Koordinat Geodetic di proyeksikan ke koordinat Cartesian dengan Proyeksi Mercator.

No	Lokasi / Gedung	Posisi (Geodetic)	Posisi (Cartesian)
1	Teknologi 3 (receiver-1)	Lat : -6.34574 Lon : 106.661 Alt : 83.7835 m	X ₁ = 683728.185 Y ₁ = 9298281.905
2	Manajemen (receiver-3)	Lat : -6.35558 Lon : 106.671 Alt : 87.1504 m	X ₂ = 684831.110 Y ₂ = 9297190.145
3	Sentra Teknologi Polimer (STP) (receiver-2)	Lat : -6.35269 Lon : 106.659 Alt : 77.0925 m	X ₃ = 683504.436 Y ₃ = 9297514.020

E. Hasil Uji Coba

Ujicoba dilakukan dengan menempatkan ADS-B transmitter diatas mobil seperti gambar 11. Pengujian hanya dilakukan pada saat mobil lagi diam di lokasi dimana signal dari ADS-B transmitternya bisa diterima oleh ketiga ADS-B receiver, hal ini sebabkan karena dikawasan puspiptek banyak gedung dan pepohonan, yang bisa mengganggu propagasi signal ADS-B. Pengujian dilakukan selama 15 menit dan diperoleh data sebanyak 96 data. Pada posisi diam ADS-B transmitter hanya mengirim data setiap 5 detik. Masih banyak data yang tidak bisa diproses karena datanya tidak bisa diterima di 3 receiver.

Data yang disimpan pada saat pengujian adalah data posisi (latitude dan longitude) berdasarkan GPS ADS-B Transmitter dan berdasarkan perhitungan MLAT seperti pada gambar 12. Kemudian data ini disajikan dalam bentuk Peta dengan skala 1:8000 dan

1:500 seperti gambar 13 dan 14. Pada gambar 14 jarak terjauh antara dua lokasi adalah 59.64 meter. Untuk melihat perbedaan data ADS-B, MLAT dan data keluaran kalman filter disajikan dalam bentuk grafik seperti gambar 15 dan gambar 16. Perbedaan jarak antara posisi yang dihasilkan oleh MLAT, Kalman Filter dan ADS-B disajikan pada gambar 17.

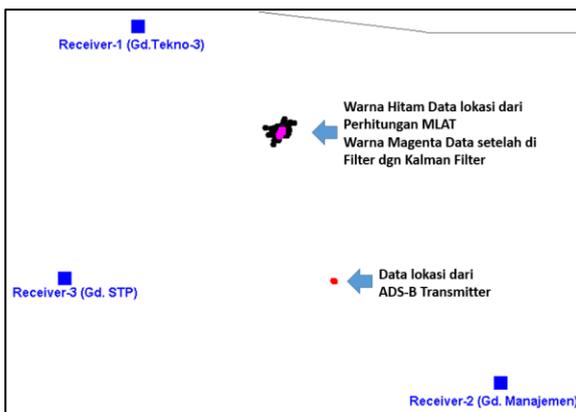


Gambar 11. ADS-B Transmitter dipasang diatas Mobil

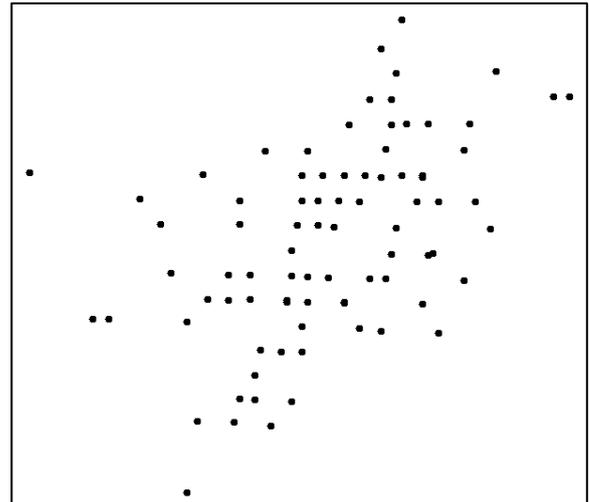
No.	ADS-B		Timestamp (second)	MLAT	
	Latitude	Longitude		Latitude	Longitude
1	-6.35276413	106.6664111	1562902753	-6.348791969	106.664914
2	-6.35276413	106.6664111	1562902763	-6.348719354	106.6649815
3	-6.352762772	106.6664018	1562902768	-6.348715104	106.6648387
4	-6.35276413	106.6663994	1562902773	-6.34889892	106.6648715
-	-6.352762772	106.6664018	1562902778	-6.348572955	106.6649714
96	-6.35276413	106.6663994	1562902783	-6.348610638	106.6649647

Gambar 12. Contoh Data Hasil Pengujian

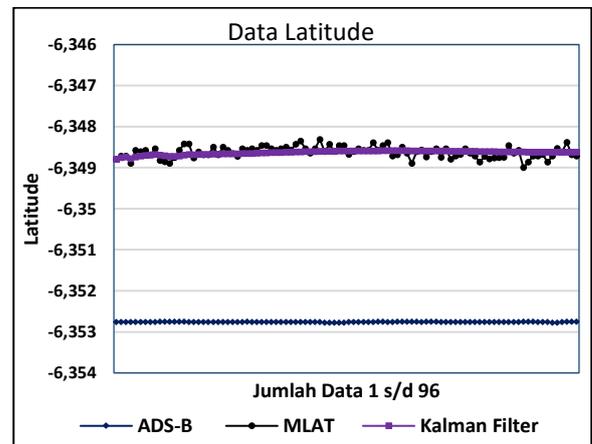
Untuk data hasil perhitungan MLAT (gambar 14) jarak terjauh antara dua posisi adalah 59.64 meter dan jarak terdekat adalah 0 meter, dan rata-rata nya sama dengan 25.56 meter. Untuk gambar 17 jarak antara data lokasi ADS-B dengan perhitungan MLAT adalah 517.82 meter, dan terdekat adalah 456.39 meter dan rata-rata 482.64 meter. Sedangkan perbedaan jarak data lokasi ADS-B dengan Keluaran Kalman Filter terjauh adalah 491.05 meter dan terdekat adalah 472.90 meter dan rata-rata adalah 480.64 meter.



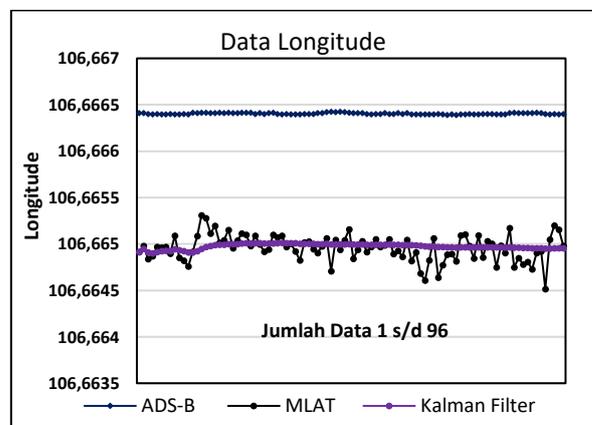
Gambar 13. Hasil perhitungan posisi MLAT pada skala 1:8000



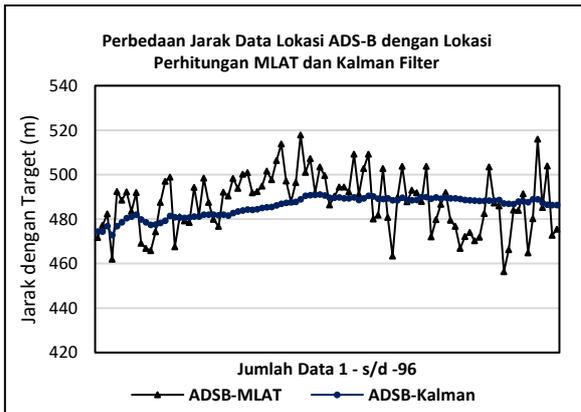
Gambar 14. Perhitungan posisi MLAT digambar pada skala 1:500



Gambar 15. Data Latitude dari ADS-B Transmitter, MLAT dan Kalman Filter



Gambar 16. Data Longitude dari ADS-B Transmitter, MLAT dan Kalman Filter



Gambar 17. Selisih jarak antara ADS-B – MLAT dan ADS-B – Kalman Filter

V. KESIMPULAN

Dari hasil pengujian terhadap rancang bangun prototipe sistem MLAT dapat disimpulkan sbb:

1. Perbedaan posisi antara data ADS-B dengan MLAT seperti yang ditunjukkan oleh gambar 15,16 dan 17 dapat disebabkan karena perbedaan ketinggian dari ke tiga receiver dan juga akurasi dari GPS itu sendiri. Perhitungan MLAT menggunakan model matematika 2-dimensi yang tidak memperhitungkan faktor ketinggian.
2. ADS-B receiver radarcape memiliki akurasi timestamp ± 50 nano second [6] yang secara teori jarak terjauh antara 2 posisi bisa mencapai 44.5 meter. Pada rancang bangun prototipe sistem MLAT ini jarak terjauhnya mencapai 59.64 meter. Walaupun ada perbedaan tetapi secara proof of concept (POC) rancang bangun sistem ini sudah sesuai dengan karakteristik sistem MLAT seperti ditunjukkan pada gambar 13 dan 14.
3. Tahap selanjutnya perlu penyempurnaan terutama pada pemilihan ADS-B receiver yang memiliki akurasi timestamp yang lebih baik dan pengukuran posisi ke-3 receiver sebaiknya menggunakan GPS dengan akurasi yang tinggi.

DAFTAR PUSTAKA

- [1] How WAM Works Diagram [Online] Tersedia dalam : https://www.pagerpower.com/news/wide-area-multilateration-wam/how_wam_works_diagram-png/ [Diakses 2 -9- 2019]
- [2] Jacek Stefanski. "Hyperbolic Position Location Estimation in the Multipath Propagation Environment", IFIP Wireless and Mobile Networking Conference, 2009.
- [3] Wide Area Multilateration, Report on EATMP TRS 131/04 Version 1.1 [Online] Tersedia dalam: https://pdfs.semanticscholar.org/6b06/9be56e15_5b32cec318a30f49a83150f5e372.pdf [Diakses 20 -9- 2019]
- [4] Proyeksi Peta [Online] Tersedia dalam : <https://aristyakristina.wordpress.com/2012/09/16/proyeksi-peta/> [Diakses 24 - 9 - 2019]
- [5] Sistem Proyeksi Peta [Online] Tersedia dalam : <http://adhipakumpulantugas.blogspot.com/2017/02/sistem-proyeksi-peta.html> [Diakses 24-9- 2019]
- [6] Radarcape Mode-S Beast miniADSB [Online] Tersedia dalam : <https://manualzz.com/doc/32242631/radarcape-mode-s-beast-miniadsb---picadsb9> [Diakses 24 September 2019]
- [7] Chan Y. T., Ho K. C., "A Simple and Efficient Estimator for Hyperbolic Location", IEEE Trans. on Signal Proc., vol. 42, no. 8, 1994, pp. 1905-1915
- [8] Foy W. H., "Position-Location Solutions by Taylor-Series Estimation", IEEE Trans. On Aero. and Elec. Systems, vol. AES-12, no. 2, 1976, pp. 187-194
- [9] Fang B. T., "Simple Solution for Hyperbolic and Related Position Fixes", IEEE Trans. on Aero. and Elec. Systems, vol. 26, no. 5, 1990, pp. 748-753
- [10] Friedlander B., "A Passive Localization Algorithm and Its Accuracy Analysis", IEEE Jour. of Oceanic Engineer., vol. OE-12, no. 1, 1987, pp. 234-245

Perancangan Teknik Kriptografi *Block Cipher* Berbasis Pola Tarian Sajojo Papua

Dwayne Jeremy Euagellino Prihanto¹, Magdalena A. Ineke Pakereng, S.Kom.²

^{1,2}Teknik Informatika, Universitas Kristen Satya Wacana

Jl.Dr. O. Notohamidjojo, Salatiga 50714, Indonesia

¹dwaynejermaine96@gmail.com, ²ineke.pakereng@staff.uksw.edu

Diterima 16 Oktober 2019

Disetujui 20 Desember 2019

Abstract — *Cryptography is a science to maintain the security and confidentiality of an information. In this research we designed Block Cipher Cryptography 64 bit Based on Tarian Sajojo Papua Pattern to build new cryptography. In this critique is designed with 10 rounds, where each round there are 4 processes. In each round there are 4 patterns for the plaintext process and 4 patterns for the key process. In second and fourth process is transformed with S-BOX table to get a more random ciphertext. Testing is also done using Avalanche Effect and Correlation value where the average character change reaches 49,69%, so it can be used as an alternative in securing data.*

Index — *Kriptografi adalah suatu ilmu untuk menjaga keamanan dan kerahasiaan suatu informasi. Dalam penelitian ini dirancang Kriptografi Block Cipher 64 bit Berbasis Pola Sajojo Papua guna membangun kriptografi baru. Dalam kriptografi ini dirancang dengan 10 putaran, dimana setiap putaran terdapat 4 proses. Pada setiap putaran terdapat 4 pola untuk proses plaintext dan 4 pola untuk proses kunci. Di proses kedua dan keempat ditransformasikan dengan tabel S-BOX untuk mendapatkan ciphertext yang lebih acak. Pengujian juga dilakukan menggunakan Avalanche Effect dan nilai Korelasi dimana rata-rata perubahan karakter mencapai 49,69% sehingga dapat digunakan sebagai alternatif dalam mengamankan data.*

I. PENDAHULUAN

Enkripsi secara eksplisit dapat diartikan sebagai suatu proses untuk mengubah pesan (informasi) sehingga tidak dapat dilihat tanpa menggunakan kunci pembuka rahasia. Teknologi ini sudah digunakan sejak lama oleh kalangan militer dan intelejen. Saat ini, teknologi enkripsi dengan beberapa modifikasi sudah diaplikasikan untuk kepentingan umum, dalam aktivitas digital seperti merahasiakan data-data penting milik perorangan maupun perusahaan. Hasil statistik dari Breach Level Index (BLI) membuktikan, sepanjang 2016 telah terjadi 1.378.509.261 kehilangan atau pencurian data di seluruh dunia, atau sama dengan 3.776.738 data per hari, dan 157,364 per jam. Dari keseluruhan pelanggaran data di 2016 hanya 4 persen pembobolan data dianggap tidak

berhasil karena data yang dicuri sudah terlebih dulu di enkripsi oleh perusahaan [1].

Maka dari itu, dapat dikatakan bahwa keamanan dalam proses pemindahan informasi sangat diperlukan. IT infrastruktur mulai gencar dalam merancang dan membangun untuk mengamankan informasi. Kriptografi hadir sebagai ilmu untuk menjaga kerahasiaan pesan/mengamankan informasi. Informasi yang dapat dibaca dan dipahami dengan bahasa tertentu diubah ke dalam bentuk sandi tertentu yang berstruktur huruf/kata/kalimat yang susah dipahami dari segi bahasa apapun. Salah satu algoritma nya adalah menggunakan algoritma Kriptografi Block Cipher. Block Cipher menggunakan kumpulan bit dengan panjang tetap yang disebut sebagai block dan kemudian dioperasikan dengan cipher kunci untuk nantinya ditransformasikan. Seiring kemajuan teknologi, makin banyak pula cara untuk memecahkan algoritma ini. Untuk itu salah satu cara untuk membuat data atau informasi menjadi lebih aman adalah dengan membuat pola atau algoritma baru untuk memodifikasi algoritma yang sudah ada.

Penelitian ini merupakan kriptografi Block cipher dengan menggunakan pendekatan pola Tarian Sajojo Papua. Dari pola-pola tersebut akan dicari korelasi terbaik yang kemudian akan digunakan sebagai proses enkripsi dan dekripsi pesan plaintext. Beberapa gerakan-gerakan dalam tarian sajojo papua dijadikan pola pertukaran kode bit di dalamnya. Sehingga kewanaman data menjadi lebih kuat dan data dapat digunakan sebagaimana mestinya.

II. TINJAUAN PUSTAKA

Penelitian sebelumnya berjudul Perancangan Kriptografi Block Cipher Berbasis pada Teknik Formasi Permainan Bola. Penelitian ini membahas mengenai perancangan kriptografi berbasis pada Teknik formasi permainan bola dapat melakukan proses enkripsi dan dekripsi dan telah memenuhi 5-tuple dari kriptosistem [3].

Penelitian sebelumnya berjudul Perancangan Kriptografi *Block Cipher* Berbasis Pola Gerakan Lempeng Tektonik Divergensi dan Konvergensi. Penelitian ini membahas mengenai perancangan

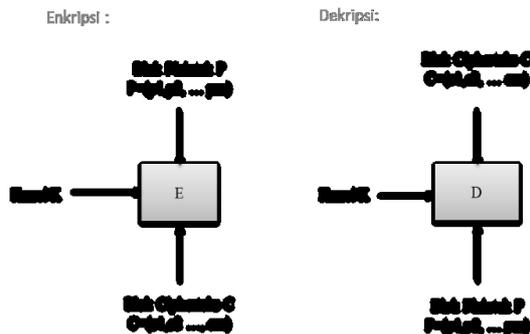
kriptografi *Block Cipher* berbasis pada pola gerakan lempeng tektonik divergensi dan konvergensi, dimana pola divergensi dijadikan dalam pertukaran kode bit pada *plaintext* sedangkan pola konvergensi digunakan pada pertukaran kode bit kunci [5].

Penelitian sebelumnya berjudul Perancangan Kriptografi *Block Cipher* dengan Langkah Permainan Engklek. Penelitian ini membahas mengenai eksperimen perancangan *Block Cipher* untuk diimplementasikan menjadi sebuah aplikasi yang dapat digunakan secara otomatis melakukan enkripsi dan dekripsi, dan penelitian ini menunjukkan bahwa permainan tradisional dari Indonesia juga dapat dijadikan alur algoritma [7].

Penelitian sebelumnya berjudul Perancangan Kriptografi *Block Cipher* Berbasis Pola Formasi Futsal 1-2-1. Penelitian ini membahas mengenai kriptografi *Block Cipher* 256-bit berbasis formasi futsal 1-2-1 dapat menunjukan ciri khas dari sebuah permainan futsal dalam sebuah team sehingga dapat menyembunyikan kerahasiaan data lebih baik [4].

Penelitian sebelumnya berjudul Perancangan Kriptografi *Block Cipher* 256-bit Berbasis pada Game SUDOKU. Penelitian ini membahas mengenai kriptografi *Block Cipher* 256-bit berbasis Game SUDOKU dapat menunjukan bahwa dari Game SUDOKU yang tersembunyi, sehingga tidak banyak orang mengira bahwa dengan Game SUDOKU ada rahasia yang tersimpan [6].

Skema proses enkripsi dan dekripsi *block cipher* secara umum digambarkan pada Gambar 1.



Gambar 1. Skema Proses Enkripsi dan Dekripsi Pada *Block Cipher* [9].

Misalkan blok *plaintext* (P) yang berukuran m bit dinyatakan sebagai

$$P = (p_1, p_2, \dots, p_n) \quad (1)$$

Blok *ciphertext* (C) dinyatakan sebagai

$$C = (c_1, c_2, \dots, c_n) \quad (2)$$

Kunci (K) dinyatakan sebagai

$$K = (k_1, k_2, \dots, k_n) \quad (3)$$

Sehingga proses enkripsi adalah

$$EK(P) = C \quad (4)$$

Dan proses dekripsi adalah

$$DK(C) = P \quad (5)$$

Sebuah sistem kriptografi harus memenuhi lima-tupel (*five-tuple*) (P, C, K, E, D) dengan kondisi [10]:

1. P adalah himpunan berhingga dari *Plaintext*.
2. C adalah himpunan berhingga dari *Ciphertext*.
3. K merupakan ruang kunci (*keyspace*), adalah himpunan berhingga dari kunci.
4. Untuk setiap $k \in K$ terdapat aturan enkripsi $e_k \in E$ dan berkorespondensi dengan aturan dekripsi $d_k \in D$. Setiap $e_k: P \rightarrow C$ dan $d_k: C \rightarrow P$ adalah fungsi sedemikian hingga $d_k(e_k(x)) = x$ untuk setiap *plaintext* $x \in P$.

Dalam pengujian menggunakan korelasi yang merupakan teknik statistik untuk mengukur kekuatan hubungan antar dua variabel dan untuk mengetahui bentuk hubungan antara dua variabel tersebut dengan hasil yang bersifat kuantitatif. Kekuatan hubungan antar dua variabel itu disebut dengan koefisien korelasi. Nilai koefisien akan selalu berada diantara -1 sampai +1. Untuk menentukan kuat atau lemahnya hubungan antara variabel yang diuji, dapat digunakan Tabel 1. [6].

TABEL I
KLASIFIKASI KOEFISIEN KORELASI

Interval Koefisien	Tingkat Hubungan
0,00 – 0,199	Sangat Rendah
0,20 – 0,399	Rendah
0,40 – 0,599	Sedang
0,60 – 0,799	Kuat
0,80 – 1,000	Sangat Kuat

Selain itu proses *block cipher* ini menggunakan operasi XOR dimana output yang dihasilkan dari proses enkripsi akan susah ditebak, karena apabila kita melihat dasar dari XOR seperti berikut :

- 0 XOR 0 = 0
- 0 XOR 1 = 1
- 1 XOR 0 = 1
- 1 XOR 1 = 0

Maka apabila hasil output adalah 0 untuk mendapatkan input nya kita tidak tahu, bisa jadi input yang dihasilkan adalah 1 atau 0. Dasar tersebut digunakan untuk melakukan kriptografi *block cipher*.

Kemudian *S-Box* (*Substitution Box*) merupakan salah satu prinsip dalam perancangan *block cipher* dimana proses *s-box* itu sendiri adalah mengganti karakter inputan dengan karakter yang sudah menjadi tetapan pada sebuah tabel. Secara teoritis, *S-Box*

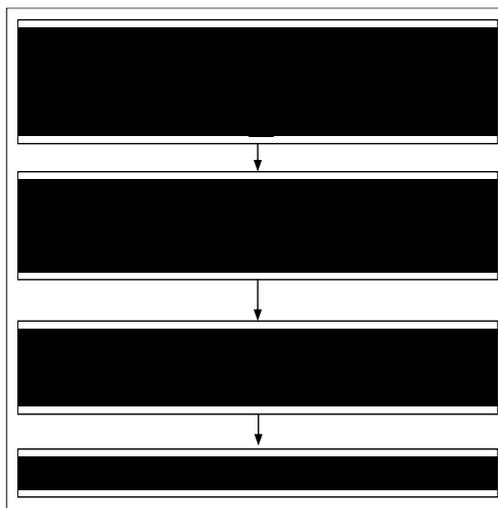
adalah satu-satunya algoritma yang mempunyai kemampuan untuk membuat hubungan yang tidak linier antara *plaintext* dan *ciphertext*. Maka dari itu, penggunaan *S-Box* ditujukan agar membuat Kriptografi *block cipher* menjadi lebih acak. Hal ini dilakukan dengan cara mensubstitusikan bilangan *hexadecimal* ke dalam tabel *S-Box* dan kemudian kita ambil output dari tabel *S-Box* berupa bilangan *hexadecimal* yang baru.

