



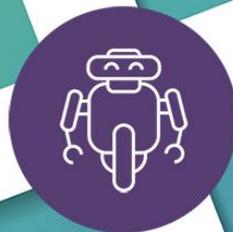
# INSYST

Journal of Intelligent System and Computation

p-ISSN: 2621-9220

e-ISSN: 2722-1962

Volume 1 Nomor 1, Oktober 2019



Published By **Lembaga Penelitian dan Pengabdian Masyarakat (LPPM)**  
**Institut Sains dan Teknologi Terpadu Surabaya (ISTTS)**  
formerly **Sekolah Tinggi Teknik Surabaya (STTS)**



Managed By  
**Departement of Informatics**  
**Institut Sains dan Teknologi Terpadu Surabaya (ISTTS)**

# INSYST

Journal of Intelligent System and Computation

Volume 01 Nomor 01 Oktober 2019

---

## **Editor in Chief:**

Dr. Yosi Kristian, S.Kom, M.Kom.  
*Institut Sains dan Teknologi Terpadu Surabaya, Indonesia*

## **Managing Editor:**

Dr. Esther Irawati Setiawan, S.Kom., M.Kom.  
*Institut Sains dan Teknologi Terpadu Surabaya, Indonesia*

Reddy Alexandro H., S.Kom., M.Kom.  
*Institut Sains dan Teknologi Terpadu Surabaya, Indonesia*

## **Editorial Board:**

Dr. Ir. Endang Setyati, M.T.  
*Institut Sains dan Teknologi Terpadu Surabaya, Indonesia*

Ir. Edwin Pramana, M.App.Sc, Ph.D  
*Institut Sains dan Teknologi Terpadu Surabaya, Indonesia*

Prof. Dr. Ir. Mauridhi Hery Purnomo, M.T.  
*Institut Sepuluh November, Indonesia*

Hindriyanto Dwi Purnomo, Ph.D.  
*Universitas Kristen Satya Wacana, Salatiga, Indonesia*

Hendrawan Armanto, S.Kom., M.Kom.  
*Institut Sains dan Teknologi Terpadu Surabaya, Indonesia*

Dr. Lukman Zaman PCSW, M.Kom.  
*Institut Sains dan Teknologi Terpadu Surabaya, Indonesia*

Dr. Diana Purwitasari, S.Kom., M.Sc.  
*Institut Sepuluh November, Indonesia*

# INSYST

Journal of Intelligent System and Computation

Volume 01 Nomor 01 Oktober 2019

---

## **Reviewer:**

Teguh Wahyono, S.Kom., M.Cs.

*Universitas Kristen Satya Wacana, Salatiga, Indonesia*

Dr. Anang Kukuh Adisusilo, ST, MT.

*Universitas Wijaya Kusuma, Surabaya, Indonesia*

Dr. I Ketut Eddy Purnama, ST., MT.

*Institut Sepuluh November, Indonesia*

Prof. Dr. Benny Tjahjono, M.Sc.

*Coventry University, United Kingdom*

Dr. Ir. Gunawan, M.Kom.

*Institut Sains dan Teknologi Terpadu Surabaya, Indonesia*

Dr. Umi Laili Yuhana S.Kom., M.Sc.

*Institut Sepuluh November, Indonesia*

Dr. Tita Karlita, S.Kom., M.Kom.

*Politeknik Elektronika Negeri Surabaya, Indonesia*

Dr. Ir. Rika Rokhana, M.T.

*Politeknik Elektronika Negeri Surabaya, Indonesia*

Dr. I Made Gede Sunarya, S.Kom., M.Cs.

*Universitas Pendidikan Ganesha, Indonesia*

Dr. Yuni Yamasari, S.Kom., M.Kom.

*Universitas Negeri Surabaya, Indonesia*

Dr. Adri Gabriel Sooai, S.T., M.T.

*Universitas Katolik Widya Mandira, Indonesia*

# INSYST

Journal of Intelligent System and Computation

Volume 01 Nomor 01 Oktober 2019

---

## Daftar Isi

<b>Multilevel Image Thresholding Memanfaatkan Firefly Algorithm, Improved Bat Algorithm, dan Symbiotic Organisms Search</b> Pickerling Pickerling, Hendrawan Armanto, Stefanus Kurniawan Bastari .....	1
<b>Augmented Reality Marker Based Tracking Visualisasi Drawing 2D ke dalam Bentuk 3D dengan Metode FAST Corner Detection</b> Nanang Wahyudi, Reddy Alexandro Harianto, Endang Setyati .....	9
<b>Klasifikasi SMS Center RSUD SMART Berdasarkan Jenis Keluhan Pelayanan Menggunakan Naive Bayes</b> Sholeh Rachmatullah, Esther Irawati Setiawan, Reddy Alexandro Harianto .....	19
<b>Klasifikasi Citra Daun Memanfaatkan Angular Partition, Edge Detection dan Neural Network</b> Elkana Lewi Santoso, Endang Setyati, Yosi Kristian .....	27
<b>Game Battle of Artificial Intelligence Berbasis Android</b> Riandika Lumaris, Hendrawan Armanto, Willy Adrianus .....	34
<b>Analisis Peramalan Stok Barang dengan Metode Weight Moving Average dan Double Exponential Smoothing pada Jovita Ms Glow Lamongan</b> Nur Nafi'iyah .....	39
<b>Analisis Sentimen Pengguna Twitter Terhadap Polemik Persepakbolaan Indonesia Menggunakan Pembobotan TF-IDF dan K-Nearest Neighbor</b> Jeremy Andre Septian, Tresna Maulana Fachrudin, Aryo Nugroho .....	43