	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
0	63	7C	77	7B	F2	6B	6F	C5	30	01	67	2B	FE	D7	AB	76
1	CA	82	C9	7D	FA	59	47	F0	AD	D4	A2	AF	9C	A4	72	C0
2	B7	FD	93	26	36	3F	F7	CC	34	A5	E5	F1	71	D8	31	15
3	04	C7	23	C3	18	96	05	9A	07	12	80	E2	EB	27	B2	75
4	09	83	2C	1A	1B	6E	5A	A0	52	3B	D6	B3	29	E3	2F	84
5	53	D1	00	ED	20	FC	B1	5B	6A	CB	BE	39	4A	4C	58	CF
6	D0	EF	AA	FB	43	4D	33	85	45	F9	02	7F	50	3C	9F	A8
7	51	A3	40	8F	92	9D	38	F5	BC	B6	DA	21	10	FF	F3	D2
8	CD	0C	13	EC	5F	97	44	17	C4	A7	7E	3D	64	5D	19	73
9	60	81	4F	DC	22	2A	90	88	46	EE	B8	14	DE	5E	0B	DB
A	E0	32	3A	0A	49	06	24	5C	C2	D3	AC	62	91	95	E4	79
B	E7	C8	37	6D	8D	D5	4E	A9	6C	56	F4	EA	65	7A	AE	08
C	BA	78	25	2E	1C	A6	B4	C6	E8	DD	74	1F	4B	BD	8B	8A
D	70	3E	B5	66	48	03	F6	0E	61	35	57	B9	86	C1	1D	9E
E	E1	F8	98	11	69	D9	8E	94	9B	1E	87	E9	CE	55	28	DF
F	8C	A1	89	0D	BF	E6	42	68	41	99	2D	0F	B0	54	BB	16

Gambar 2. Tabel S-Box

III. METODE PENELITIAN

Secara umum penelitian terbagi ke dalam 4 (empat) tahapan, yaitu: (1) tahap identifikasi masalah, (2) tahap perancangan, (3) tahap implementasi dan analisis hasil, (4) tahap pelaporan dari hasil penelitian.



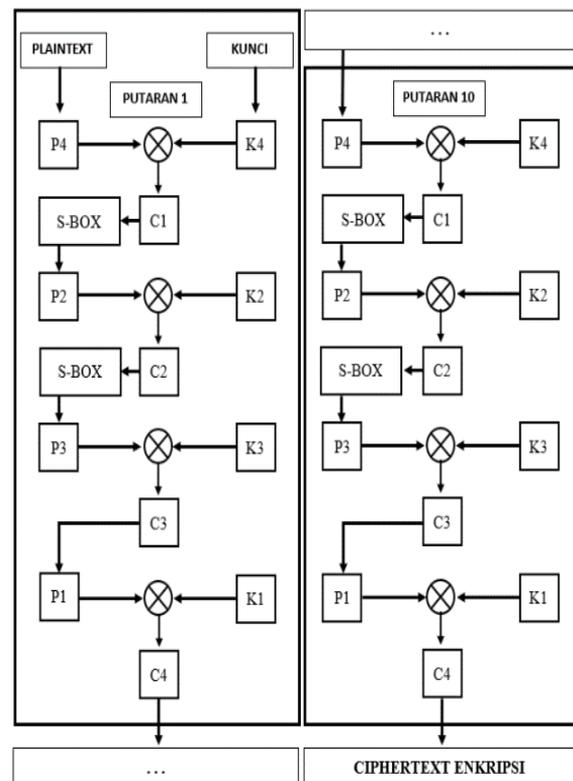
Gambar 3. Tahapan Penelitian

Tahapan penelitian tersebut dapat dijelaskan sebagai berikut.

1. Identifikasi Masalah: pada tahap ini dilakukan analisis terhadap permasalahan yang ada, yaitu perancangan kriptografi *block cipher* menggunakan pola Tarian Sajojo Papua.
2. Perancangan: tahapan selanjutnya adalah perancangan kriptografi *block cipher* dengan menggunakan pola Tarian Sajojo Papua.

3. Implementasi dan Analisis Hasil: setelah perancangan selesai kemudian dilakukan uji coba dan analisis dari kriptografi yang telah dibuat.
4. Laporan Hasil Penelitian: tahap terakhir adalah penulisan penelitian yang sudah dilakukan dalam bentuk laporan.

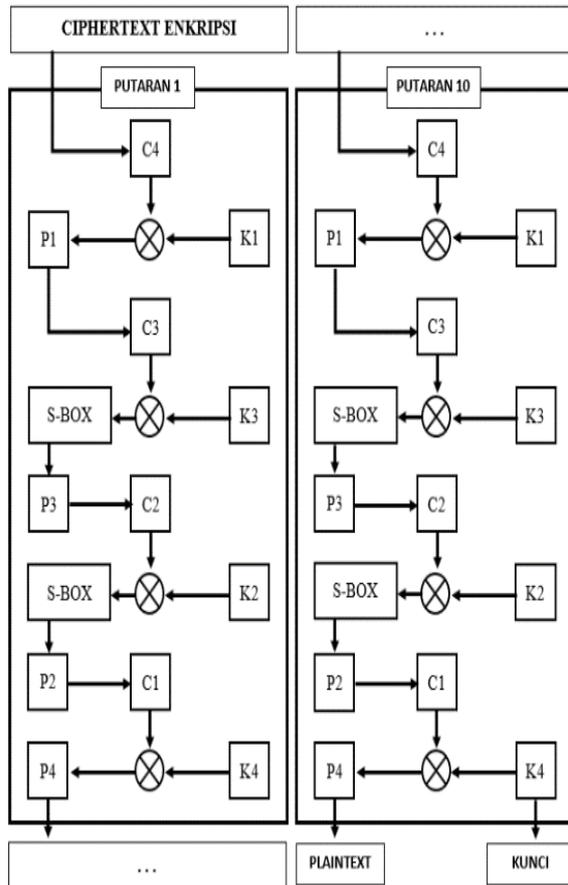
Dalam perancangan kriptografi *block cipher* pada pola Tarian Sajojo Papua ini dilakukan dua proses yaitu enkripsi dan proses dekripsi. Enkripsi dan dekripsi itu sendiri dilakukan dalam 10 putaran. 10 putaran untuk enkripsi dan 10 putaran untuk dekripsi. Di setiap putaran terdapat 4 proses.



Gambar 4. Proses Alur Enkripsi

Langkah-langkah proses enkripsi dapat dijabarkan sebagai berikut: a) Menyiapkan *plaintext*; b) Mengubah *plaintext* menjadi biner sesuai dalam tabel ASCII; c) Dalam proses enkripsi, *plaintext* dan kunci akan melewati empat proses pada setiap putaran, yaitu : 1) Putaran pertama *Plaintext* 1 (P1) melakukan transformasi dengan pola tarian Sajojo Papua dan di XOR dengan Kunci 1 (K1) menghasilkan *Ciphertext* 1 (C1); 2) *Plaintext* 2 (P2) melakukan transformasi dengan pola tarian Sajojo Papua dan di XOR dengan Kunci 2 (K2) menghasilkan *Ciphertext* 2 (C2), dan tahapan tersebut akan berlanjut sampai empat proses yang menghasilkan *Ciphertext* 4 (C4) ; 3) *Ciphertext* 4 (4) masuk pada putaran kedua dengan alur proses yang sama dengan putaran pertama, dan tahapan tersebut

akan berlanjut sampai putaran ke-20 yang menghasilkan *Ciphertext* Akhir.



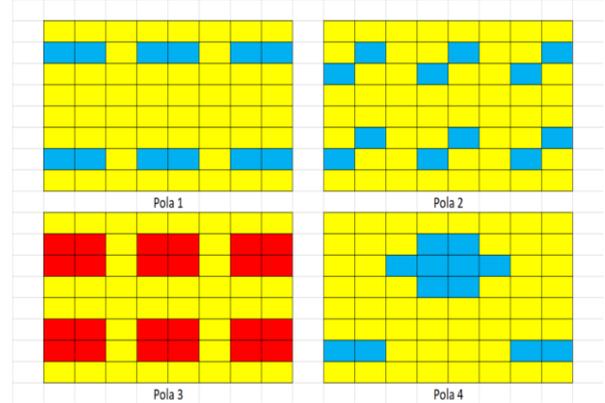
Gambar 5. Proses Alur Deskripsi

Gambar 5 menunjukkan alur proses dekripsi, langkah-langkah proses dekripsi tersebut dijelaskan sebagai berikut: a) Menyiapkan *ciphertext* dan kunci; b) Mengubah *ciphertext* dan kunci menjadi biner sesuai dalam tabel ASCII; c) dalam perancangan dekripsi, *ciphertext* dan kunci akan melewati empat proses pada setiap putaran; d) Putaran pertama *Ciphertext* (C) diproses dengan pola dan di XOR dengan Kunci 4 (K4) dari putaran 20, menghasilkan P4; d) P4 tersebut kemudian menjadi C3 di putaran 20; e) Masuk pada putaran dua, C3 diproses dengan pola dan di XOR dengan Kunci 3 (K3) dari putaran 20, menghasilkan P3; Proses tersebut berlanjut sampai ke putaran 10 sehingga menghasilkan *Plaintext* akhir.

IV. HASIL DAN PEMBAHASAN

Dalam bagian ini akan membahas tentang algoritma perancangan kriptografi block cipher 64bit berbasis pola Tarian Sajojo Papua secara lebih rinci.

Dalam algoritma ini pola yang terdapat pada contoh Tarian Sajojo Papua digunakan sebagai proses pemasukan dan pengambilan bit. Pola tersebut ditunjukkan pada Gambar 5.



Gambar 6. Tahapan Penelitian

Pada Gambar 6 menunjukan empat pola yang berbeda, dimana pola-pola tersebut menunjukan pola-pola yang terdapat pada tarian Sajojo Papua. Berdasarkan pola-pola yang sudah dirancang, dilakukan pengujian korelasi dengan mengkombinasikan urutan pola untuk menemukan nilai korelasi terbaik. Pengujian dilakukan menggunakan contoh plaintext “GWENNI01” menggunakan kunci “DAPHNE06”.

Berdasarkan hasil pengujian korelasi, maka hasil terkecil yang akan digunakan sebagai acuan perancangan dalam proses enkripsi dan dekripsi.

TABEL II
HASIL KORELASI SETIAP KOMBINASI POLA TARIAN SAJOJO PAPUA

RATA-RATA NILAI KORELASI			
POLA	RATA-RATA	POLA	RATA-RATA
1-2-3-4	0,009425450	3-1-2-4	0,161300639
1-2-4-3	0,017899859	3-1-4-2	0,170749096
1-3-2-4	0,090862916	3-2-1-4	0,582125035
1-3-4-2	0,207041609	3-2-4-1	0,645915328
1-4-2-3	0,220356177	3-4-1-2	0,591970156
1-4-3-2	0,180880160	3-4-2-1	0,556334151
2-1-3-4	0,361258097	4-1-2-3	0,429722001
2-1-4-3	0,313262995	4-1-3-2	0,506564942
2-3-1-4	0,038877346	4-2-1-3	0,146680735
2-3-4-1	0,659971578	4-2-3-1	0,006651533
2-4-1-3	0,130867750	4-3-1-2	0,046562621
2-4-3-1	0,502260685	4-3-2-1	0,490564999

Tabel 2 menunjukan hasil kombinasi pola dan mendapatkan nilai korelasi terbaik pada kombinasi pola 4-2-3-1. Kombinasi ini yang akan digunakan untuk melanjutkan proses enkripsi hingga putaran ke-10 untuk menghasilkan *ciphertext*.

Telah dijelaskan bahwa perancangan kriptografi ini dilakukan sebanyak 10 putaran, dan disetiap putaran memiliki 4 proses untuk mendapatkan hasil akhir yaitu *ciphertext*. Proses pertama plaintext dan kunci diubah kedalam bentuk ASCII kemudian diubah lagi

kedalam biner. Kemudian bit-bit *plaintext* diproses dengan pola pemasukan dan pengambilan kedalam kolom matriks 8x8 menggunakan bagian dari pola tarian yang berbeda-beda pada setiap proses. Kemudian di setiap proses dilakukan X-OR dari Plaintext (P) dan kunci (K) menghasilkan ciphertext (C) sampai proses keempat di setiap putaran. Kemudian diulang terus sampai putaran ke-10 dan hingga menghasilkan Ciphertext akhir.

Untuk menjelaskan secara detail proses pemasukan bit dalam matriks maka diambil proses 1 pada putaran 1 sebagai contoh. Misalkan angka 1 merupakan inialisasi setiap bit yang merupakan hasil konversi plaintext maka urutan bit adalah sebagai berikut 1, 2, 3, 4,64.

1	2	3	4	36	35	34	33	1	8	9	16	17	24	25	32
8	7	6	5	37	38	39	40	2	7	10	15	18	23	26	31
9	10	11	12	44	43	42	41	3	6	11	14	19	22	27	30
16	15	14	13	45	46	47	48	4	5	12	13	20	21	28	29
17	18	19	20	52	51	50	49	61	60	53	52	45	44	37	36
24	23	22	21	53	54	55	56	62	59	54	51	46	43	38	35
25	26	27	28	60	59	58	57	63	58	55	50	47	42	39	34
32	31	30	29	61	62	63	64	64	57	56	49	48	41	40	33
Pola Kunci 1								Pola Kunci 2							
33	34	35	36	29	30	31	32	33	40	41	48	49	56	57	64
40	39	38	37	28	27	26	25	34	39	42	47	50	55	58	63
41	42	43	44	21	22	23	24	35	38	43	46	51	54	59	62
48	47	46	45	20	19	18	17	36	37	44	45	52	53	60	61
49	50	51	52	13	14	15	16	29	28	21	20	13	12	5	4
56	55	54	53	12	11	10	9	30	27	22	19	14	11	6	3
57	58	59	60	5	6	7	8	31	26	23	18	15	10	7	2
64	63	62	61	4	3	2	1	32	25	24	17	16	9	8	1
Pola Kunci 3								Pola Kunci 4							

Gambar 7. Pola Pemasukan Kunci

1	2	3	4	5	6	7	8
9	10	11	12	13	14	15	37
16	17	18	19	20	21	38	39
22	23	24	25	26	40	41	42
27	28	29	30	43	44	45	46
31	32	33	47	48	49	50	51
34	35	52	53	54	55	56	57
36	58	59	60	61	62	63	64

Gambar 8. Pola Ambil Semua Kunci

13	20	27	34	39	44	51	58
14	21	28	1	2	45	52	59
15	22	3	4	5	6	53	60
16	23	29	7	8	46	54	61
17	24	30	35	40	47	55	62
18	25	31	36	41	48	56	63
9	10	32	37	42	49	11	12
19	26	33	38	43	50	57	64



Gambar 9. Pola Pemasukan Plaintext dari Pola 4 Untuk Proses 1 di Setiap Putaran

58	57	44	43	34	33	20	19
59	56	45	8	7	32	21	18
60	55	3	4	5	6	22	17
61	54	46	2	1	31	23	16
62	53	47	42	35	30	24	15
63	52	48	41	36	29	25	14
12	11	49	40	37	28	10	9
64	51	50	39	38	27	26	13

Gambar 10. Pola Pengambilan Plaintext dari Pola 4 Untuk Proses 1 di Setiap Putaran

Gambar 9 merupakan pola masuk dari pola 4 yang digunakan untuk memasukkan setiap 8 bit dari karakter plaintext, kemudian pola tersebut diimplementasikan ke dalam excel. Dari pola tersebut kemudian diambil bitnya sesuai pola ambil pada Gambar 10 sehingga menghasilkan bit bit yang nantinya akan di XOR kan dengan kunci yang sebelumnya sudah dimasukkan ke pola masuk seperti pada Gambar 7 dan sudah diambil menggunakan pola ambil seperti Gambar 8, sehingga menghasilkan Ciphertext 1.

13	24	25	38	39	52	53	64
14	2	26	37	4	51	54	6
1	23	27	3	40	50	5	63
15	22	28	36	41	49	55	62
16	21	29	35	42	48	56	61
17	11	30	34	9	47	57	7
12	20	31	10	43	46	8	60
18	19	32	33	44	45	58	59



Gambar 11. Pola Pemasukan Plaintext dari Pola 2 Untuk Proses 2 di Setiap Putaran

59	58	45	44	33	32	19	18
60	8	46	43	10	31	20	12
7	57	47	9	34	30	11	17
61	56	48	42	35	29	21	16
62	55	49	41	36	28	22	15
63	5	50	40	3	27	23	1
6	54	51	4	37	26	2	14
64	53	52	39	38	25	24	13

Gambar 12. Pola Pengambilan Plaintext dari Pola 2 Untuk Proses 2 di Setiap Putaran

Gambar 11 merupakan pola masuk dari pola 2 yang digunakan untuk memasukkan setiap 8 bit dari karakter plaintext, kemudian pola tersebut diimplementasikan ke dalam excel. Dari pola tersebut kemudian dilakukan proses S-BOX lalu hasilnya dalam bentuk biner diambil sesuai pola ambil pada Gambar 12 sehingga menghasilkan bit bit yang nantinya akan di XOR kan dengan kunci yang sebelumnya sudah dimasukkan ke pola masuk seperti pada Gambar 7 dan sudah diambil menggunakan pola ambil seperti Gambar 8, sehingga menghasilkan Ciphertext 2.

25	26	27	28	29	30	31	32
1	2	34	5	6	33	9	10
4	3	35	8	7	36	12	11
44	43	42	41	40	39	38	37
45	46	47	48	49	50	51	52
13	14	54	17	18	53	21	22
16	15	55	20	19	56	24	23
64	63	62	61	60	59	58	57



Gambar 13. Pola Pemasukan Plaintext dari Pola 3 Untuk Proses 3 di Setiap Putaran

57	58	59	60	61	62	63	64
4	3	56	8	7	55	12	11
1	2	53	5	6	54	9	10
52	51	50	49	48	47	46	45
37	38	39	40	41	42	43	44
16	15	36	20	19	35	24	23
13	14	33	17	18	34	21	22
32	31	30	29	28	27	26	25

Gambar 14. Pola Pengambilan Plaintext dari Pola 3 Untuk Proses 3 di Setiap Putaran

Gambar 13 merupakan pola masuk dari pola 3 yang digunakan untuk memasukkan setiap 8 bit dari karakter plaintext, kemudian pola tersebut diimplementasikan ke dalam excel. Dari pola tersebut kemudian diambil bitnya sesuai pola ambil pada Gambar 14 sehingga menghasilkan bit bit yang nantinya akan di XOR kan dengan kunci yang sebelumnya sudah dimasukkan ke pola masuk seperti pada Gambar 7 dan sudah diambil menggunakan pola ambil seperti Gambar 8, sehingga menghasilkan Ciphertext 3.

13	14	15	16	17	18	19	20
1	2	22	3	4	21	5	6
23	24	25	26	27	28	29	30
38	37	36	35	34	33	32	31
39	40	41	42	43	44	45	46
54	53	52	51	50	49	48	47
7	8	55	9	10	56	11	12
64	63	62	61	60	59	58	57



Gambar 15. Pola Pemasukan Plaintext dari Pola 1 Untuk Proses 4 di Setiap Putaran

57	58	59	60	61	62	63	64
12	11	56	10	9	55	8	7
47	48	49	50	51	52	53	54
46	45	44	43	42	41	40	39
31	32	33	34	35	36	37	38
30	29	28	27	26	25	24	23
6	5	21	4	3	22	2	1
20	19	18	17	16	15	14	13

Gambar 16. Pola Pengambilan Plaintext dari Pola 1 Untuk Proses 4 di Setiap Putaran

Gambar 15 merupakan pola masuk dari pola yang digunakan untuk memasukkan setiap 8 bit dari karakter plaintext, kemudian pola tersebut diimplementasikan ke dalam excel. Dari pola tersebut kemudian dilakukan proses S-BOX lalu hasilnya dalam bentuk biner diambil bitnya sesuai pola ambil pada Gambar 16 sehingga menghasilkan bit bit yang nantinya akan di XOR kan dengan kunci yang sebelumnya sudah dimasukkan ke pola masuk seperti pada Gambar 7 dan sudah diambil menggunakan pola

ambil seperti Gambar 8, sehingga menghasilkan Ciphertext 4.

Proses enkripsi putaran 1 telah selesai, kemudian dilakukan proses yang sama secara terus-menerus hingga putaran ke-10 untuk mendapatkan ciphertext akhir. Di semua putaran dilakukan proses transformasi dengan menggunakan Tabel S-BOX pada Plaintext 2 dan Plaintext 4.

TABEL III
TABEL PERUBAHAN P2 DAN P3 SETIAP PUTARAN SETELAH DILAKUKAN PROSES S-BOX

Putaran	Plaintext	Hexa Sebelum Proses S-BOX	Hexa Sesudah Proses S-BOX
1	P2	D7C260575AE8A4B7	0DA890DA46C81D20
	P3	3A8B0414462B9658	A2CE309B980B355E
2	P2	9B2CAAF8CF856C993	E8426355E1B91222
	P3	4418925676728EBA	863474B90F1EE6C0
3	P2	ABCC8FA4EB545531	0E27731D3CFDED2E
	P3	287616759DD1095F	EE0FFF3F75514084
4	P2	3B7B018CF82E6EB0	490309F0E1CC345FC
	P3	62D72F3FD21AD212	AB0D4E257F437F39
5	P2	FF6E056DFB3B156A	7D4536B363492F58
	P3	48038E1BDF34653F	D4D5E644EF28BC25
6	P2	C93BC52B54E0E92D	1249070BFDA0EBFA
	P3	CE4C6C0D1ED78E5A	EC5DB8F3E90DE646
7	P2	3538F08290DC36AB	D97617119693240E
	P3	8979F330CA6380B3	F2AF7E0810003A4B
8	P2	ED9A12C38B376BAF	53373933CEB2051B
	P3	29F13D8A1D614C01	4C2B8BCFDE8D5D09
9	P2	761638DDFF01159F	0FFF76C97D092F6E
	P3	5B381E358DF4BD46	5776E9D9B4BACD98
10	P2	04E184A4299BD7CF	30E04F1D4CE80D5F
	P3	104348FAD6F6A669	7C64D4144AD6C5E4

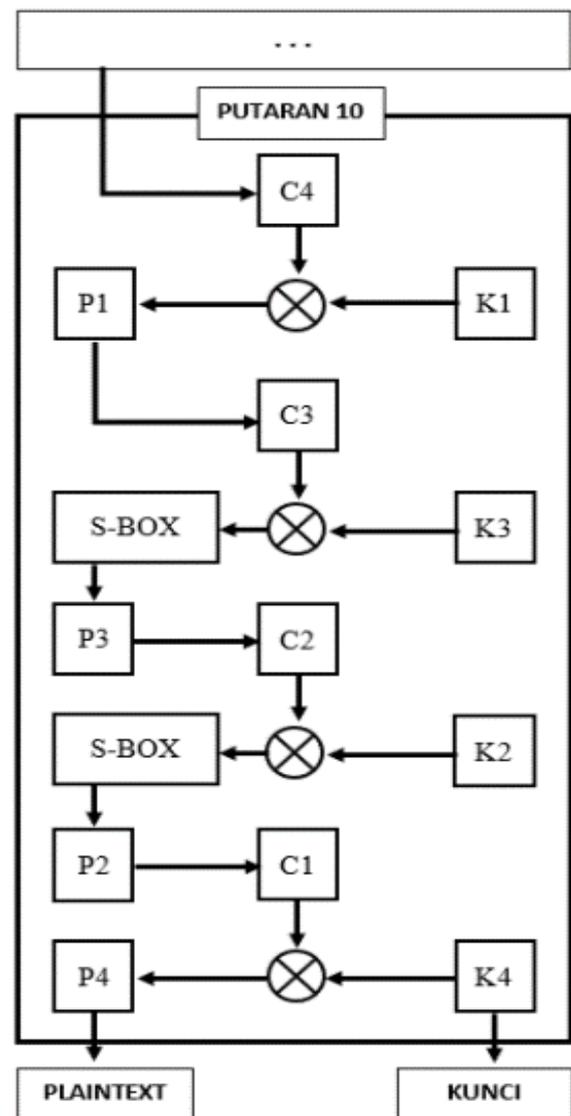
Tabel 3 merupakan hasil dari proses S-BOX yang dilakukan pada setiap putaran untuk proses Plaintext 2 dan Plaintext 3. Proses S-BOX dilakukan agar Ciphertext yang dihasilkan pada setiap akhir putaran menjadi lebih acak.

Untuk pengujian algoritma dilakukan dengan mengambil contoh plaintext GWENNI01 dan kunci adalah DAPHNE06. Kemudian dilakukan proses enkripsi sebanyak 10 putaran, dan disetiap putaran enkripsi akan mendapatkan ciphertext (C) berupa char dan konversi hexadesimal. Hasil enkripsi dari putaran ke 10 adalah final ciphertext ditunjukkan pada Tabel 4.

TABEL IV
HASIL CIPHERTEXT SETIAP PUTARAN PADA PROSES ENKRIPSI

Putaran	Hasil Hexadesimal	Hasil Char
1	F147964FA14D0BA	ñG-úĐ°
2	5ABE15AEF08410BA	Z¾®ð,,°
3	963A369D4E63CCA	cfiÔæ<Ê
4	AAF663585EC479B7	ªöcX^Äy·
5	4A12A1A9128BDDF8	J¡©<Ýø
6	99B4DA29896B65E8	™'Ú)%økeè
7	96251D2AFD13FDA6	-%*ýý
8	1CB74C7C73E3A6E	·ÇÇ>:n
9	2EC1B0F0CAFEE755	.Ä°ðËþçU
10	EA14FD29794750ED	êý)yGPí

Kemudian masuk ke proses dekripsi. Proses dekripsi adalah proses merubah ciphertext menjadi plaintext awal. Dekripsi dilakukan sama seperti enkripsi, tetapi dekripsi dimulai dari putaran ke-10 menuju putaran ke-1 untuk mendapatkan *plaintext* awal.



Gambar 17. Skema Proses Dekripsi

Gambar 17 menjelaskan alur dekripsi. Pola pengambilan pada proses enkripsi akan menjadi pola pemasangan pada proses dekripsi, sedangkan pola pemasangan pada enkripsi akan digunakan sebagai pola pengambilan pada dekripsi. Proses dekripsi dimulai dari memasukkan ciphertext ke kolom matrik 8x8 C4 kemudian di-XOR dengan K1 pada proses keempat menghasilkan P1. Kemudian P1 akan digunakan sebagai C3 kemudian di XOR dengan K3 dan menghasilkan P3. P3 kemudian dilakukan proses S-BOX lalu akan digunakan sebagai C2 pada proses berikutnya. Setelah itu, C2 di-XOR dengan K2 menghasilkan P2. Kemudian P2 dilakukan proses S-BOX lalu akan digunakan sebagai C1 pada proses selanjutnya. C1 kemudian di-XOR dengan K4 menghasilkan P4, proses itu dilakukan berulang-ulang sebanyak 10 putaran sesuai dengan banyaknya putaran enkripsi dan hasil akhir dari dekripsi putaran ke-10 adalah *plaintext* awal.

TABEL V
ALGORITMA ENKRIPSI DAN DEKRIPSI

No	Proses Enkripsi	No.	Proses Dekripsi
1.	Masukkan <i>plaintext</i>	1.	Masukkan <i>ciphertext</i>
2.	<i>Plaintext</i> diubah ke DECIMAL	2.	<i>Ciphertext</i> diubah ke DECIMAL
3.	DECIMAL diubah ke BINER	3.	DECIMAL diubah ke BINER
4.	Bit BINER dimasukkan ke kolom matriks 8x8 P4 dengan pola pemasukan <i>plaintext</i>	4.	Bit BINER dimasukkan ke kolom matriks 8x8 C4 dengan pola pemasukan <i>plaintext</i>
5.	Bit pada kolom matrik diambil menggunakan pola pengambilan pola 4	5.	C4 di-XOR dengan K4 menghasilkan P1
6.	Bit pengambilan dimasukkan lagi kedalam matrik mendapatkan hasil akhir P4	6.	P1 diproses dengan pola pemasukan <i>plaintext</i>
7.	P4 di-XOR dengan K4 menghasilkan C1	7.	Hasil proses P1 dimasukkan kedalam matriks 8x8 lagi dengan pola pengambilan pola 1
8.	C1 menjadi P2 untuk proses selanjutnya	8.	P1 menjadi C3 untuk proses selanjutnya
9.	Bit pada kolom matrik diambil menggunakan pola pengambilan pola 2	9.	C3 di-XOR dengan K3 menghasilkan P3
10.	Bit pengambilan dimasukkan lagi kedalam matrik mendapatkan hasil akhir P2	10.	P3 diproses dengan pola pemasukan <i>plaintext</i>
11.	P2 dilakukan	11.	P3 dilakukan
			proses S-BOX.
12.			P2 di-XOR dengan K2 menghasilkan C2
			proses S-BOX
12.		12.	Hasil proses P3 dimasukkan kedalam matriks 8x8 lagi dengan pola pengambilan pola 3
13.			C2 menjadi P3 untuk proses selanjutnya
13.		13.	P3 menjadi C2 untuk proses selanjutnya
14.			Bit pada kolom matrik diambil menggunakan pola pengambilan pola 3
14.		14.	C2 di-XOR dengan K2 menghasilkan P2
			Bit pengambilan dimasukkan lagi kedalam matrik mendapatkan hasil akhir P3
15.		15.	P2 diproses dengan pola pemasukan <i>plaintext</i>
16.			P3 dilakukan proses S-BOX.
16.		16.	P2 dilakukan proses S-BOX
17.			P3 di-XOR dengan K3 menghasilkan C3
17.		17.	Hasil proses P2 dimasukkan kedalam matriks 8x8 lagi dengan pola pengambilan pola 2
18.			C3 menjadi P1 untuk proses selanjutnya
18.		18.	P2 menjadi C1 untuk proses selanjutnya
19.			Bit pada kolom matrik diambil menggunakan pola pengambilan pola 1
19.		19.	C1 di-XOR dengan K4 menghasilkan P4
20.			Bit pengambilan dimasukkan lagi kedalam matrik mendapatkan hasil akhir P1
20.		20.	P4 diproses dengan pola pemasukan <i>plaintext</i>
21.			P1 di-XOR dengan K1 menghasilkan C1
21.		21.	Hasil proses P4 dimasukkan kedalam matriks 8x8 lagi dengan pola pengambilan

			pola 4
22.	C1 diubah ke DECIMAL	22.	P4 diubah ke DECIMAL
23.	DECIMAL diubah ke CHAR untuk mendapatkan <i>Ciphertext</i> akhir.	23.	DECIMAL diubah ke CHAR untuk mendapatkan <i>Plaintext</i> awal.

Tabel 5 merupakan algoritma proses enkripsi dan dekripsi secara menyeluruh. Proses enkripsi menghasilkan *Ciphertext* akhir, dan proses dekripsi menghasilkan *Plaintext* awal.

Algoritma proses Kunci (key), dijelaskan sebagai berikut:

1. Masukkan Kunci
2. Kunci diubah ke DECIMAL
3. DECIMAL ke BINER
4. Bit BINER dimasukkan ke kolom K4 dengan pola pemasukan Kunci
5. Bit kunci diambil dengan pola pengambilan Kunci
6. BINER hasil pengambilan dimasukkan kedalam kolom matrik K4
7. $K4 = K2$
8. K2 dimasukkan ke kolom matrik K2 dengan pola pemasukan
9. Bit kunci diambil dengan pola pengambilan Kunci
10. BINER hasil pengambilan dimasukkan kedalam kolom matrik K2
11. $K2 = K3$
12. K3 dimasukkan ke kolom matrik K3 dengan pola pemasukan
13. Bit kunci diambil dengan pola pengambilan Kunci
14. BINER hasil pengambilan dimasukkan kedalam kolom matrik K3
15. $K3 = K1$
16. K1 dimasukkan ke kolom matrik K1 dengan pola pemasukan
17. Bit kunci diambil dengan pola pengambilan Kunci
18. BINER hasil pengambilan dimasukkan kedalam kolom matrik K1

Dari setiap putaran, tentunya akan menghasilkan nilai korelasi antara *plaintext* dengan *ciphertext* yang bertujuan untuk menilai seberapa acak hasil enkripsi yang berupa *ciphertext* dengan *plaintext* awal pada masing-masing putaran. Nilai korelasi itu sendiri berkisaran 1 sampai -1 dimana jika nilai korelasi mendekati 0, maka *plaintext* dan *ciphertext* tidak memiliki nilai yang berhubungan. Akan tetapi jika nilai korelasi mendekati 1 atau -1, maka nilai dari korelasi itu sangat berhubungan.

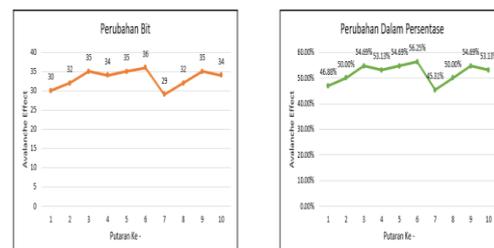
TABEL VI
NILAI KORELASI SETIAP PUTARAN

Putaran	Nilai Korelasi
1	-0.537021257
2	-0.091528429
3	0.212142943
4	0.295297477
5	0.179997713
6	0.097581066
7	0.234471714
8	-0.847113678
9	-0.610498822
10	0.104262947

Tabel 6 menunjukkan nilai korelasi pada setiap putaran dan dapat disimpulkan bahwa algoritma Kriptografi Block Cipher berbasis pola Tarian Sajojo Papua memiliki korelasi yang lemah dan menghasilkan nilai korelasi yang acak. Kemudian pengujian Avalanche Effect dilakukan agar dapat mengetahui nilai perubahan bit yang ada ketika *plaintext* diubah. Pengujian dilakukan dengan mengubah karakter yang terdapat pada *plaintext* awal, dan tentunya akan menghasilkan perbedaan pada setiap putarannya.

Pada umumnya, bit pada *ciphertext* akan mengalami perubahan dari jumlah bit pada *plaintext* sebesar 50%. Suatu Avalanche Effect dikatakan baik jika perubahan bit yang dihasilkan berkisar antara 45% - 60% (sekitar separuhnya) [11].

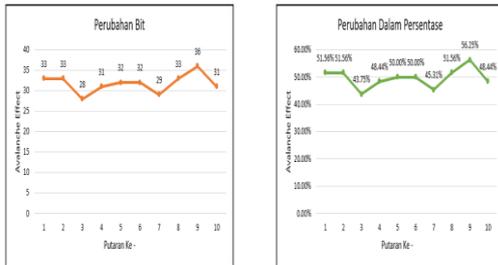
	1	2	3	4	5	6	7	8	9	10	BATA-RATA
Persentase Perubahan	46,88%	50,00%	54,69%	51,13%	54,69%	56,25%	45,33%	50,00%	54,69%	51,13%	51,88%
Perubahan Bit	30	32	35	34	35	36	29	32	35	34	31,1



Gambar 18. Grafik Avalanche Effect dari *Plaintext* "GWENNI01"

Gambar 18 adalah hasil dari pengujian Avalanche Effect dimana *plaintext* awal yang digunakan adalah "GWENNI01". Pada putaran keenam perubahan bit yang terjadi tidak terlalu besar yaitu 56,25%. Dengan ini berarti terdapat perubahan bit yang baik, namun untuk nilai Avalanche Effect dapat dikatakan tidak begitu baik jika persentase jauh dari angka 50%. Berdasarkan hasil putaran pertama hingga putaran kesepuluh, dapat disimpulkan bahwa rata-rata hasil pengujian Avalanche Effect ini yaitu sebesar 51,88% yang berarti termasuk kategori sangat baik [11].

	1	2	3	4	5	6	7	8	9	10	RATA-RATA
Persentase Perubahan	51,56%	51,56%	43,75%	48,44%	50,00%	50,00%	45,21%	51,56%	56,25%	48,44%	49,69%
Perubahan bit	33	33	28	31	32	32	29	33	36	31	31,8



Gambar 19. Grafik Avalanche Effect dari Plaintext “VIONA997”

Pada Gambar 19 merupakan hasil dari pengujian Avalanche Effect dari plaintext awal “VIONA997”, menghasilkan perubahan bit yang tidak terlalu tinggi dimana paling tinggi 56,25% pada putaran kesembilan. Nilai Avalanche Effect yang dihasilkan dari plaintext awal “VIONA997” lebih baik dibandingkan dengan plaintext awal “GWENNI01” yaitu sebesar 49,69% dengan kategori sangat baik[11].

V. SIMPULAN

Berdasarkan penelitian yang dilakukan, dapat disimpulkan bahwa kriptografi *block cipher* 64 bit berbasis pola Tarian Sajojo Papua dapat dikatakan sebagai sistem kriptografi. Dalam proses enkripsi, rancangan kriptografi *block cipher* berbasis pola Tarian Sajojo Papua ini menghasilkan output yang acak sehingga dapat digunakan sebagai alternatif dalam pengamanan data. Dalam pengujian *avalanche effect* yang dilakukan pun menunjukkan bahwa proses enkripsi di setiap putaran memiliki rata-rata perubahan yang mencapai 49,69% pada plaintext “VIONA997” dibandingkan dengan plaintext “GWENNI01” memiliki rata-rata perubahan yang mencapai 51,88% yang berarti algoritma kriptografi ini berhasil dan termasuk ke dalam kategori yang sangat baik. Penelitian ini sudah sangat baik apabila dibandingkan dengan penelitian terdahulu yang

kebanyakan mempunyai nilai rata-rata *avalanche effect* sama-sama mendekati angka 50% yang berarti algoritma kriptografinya termasuk sangat baik.

VI. DAFTAR PUSTAKA

- [1] Berita Satu, “Teknologi Enkripsi, Solusi Terbaik Pengamanan Data,” [Online]. Available: <http://www.beritasatu.com/ipitek/426799-teknologi-enkripsi-solusi-terbaik-pengamanan-data.html>. [Accessed 29 November 2018].
- [2] Humaira, Rafiq, dkk “Kriptanalisis dengan Metode Brute Force pada Graphics Processing Unit”, Hal. 2–5, Bandung
- [3] F. D. Paliama, “Perancangan Kriptografi Block Cipher Berbasis Pada Teknik Formasi Permainan Bola Perancangan Kriptografi Block Cipher Berbasis Pada Teknik Formasi Permainan Bola,” Universitas Kristen Satya Wacana, 2016.
- [4] N. M. Louhenapessy, “Perancangan Kriptografi Block Cipher Berbasis Pola Formasi Futsal 1-2-1,” Universitas Kristen Satya Wacana, 2016.
- [5] B. L. Setiyadi, “Perancangan Kriptografi Block Cipher Berbasis Pada Pola Gerakan Lempeng Tektonik Divergensi dan Konvergensi Program Studi Teknik Informatika Fakultas Teknologi Informasi Universitas Kristen Satya Wacana Salatiga Movember 2016 Perancangan Kriptografi Block,” Universitas Kristen Satya Wacana, 2016.
- [6] Mahendra, Dwi Putera. 2016. Perancangan Kriptografi Block Cipher Menggunakan Pola Game SUDOKU. Universitas Kristen Satya Wacana: Salatiga.
- [7] K. D. Cahyono, “Perancangan Kriptografi Block Cipher dengan Langkah Permainan Engklek,” Universitas Kristen Satya Wacana, 2016.
- [8] Munir, R., 2006. Kriptografi. Bandung: Informatika.
- [9] J. Leodrian, “Pengaruh Perubahan Ciphertext Terhadap Perancangan Kriptografi Block Cipher 64 Bit Berbasis Pola Ikatan Jimbe Dengan Menggunakan Kombinasi S-Box,” Universitas Kristen Satya Wacana, 2016.
- [10] Sugiyono. 2009. “Metode Penelitian Bisnis (Pendekatan Kuantitatif, Kualitatif, dan R&D)”. Bandung : Alfabeta.
- [11] Sugianto. 2016. Pengembangan Algoritma Advanced Encryption Standard pada Sistem Keamanan SMS Berbasis Android Menggunakan Algoritma Vigenere. Institut Teknologi Adhi Tama: Surabaya.

Implementasi Jaringan Saraf Tiruan Backpropagation untuk Pengenalan Karakter pada Dokumen Tercetak

Dian Puspita Chandra¹, Alethea Suryadibrata²

Program Studi Informatika, Universitas Multimedia Nusantara, Tangerang, Indonesia

Diterima 30 Oktober 2019

Disetujui 20 Desember 2019

Abstract--Digital documents from the scanner device cannot be edited. To be able to edit digital documents, Optical Character Recognition (OCR) technology is needed. This research was conducted with the aim of implementing backpropagation artificial neural networks in printed documents and to find out how the accuracy of the implementation of backpropagation artificial neural networks in printed documents. This research uses multilayer networks with three layers. The input layer consists of 225 nodes with 15×15 pixels digital image as input, hidden layer consists of 110 nodes, and the output layer consists of 54 nodes representing A-Z, a-z, point punctuation (.), and comma punctuation (,). The learning rate used in this research is 0,29. The average accuracy level obtained from the implementation of backpropagation artificial neural networks in this research was 94 % for Ms Arial Unicode font type, 96,6 % for Tahoma font type, and 94 % for Times New Roman font type.

Index Terms--Artificial Neural Network, Backpropagation, Image Processing, Optical Character Recognition

I. PENDAHULUAN

Perkembangan ilmu pengetahuan dan teknologi telah memberikan banyak perubahan dalam kehidupan manusia. Salah satu perubahan yang terjadi adalah diterapkannya *paperless office* pada suatu perusahaan atau instansi. *Paperless office* merupakan pengurangan pemakaian kertas dengan mendigitalkan dokumen yang dimiliki suatu perusahaan atau instansi sehingga mengurangi pemakaian kertas secara fisik. Menurut Susilawati [1], konsep *paperless office* dapat menunjang kinerja suatu lembaga pada masa kini dan yang akan datang, karena konsep digitalisasinya sangat menunjang dalam penyampaian arus informasi secara cepat, tepat, lengkap, dan yang paling penting efisien.

Untuk dapat mendigitalkan dokumen tercetak, suatu perusahaan atau instansi dapat menggunakan alat pemindai (*scanner*). Akan tetapi, dokumen digital yang dihasilkan dari alat pemindai tidak dapat diedit. Sehingga, untuk mengubah atau memperbaiki dokumen digital yang dihasilkan dari alat pemindai, suatu perusahaan atau instansi membutuhkan

teknologi *Optical Character Recognition* (OCR) untuk dapat mengedit dokumen digital tersebut.

OCR dapat digunakan untuk mengenali teks yang ada pada dokumen tercetak. Menurut Mithe dkk. [2], OCR adalah teknologi yang memungkinkan berbagai jenis dokumen dikonversi menjadi data yang dapat diedit dan dicari. OCR telah muncul sebagai area penelitian utama sejak tahun 1950 [3]. Sampai saat ini, OCR telah banyak diteliti dengan berbagai metode. Li dkk. [4] melakukan penelitian mengenai OCR untuk pengenalan karakter pada naskah bahasa tradisional yang dinamai New Tai Lue menggunakan *backpropagation neural network* dengan akurasi akhir sebesar 98,04%. Selain penelitian yang dilakukan Li dkk. [4], Radhiah dkk. [5] melakukan penelitian untuk membandingkan tingkat akurasi pengenalan karakter pada huruf arab tercetak dari dua metode yang berbeda yaitu *backpropagation neural network* dengan *hidden markov model*. Hasil akhir penelitian tersebut menunjukkan bahwa tingkat akurasi dari pengenalan karakter arab terisolasi dengan metode *backpropagation neural network* lebih tinggi yaitu sebesar 100% untuk *font* Arial Unicode Ms, Tahoma, dan Times New Roman. Sementara, tingkat akurasi dari pengenalan karakter arab terisolasi dengan metode *hidden markov model* adalah sebesar 74% untuk *font* Arial Unicode Ms, 61% untuk *font* Tahoma, dan 77% untuk *font* Times New Roman.

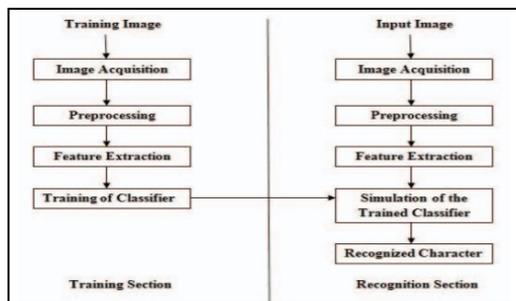
Berdasarkan latar belakang tersebut, diimplementasikan jaringan saraf tiruan *backpropagation* untuk pengenalan karakter dalam penelitian ini. Pengenalan karakter akan dilakukan pada dokumen tercetak yang sudah berupa citra digital. Citra digital akan diproses dengan operasi morfologi dilasi dan erosi, teknik segmentasi *thresholding*, deteksi tepi *canny*, dan penghalusan citra dengan filter *gaussian*.

II. DASAR TEORI

A. *Optical Character Recognition* (OCR)

Optical Character Recognition (OCR) merupakan teknologi untuk pengenalan karakter yang biasa digunakan untuk mengubah teks dalam citra digital

menjadi format teks yang dapat diolah komputer. Dalam penelitian Afroge dkk. [6], OCR terdiri dari dua bagian yaitu bagian pelatihan dan pengenalan. Selain itu, Afroge dkk. [6] menunjukkan bahwa sistem OCR terdiri dari image acquisition, preprocessing, dan feature extraction, serta training of the classifier pada bagian pelatihan dan simulation of the trained classifier pada bagian pengenalan. Ilustrasi sistem OCR dapat dilihat pada Gambar 1.



Gambar 1. Sistem *Optical Character Recognition*

Dalam penelitian Afroge dkk. [6], *feature extraction* digunakan untuk mengekstrak ciri numerik dari data asli. Menurut penelitian yang dilakukan oleh Nugroho [7], *feature extraction* dapat tidak dilakukan namun mengakibatkan proses pelatihan membutuhkan waktu yang lebih lama.

B. Thresholding

Metode *thresholding* adalah proses mengubah citra berderajat keabuan menjadi citra biner atau hitam putih sehingga dapat diketahui daerah mana yang termasuk objek dan *background* dari citra secara jelas [8]. Menurut Hietania dkk. [9], *thresholding* merupakan teknik yang sederhana dan efektif untuk segmentasi citra. *Thresholding* diklasifikasikan menjadi dua yaitu *global thresholding* dan *local thresholding (adaptive)*.

- *Global Thresholding*

Global thresholding merupakan salah satu teknik *thresholding* yang paling sederhana. Apabila nilai ambang t bergantung hanya pada satu nilai aras keabuan $f(y, x)$, *thresholding* disebut global [10].

- *Local Thresholding*

Local thresholding merupakan salah satu teknik *thresholding* yang menghitung nilai *threshold* secara lokal. *Thresholding* disebut lokal jika nilai ambang t bergantung pada $f(y, x)$ dan $g(y, x)$ dengan $g(y, x)$ menyatakan properti citra lokal pada titik (y, x) [10].

C. Operasi Morfologi

Menurut Kadir dan Susanto [10], operasi morfologi merupakan operasi yang umum dikenakan pada citra biner (hitam-putih) untuk mengubah struktur bentuk objek yang terkandung dalam citra.

- *Dilasi*

Dilasi merupakan salah satu operasi morfologi yang digunakan untuk mendapatkan efek pelebaran dengan menambah piksel pada citra digital.

- *Erosi*

Erosi merupakan salah satu operasi morfologi yang digunakan untuk mendapatkan efek penipisan dengan mengurangi piksel pada citra digital.

D. Operator Canny

Operator *Canny* merupakan operator deteksi tepi yang dikemukakan pada tahun 1986 oleh John F. Canny. Menurut Hermawati [11], operator ini diformulasikan dengan tiga tujuan utama yaitu *optimal detection* dengan tidak adanya respon yang salah (palsu), *good localization* dengan jarak minimal antara posisi tepi sebenarnya dengan posisi yang terdeteksi, dan *single response* dengan mengeliminasi *multiple response* menjadi satu garis tepi saja.

E. Gaussian Blur

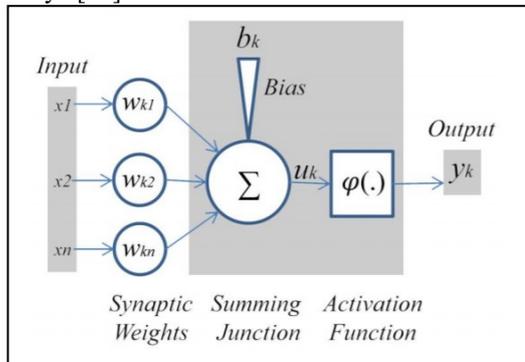
Gaussian blur adalah filter blur yang menempatkan warna transisi yang signifikan dalam sebuah *image*, kemudian membuat warna-warna pertengahan untuk menciptakan efek lembut pada sisi-sisi sebuah *image* [12]. *Gaussian blur* memberikan efek blur dengan menggunakan filter *gaussian*. Filter *gaussian* merupakan filter linear dengan nilai pembobotan pada piksel-piksel yang dipilih berdasarkan bentuk fungsi *gaussian*. Filter *Gaussian* biasanya digunakan dalam pengolahan citra digital untuk penghalusan citra atau penghilangan derau (*noise*).

F. Jaringan Saraf Tiruan

Jaringan saraf tiruan adalah paradigma pengolahan informasi yang terinspirasi oleh sistem saraf secara biologis, seperti informasi pada otak manusia [13]. Jaringan saraf tiruan merupakan salah satu upaya manusia untuk memodelkan fungsi dari sistem saraf manusia dalam memproses informasi untuk melaksanakan suatu tugas. Jaringan saraf tiruan tersusun atas sejumlah neuron yang dihubungkan oleh *link*. Setiap *link* memiliki bobot yang digunakan untuk mengirimkan informasi yang diterima dari satu neuron ke neuron lain yang terhubung. Bobot tersebut akan selalu diperbaharui selama proses pembelajaran.

Satu sel saraf dalam jaringan saraf tiruan terdiri dari tiga bagian yaitu *summing function*, fungsi aktivasi, dan *ouput*. *Summing function* adalah fungsi yang digunakan untuk menjumlahkan nilai bobot yang ada pada setiap *link* yang terhubung dengan neuron. Sedangkan, fungsi aktivasi adalah fungsi yang digunakan untuk mengaktifkan neuron. Informasi input pada jaringan saraf tiruan akan dikirimkan ke setiap neuron dengan nilai bobot tertentu. Kemudian, nilai bobot akan dijumlahkan

dengan *summing function* dan hasil dari penjumlahan akan dibandingkan dengan suatu nilai ambang (*threshold*) tertentu melalui fungsi aktivasi setiap neuron. Apabila input tersebut melewati suatu nilai ambang (*threshold*) tertentu, maka neuron tersebut akan diaktifkan. Jika tidak, maka neuron tersebut tidak akan diaktifkan. Apabila neuron tersebut diaktifkan, maka neuron tersebut akan mengirimkan *output* melalui bobot-bobot *outputnya* ke semua neuron yang berhubungan dengannya, demikian seterusnya [14].



Gambar 2. Model Saraf

Jaringan saraf tiruan memiliki beberapa lapisan yaitu lapisan input, lapisan tersembunyi (*hidden layer*), dan lapisan *output*. Pada lapisan input, setiap unit menerima pola inputan data dari luar. Pada lapisan *output*, *output* yang dihasilkan dari lapisan ini akan menjadi solusi jaringan saraf tiruan terhadap suatu tugas. Sementara, lapisan tersembunyi (*hidden layer*) adalah lapisan yang menghubungkan lapisan input dan lapisan *output*. *Output* yang dihasilkan lapisan tersembunyi (*hidden layer*) tidak dapat diamati secara langsung.

G. Backpropagation

Backpropagation merupakan algoritma pembelajaran yang terawasi dan biasanya digunakan oleh perceptron dengan banyak lapisan untuk mengubah bobot-bobot yang terhubung dengan neuron-neuron yang ada pada lapisan tersembunyinya [15]. Proses pelatihan jaringan dengan algoritma ini terdiri dari tiga tahap yaitu tahap perambatan maju (*forward propagation*), tahap perambatan balik (*backpropagation*), dan tahap perubahan bobot dan bias. *Forward propagation* digunakan untuk membawa data pada input sampai ke *output*. Sedangkan, *backpropagation* digunakan untuk menyesuaikan bobot-bobot dan bias berdasarkan nilai *error* yang didapatkan dengan arah mundur.

Langkah-langkah yang dilakukan pada pada proses perubahan bobot adalah sebagai berikut :

1. Inisialisasi nilai bobot dengan menetapkan nilai bobot awal secara acak.
2. Lakukan *forward propagation* selama kondisi berhenti bernilai salah atau nilai *epoch* (set

pelatihan pada jaringan) lebih kecil dari nilai maksimum *epoch*.

Langkah-langkah yang dilakukan pada tahap *forward propagation* :

1. Setiap unit input (X_i , $i = 1, 2, \dots, n$) menerima sinyal X_i dan mengirim sinyal tersebut ke semua unit pada lapisan tersembunyi (*hidden layer*).
2. Setiap unit tersembunyi (Z_j , $j = 1, 2, \dots, p$) menjumlahkan bobot sinyal input dengan persamaan :

$$Zin_j = V_{0j} + \sum_{i=1}^n X_i V_{ij} \quad (1)$$

gunakan fungsi aktivasi untuk menghitung sinyal *output* :

$$Z_j = f(Zin_j) \quad (2)$$

3. Setiap unit tersembunyi (Y_k , $k = 1, 2, \dots, m$) menjumlahkan bobot sinyal input dengan persamaan :

$$Yin_k = W_{0k} + \sum_{j=1}^p Z_j W_{jk} \quad (3)$$

gunakan fungsi aktivasi untuk menghitung sinyal *output* :

$$Y_k = f(Yin_k) \quad (4)$$

setelah itu, kirim sinyal ke lapisan *output*.

Langkah-langkah yang dilakukan pada tahap *backpropagation* :

1. Setiap unit *output* (Y_k , $k = 1, 2, \dots, m$) menerima pola target yang sesuai pada pola input pelatihan, hitung informasi *error* dengan persamaan :

$$\delta_k = (t_k - Y_k) f'(Yin_k) \quad (5)$$

hitung koreksi nilai bobot yang akan digunakan untuk memperbaharui bobotnya dengan persamaan :

$$\Delta W_{jk}(t) = \alpha \delta_k Z_j \quad (6)$$

hitung koreksi nilai bias yang akan digunakan untuk memperbaharui biasnya dengan persamaan :

$$\Delta W_{0k}(t) = \alpha \delta_k \quad (7)$$

Kirimkan nilai δ_k ke setiap unit yang berada pada lapisan dibawahnya.

2. Setiap unit pada lapisan tersembunyi (Z_j , $j = 1, 2, \dots, p$) jumlahkan hasil perubahan input dari setiap unit pada lapisan diatasnya dengan persamaan :

$$\delta in_j = \sum_{k=1}^m \delta_k W_{jk} \quad (8)$$

kalikan nilai dengan turunan dari fungsi aktivasinya untuk menghitung informasi *error* :

$$\delta_j = \delta in_j f'(Zin_j) \quad (9)$$

hitung koreksi nilai bobot dengan persamaan :

$$\Delta V_{jk}(t) = \alpha \delta_j X_i \quad (10)$$

hitung koreksi nilai bias dengan persamaan :

$$\Delta V_{0k}(t) = \alpha \delta_j \quad (11)$$

Langkah-langkah yang dilakukan pada tahap perubahan bobot dan bias :

1. Setiap unit *output* (Y_k , $k = 1, 2, \dots, m$) perbaharui nilai bias dan nilai bobot ($j = 0, \dots, p$) dengan persamaan :

$W_{jk}(baru) = W_{jk}(lama) + \Delta W_{jk}$ (12)
 untuk setiap unit lapisan tersembunyi ($Z_j, j = 1, 2, \dots, p$) perbaharui nilai bias dan nilai bobot ($i = 0, \dots, n$) dengan persamaan :

$$V_{ij}(baru) = V_{ij}(lama) + \Delta V_{ij} \quad (13)$$

2. Tes kondisi berhenti.

H. Fungsi Aktivasi

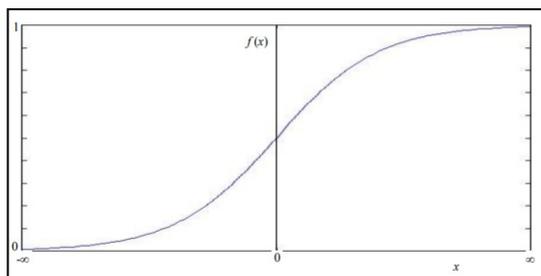
Fungsi aktivasi adalah fungsi yang digunakan pada jaringan saraf tiruan untuk mengaktifkan neuron. Menurut Suhardi [16], karakteristik yang harus dimiliki oleh fungsi aktivasi jaringan backpropagation antara lain harus kontinyu, terdiferensialkan, dan tidak menurun secara monotonis (*monotonically non-decreasing*). Beberapa fungsi aktivasi yang biasa digunakan dalam algoritma backpropagation diantaranya :

1. Fungsi Sigmoid

Fungsi ini memiliki jangkauan nilai antara 0 sampai 1. Fungsi sigmoid biner diilustrasikan pada Gambar 3. Fungsi sigmoid didefinisikan dengan persamaan pada Persamaan (14) dan fungsi turunan sigmoid didefinisikan dengan persamaan pada Persamaan (15).

$$y = f(x) = \frac{1}{(1 + e^{-x})} \quad (14)$$

$$y = f'(x) = \frac{1}{(1 + e^{-x})} \times (1 - \frac{1}{(1 + e^{-x})}) \quad (15)$$



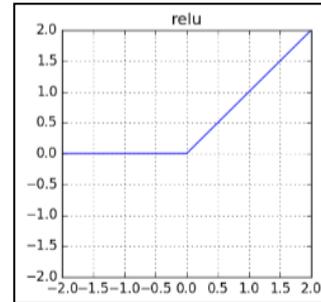
Gambar 3. Fungsi Aktivasi Sigmoid

2. Fungsi Rectified Linear Unit (ReLU)

Rectified linear unit (ReLU) adalah salah satu fungsi aktivasi yang memiliki perhitungan yang sederhana. Menurut Wibawa [17], kelebihan ReLU akan muncul saat berhadapan dengan jaringan yang memiliki neuron yang banyak sehingga dapat mengurangi waktu training dan testing dengan signifikan. Fungsi ReLU diilustrasikan pada Gambar 4. Fungsi ReLU didefinisikan dengan persamaan pada Persamaan (16) dan fungsi turunan ReLU didefinisikan dengan persamaan pada Persamaan (17).

$$y = f(x) = \max(0, x) \quad (16)$$

$$y = f'(x) = \begin{cases} 1, & x > 0 \\ 0, & x \leq 0 \end{cases} \quad (17)$$



Gambar 4. Fungsi Aktivasi ReLU

III. PERANCANGAN SISTEM

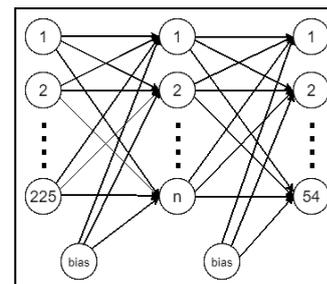
A. Perancangan Jaringan Saraf Tiruan

Arsitektur jaringan saraf tiruan yang digunakan dalam penelitian adalah jaringan lapisan banyak (*multilayer network*). Jaringan ini terdiri dari tiga lapisan yaitu lapisan input, lapisan tersembunyi, dan lapisan *output*. Lapisan input terdiri dari 225 node yang merupakan piksel-piksel dari citra input yang sudah dinormalisasi menjadi 15×15 piksel. Lapisan *output* terdiri dari 54 node yang merepresentasikan karakter A-Z, a-z, serta tanda baca titik (.) dan koma (,). Berdasarkan Persamaan (18), jumlah node yang digunakan pada lapisan tersembunyi adalah 110. Sementara, learning rate yang digunakan berdasarkan pada Persamaan (19) adalah 0,29. Perhitungan jumlah node pada lapisan tersembunyi dan learning rate mengacu pada penelitian yang dilakukan dalam penelitian [18].

$$N^{(h)} = \sqrt{N^{(i)}N^{(o)}} \quad (18)$$

$$\eta = 32/\sqrt{N^{(i)}N^{(o)}} \quad (19)$$

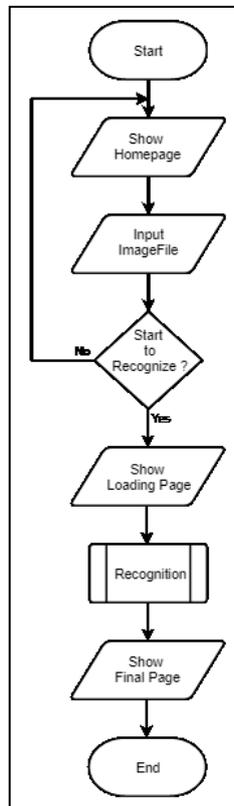
Fungsi aktivasi yang digunakan pada lapisan tersembunyi adalah ReLU dan fungsi aktivasi yang digunakan pada lapisan *output* adalah sigmoid. Pada tahap pelatihan, digunakan seluruh algoritma *backpropagation* untuk mendapatkan bobot dan bias optimal. Sedangkan, pada tahap pengenalan hanya dilakukan *forward propagation* dengan bobot dan bias yang sudah didapat pada tahap pelatihan. Arsitektur jaringan saraf tiruan yang digunakan dalam penelitian ini dapat dilihat pada Gambar 5.



Gambar 5. Arsitektur Jaringan Saraf Tiruan

B. Flowchart

Gambar 6. menunjukkan alur proses aplikasi secara keseluruhan. Ketika aplikasi dijalankan, halaman utama ditampilkan. Ketika *image file* sudah dipilih dan tombol *start to recognize* ditekan maka halaman *loading* ditampilkan dan proses *recognition* dijalankan. Setelah proses *recognition* selesai dijalankan maka halaman akhir ditampilkan.



Gambar 6. Flowchart Aplikasi

Pada proses *recognition* akan dilakukan pemrosesan gambar yang dimulai dengan pengubahan gambar menjadi *grayscale image*. Setelah input diubah menjadi *grayscale image*, proses *line extraction* dijalankan. Proses *line extraction* mengembalikan jumlah baris dan *array* yang berisi *image file* yang sudah dipotong per baris. Setelah proses *line extraction* dilakukan, dijalankan iterasi sebanyak jumlah baris. Selama iterasi dilakukan, proses *word extraction* dijalankan. Proses *word extraction* mengembalikan jumlah kata dan *array* yang berisi *image file* yang sudah dipotong per kata. Selanjutnya, dijalankan iterasi sebanyak jumlah kata. Selama iterasi dilakukan, proses *letter extraction* dijalankan. Proses *letter extraction* mengembalikan jumlah huruf dan *array* yang berisi *image file* yang sudah dipotong per huruf. Setelah itu, dijalankan iterasi sebanyak jumlah kata. Selama iterasi dilakukan, dijalankan proses *normalize* dan *forward propagation*. Proses *normalize* akan mengembalikan *image file* yang sudah dinormalisasi menjadi 15×15 piksel. *Image file* yang dikembalikan setelah proses

normalize dijadikan input pada proses *forward propagation* untuk dikenali. Proses *forward propagation*, mengembalikan *array* yang berisi *output* dari lapisan *output* dan lapisan tersembunyi. Setelah proses *forward propagation* selesai dilakukan, nilai maksimum dari *array* yang berisi *output* dari lapisan *output* dijadikan indeks untuk diklasifikasikan sebagai karakter.

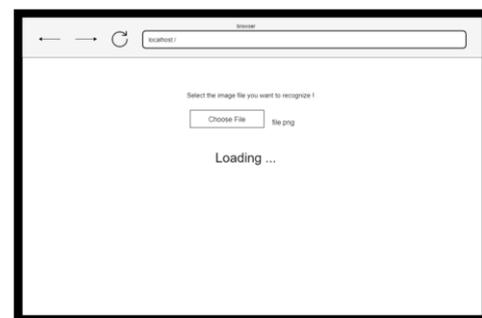
C. Rancangan Antarmuka Aplikasi

Rancangan antarmuka untuk implementasi jaringan saraf tiruan *backpropagation* untuk pengenalan karakter pada dokumen tercetak terdiri dari halaman awal, halaman *loading*, dan halaman akhir. Berikut ini merupakan rancangan antarmuka untuk implementasi jaringan saraf tiruan *backpropagation* untuk pengenalan karakter pada dokumen tercetak.



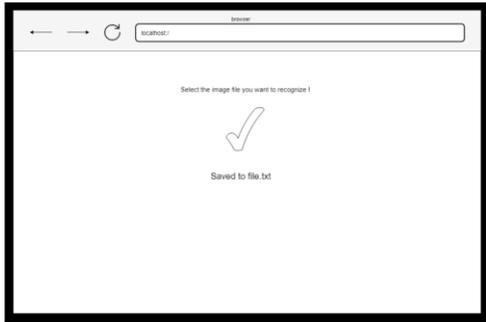
Gambar 7. Rancangan Halaman Awal

Gambar 7. merupakan rancangan antarmuka halaman awal aplikasi. Pada halaman ini terdapat tombol *choose file* dan *start to recognize*. Tombol *choose file* digunakan untuk memilih *image file* yang ingin dikenali. Sedangkan, tombol *start to recognize* digunakan untuk memulai pengenalan pada *image file*.



Gambar 8. Rancangan Halaman Loading

Gambar 8. merupakan rancangan antarmuka halaman *loading* aplikasi. Halaman ini ditampilkan saat proses pengenalan sedang dilakukan.



Gambar 9. Rancangan Halaman Akhir

Gambar 9. merupakan rancangan antarmuka halaman akhir aplikasi. Halaman ini ditampilkan saat proses pengenalan sudah selesai dilakukan.

Format persamaan merupakan suatu pengecualian di dalam spesifikasi *template* ini. Anda harus menentukan apakah akan menggunakan jenis tulisan Times New Roman atau Symbol (jangan jenis tulisan yang lain). Bila Anda membuat beberapa persamaan berbeda, akan lebih baik bila Anda mempersiapkan persamaan tersebut sebagai gambar dan menyisipkannya ke dalam artikel Anda setelah diberi *style*.

Beri penomoran untuk persamaan Anda secara berurutan. Nomor persamaan berada dalam tanda kurung seperti (1), dan diletakkan pada bagian kanan dengan menggunakan suatu *right tab stop*.

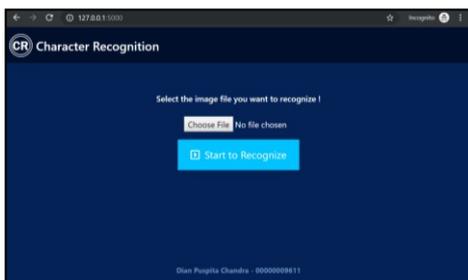
$$\int_0^{r_2} F(r, \phi) dr d\phi = [\sigma r_2 / (2\mu_0)] \quad (1)$$

Perhatikan bahwa persamaan di atas diposisikan di bagian tengah dengan menggunakan suatu *center tab stop*. Pastikan bahwa simbol-simbol yang digunakan dalam persamaan Anda didefinisikan sebelum atau sesudah persamaan. Gunakan "(1)," bukan "Persamaan (1)," kecuali pada awal sebuah kalimat, seperti "Persamaan (1) merupakan"

IV. IMPLEMENTASI DAN UJI COBA

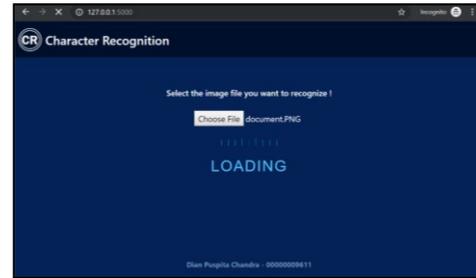
A. Implementasi Antarmuka

Gambar 10. menunjukkan hasil implementasi halaman awal. Pada halaman ini terdapat dua tombol yaitu *choose file* dan *start to recognize*. Tombol *choose file* digunakan untuk memilih *image file* yang ingin dikenali. Sedangkan, tombol *start to recognize* digunakan untuk memulai pengenalan pada *image file*.



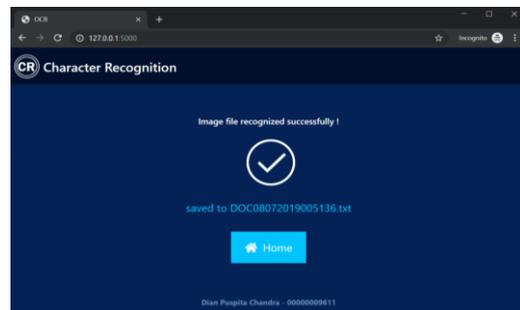
Gambar 10. Hasil Implementasi Halaman Awal

Gambar 11. menunjukkan hasil implementasi halaman *loading*. Halaman ini ditampilkan ketika tombol *start to recognize* pada halaman utama ditekan. Halaman ini ditampilkan sampai proses pengenalan selesai dilakukan.



Gambar 11. Hasil Implementasi Halaman Loading

Gambar 12. merupakan hasil implementasi halaman akhir. Halaman ini menampilkan nama *file* yang digunakan untuk menyimpan hasil pengenalan dan tombol *home* yang akan menampilkan halaman awal jika ditekan. Halaman ini ditampilkan setelah halaman *loading* setelah proses pengenalan selesai dilakukan.



Gambar 12. Hasil Implementasi Halaman Akhir

B. Uji Coba Jaringan Saraf Tiruan

Uji coba jaringan saraf tiruan digunakan untuk melihat kebenaran dari implementasi kode jaringan saraf tiruan *backpropagation* yang telah dibuat. Jaringan saraf tiruan diuji dengan menjalankan logika AND, OR, dan XOR. Arsitektur jaringan yang digunakan untuk uji coba adalah jaringan lapisan banyak (*multilayer network*) dengan tiga lapisan yaitu lapisan input dengan dua node, lapisan tersembunyi dengan tiga node, dan lapisan *output* dengan dua node yang merepresentasikan 0 dan 1. *Learning rate* yang digunakan adalah 0,1 dengan target *error* 0,01. Jaringan akan dilatih sampai total *error* lebih kecil dari target *error*.

X1	X2	T
0	0	0
1	0	1
0	1	1
1	1	1

Gambar 13. Tabel Kebenaran Logika OR

Gambar 13. merupakan gambar tabel kebenaran logika OR. Pada logika OR, *output* bernilai nol jika kedua input bernilai nol. Pelatihan jaringan saraf tiruan dengan logika OR dilakukan dengan *epoch* sebanyak 161 dengan semua hasil uji benar. Hasil uji logika OR dapat dilihat pada Gambar 14.

Input	Output	Output pada Lapisan Output
[0. 0.]	0	[0.6980956521367753, 0.5521530076446559]
[0. 1.]	1	[0.20347556695954708, 0.7530506102134477]
[1. 0.]	1	[0.13256140609376277, 0.8243350577763973]
[1. 1.]	1	[0.03692253339567626, 0.9082554483297861]

Gambar 14. Hasil Uji Logika OR

X ₁	X ₂	T
0	0	0
1	0	1
0	1	1
1	1	0

Gambar 15. Tabel Kebenaran Logika XOR

Gambar 15. merupakan gambar tabel kebenaran logika XOR. Pada logika XOR, *output* bernilai nol jika kedua input bernilai sama. Pelatihan jaringan saraf tiruan dengan logika XOR dilakukan dengan *epoch* sebanyak 1794 dengan semua hasil uji benar. Hasil uji logika XOR dapat dilihat pada Gambar 16.

Input	Output	Output pada Lapisan Output
[0. 0.]	0	[0.91498314366016, 0.08824166371790268]
[0. 1.]	1	[0.08708269277816834, 0.9099110503987154]
[1. 0.]	1	[0.06800521021736916, 0.9285870375298074]
[1. 1.]	0	[0.9311800476349783, 0.07166826871819867]

Gambar 16. Hasil Uji Logika XOR

Gambar 17. merupakan gambar tabel kebenaran logika AND. Pada logika AND, *output* bernilai satu jika kedua input bernilai satu. Pelatihan jaringan saraf tiruan dengan logika AND dilakukan dengan *epoch* sebanyak 2200 dengan semua hasil uji benar. Hasil uji logika AND dapat dilihat pada Gambar 18.

X ₁	X ₂	T
0	0	0
1	0	0
0	1	0
1	1	1

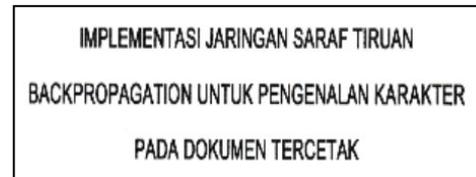
Gambar 17. Tabel Kebenaran Logika AND

Input	Output	Output pada Lapisan Output
[0. 0.]	0	[0.9997257418370317, 0.00030755987632060414]
[0. 1.]	0	[0.9468622708217166, 0.053811517802038585]
[1. 0.]	0	[0.9380910725429703, 0.06540355640372532]
[1. 1.]	1	[0.06896374307797372, 0.9282451695412202]

Gambar 18. Hasil Uji Logika AND

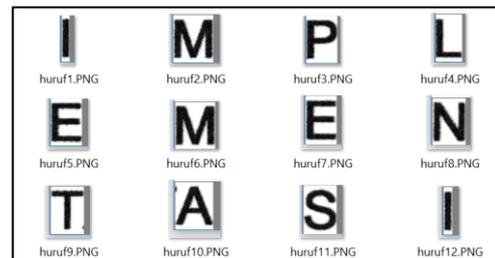
C. Evaluasi Hasil

Pengujian aplikasi dilakukan dengan menggunakan dokumen tercetak yang sudah dijadikan citra digital. Proses perubahan dokumen tercetak menjadi citra digital dilakukan menggunakan alat pemindai (*scanner*). Gambar 19. merupakan salah satu citra digital dengan *font* Arial Unicode Ms yang dijadikan sebagai input.



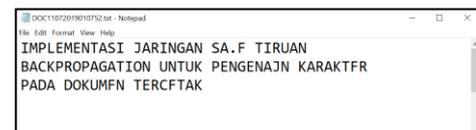
Gambar 19. Citra dengan Font Arial Unicode Ms

Citra akan diproses dengan proses *line extraction*, *word extraction*, dan *letter extraction* untuk mendapatkan potongan huruf dari citra. Citra yang sudah dipotong disimpan ke dalam *array* kemudian citra dinormalisasi dan dijadikan input pada proses *forward propagation* untuk dikenali sebagai karakter. Gambar 20. Merupakan hasil dari proses *letter extraction* dari citra kata pertama dari proses *word extraction* dan citra baris pertama dari proses *line extraction*.



Gambar 20. Hasil Proses Letter Extraction

Setelah proses pengenalan selesai dilakukan, hasil dari pengenalan karakter ditulis ke dalam *file*. *File* tersebut dijadikan sebagai *output* dari aplikasi dengan ekstensi .txt yang dapat dilihat pada Gambar 21.



Gambar 21. Hasil Output Aplikasi

Jumlah data *training* yang digunakan untuk pelatihan jaringan saraf tiruan adalah 486 data. Jumlah data *training font* Arial Unicode Ms adalah 162 data dengan 54 karakter yang masing-masing berjumlah tiga data. Jumlah data *training font* Tahoma adalah 162 data dengan 54 karakter yang masing-masing berjumlah tiga data. Jumlah data *training font* Times New Roman adalah 162 data dengan 54 karakter yang masing-masing berjumlah tiga data.

Pengujian dilakukan dengan 15 dokumen tercetak yang sudah berupa citra digital dengan jenis *font* Arial Unicode Ms, Tahoma, dan Times New Roman masing-masing sebanyak lima. Seluruh citra digital dihitung tingkat akurasi dengan membagi jumlah karakter uji yang dikenali dengan jumlah karakter uji keseluruhan. Akurasi yang dihasilkan dari hasil penelitian dapat dilihat pada Tabel 1.

Tabel 1. Hasil Uji Coba

Nama	Font	Persentase Tingkat Akurasi
Citra Input 1	Arial Unicode Ms	92 %
Citra Input 2	Arial Unicode Ms	91 %
Citra Input 3	Arial Unicode Ms	98 %
Citra Input 4	Arial Unicode Ms	94 %
Citra Input 5	Arial Unicode Ms	97 %
Citra Input 6	Tahoma	96 %
Citra Input 7	Tahoma	97 %
Citra Input 8	Tahoma	98 %
Citra Input 9	Tahoma	96 %
Citra Input 10	Tahoma	96 %
Citra Input 11	Times New Roman	95 %
Citra Input 12	Times New Roman	94 %
Citra Input 13	Times New Roman	93 %
Citra Input 14	Times New Roman	95 %
Citra Input 15	Times New Roman	93 %

Nilai rata-rata tingkat akurasi dari setiap *font* didapat dengan perhitungan sebagai berikut.

$$\text{Akurasi Arial Unicode Ms} = \frac{(0,92 + 0,91 + 0,98 + 0,94 + 0,97)}{5} \times 100 \%$$

$$\text{Akurasi Arial Unicode Ms} = 94 \%$$

$$\text{Akurasi Tahoma} = \frac{(0,96 + 0,97 + 0,98 + 0,96 + 0,96)}{5} \times 100 \%$$

$$\text{Akurasi Tahoma} = 96,6 \%$$

$$\text{Akurasi Times New Roman} = \frac{(0,95 + 0,94 + 0,93 + 0,95 + 0,93)}{5} \times 100 \%$$

$$\text{Akurasi Times New Roman} = 94 \%$$

Jenis *font* Tahoma menghasilkan rata-rata tingkat akurasi yang paling tinggi yaitu sebesar 96,6 % disusul dengan rata-rata tingkat akurasi jenis *font* Arial Unicode Ms sebesar 94 % dan Jenis *font* Times New Roman sebesar 94 %.

V. SIMPULAN

Aplikasi pengenalan karakter pada dokumen tercetak dibuat dengan bahasa pemrograman python

dan OpenCV *library* untuk bagian pengolahan citra. Arsitektur jaringan saraf tiruan yang digunakan adalah jaringan lapisan banyak (*multilayer network*) dengan tiga lapisan. Lapisan input terdiri dari 225 node dengan masukan berupa citra digital berukuran 15 × 15 piksel, lapisan tersembunyi terdiri dari 110 node, dan lapisan *output* terdiri dari 54 node yang merepresentasikan karakter A-Z, a-z, serta tanda baca titik (.) dan koma (,). *Learning rate* yang digunakan dalam penelitian ini adalah 0,29. Jaringan dibuat agar bisa mengenali karakter dengan jenis *font* Arial Unicode Ms, Tahoma, dan Times New Roman. Rata-rata tingkat akurasi yang didapat adalah 94 % untuk jenis *font* Arial Unicode Ms, 96,6 % untuk jenis *font* Tahoma, dan 94 % untuk jenis *font* Times New Roman.

DAFTAR PUSTAKA

- [1] R. Susilawati. (2012). Strategi Paperless Office di STIE Nusa Megarkencana. *Jurnal Ekonomi dan Kewirausahaan*, [online] 8(16), hal.61-69. Tersedia di: <http://ejournal.stienusa.ac.id/index.php/ekowir/article/view/35>
- [2] R. Mithe, S. Indalkar, dan N. Divekar. (2013). Optical Character Recognition. *International Journal of Recent Technology and Engineering*, [online] 2(1), hal.72-75. Tersedia di: <https://pdfs.semanticscholar.org/6a4b/4f04d5ce3c3592832eb40c23cc8fc5a9131e.pdf>
- [3] P. Bairagi. (2018). Optical Character Recognition for Hindi. *International Research Journal of Engineering and Technology*, [online] 5(5), hal.3968-3973. Tersedia di: <https://www.irjet.net/archives/V5/I5/IRJET-V5I5846.pdf>
- [4] D. Li, P. Yu, H. Li, dan G. Peng, "Printed New Tai Lue Character Recognition Based on BP Neural Network," dalam *IEEE International Conference on Signal and Image Processing*, 2016.
- [5] A. Radhiah, C. Machbub, E. Hidayat, dan A. Prihatmanto, "Printed Arabic Letter Recognition Based On Image," dalam *International Conference on Signals and Systems*, 2018.
- [6] S. Afroge, B. Ahmed, dan F. Mahmud, "Optical Character Recognition using Back Propagation Neural Network," dalam *International Conference on Electrical, Computer & Telecommunication Engineering*, 2016.
- [7] M. Nugroho, "Perbandingan Beberapa Metode Praproses Pada Pengenalan Karakter Tulisan Tangan Menggunakan Jaringan Syaraf Tiruan Propagasi Balik," Skripsi, Jurusan Ilmu Komputer, Institut Pertanian Bogor. Indonesia, 2003.
- [8] I. Malik, "Pembuatan Aplikasi Pengolahan Citra Teks Inggris Dan Penerjemahannya Ke Dalam Bahasa Indonesia Berbasis Sistem Operasi Android," Skripsi, Universitas Komputer Indonesia. Indonesia, 2014.
- [9] F. Hietania, W. Santiyasa, dan I. Dwidasmara. (2012). Implementasi Backpropagation Dalam Pengolahan Citra Teks Tulisan Tangan Menjadi Teks Digital. *Jurnal Elektronik Ilmu Komputer Udayana*, [online] 1(1), hal.1-10. Tersedia di: <https://ojs.unud.ac.id/index.php/JLK/article/view/2789>
- [10] A. Kadir, dan A. Susanto, *Teori dan Aplikasi Pengolahan Citra*. Yogyakarta: ANDI, 2013.
- [11] F. Hermawati, *Pengolahan Citra Digital Konsep dan Teori*. Yogyakarta: ANDI, 2013.
- [12] A. Wedianto, H. Sari, dan Y. Suzantri. (2016). Analisa Perbandingan Metode Filter Gaussian, Mean dan Median Terhadap Reduksi Noise. *Jurnal Media Infotama*, [online]

- 12(1), hal.21-30. Tersedia di: <https://jurnal.unived.ac.id/index.php/jmi/article/view/269>
- [13] T. Sutojo, E. Mulyanto, dan V. Suhartono, *Kecerdasan Buatan*. Yogyakarta: ANDI, 2011.
- [14] E. Handoyo, dan L. Susanto. (2011). Penerapan Jaringan Syaraf Tiruan metode Propagasi Balik Dalam Pengenalan Tulisan Tangan Huruf Jepang Jenis Hiragana dan Katakana. *Jurnal Informatika*, [online] 7(1), hal.39-55. Tersedia di : <https://repository.maranatha.edu/373/>
- [15] I. Wirawan, *Metode Penalaran dalam Kecerdasan Buatan*. Depok: Rajawali Pers, 2017, hal.135-186.
- [16] I. Suhardi. (2007). Evaluasi Pengaruh Fungsi Aktivasi Dan Parameter Kemiringannya Terhadap Unjuk Kerja Pengenalan Jaringan Syaraf Tiruan. *Jurnal Ilmiah Semesta Teknika*, [online] 10(1), hal.53-68. Tersedia di: <http://journal.umy.ac.id/index.php/st/article/view/854>
- [17] M. Wibawa. (2017). Pengaruh Fungsi Aktivasi, Optimisasi dan Jumlah Epoch Terhadap Performa Jaringan Saraf Tiruan. *Jurnal Sistem Dan Informatika (JSI)*, [online] 11(2), hal.167-174. Tersedia di: <https://jsi.stikombali.ac.id/index.php/jsi/article/view/129>
- [18] K. Shibata, dan Y. Ikeda, "Effect of number of hidden neurons on learning in large-scale layered neural networks," dalam *ICROS-SICE International Joint Conference*, 2009.

Penerapan Metode Fuzzy AHP untuk Sistem Pendukung Keputusan Pemilihan Pemasok Terbaik

Fenina Adline Twince Tobing¹, Muhammad Iqbal Dzulhaq², Ramzi Fajar Sidiq³

¹Program Studi Informatika, Fakultas Teknik dan Informatika Universitas Multimedia Nusantara Tangerang, Banten

²Program Studi Teknik Informatika STMIK Bina Sarana Global Tangerang, Banten

³Program Studi Sistem Informasi STMIK Bina Sarana Global Tangerang, Banten

¹fenina.tobing@umn.ac.id, ²miqbaldzulhaq@stmikglobal.ac.id, ³ramzifajar@gmail.com

Diterima 31 Oktober 2019

Disetujui 20 Desember 2019

Abstrak—Keberadaan pemasok sangat menentukan kelancaran proses produksi dan produk yang akan dihasilkan. Masalah yang dihadapi perusahaan adalah proses evaluasi atau pemilihan pemasok terbaik masih bersifat manual, sulitnya menentukan pemasok mana yang memiliki performansi terbaik, keterlambatan datangnya bahan baku mengakibatkan keterlambatan proses produksi yang berujung pada tidak tercapainya jumlah produksi. Pada penelitian ini dibangun Sistem Pendukung Keputusan (SPK) untuk pemilihan pemasok terbaik menggunakan 4 kriteria yang disesuaikan dengan kondisi dan kebutuhan perusahaan yaitu harga, stok, *delivery*, dan mutu. Masing-masing kriteria tersebut akan dibobotkan menggunakan metode *Fuzzy AHP*. Pada implementasinya dihasilkan bobot kriteria harga sebesar 0,632, stok 0,352, *delivery* 0,084, mutu 0,107. Hasil pembobotan ini akan digunakan untuk mendukung penilaian pemasok terbaik dan akan diperoleh nilai masing-masing pemasok sehingga diketahui siapa pemasok terbaik.

Kata Kunci: Sistem Pendukung Keputusan, *Fuzzy AHP*, Pemasok.

I. PENDAHULUAN

Persaingan didunia bisnis saat ini terasa semakin ketat, terutama semenjak perjanjian perdagangan bebas antara negara-negara ASEAN dan Cina mulai diberlakukan pada tanggal 1 Januari 2010. Seiring dengan berlakunya ACFTA (*ASEAN-China Free Trade Agreement*) mulai timbul kekhawatiran masyarakat Indonesia terkait produk lokal yang rawan tergerus produk import Cina dan persaingan berbagai sektor industri yang akan semakin ketat. Diperkirakan ada sepuluh sektor industri yang paling dirugikan dalam perjanjian perdagangan tersebut, salah satunya adalah sektor industri tekstil Indonesia.

Masalah yang dihadapi perusahaan adalah proses evaluasi atau pemilihan pemasok terbaik yang masih bersifat manual. Perusahaan mengalami kesulitan dalam menentukan pemasok mana yang memiliki performansi terbaik dari segi harga, stok, *delivery* dan mutu, sehingga perusahaan bisa memprioritaskan

pemasok tersebut dalam memenuhi bahan baku yang dibutuhkan. Selain itu, perusahaan juga mengalami keterlambatan datangnya bahan baku yang disediakan oleh perusahaan lain, kejadian ini mengakibatkan keterlambatan proses produksi yang berujung pada tidak tercapainya jumlah produksi.

Penelitian yang telah dilakukan oleh Arifin, dkk dengan merancang sebuah aplikasi berbasis web, menyediakan database sehingga pemakai dapat mengolah data masukan sistem sebagaimana diperlukan oleh perusahaan, sehingga proses biaya produksi yang terdiri dari biaya bahan baku, biaya tenaga kerja, dan biaya overhead pabrik yang disajikan dalam bentuk laporan biaya produksi sehingga manager produksi dapat mengetahui secara jelas proses produksi yang terkait dalam periode tertentu menjadi tolak ukur dalam pengambilan keputusan.^[1]

Dari permasalahan tersebut, maka perlu adanya sebuah sistem pendukung keputusan yang dapat membantu proses pengambilan keputusan pemilihan pemasok terbaik dengan metode yang diusulkan adalah *Fuzzy Analytical Hierarchy Process* (F-AHP) yang digunakan untuk pemilihan suatu alternatif dan penyesuaian masalah dengan menggabungkan konsep teori *fuzzy* dan analisis terstruktur hierarki.

II. LANDASAN TEORI

A. Sistem Pendukung Keputusan

“Sistem Pendukung Keputusan sebagai sistem berbasis komputer yang terdiri dari tiga komponen yang saling berinteraksi, sistem bahasa (mekanisme untuk memberikan komunikasi antara pengguna dan komponen sistem pendukung keputusan lain), sistem pengetahuan (respositori pengetahuan domain masalah yang ada pada sistem pendukung keputusan atau sebagai data atau sebagai prosedur), dan sistem pemrosesan masalah (hubungan antara dua komponen lainnya, terdiri dari satu atau lebih kapabilitas

manipulasi masalah umum yang diperlukan untuk pengambilan keputusan)”^[2].

B. Adobe Dreamweaver

“Adobe Dreamweaver adalah aplikasi desain dan pengembangan web yang menyediakan editor WYSIWYG (What Your See Is What Your Get) visual (lebih dikenal sebagai Design View) dan kode editor dengan fitur standar seperti syntax highlighting, code completion, dan code collapsing serta fitur lebih canggih seperti real-time syntax checking dan code introspection untuk menghasilkan petunjuk kode untuk membantu pengguna dalam menulis kode.”^[3].

C. Evaluasi Kinerja Pemasok dan Kerja Sama Dengan Pemasok

Martono mengungkapkan: “Setelah memilih dan bekerja sama dengan pemasok perusahaan melakukan penilaian terhadap kinerja pemasok secara berkala (misalnya setiap 3 bulan atau 6 bulan sekali).” Proses penilaian dilakukan secara terbuka dengan semua pemasok agar masing-masing saling mengetahui kinerja mana yang terbaik sehingga bisa dilakukan perbaikan bagi pemasok yang kurang baik.

Martono menguraikan bahwa: “prinsip-prinsip kerja sama dengan pemasok:”

1. Mempunyai tujuan yang sama.
2. Saling menguntungkan.
3. Saling percaya dan terbuka.
4. Kerja sama jangka panjang.
5. Perbaikan berkesinambungan dalam hal mutu dan biaya^[4].

D. Proses Hierarki Analitik

Marimin dan Maghfiroh menjelaskan, “Terdapat tiga prinsip dalam memecahkan persoalan dengan analisis logis eksplisir, yaitu:”^[5].

1. Penyusunan Hierarki
2. Penentuan Prioritas
3. Konsistensi Logis

E. Proses Fuzzy AHP

Tahapan pertama dalam proses fuzzy AHP yaitu melakukan perbandingan berpasangan dengan menggunakan verbal judgement kemudian proses konversi.

	V1	V2	V3
V1	Equ	Mod	Vstr
V2	Non-mod	Equ	Mod
V3	Non vstr	Not-mod	Equ

(1)

$(1, m, u) \text{ invers} \rightarrow (1/u, 1/m, 1/l)$
 Example :
 V1 vs V2 \rightarrow Mod = (1, 3, 5)
 V2 vs V1 \rightarrow not-mod = (1/5, 1/3, 1/1) = (0.2, 0.33, 1)

	V1	V2	V3
V1	1, 1, 3	1, 3, 5	5, 7, 9
V2	0.2, 0.33, 1	1, 1, 3	1, 3, 5
V3	0.11, 0.14, 0.2	0.2, 0.33, 1	1, 1, 3

(3)

Gambar 1. Skema Proses Perbandingan Berpasangan

Setelah itu dilakukan proses perhitungan titik tengah atau centroid measurement dengan membaginya menjadi tiga jenis nilai (*lower, middle dan upper*).

Untuk nilai *lower* parameter V1 adalah:

$$l_{V1} = \sqrt[3]{1 \times 1 \times 5} = \sqrt[3]{5} = 1.71$$

Untuk nilai *middle* parameter V1 adalah:

$$m_{V1} = \sqrt[3]{1 \times 3 \times 5} = \sqrt[3]{15} = 2.76$$

Untuk nilai *upper* parameter V1 adalah:

$$u_{V1} = \sqrt[3]{3 \times 5 \times 9} = \sqrt[3]{135} = 5.13$$

Untuk nilai *lower* parameter V2 adalah:

$$l_{V2} = \sqrt[3]{0.2 \times 1 \times 1} = \sqrt[3]{0.2} = 0.58$$

Untuk nilai *middle* parameter V2 adalah:

$$m_{V2} = \sqrt[3]{0.33 \times 1 \times 3} = \sqrt[3]{0.99} = 1.00$$

Untuk nilai *upper* parameter V2 adalah:

$$u_{V2} = \sqrt[3]{1 \times 3 \times 5} = \sqrt[3]{15} = 2.47$$

Untuk nilai *lower* parameter V3 adalah:

$$l_{V3} = \sqrt[3]{0.11 \times 0.2 \times 1} = \sqrt[3]{0.02} = 0.28$$

Untuk nilai *middle* parameter V3 adalah:

$$m_{V3} = \sqrt[3]{0.14 \times 0.33 \times 1} = \sqrt[3]{0.05} = 0.36$$

Untuk nilai *upper* parameter V3 adalah:

$$u_{V3} = \sqrt[3]{0.2 \times 1 \times 3} = \sqrt[3]{0.6} = 0.84$$

Selanjutnya dilakukan proses normalisasi yaitu dengan cara menjumlahkan setiap nilai untuk masing masing nilai *lower, middle dan upper*.

$$l_{total} = 1.71 + 0.58 + 0.28 = 2.57$$

$$m_{total} = 2.76 + 1 + 0.36 = 4.12$$

$$u_{total} = 5.13 + 2.47 + 0.84 = 8.44$$

Kemudian dilakukan proses normalisasi khusus untuk fungsi keanggotaan segitiga, nilai pembagi *upper* digunakan untuk membagi nilai *lower* dari parameter yang bersangkutan, dan nilai pembagi *lower* digunakan untuk membagi nilai *upper* dari parameter yang bersangkutan, sedangkan untuk nilai pembagi *middle* adalah tetap digunakan untuk membagi nilai *middle* dari parameter yang bersangkutan.

Untuk parameter V1 = (1.71, 2.76, 5.13) :
 (1.71/8.44, 2.76/4.12, 5.13/2.57) = (0.20, 0.67, 2.00)

Untuk parameter V2 = (0.58, 1, 2.47) :
 (0.58/8.44, 1/4.12, 2.47/2.57) = (0.07, 0.24, 0.96)

Untuk parameter V3 = (0.28, 0.36, 0.84) :
 (0.28/8.44, 0.36/4.12, 0.84/2.57) = (0.03, 0.09, 0.33)

Setelah mendapatkan nilai normal untuk setiap parameternya, maka proses terakhir adalah proses defuzzyfikasinya, tentu untuk mendapatkan nilai crisp output (Cr). Persamaan yang digunakan adalah persamaan dimana α adalah degree of optimism (bernilai $0 \leq \alpha \leq 1$, biasanya bernilai 0,5), l adalah *lower*, m adalah *middle*, dan u adalah *upper*.

$$Cr = 0,5 (\alpha u + m + (1-\alpha) l) \tag{4}$$

Seperti halnya pada proses normalisasi, nilai *upper* dan *lower* dipertukarkan.^[6]

III. METODE PENELITIAN

A. Objek Penelitian

Penelitian dilakukan pada PT Gee San Indonesia di Jl. Raya Serang Km.11 RT.02 RW.01 Desa Bunder Kec. Cikupa Tangerang - Banten 15710. Pada sistem yang berjalan staff administrasi mengisi form daftar evaluasi pemasok sesuai dengan data pemasok yang akan dievaluasi lalu memberikan hasil form data evaluasi pemasok yang telah diisi kepala general affair, general affair melakukan evaluasi pemasok dari data yang telah diisi oleh staff administrasi, Kemudian general affair membuat rekap hasil evaluasi pemasok dari data yang sudah diisi staff administrasi, setelah itu general affair menyerahkan dan melaporkan rekap hasil evaluasi pemasok kepada general manager, general manager menerima rekap hasil evaluasi pemasok yaitu laporan evaluasi pemasok dari general affair, general manager memeriksa laporan evaluasi pemasok apabila laporan sudah sesuai dengan kriteria maka laporan evaluasi pemasok akan di setujui dan apabila laporan belum sesuai dengan kriteria maka general manager akan mengembalikan laporan evaluasi pemasok untuk di lakukan evaluasi kembali oleh general affair.

B. Masalah yang Dihadapi

Di dalam penelitian yang dilakukan penulis dalam sistem yang sedang berjalan, penulis menemukan beberapa masalah yang terjadi pada PT Gee San Indonesia:

- Pada proses pemilihan pemasok terbaik masih bersifat manual belum terkomputerisasi.
- Sulitnya menentukan pemasok mana yang memiliki performansi terbaik dari segi harga, stok, *delivery*, dan mutu.
- Ketidakpastian kedatangan bahan baku yang dipesan kepada pemasok sehingga mengakibatkan keterlambatan proses produksi.

C. Alternatif Pemecahan Masalah

Setelah mengamati dan meneliti dari beberapa permasalahan yang terjadi pada sistem yang berjalan, penulis mengusulkan beberapa alternatif pemecahan dari permasalahan yang dihadapi, antara lain:

- Sistem Pendukung Keputusan penilaian pemasok terbaik menggunakan metode *Fuzzy AHP* yang digunakan untuk menentukan pemasok terbaik di PT Gee San Indonesia dengan berbasis *web* agar mempermudah dalam pengaksesan lebih cepat dan efektif.
- Proses penginputan data pemasok akan masuk pada *database* dan kriteria yang diberikan perusahaan akan dijadikan acuan dan dibuat perhitungannya menggunakan metode *fuzzy AHP*, sehingga dapat

membantu proses pengambilan keputusan dengan tepat.

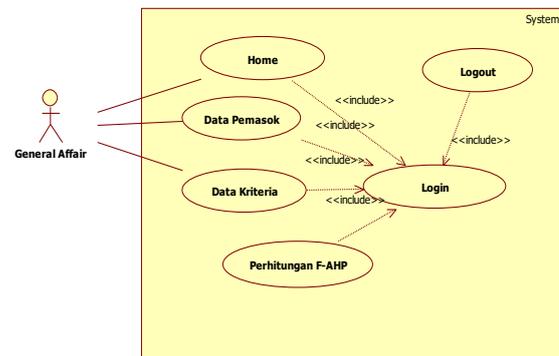
IV. HASIL DAN PEMBAHASAN

A. Usulan Prosedur Yang Baru

Pada analisa sistem baru, akan dibangun suatu Sistem Pendukung Keputusan (SPK) pemilihan pemasok terbaik dengan menerapkan metode F-AHP. Sistem akan menerima *input* (data masukan) berdasarkan kriteria yang telah disepakati dengan pihak perusahaan dan penilaian pemasok (alternatif). Kemudian akan diproses dengan menerapkan perhitungan F-AHP dan menghasilkan *output* (data keluaran) perangkingan alternatif berupa bobot penilaian calon pemasok terbaik beserta hasil nilai keputusannya yang berupa daftar rangking.

B. Diagram Rancangan Sistem

Rancangan sistem ini adalah tahapan perancangan sistem yang akan dibentuk yang berupa penggambaran proses-proses suatu elemen-elemen dari suatu komponen, proses perancangan ini merupakan suatu tahapan awal dari perancangan aplikasi dari sistem pendukung keputusan pemilihan pemasok terbaik (Gambar 3).



Gambar 2. Use Case Diagram yang diusulkan

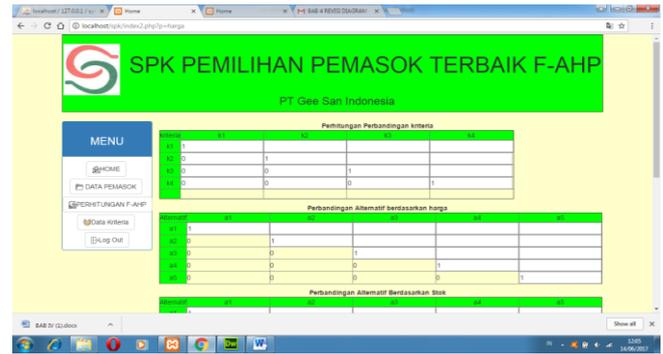
Proses perhitungan F-AHP ditentukan melalui beberapa tahapan dalam penilaian pemilihan pemasok terbaik.

- Menentukan data kriteria dan alternatif.

Tabel 1. Data Kriteria dan Alternatif

No	Kode Kriteria	Nama Kriteria	Kode Alternatif	Alternatif
1	K1	Harga	A1	PT Global Fiberindo
2	K2	Stok barang	A2	PT Trikarsa Fiberindo
3	K3	Delivery	A3	PT Hasil Damai Textile
4	K4	Mutu	A4	PT Eco Fiber
			A5	PT Hansung Fiber

2. Melakukan pemodelan *Fuzzy* AHP berdasarkan kriteria dan alternatif.
3. Melakukan representasi kedalam struktur hierarki.
4. Melakukan penilaian setiap tingkat hierarki dari masing-masing kriteria.
5. Melakukan konversi *fuzzy*.
6. Melakukan perbandingan berpasangan F-AHP.
7. Melakukan konversi *fuzzy* pada perbandingan berpasangan F-AHP.
8. Menentukan *centroid measurement*.
9. Menghitung nilai total *l*, *m*, dan *u* dari setiap kriteria.
10. Menentukan nilai sintesis *fuzzy*.
11. Menghitung *crisp output*.



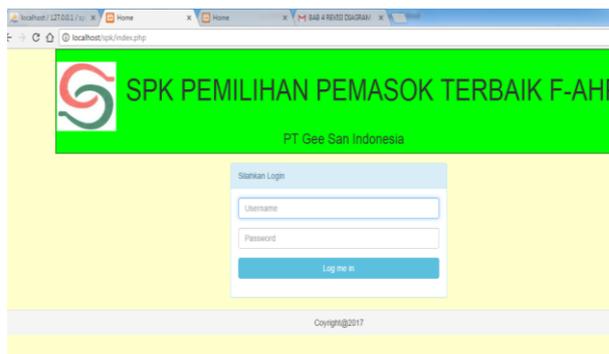
Gambar 4. Tampilan Perhitungan F-AHP

Proses perhitungan F-AHP dimulai dengan general affair memasukkan *username* dan *password*. Kemudian general affair akan melakukan perhitungan F-AHP dengan menginput perbandingan berpasangan berdasarkan kriteria dan perbandingan berpasangan alternatif berdasarkan harga, stok, delivery dan mutu. Kemudian general affair menekan tombol hitung untuk melakukan proses perhitungan dan mengetahui status konsistensi, jika status konsistensi tidak konsisten maka general affair harus melakukan perhitungan ulang sampai mendapatkan hasil yang konsisten.

Pada gambar 4 terdapat inputan yang harus di isi dengan data nilai *fuzzy* triangular, data kriteria yang digunakan terdapat 4 kriteria: harga (K1), stok (K2), *delivery* (K3), mutu (K4) dan 5 alternatif: PT Global Fiberindo (A1), PT Trikarsa Fiberindo (A2), PT Hasil Damai *Textile* (A3), PT Eco Fiber (A4), PT Hansung Fiber (A5). Dilakukan penilaian setiap tingkat hierarki dinilai melalui perbandingan berpasangan kriteria dan penilaian tingkat hierarki alternatif berdasarkan kriteria harga, stok, *delivery* dan mutu sampai kolom terisi semua selanjutnya hitung penilaian dengan klik form hitung hingga hasil konsistensi CR<0,1. Jika CR>0,1 maka harus dilakukan perhitungan ulang samapi konsistensi CR terpenuhi yaitu <0,1. Perhitungan nilai bobot yang telah diproses akan menampilkan bobot global kriteria dan alternatif kemudian bobot global dicetak untuk digunakan sebagai laporan hasil perhitungan penilaian pemasok terbaik yang akan digunakan sebagai bahan pertimbangan untuk membantu dalam penunjang keputusan.

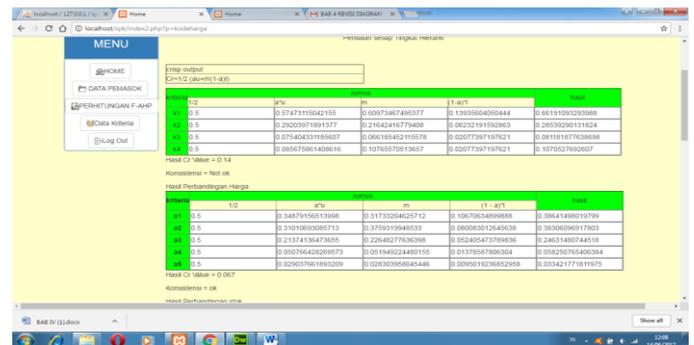
Apabila sudah konsisten, general affair akan mengetahui hasil dari perhitungan dan dapat mencetak laporan perhitungan penilaian pemasok terbaik.

C. Rancangan Tampilan
a. Tampilan Sistem



Gambar 3. Tampilan Login

Pada gambar 3 menampilkan menu untuk login, dimana terdapat kolom *username* dan *password* yang harus di input untuk dapat mengakses masuk ke dalam sistem.



Gambar 5. Tampilan Hasil Perhitungan F-AHP

Pada gambar 5 merupakan tampilan hasil perankingan F-AHP. Hasil perhitungan diatas dapat diperoleh hasil perankingan global dari setiap kriteria dan alternatif sebagai berikut:

Tabel 2. Perankingan Global

Globa l	K1	K2	K3	K4	Bobot Globo l	Rangkin g
0.5	0.34879156113968	0.3173204225712	0.19679634899888	0.3064148019796		
0.5	0.31610002051715	0.375911984853	0.08000512648298	0.3020004917953		
0.5	0.232174136473656	0.2048577536298	0.65248547379493	0.24631480744518		
0.5	0.050766432829573	0.051949224480155	0.013785878006304	0.086202765406384		
0.5	0.026037661893209	0.02830358045446	0.0095019236852958	0.033421771811975		

Bobot	0,63 2	0,35 2	0,08 4	0,10 7		
	Alternatif					
A1	0,38	0,42 7	0,37 1	0,46 8	0,472	1
A2	0,38 9	0,29 6	0,29 4	0,32 5	0,409	2
A3	0,24 2	0,18 8	0,25 9	0,19 9	0,262	3
A4	0,05 4	0,15 3	0,12 6	0,17 1	0,117	5
A5	0,03 3	0,17 8	0,17 8	0,18 3	0,118	4

Perhitungan bobot global

$$\begin{aligned}
 A1 &= (0,632 \times \mathbf{0,380}) + (0,352 \times \mathbf{0,427}) + (0,084 \times \mathbf{0,371}) \\
 &+ (0,107 \times \mathbf{0,468}) = \mathbf{0,472}. \\
 A2 &= (0,632 \times \mathbf{0,389}) + (0,352 \times \mathbf{0,296}) + (0,084 \times \mathbf{0,294}) \\
 &+ (0,107 \times \mathbf{0,325}) = \mathbf{0,409}. \\
 A3 &= (0,632 \times \mathbf{0,242}) + (0,352 \times \mathbf{0,188}) + (0,084 \times \mathbf{0,259}) \\
 &+ (0,107 \times \mathbf{0,199}) = \mathbf{0,262}. \\
 A4 &= (0,632 \times \mathbf{0,054}) + (0,352 \times \mathbf{0,153}) + (0,084 \times \mathbf{0,126}) \\
 &+ (0,107 \times \mathbf{0,171}) = \mathbf{0,117}. \\
 A5 &= (0,632 \times \mathbf{0,033}) + (0,352 \times \mathbf{0,178}) + (0,084 \times \mathbf{0,178}) \\
 &+ (0,107 \times \mathbf{0,183}) = \mathbf{0,118}.
 \end{aligned}$$

V.KESIMPULAN

Berdasarkan penelitian sistem pendukung keputusan pemilihan pemasok terbaik menggunakan metode fuzzy AHP di PT Gee San Indonesia, maka dapat diambil beberapa kesimpulan sebagai berikut :

1. Sistem yang akan dirancang yaitu sistem pendukung keputusan pemilihan pemasok terbaik dengan menggunakan metode F-AHP. Sistem akan menerima *input* (data masukan) kriteria-kriteria dan penilaian pemasok (alternatif). Kemudian akan diproses dengan menerapkan perhitungan F-AHP dan menghasilkan *output* (data keluaran) perankingan alternatif berupa bobot penilaian calon pemasok terbaik beserta hasil nilai keputusannya berupa daftar ranking Sistem pendukung keputusan pemilihan pemasok terbaik dengan menggunakan metode F-AHP akan ditentukan dalam beberapa tahapan yaitu:
 - a. Menentukan data kriteria dan alternatif.
 - b. Melakukan pemodelan *Fuzzy* AHP berdasarkan kriteria dan alternatif.
 - c. Melakukan representasi kedalam struktur hierarki.
 - d. Melakukan penilaian setiap tingkat hierarki dari masing-masing kriteria.
 - e. Melakukan konversi *fuzzy*.
 - f. Melakukan perbandingan berpasangan F-AHP.
 - g. Melakukan konversi *fuzzy* pada perbandingan berpasangan F-AHP.
 - h. Menentukan *centroid measurement*.

- i. Menghitung nilai total *l*, *m*, dan *u* dari setiap kriteria.
 - j. Menentukan nilai sintesis fuzzy.
 - k. Menghitung *crisp output*.
 - l. Perankingan Bobot Global.
2. Berdasarkan hasil pembobotan yang dilakukan dengan metode *Fuzzy Analytical Hierarchy Proses* (F-AHP), PT Global Fiberindo mempunyai nilai bobot 0,472 paling tinggi dibandingkan empat pemasok lainnya hal ini berarti bahwa pemasok yang mempunyai kinerja terbaik adalah PT Global Fiberindo.

DAFTAR PUSTAKA

- [1] Arifin, A., Tobing, F.A.T. and Apriliani, A., 2015. Aplikasi Akumulasi Biaya Pabrikasi Dengan Metode Proses Studi Kasus PT. Vitra Graha Interia. *JURNAL SISFOTEK GLOBAL*, 5(1).
- [2] D. Nofriansyah, *Konsep Data Mining Vs sistem Pendukung Keputusan*, Yogyakarta: Deepublish, 2015.
- [3] H. Bekti, *Dreamweaver CS6, CSS dan JQuery*, Yogyakarta: Penerbit Andi: 2015.
- [4] R. Martono, *Manajemen Logistik Terintegrasi*, Jakarta Pusat: Penerbit PPM, 2015.
- [5] Marimin dan N. Maghfiroh, *Aplikasi Teknik Pengambilan Keputusan dalam Manajemen Rantai Pasok*, Bogor: PT Penerbit IPB Prress, 2010.
- [6] Marimin, dkk, "*Teknik dan Analisis Pengambilan Keputusan Fuzzy dalam Manajemen Rantai Pasok*", Bogor: PT Penerbit IPB Press, 2013
- [7] Y. Murya, *Project PHP dan MySQL*, Jasakom, 2014.
- [8] D. Nugeraha, *Sistem Penunjang Keputusan Filosofi, Teori dan Implementasi*, Yogyakarta: Garudhawaca, 2017.

Analisis Kinerja Algoritma Quick Double Merge Sort Paralel Menggunakan openMP

I Nyoman Aditya Yudiswara¹, Abba Suganda²

Computer Science Department, BINUS Graduate Program Master of Computer Science
Bina Nusantara University, Jakarta, Indonesia 11480
i.yudiswara@binus.edu¹, agirsang@binus.edu²

Diterima 11 November 2019

Disetujui 20 Desember 2019

Abstract — Processor technology currently tends to increase the number of cores more than increasing the clock speed. This development is very useful and becomes an opportunity to improve the performance of sequential algorithms that are only done by one core. This paper discusses the sorting algorithm that is executed in parallel by several logical CPUs or cores using the openMP library. This algorithm is named QDM Sort which is a combination of sequential quick sort algorithm and double merge algorithm. This study uses a data parallelism approach to design parallel algorithms from sequential algorithms. The data used in this study are the data that have not been sorted and also the data that has been sorted is integer type which is stored in advance in a file. The parameter measured to determine the performance of the QDM Sort algorithm is speedup. In a condition where a large amount of data is above 4096 and the number of threads in QDM Sort is the same as the number of logical CPUs, the QDM Sort algorithm has a better speedup compared to the other parallel sorting algorithms discussed in this study. For small amounts of data it is still better to use sequential sorting algorithm.

Keywords : *core, double merge, logical CPU, QDM sort, quick sort, speedup, thread*

I. PENDAHULUAN

Mulai tahun 2004 trend dari teknologi processor lebih menambah jumlah core dari pada meningkatkan clock speed. Dalam sebuah processor terdiri dari beberapa core, dan di dalam core bisa menjalankan lebih dari satu thread. Perkembangan ini sangat bermanfaat dan menjadi sebuah kesempatan untuk meningkatkan kinerja pada sebuah algoritma sekuensial yang hanya dikerjakan oleh satu CPU atau satu core dengan cara memodifikasi algoritma tersebut agar bisa dikerjakan secara paralel oleh beberapa core. Salah satu algoritma sekuensial yang sering dipakai pada sebuah program adalah algoritma sorting. Sudah banyak algoritma sorting yang dikembangkan baik yang sekuensial maupun yang paralel, namun ruang untuk lebih meningkatkan lagi kinerja algoritma sorting khususnya algoritma sorting paralel masih terbuka

lebar sejalan dengan perkembangan arsitektur dan organisasi komputer.

Paper ini membahas mengenai gabungan algoritma sekuensial sorting yang sudah ada yaitu quick sort dan merge sort yang akan dimodifikasi menjadi algoritma paralel, yang selanjutnya penulis sebut dengan nama Quick Double Merge Sort (QDM Sort). Untuk menjadikan algoritma sekuensial menjadi algoritma paralel dapat menggunakan dua pendekatan yaitu data parallelism dan control parallelism. Data parallelism adalah metode dengan membagi data menjadi beberapa blok data yang kemudian semua blok data dikerjakan secara paralel. Control parallelism adalah membagi instruksi menjadi beberapa sub-instruksi yang kemudian dijalankan secara paralel. QDM Sort menggunakan kedua pendekatan diatas agar dapat mengoptimalkan pemakaian semua core yang ada pada prosesor.

Kinerja algoritma QDM Sort diukur dengan menghitung nilai speedup dan efisiensi, dengan merubah faktor faktor yang mempengaruhinya antara lain jumlah data yang diurut, jumlah thread pada program QDM Sort dan jumlah core pada prosesor komputer. Paper ini bertujuan untuk mengetahui seberapa besar peningkatan kecepatan (speedup) dan efisiensi dari algoritma QDM Sort dibandingkan dengan algoritma sorting sekuensial dan paralel yang lainnya.

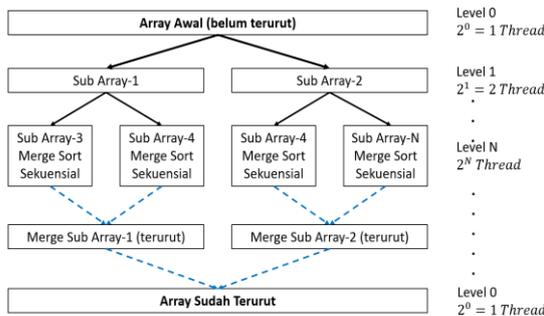
II. TINJAUAN PUSTAKA

A. Merge Sort Paralel

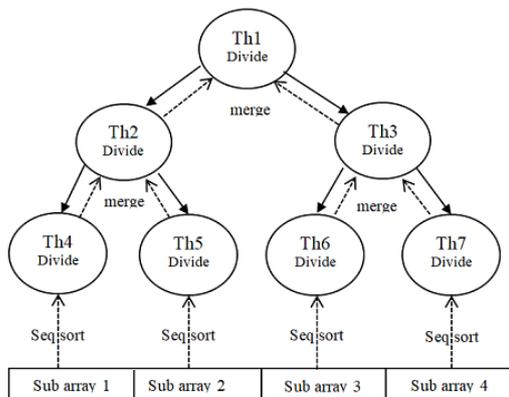
Algoritma merge sort paralel dijelaskan oleh Atanas Radenski [1] yang secara garis besar cara kerja algoritmanya seperti gambar 1. Array akan dibagi menjadi sub-sub array sampai level N dimana $2^N =$ jumlah thread. Jika sudah sampai pada level N setiap sub-array akan dilakukan pengurutan oleh masing-masing thread dengan menggunakan algoritma merge sort sekuensial. Setelah setiap thread selesai melakukan pengurutan kemudian dilakukan merge secara paralel oleh sejumlah thread mulai dari $2^{(N-1)}$ thread sampai terakhir hanya oleh 1 thread. Hasil penelitiannya menunjukkan nilai speedup dijalankan pada komputer dengan 2, 4 dan 8 core adalah 1.7, 2.6

dan 3.2 kali.

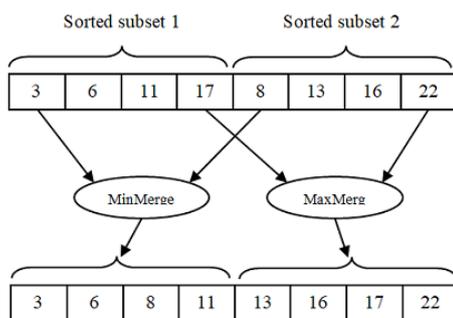
Pada penilitan Ahmet Uyar [2] menjelaskan tentang pengembangan algoritma paralel merge sort, dimana pada penelitiannya menggunakan dua thread untuk melakukan operasi merge secara bersamaan (double merge). Thread pertama akan melakukan proses merge min dan thread ke dua akan melakukan merge max seperti gambar 2 dan gambar 3. Hasilnya setelah diimplementasikan dengan menggunakan bahasa pemrograman Java peningkatan kecepatan sebanyak 20% - 30% jika dibandingkan dengan merge sort yang diimplementasikan di Java Library.



Gambar 1. Merge Sort Paralel



Gambar 2

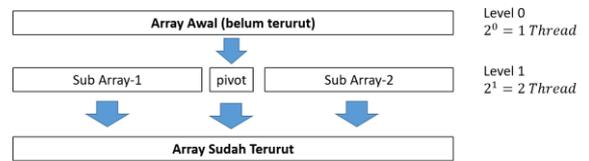


Gambar 3

B. Quick Sort Paralel

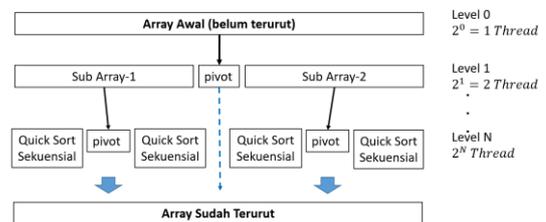
Pada penilitan Yong Liu [3] dan Sinan Sameer M.[4] menjelaskan tentang pengembangan algoritma paralel quick sort. Sebenarnya kedua penelitian diatas

hampir sama menggunakan maksimal dua thread seperti pada gambar 4 dibawah. Pencarian pivot pertama kali dilakukan oleh satu thread dan setelah itu masing-masing sub-array dilakukan quick sort dengan algoritma sekuensial oleh dua thread secara paralel.



Gambar 4. Paralel Quick Sort

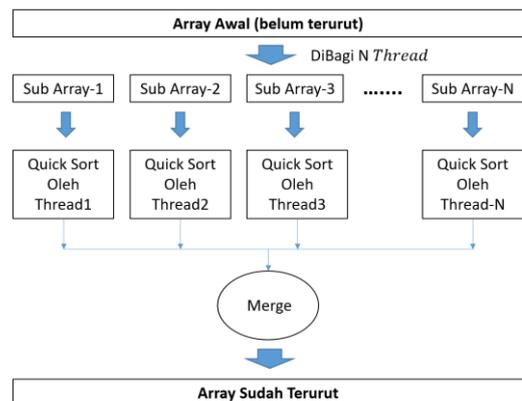
Pada penelitian Daniel Langr [5] menjelaskan algoritma quick sort paralel dengan menggunakan sejumlah thread sampai level N sehingga terdapat sebanyak 2^N sub-array yang akan di sorting secara paralel oleh masing-masing thread dengan menggunakan algoritma quick sort sekuensial, seperti gambar 5 dibawah.



Gambar 5. Quick Sort Paralel

C. Hybrid Quick Sort

Algoritma paralel hybrid quick sort dijelaskan oleh Kil Jae Kim [6] seperti pada gambar 6 dibawah. Mula-mula array yang belum terurut dibagi dengan jumlah thread, kemudian masing-masing sub-array ini dilakukan sorting oleh setiap thread secara bersamaan dengan algoritma quick sort sekeunsial. Sub-array yang sudah terurut ini kemudian dilakukan merge atau penggabungan maka secara keseluruhan akan diperoleh array yang sudah terurut.



Gambar 6. Quick Merge Sort

III. METODE PENELITIAN

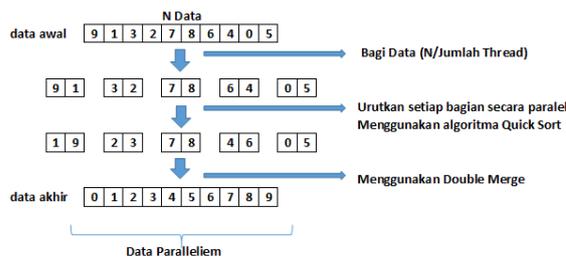
Penelitian ini dibagi menjadi beberapa tahap sebagai berikut :

- ✓ Identifikasi masalah
- ✓ Pendekatan penyelesaian masalah
- ✓ Metode yang diusulkan
- ✓ Implementasi (Coding)
- ✓ Pengukuran kinerja
- ✓ Evaluasi hasil & kesimpulan

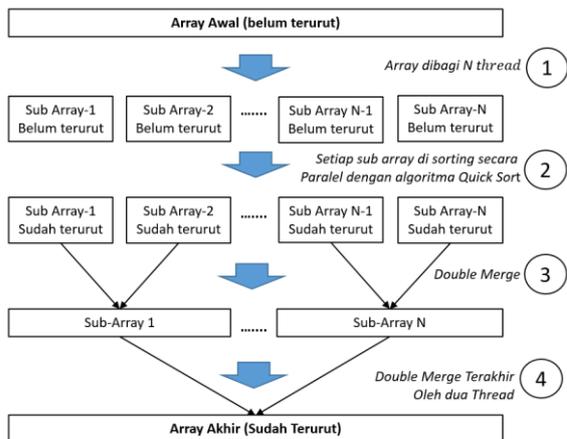
Untuk merancang algoritma paralel umumnya menggunakan pendekatan data parallelism, task parallelism atau kombinasi antara data parallelism dan task parallelism. Perancangan algoritma QDM Sort paralel menggunakan pendekatan *Data Parallelism* seperti gambar 7. Pendekatan ini pada dasarnya mencoba untuk mengoptimalkan pemakaian semua core yang ada pada prosesor komputer pada saat menjalankan program QDM Sort. Parameter yang diukur untuk mengetahui kinerja algoritma QDM Sort adalah speedup yang dihitung dengan menggunakan rumus sbb:

$$Speedup = \frac{T_s}{T_p} \dots\dots\dots (1)$$

T_s adalah waktu eksekusi pada program dengan algoritma sekuensial dan T_p adalah waktu eksekusi pada program dengan algoritma paralel untuk menyelesaikan masalah yang sama.



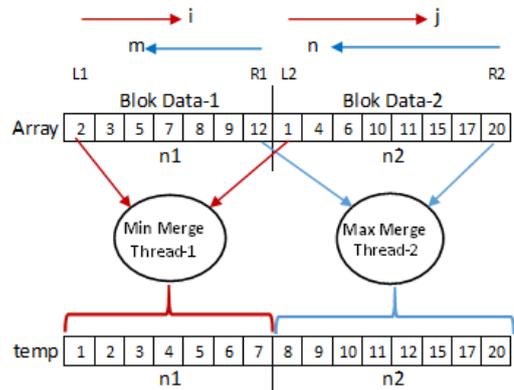
Gambar 7. Data Parallelism



Gambar 8. QDM Sort

Metode yang diusulkan untuk diimplementasikan pada QDM Sort adalah seperti gambar 8. Jika jumlah data yang akan diurut pada array adalah N, maka pada

tahap awal data dibagi menjadi N/T sub-array, dimana T jumlah thread pada program QDM Sort. Kemudian setiap sub-array akan dilakukan sorting secara bersamaan oleh masing-masing thread. Setelah semua sub-array terurut kemudian dilakukan merge dengan menggunakan double merge, yaitu min-merge dan max-merge seperti gambar 9. Min Merge melakukan merge mulai dari data terkecil ke data lebih besar sebanyak n1 dan Max Merge melakukan merge mulai data terbesar ke data yang lebih kecil sebanyak n2. Min Merge dan Max Merge dilakukan secara bersamaan oleh 2 thread, sehingga pada tahap merge yang paling terakhir akan dilakukan oleh 2 thread. Jadi dengan double merge akan dilakukan tahapan merge sebanyak $\log_2(N)$, dengan jumlah thread pada tiap tahapan adalah N, N/2 sampai 2 thread. Jadi operasi merge yang terakhir akan dilakukan oleh 2 thread.



Gambar 9. Proses Double Merge

IV. IMPLEMENTASI DAN HASIL

A. Ujicoba

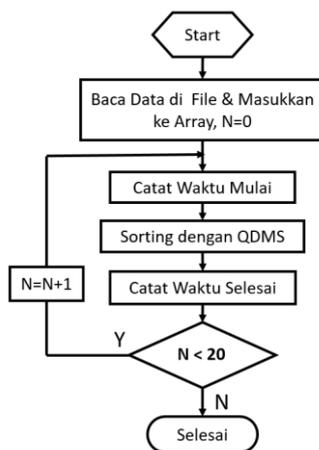
Komputer yang digunakan untuk percobaan pada penelitian ini seperti table-1 dibawah.

Tabel 1. Spesifikasi Komputer Untuk Percobaan

No	Spesifikasi Komputer	
1	Komputer	Desktop Acer Veriton M4610
	CPU	Intel i7 2600, 3.4 GHz, 4 Core 8 Thread (8 Logical CPU)
	RAM	DDR3 16 GByte PC3-12800
	Sistem Operasi	Linux Ubuntu 64-bit, 18.04
2	Komputer	All In One MSI
	CPU	Intel i3-2120, 3.3 GHz, 2 Core, 4 threads (4 Logical)
	RAM	8 GByte
	Sistem Operasi	Linux Ubuntu 64-bit 18.04
3	Komputer	Laptop Dell Inspiron 1470
	CPU	Core2Duo U9400 1.4GHz, 2 Core (2 logical CPU)
	RAM	4 GByte
	Sistem Operasi	Linux Mint 19.1 Tessa

Data yang digunakan untuk mengetahui kinerja algoritma QDM Sort adalah data integer yang nilai nya random dengan jumlah data nya bervariasi mulai dari 256, 512, 1024, 2048, 4096, 8192, 16384, 32768, 49152, 65536, 131072, 262144, 393216, 524288, 786432 dan 1048576, dan data ini masing-masing disimpan pada sebuah file.

Membaca data di file dan memasukkan data ke array tidak dihitung dalam pencatatan waktu eksekusi yang diperlukan oleh program seperti gambar 10. Pencatatan run time program dilakukan beberapa kali (lebih dari 20 kali) dan dipilih waktu run time yang paling cepat / kecil. Pada saat pencatatan waktu run time tidak ada program lain yang dijalankan dan komunikasi internet/wifi juga dimatikan.



Gambar 10

B. Implementasi

Program QDM Sort diimplementasikan menggunakan bahasa pemrograman C dan library openMP versi 3.1, menggunakan Integrated Development Environment (IDE) CodeBlocks versi 17.12, menggunakan sistem operasi Linux dan compiler GCC. Algoritmanya seperti potongan program dibawah ini.

- ✓ Baris 1 adalah structure untuk menyimpan index yang terendah (low) dan index tertinggi (high) dari setiap sub-array
- ✓ Baris 5 adalah nama fungsinya (QDMS) dengan parameter A, Size, T dan N. A adalah Array yang akan di sorting, Size adalah jumlah data dari Array, T adalah Array yang besarnya sama dengan A yang digunakan untuk menampung data sementara dan N adalah jumlah thread.
- ✓ Pada baris 10 dihitung jumlah elemen dari setiap sub-array, kecuali sub-array yang terakhir ukurannya bisa berbeda.
- ✓ Baris 14 - 28 adalah area paralel dengan jumlah thread ditentukan oleh variable N
- ✓ Baris 15 setiap thread akan membaca ID nya masing-masing yang penomorannya dimulai dari 0 s/d N-1
- ✓ Baris 17-20 pada sub-array terakhir dilakukan sorting dengan algoritma Quick Sort sekuensial, dan

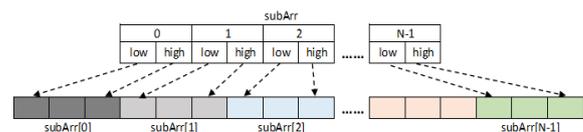
nilai index terkecil dan terbesar dari sub-array disimpan pada array struct subArray.

- ✓ Pada baris 23–27 semua thread akan melakukan sorting pada masing-masing sub-array dan sekaligus menyimpan nilai index terkecil dan indeks terbesar dari sub-array pada array struct subArray.
- ✓ Pada baris 32-36 semua sub-array yang sudah diurut dengan algoritma quick sort sekuensial dilakukan merge dengan metode double merge secara paralel.

```

1 | typedef struct subArray{
2 |     int low; int high;
3 | }SA;
4 |
5 | void QDMS(int A[],int Size,int T[],int N)
6 | {
7 |     int i, j, sizeSA;
8 |     SA *subArr;
9 |     subArr=(SA*)malloc(sizeof(SA)* N);
10 |    sizeSA=round((float)Size/(float)N);
11 |
12 |    #pragma omp parallel num_threads(N)
13 |    {
14 |        int low, high;
15 |        int tid = omp_get_thread_num();
16 |        if(tid == (N-1)){
17 |            low = tid*sizeSA; high=Size-1;
18 |            QuickSortSerial(A, low, high);
19 |            subArr[tid].low = low;
20 |            subArr[tid].high = high;
21 |        }
22 |        else{
23 |            low = tid*sizeSA;
24 |            high = tid*sizeSA + sizeSA-1;
25 |            QuickSortSerial(A, low, high);
26 |            subArr[tid].low = low;
27 |            subArr[tid].high = high;
28 |        }
29 |    }
30 |
31 |    while(N >= 2){
32 |        DoubleMerge(A,Size,T,subArr,N);
33 |        N=N/2;
34 |        for(j=0; j < N; j++){
35 |            subArr[j].low = subArr[j*2].low;
36 |            subArr[j].high= subArr[j*2+1].high;
37 |        }
38 |    }
39 | }
40 |
  
```

Gambar 11. Algoritma QDMS dengan openMP



Gambar 12. Array of structure digunakan untuk menyimpan index low dan high pada setiap sub-array

C. Hasil Uji Coba

Sebagai bahan evaluasi ada 7 program sorting yang di buat dan dijalankan untuk diambil datanya yaitu program QDM sort, Quick sort paralel, Merge sort paralel, Quick sort serial, Merge sort serial, Quick

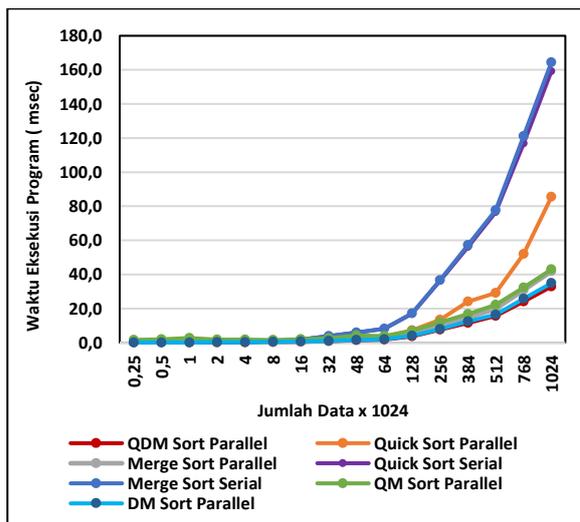
Merge sort paralel, Double merge sort paralel dan seperti yang dijelaskan pada tinjauan pustaka diatas

dan yang diusulkan pada penelitian ini yaitu Quick Double Merge Sort (QDM Sort).

Table 1. Waktu eksekusi dari algoritma sorting yang digunakan pada penelitian

No	Jumlah Data Integer	Execution Time (mikro second)						
		QDM Sort Paralel	Quick Sort Paralel	Merge Sort Paralel	Quick Sort Serial	Merge Sort Serial	Quick Merge Sort Paralel	Double Merge Sort Paralel
1	256	137	93	79	11	16	1568.0	160.0
2	512	112	83	86	31	37	2016.0	182.0
3	1024	122	120	97	80	94	2645.0	161.0
4	2048	136	156	142	173	240	1812.0	203.0
5	4096	177	222	242	380	364	1748.0	256.0
6	8192	300	537	372	823	799	1564.0	395.0
7	16384	495	684	604	1791	1744	1960.0	618.0
8	32768	917	1889	1133	3795	3762	2458.0	1074.0
9	49152	1363	2620	1721	5934	5906	4209.0	1606.0
10	65536	1836	3583	2280	8109	8075	3885.0	2041.0
11	131072	3658	7255	4633	17000	17221	6860.0	3982.0
12	262144	7570	13495	9554	36260	36628	11857.0	7904.0
13	393216	11600	24033	14752	56150	57177	16801.0	12504.0
14	524288	15685	29186	19811	76473	77693	22147.0	16372.0
15	786432	24118	51959	30911	116827	121152	32267.0	25807.0
16	1048576	32897	85454	41829	159197	164433	42956.0	34879.0

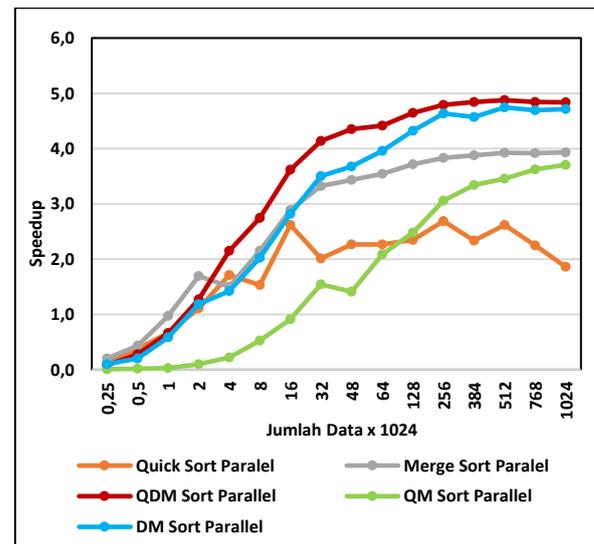
Semua algoritma sorting pada Tabel-2 dijalankan pada komputer dengan spesifikasi seperti Tabel-1 no.1 yaitu komputer dengan 8 logical CPU. Hasil ujicoba untuk waktu eksekusi ke tujuh program tersebut ditampilkan pada Tabel-2 dan juga ditampilkan dalam bentuk grafik seperti gambar 13 dan gambar 14.



Gambar 13. Waktu eksekusi vs jumlah data

Pada gambar 13 menunjukkan waktu eksekusi versus jumlah data untuk beberapa algoritma sorting, dimana pada program sorting serial hanya menggunakan 1 logical CPU sedangkan algoritma sorting paralel menggunakan 8 logical CPU. Berdasarkan hasil pada Tabel-2 dan gambar 13 program Quick Double Merge Sort (QDM Sort) untuk jumlah

data diatas 16 x 1024 memerlukan waktu yang lebih cepat dibandingkan dengan program sorting lainnya.



Gambar 14. Speedup vs jumlah data

Pada gambar 14 menunjukkan besarnya nilai pertambahan kecepatan (speedup) versus jumlah data untuk semua program sorting paralel yang digunakan pada penelitian ini. Speedup dihitung dengan menggunakan rumus (1) diatas. Untuk menghitung speedup QDM sort paralel, Quick sort paralel dan Quick merge sort paralel nilai T_s diambil dari waktu eksekusi quick sort serial. Sedangkan untuk menghitung speedup double merge sort paralel dan merge sort paralel nilai T_s diambil dari nilai waktu eksekusi merge sort serial. Berdasarkan Tabel-2 dan gambar 14 untuk jumlah data diatas 4 x 1024, QDM

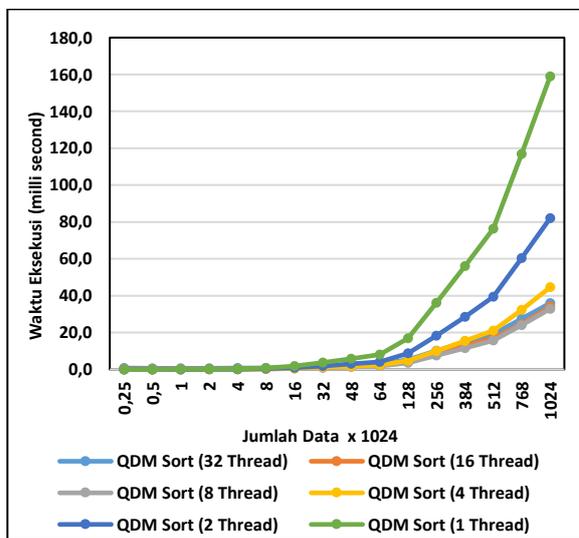
Sort memiliki nilai speedup yang lebih besar dibandingkan dengan program lainnya.

dijalankan pada komputer dengan spesifikasi seperti Tabel-1 no.1.

Tabel-3 berisi waktu eksekusi program QDM sort dengan jumlah thread yang berbeda-beda yang

Table 3. Waktu eksekusi program QDM Sort untuk jumlah thread yang berbeda

No.	Jumlah Data x 1024	Waktu eksekusi (milli second)					
		QDM Sort (32 Thread)	QDM Sort (16 Thread)	QDM Sort (8 Thread)	QDM Sort (4 Thread)	QDM Sort (2 Thread)	QDM Sort (1 Thread)
1	0,25	0,8	0,364	0,137	0,024	0,007	0,011
2	0,5	0,678	0,32	0,112	0,028	0,014	0,031
3	1	0,719	0,322	0,122	0,047	0,033	0,079
4	2	0,685	0,333	0,136	0,083	0,085	0,173
5	4	0,755	0,401	0,177	0,158	0,193	0,38
6	8	0,901	0,528	0,3	0,303	0,428	0,823
7	16	1,2	0,766	0,495	0,617	0,925	1,791
8	32	1,698	1,295	0,917	1,134	1,956	3,795
9	48	2,256	1,846	1,363	1,708	3,028	5,934
10	64	2,91	2,448	1,836	2,325	4,131	8,109
11	128	5,4	4,326	3,658	4,818	8,775	16,997
12	256	10,145	9,173	7,57	10,012	18,381	36,221
13	384	14,676	12,863	11,6	15,531	28,553	56,09
14	512	19,134	16,976	15,685	21,077	39,542	76,42
15	768	27,439	24,605	24,118	32,331	60,471	116,827
16	1024	36,028	34,177	32,897	44,59	82,066	159,015



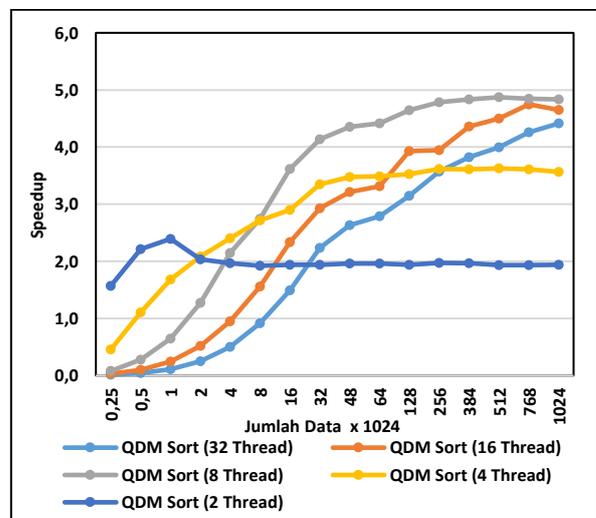
Gambar 15. Waktu Eksekusi vs Jumlah Data vs Jumlah Thread

Representasi dalam bentuk grafik untuk tabel-3 dapat dilihat pada gambar 15 dan 16. Gambar 15 menunjukkan pengaruh jumlah data dan jumlah thread pada program QDM sort terhadap waktu eksekusi. Dari Table-3 dan gambar 15 dapat dilihat untuk jumlah data yang besarnya diatas 8 x 1024 dan untuk program QDM sort yang memiliki jumlah thread sama dengan jumlah logical CPU nya memiliki waktu eksekusi lebih cepat. Karena program QDM sort ini dijalankan pada komputer dengan 8 logical CPU maka program QDM sort dengan 8 thread memiliki waktu eksekusi paling cepat dibandingkan dengan jumlah thread yang lainnya.

Gambar 16 menunjukkan pengaruh jumlah data dan jumlah thread pada program QDM sort terhadap

speedup. Speedup dihitung dengan menggunakan rumus (1) diatas. Untuk menghitung speedup QDM sort dengan jumlah thread 2, 4, 8, 16 dan 32 nilai T_s diambil dari waktu eksekusi QDM sort untuk 1 thread.

Pada gambar 16 terlihat untuk data yang besarnya diatas 8 x 1024 dan untuk program QDM Sort yang memiliki jumlah thread sama dengan jumlah logical CPU akan memiliki speedup yang yang paling besar. Program QDM sort dengan 8 thread untuk jumlah data lebih besar dari 8x1024 memiliki speedup paling tinggi.



Gambar 16. Speedup vs Jumlah Data vs Jumlah Thread

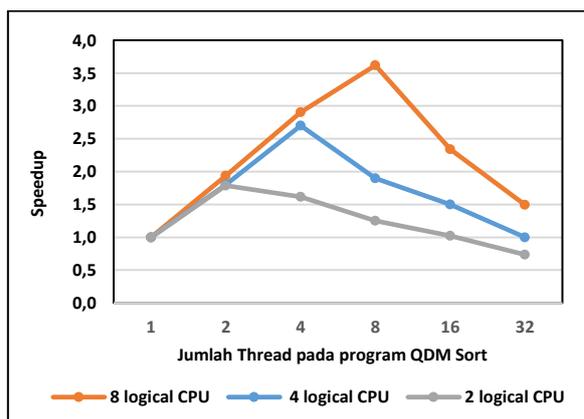
Tabel-4 berisi data hasil perhitungan speedup dan efisiensi program QDM sort dengan jumlah thread yang berbeda-beda dan dengan jumlah data yang diurut

sebesar 16x1024, dan dijalankan pada tiga buah komputer yang berbeda dengan spesifikasi seperti pada tabel-1. Representasi dalam bentuk grafik untuk Tabel-4 dapat dilihat pada gambar 17 dan 18. Kedua gambar ini menunjukkan speedup dan efisiensi yang optimal

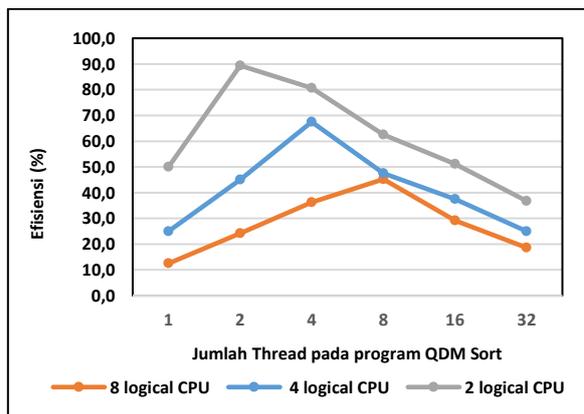
didapat jika jumlah data yang diurut cukup besar diatas 16384 dan jumlah thread pada program QDM sort sama dengan jumlah logical CPU pada komputer.

Tabel 4 – Speedup dan Efisiensi program QDM sort pada 3 komputer yang beda

No	Jumlah Thread	Komputer dengan 8 logical CPU		Komputer dengan 4 logical CPU		Komputer dengan 2 logical CPU	
		Speedup	Efisiensi	Speedup	Efisiensi	Speedup	Efisiensi
1	1	1.000	12.500	1.000	25.000	1.000	50.000
2	2	1.936	24.203	1.800	45.000	1.788	89.394
3	4	2.903	36.284	2.700	67.500	1.614	80.724
4	8	3.618	45.227	1.900	47.500	1.251	62.547
5	16	2.338	29.227	1.500	37.500	1.023	51.164
6	32	1.493	18.656	1.000	25.000	0.735	36.770



Gambar 17 – Speedup vs Jumlah Thread pada program vs Jumlah logical CPU



Gambar 18 – Efisiensi vs Jumlah Thread pada program vs Jumlah logical CPU

V. KESIMPULAN

A. Kesimpulan

Dari hasil ujicoba diatas dapat disimpulkan sbb :

1. QDM Sort memiliki waktu eksekusi yang lebih cepat dibandingkan dengan algoritma Sorting paralel lainnya yang digunakan sebagai referensi pada penelitian ini, untuk jumlah data yang besar diatas diatas 4096. Sedangkan untuk jumlah data yang kecil lebih baik menggunakan algoritma sorting sekuensial.

2. QDM Sort memiliki speedup yang optimal jika jumlah data yang besar diatas 4096 dan jumlah thread-nya sama dengan jumlah logical CPU yang dimiliki oleh prosesor komputer.
3. Untuk mendapatkan speedup yang optimal, program QDM Sort secara otomatis akan menyesuaikan jumlah threadnya sama dengan jumlah logical CPU, dan jika datanya kecil (dibawah 4096) program QDM sort secara otomatis akan menggunakan algoritma quick sort sekuensial.

B. Saran

Program QDM sort ini masih bisa ditingkatkan speedupnya jika pada tahapan proses merge nya di setiap tahapnya dapat dilakukan oleh semua logical CPU yang dimiliki oleh prosesor. QDM sort baru bisa menggunakan logical CPU pada tahapan merge mulai dari N , $N/2$, $N/4$ dan pada tahap terakhir dilakukan oleh 2 logical CPU (N =Jumlah logical CPU). Disarankan agar penelitian ini bisa dilanjutkan dengan memanfaatkan semua resources (logical CPU) di setiap tahap dari tahapan proses merge.

REFERENSI

- [1] Atanas Radenski, Shared Memory, Message Passing, and Hybrid Merge Sorts for Standalone and Clustered SMPs. Proc. PDPTA'11, the 2011 International Conference on Parallel and Distributed Processing Techniques and Applications, CSREA Press (H. Arabnia, Ed.), 2011, pp. 367 - 373.
- [2] Uyar, A., Parallel Merge Sort with Double Merging. IEEE 8th International Conference on Application of Information and Communication Technologies (AICT), 2014
- [3] Yong Liu, Y. Y. (2013). Quick-MergeSort Algorithm Based on Multi-coreLinux. International Conference on Mechatronic
- [4] Sinan Sameer Mahmood Al-Dabbagh, N. H., Parallel QuickSort Algorithm using OpenMP. International Journal of Computer Science & Mobile Computing (IJCSMCC), Vol.5 Issue.6, June- 2016, pg. 372-381.
- [5] Daniel Langr, Pavel Tvrdik and Ivan Simecek, AQsort : Scalable Multi-Array In-Place Sorting With OpenMP. Scalable Computing: Practice And Experience, Volume 17, Number 4, 2016, pp. 369–391. <http://www.scepe.org>
- [6] Kil Jae Kim, Seong Jin Cho and Jae-Wook Jeon, Parallel Quick Sort Algorithms Analysis using OpenMP 3.0 in Embedded

-
- System. 11th International Conference on Control, Automation and Systems Oct. 26-29, 2011 in KINTEX, Gyeonggi-do, Korea
- [7] Sciences, Electric Engineering and Computer (MEC), 2013, 1578 - 1583.
- [8] Task Parallelism vs Data Parallelism [online] Tersedia dalam : <https://www.allprogrammingtutorials.com/tutorials/task-parallelism-vs-data-parallelism.php> [diakses 1-Nop-2019]

PEDOMAN PENULISAN JURNAL ULTIMATICS, ULTIMA INFOSYS, DAN ULTIMA COMPUTING

1. Kriteria Naskah

- Naskah belum pernah dipublikasikan atau tidak dalam proses penyuntingan di jurnal berkala lainnya.
- Naskah yang dikirimkan dapat berupa naskah hasil penelitian atau konseptual.

2. Pengetikan Naskah

- Naskah diketik dengan jarak spasi antar baris 1 pada halaman ukuran A4 (21 cm x 29,7 cm), margin kiri-atas 3 cm dan kanan-bawah 2 cm, dengan jenis tulisan Times New Roman.
- Naskah dapat ditulis dalam Bahasa Indonesia atau Bahasa Inggris.
- Jumlah halaman untuk tiap naskah dibatasi dengan jumlah minimal 4 halaman dan maksimal 8 halaman.

3. Format Naskah

- Komposisi naskah terdiri dari Judul, Abstrak, Kata Kunci, Pendahuluan, Metode, Hasil Penelitian dan Pembahasan, Simpulan, Lampiran, Ucapan Terima Kasih, dan Daftar Pustaka.
- Judul memiliki jumlah kata maksimal 15 kata dalam Bahasa Indonesia atau maksimal 12 kata dalam Bahasa Inggris (termasuk subjudul bila ada).
- Abstrak ditulis dengan Bahasa Inggris paling banyak 200 kata, meskipun bahasa yang digunakan dalam penyusunan naskah adalah Bahasa Indonesia. Isi abstrak sebaiknya mengandung argumentasi logis, pendekatan pemecahan masalah, hasil yang dicapai, dan simpulan singkat.
- Kata Kunci ditulis dengan Bahasa Inggris dalam satu baris, dengan jumlah kata antara 4 sampai 6 kata.
- Pendahuluan berisi latar belakang dan tujuan penelitian.
- Metode dapat diuraikan secara terperinci dan dibedakan menjadi beberapa bab maupun subbab yang terpisah.
- Hasil dan Pembahasan disajikan secara sistematis sesuai dengan tujuan penelitian.
- Simpulan menyajikan intisari hasil penelitian yang telah dilaksanakan. Saran pengembangan untuk penelitian selanjutnya juga dapat diberikan di sini.

- Lampiran dan Ucapan Terima Kasih dapat dijabarkan setelah Simpulan secara singkat dan jelas.
- Daftar Pustaka yang dirujuk dalam naskah harus dituliskan di bagian ini secara kronologis berdasarkan urutan kemunculannya. Cara penulisannya mengikuti cara penulisan jurnal dan transaction IEEE.
- Template naskah telah disediakan dan dapat diminta dengan menghubungi surel redaksi.

4. Penulisan Daftar Pustaka

- Artikel Ilmiah:
N. Penulis, "Judul artikel ilmiah," *Singkatan Nama Jurnal*, vol. x, no. x, hal. xxx-xxx, Sept. 2013.
- Buku
N. Penulis, "Judul bab di dalam buku," di dalam *Judul dari Buku*, edisi x. Kota atau Negara Penerbit: Singkatan Nama Penerbit, tahun, bab x, subbab x, hal. xxx-xxx.
- Laporan
N. Penulis, "Judul laporan," *Singkatan Nama Perusahaan, Kota Perusahaan, Singkatan Nama Negara, Laporan xxx, tahun*.
- Buku Manual/ *handbook*
Nama dari Buku Manual, edisi x, *Singkatan Nama Perusahaan, Kota Perusahaan, Singkatan Nama Negara, tahun*, hal. xxx-xxx.
- Prosiding
N. Penulis, "Judul artikel," di dalam *Nama Konferensi Ilmiah*, Kota Konferensi, *Singkatan Nama Negara (jika ada), tahun*, hal. xxx-xxx.
- Artikel yang Disajikan dalam Konferensi
N. Penulis, "Judul artikel," disajikan di *Nama Konferensi, Kota Konferensi, Singkatan Nama Negara, tahun*.
- Paten
N. Penulis, "Judul paten," HKI xxxxxx, 01 Januari 2014.
- Tesis dan Disertasi
N. Penulis, "Judul tesis," M.Sc. thesis, *Singkatan Departemen, Singkatan*

Universitas, Kota Universitas, Singkatan Nama Negara, tahun.

N. Penulis, "Judul disertasi," Ph.D. dissertation, Singkatan Departemen, Singkatan Universitas, Kota Universitas, Singkatan Nama Negara, tahun.

- Belum Terbit
N. Penulis, "Judul artikel," belum terbit.

N. Penulis, "Judul artikel," Singkatan Nama Jurnal, proses cetak.

- Sumber online
N. Penulis. (tahun, bulan tanggal). Judul (edisi) [Media perantara]. Alamat situs: [http://www.\(URL\)](http://www.(URL))

N. Penulis. (tahun, bulan). Judul. Jurnal [Media perantara]. *volume(issue)*, halaman jika ada. Alamat situs: [http://www.\(URL\)](http://www.(URL))

Catatan: media perantara dapat berupa media online, CD-ROM, USB, dan sebagainya.

5. Pengiriman Naskah Awal

- Para penulis dapat mengirimkan naskah hasil penelitiannya dalam bentuk .doc atau .pdf melalui surel ke umnjurnal@gmail.com dengan subjek sesuai Jurnal yang dipilih.
- Seluruh isi naskah yang dikirimkan harus memenuhi syarat dan ketentuan yang ditentukan.
- Kami akan menjaga segala kerahasiaan dan Hak Cipta karya Anda.
- Sertakan biodata penulis pertama yang lengkap, meliputi nama, alamat kantor, alamat penulis, telpon kantor/ rumah dan hp, serta No NPWP (bagi yang memiliki NPWP).

6. Penilaian Naskah

- Seluruh naskah yang diterima akan melalui serangkaian tahap penilaian yang melibatkan mitra bestari.
- Setiap naskah akan direview oleh minimal 2 orang mitra bestari.
- Rekomendasi dari mitra bestari yang akan menentukan apakah sebuah naskah diterima, diterima dengan revisi minor, diterima dengan revisi major, atau ditolak.

7. Pengiriman Naskah Final

- Naskah yang diterima untuk diterbitkan akan diinformasikan melalui surel redaksi.
- Penulis berkewajiban memperbaiki setiap kesalahan yang ditemukan sesuai saran dari mitra bestari.
- Naskah final yang telah direvisi dapat dikirimkan kembali ke surel redaksi beserta hasil scan Copyright Transfer Form yang telah ditandatangani.

8. Copyright dan Honorarium

- Penulis yang naskahnya dimuat harus membaca dan menyetujui isi Copyright Transfer Form kepada redaksi.
- Copyright Transfer Form harus ditandatangani oleh penulis pertama naskah.
- Naskah yang dimuat akan mendapatkan honorarium sebesar Rp 1.000.000,- per naskah, setelah dipotong pajak 2.5% (bila penulis pertama yang memiliki NPWP) dan 3% (tanpa NPWP).
- Honorarium akan ditransfer ke rekening penulis pertama (tidak dapat diwakilkan) paling lambat 2 minggu setelah jurnal naik cetak dan siap didistribusikan.
- Penulis yang naskahnya dimuat akan mendapatkan copy jurnal sebanyak 2 eksemplar.

9. Biaya Tambahan

- Permintaan tambahan copy jurnal harus dibeli seharga Rp 50.000,- per copy.
- Permintaan penambahan jumlah halaman dalam naskah (maksimal 8 halaman) akan dikenai biaya sebesar Rp 25.000,- per halaman.

10. Alamat Redaksi

d.a. Koordinator Riset
Fakultas Teknologi Informasi dan Komunikasi
Universitas Multimedia Nusantara
Gedung Rektorat Lt.6
Scientia Garden, Jl. Boulevard Gading Serpong,
Tangerang, Banten -15333
Surel: ftijurnal@umn.ac.id

Judul Paper

Sub Judul (jika diperlukan)

Nama Penulis A¹, Nama Penulis B², Nama Penulis C²

¹ Baris pertama (dari afiliasi): nama departemen organisasi, nama organisasi, kota, negara
Baris kedua: alamat surel jika diinginkan

² Baris pertama (dari afiliasi): nama departemen organisasi, nama organisasi, kota, negara
Baris kedua: alamat surel jika diinginkan

Diterima dd mmmmm yyyy

Disetujui dd mmmmm yyyy

Abstract—This electronic document is a “live” template which you can use on preparing your IJNMT paper. Use this document as a template if you are using Microsoft Word 2007 or later. Otherwise, use this document as an instruction set. Do not use symbol, special characters, or Math in Paper Title and Abstract. Do not cite references in the abstract.

Index Terms—enter key words or phrases in alphabetical order, separated by commas

I. PENDAHULUAN

Dokumen ini, dimodifikasi dalam MS Word 2007 dan disimpan sebagai dokumen Word 97-2003, memberikan panduan yang diperlukan oleh penulis untuk mempersiapkan dokumen elektroniknya. Margin, lebar kolom, jarak antar baris, dan jenis-jenis format lainnya telah disisipkan di sini. Penulis berkewajiban untuk memastikan dokumen yang dipersiapkannya telah memenuhi format yang disediakan.

Isi Pendahuluan mengandung latar belakang, tujuan, identifikasi masalah dan metode penelitian yang dipaparkan secara tersirat (implisit). Kecuali bab Pendahuluan dan Simpulan, penulisan judul bab sebaiknya eksplisit sesuai dengan isi yang dijelaskan, tidak harus implisit dinyatakan sebagai Dasar Teori, Perancangan, dan sebagainya.

II. PENGGUNAAN YANG TEPAT

A. Memilih Template

Pertama, pastikan Anda memiliki *template* yang tepat untuk artikel Anda. *Template* ini ditujukan untuk Jurnal ULTIMATICS, ULTIMA InfoSys, dan ULTIMA Computing. *Template* ini menggunakan ukuran kertas A4.

B. Mempertahankan Keutuhan Format

Template ini digunakan untuk mem-format artikel dan *style* isi artikel Anda. Seluruh margin, lebar kolom, jarak antar baris, dan jenis tulisan telah diberikan, jangan diubah.

III. PERSIAPKAN ARTIKEL ANDA

Sebelum Anda mulai mem-format artikel Anda, tulislah terlebih dahulu artikel Anda dan simpan sebagai *text file* lainnya. Setelah selesai baru lakukan pencocokkan *style* dokumen. Jangan tambahkan nomor halaman di bagian manapun dari dokumen ini. Perhatikan pula beberapa hal berikut saat melakukan pengecekan tulisan.

A. Singkatan

Definisikan singkatan pada saat pertama kali digunakan di dalam isi tulisan, walaupun singkatan tersebut telah didefinisikan di dalam abstrak. Singkatan seperti IEEE, SI, MKS, CGS, sc, dc, dan rms tidak harus didefinisikan. Singkatan yang menggunakan tanda titik tidak boleh diberi spasi, seperti “C.N.R.S.”, bukan “C. N. R. S.” Jangan gunakan singkatan di dalam Judul Artikel atau Judul Bab, kecuali tidak dapat dihindari.

B. Unit

- Gunakan baik SI (MKS) atau CGS sebagai unit primer.
- Jangan menggabungkan kepanjangan dan singkatan dari unit, yang tepat seperti “Wb/m²” atau “webers per meter persegi,” bukan “webers/m².”
- Gunakan angka nol di depan suatu bilangan desimal, seperti “0,25” bukan “.25.”

C. Persamaan

Format persamaan merupakan suatu pengecualian di dalam spesifikasi *template* ini. Anda harus menentukan apakah akan menggunakan jenis tulisan Times New Roman atau Symbol (jangan jenis tulisan yang lain). Bila Anda membuat beberapa persamaan berbeda, akan lebih baik bila Anda mempersiapkan persamaan tersebut sebagai gambar dan menyisipkannya ke dalam artikel Anda setelah diberi *style*.

Beri penomoran untuk persamaan Anda secara berurutan. Nomor persamaan berada dalam tanda kurung seperti (1), dan diletakkan pada bagian kanan dengan menggunakan suatu *right tab stop*.

$$\int_0^{r_2} F(r, \phi) dr d\phi = [\sigma r_2 / (2\mu_0)] \quad (1)$$

Perhatikan bahwa persamaan di atas diposisikan di bagian tengah dengan menggunakan suatu *center tab stop*. Pastikan bahwa simbol-simbol yang digunakan dalam persamaan Anda didefinisikan sebelum atau sesudah persamaan. Gunakan “(1),” bukan “Persamaan (1),” kecuali pada awal sebuah kalimat, seperti “Persamaan (1) merupakan”

D. Beberapa Kesalahan Umum

- Perhatikan tata cara penulisan Bahasa Indonesia yang benar, perhatikan penggunaan kata depan dan kata sambung yang tepat, seperti “di depan” dan “disampaikan”.
- Kata-kata asing yang belum diserap ke dalam Bahasa Indonesia dapat dicetak miring, atau diberi garis bawah, atau dicetak tebal (pilih salah satu), seperti “*italic*”, “underlined”, “**bold**”.
- Prefiks seperti “non”, “sub”, “micro”, “multi”, dan “ultra” bukan kata yang berdiri sendiri, oleh karenanya harus digabung dengan kata yang mengikutinya, biasanya tanpa tanda hubung, seperti “subsistem”.

IV. MENGGUNAKAN TEMPLATE

Setelah naskah artikel Anda selesai di-*edit*, artikel Anda dapat dipersiapkan untuk *template*. Gandakan template ini dengan menggunakan perintah Save As dan simpan dengan penamaan berikut:

- ULTIMATICS_namaPenulis1_judulArtikel.
- ULTIMAInfoSys_namaPenulis1_judulArtikel.
- ULTIMAComputing_namaPenulis1_judulArtikel.

Selanjutnya Anda dapat meng-*import* artikel Anda dan mempersiapkannya sesuai *template* yang diberikan. Perhatikan beberapa hal berikut pada saat melakukan pengecekan.

A. Penulis dan Afiliasi

Template ini didesain untuk tiga penulis dengan dua afiliasi yang berbeda. Penamaan afiliasi yang sama tidak perlu berulang, cukup afiliasi yang berbeda yang ditambahkan. Berikan alamat surel resmi afiliasi atau penulis jika diinginkan.

B. Penamaan Judul Bab dan Subbab

Bab merupakan suatu perangkat organisatorial yang memandu pembaca untuk membaca isi artikel

Anda. Terdapat dua jenis bab: bab utama (bab) dan subbab.

Bab utama mengidentifikasi komponen-komponen yang berbeda dalam artikel Anda dan tidak memiliki hubungan isi yang erat satu sama lainnya. Sebagai contoh PENDAHULUAN, DAFTAR PUSTAKA, dan UCAPAN TERIMA KASIH. Penulisan judul bab utama menggunakan huruf kapital dan penomoran angka Romawi.

Subbab merupakan isi yang dijabarkan lebih terstruktur dan memiliki relasi yang kuat. Penamaan subbab ditulis dengan menggunakan cara penulisan judul kalimat utama (*Capitalize Each Word*) dan penomorannya menggunakan huruf alfabet kapital secara berurutan. Untuk subsubbab, penamaan dan penomoran mengikuti cara penamaan dan penomoran subbab diikuti angka Arab, seperti “A.1 Penulis”, “A.1.1 Afiliasi Penulis”.

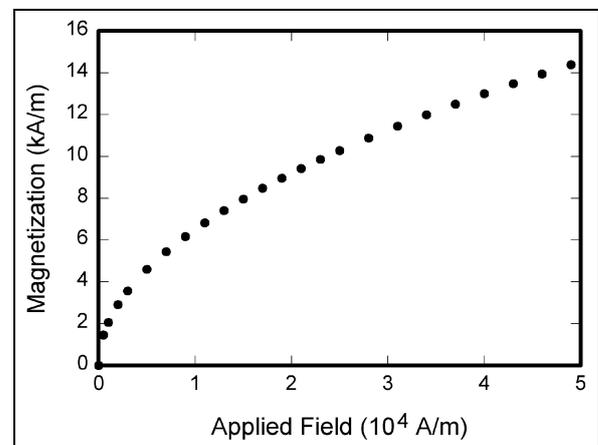
C. Gambar dan Tabel

Letakkan gambar dan tabel di atas atau di bawah kolom. Hindari posisi di tengah kolom. Gambar dan tabel yang besar dapat mengambil area dua kolom menjadi satu kolom. Judul gambar harus diletakkan di bawah gambar, sedangkan judul tabel harus diletakkan di atas tabel. Masukkan gambar dan tabel setelah mereka dirujuk di dalam isi artikel.

Tabel 1. Contoh tabel

Table Head	Table Column Head		
	Table column subhead	Subhead	Subhead
copy	More table copy		

Penamaan judul gambar dan tabel menggunakan cara penulisan kalimat biasa (*Sentence case*). Berikan jarak baris sebelum dan sesudah gambar atau tabel dengan kalimat penyertanya.



Gambar 1. Contoh gambar

V. SIMPULAN

Bagian simpulan bukan merupakan keharusan. Meskipun suatu simpulan dapat memberikan gambaran mengenai intisari artikel Anda, jangan menduplikasi abstrak sebagai simpulan Anda. Sebuah simpulan dapat menekankan pada pentingnya penelitian yang Anda lakukan atau saran pengembangan penelitian selanjutnya yang dapat dikerjakan.

LAMPIRAN

Jika diperlukan, Anda dapat menyisipkan lampiran-lampiran yang digunakan dalam artikel Anda sebelum UCAPAN TERIMA KASIH.

UCAPAN TERIMA KASIH

Di bagian ini Anda dapat memberikan pernyataan atau ungkapan terima kasih pada pihak-pihak yang telah membantu Anda dalam pelaksanaan penelitian yang Anda lakukan.

DAFTAR PUSTAKA

Untuk penamaan daftar pustaka, gunakan tanda kurung siku, seperti [1], secara berurutan dari awal rujukan dilakukan. Untuk merujuknya dalam kalimat, cukup gunakan [2], bukan "Rujukan [3]", kecuali di awal sebuah kalimat, seperti "Rujukan [3] menggambarkan"

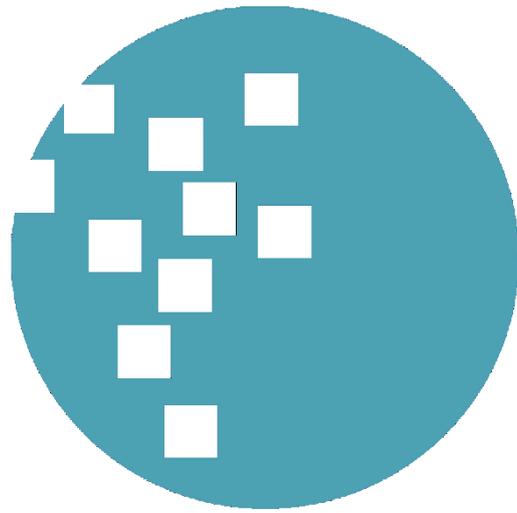
Penomoran catatan kaki dilakukan secara terpisah dengan *superscripts*. Letakkan catatan kaki tersebut di

bawah kolom dimana catatan kaki tersebut dirujuk. Jangan letakkan catatan kaki di dalam daftar pustaka.

Kecuali terdapat enam atau lebih penulis, jabarkan nama penulis tersebut satu-satu, jangan gunakan "dkk". Artikel yang belum diterbitkan, meskipun sudah dikirim untuk diterbitkan, harus ditulis "belum terbit" [4]. Artikel yang sudah dikonfirmasi untuk diterbitkan, namun belum terbit, harus ditulis "proses cetak" [5]. Gunakan cara penulisan kalimat (*Sentence case*) untuk penulisan judul artikel.

Untuk artikel yang diterbitkan dalam jurnal terjemahan, tuliskan terlebih dahulu rujukan hasil terjemahannya, diikuti dengan jurnal aslinya [6].

- [1] G. Eason, B. Noble, dan I.N. Sneddon, "On certain integrals of Lipschitz-Hankel type involving products of Bessel functions," *Phil. Trans. Roy. Soc. London*, vol. A247, hal. 529-551, April 1955.
- [2] J. Clerk Maxwell, *A Treatise on Electricity and Magnetism*, 3rd ed., vol. 2. Oxford: Clarendon, 1892, hal.68-73.
- [3] I.S. Jacobs dan C.P. Bean, "Fine particles, thin films and exchange anisotropy," in *Magnetism*, vol. III, G.T. Rado and H. Suhl, Eds. New York: Academic, 1963, hal. 271-350.
- [4] K. Elissa, "Title of paper if known," belum terbit.
- [5] R. Nicole, "Title of paper with only first word capitalized," *J. Name Stand. Abbrev.*, proses cetak.
- [6] Y. Yorozu, M. Hirano, K. Oka, dan Y. Tagawa, "Electron spectroscopy studies on magneto-optical media and plastic substrate interface," *IEEE Transl. J. Magn. Japan*, vol. 2, hal. 740-741, Agustus 1987 [Digests 9th Annual Conf. Magnetism Japan, hal. 301, 1982].
- [7] M. Young, *The Technical Writer's Handbook*. Mill Valley, CA: University Science, 1989.



UMN

**UNIVERSITAS
MULTIMEDIA
NUSANTARA**

ISSN 2355-3286



9 772355 328009



Universitas Multimedia Nusantara
Scientia Garden Jl. Boulevard Gading Serpong, Tangerang
Telp. (021) 5422 0808 | Fax. (021) 5422 0800