# Multilevel Image Thresholding Memanfaatkan Firefly Algorithm, Improved Bat Algorithm, dan Symbiotic Organisms Search

C. Pickerling, *Informatika Institut Sains dan Teknologi Terpadu Surabaya*,  
 Hendrawan Armanto, *Informatika Institut Sains dan Teknologi Terpadu Surabaya*,  
 dan Stefanus Kurniawan B., *Informatika Institut Sains dan Teknologi Terpadu Surabaya*

**Abstrak** — Multilevel image thresholding adalah teknik penting dalam pemrosesan gambar yang digunakan sebagai dasar image segmentation dan teknik pemrosesan tingkat tinggi lainnya. Akan tetapi, waktu yang dibutuhkan untuk pencarian bertambah secara eksponensial setara dengan banyaknya threshold yang diinginkan. Algoritma metaheuristic dikenal sebagai metode optimal untuk memecahkan masalah perhitungan yang rumit. Seiring dengan berkembangnya algoritma metaheuristic untuk memecahkan masalah perhitungan, penelitian ini menggunakan tiga algoritma metaheuristic, yaitu Firefly Algorithm (FA), Symbiotic Organisms Search (SOS), dan Improved Bat Algorithm (IBA). Penelitian ini menganalisis solusi optimal yang didapatkan dari percobaan masing-masing algoritma. Hasil uji coba masing-masing algoritma saling dibandingkan untuk menentukan kelemahan dan kelebihan setiap algoritma berdasarkan performanya. Hasil uji coba menyatakan tiga algoritma tersebut memiliki performa berbeda dalam optimisasi multilevel image thresholding.

**Kata Kunci** — Multilevel Image Thresholding, Threshold, Metaheuristic

## I. PENDAHULUAN

Dewasa ini, kebutuhan untuk pengolahan gambar semakin berkembang seiring dengan kemajuan teknologi. Pengolahan ini memegang peran yang penting dalam analisis dan interpretasi gambar di berbagai aspek kehidupan seperti medis, navigasi, deteksi kejadian secara otomatis, pengamatan, pengenalan tekstur beserta pola dan deteksi kecacatan. Pengembangan teknik citra digital dan teknologi komputasi membawa kemajuan pada ilmu pencitraan.

Salah satu teknik penting dalam ilmu pencitraan adalah image segmentation. Image segmentation membagi gambar menjadi beberapa kelompok dengan kemiripan sifat seperti intensitas, warna, dan kontur. Biasanya, image segmentation melambangkan tahapan awal pemahaman gambar dan dilanjutkan ke metode tingkat lanjut seperti ekstraksi fitur,

interpretasi semantik, pengenalan gambar, dan klasifikasi benda.

Multilevel image thresholding merupakan teknik umum dalam menjalankan image segmentation. Gambar yang dihasilkan gadget menghasilkan resolusi yang berbeda sesuai spesifikasi kamera yang dipakai. Kamera dengan resolusi tinggi dapat menghasilkan gambar dengan ukuran besar. Multilevel image thresholding membutuhkan waktu yang lama dan juga banyak perhitungan dikarenakan besarnya ukuran gambar dan banyaknya threshold.

Masalah waktu dan kebutuhan perhitungan yang tinggi akan menyebabkan proses pengolahan gambar tidak efisien. Untuk mengatasi masalah ini, algoritma metaheuristic diharapkan mampu mempercepat konvergensi dan mengurangi waktu perhitungan. Sebagian algoritma metaheuristic berasal dari perilaku makhluk hidup dan penerapan sistem di alam. Algoritma metaheuristic yang mendapat inspirasi dari makhluk hidup seperti Firefly Algorithm, Improved Bat Algorithm, dan Symbiotic Organisms Search adalah algoritma yang belum banyak diterapkan untuk penyelesaian multilevel image thresholding. Pada penelitian ini akan dibahas implementasi Firefly Algorithm, Improved Bat Algorithm, dan Symbiotic Organisms Search untuk multilevel image thresholding

## II. TINJAUAN PUSTAKA

Penelitian ini menggunakan beberapa tinjauan pustaka sebagai dasar agar penelitian dapat berjalan sesuai arah yang tepat. Tinjauan pustaka yang digunakan antara lain:

### A. OTSU Thresholding [1]

Teknik thresholding melakukan pemrosesan gambar berdasarkan informasi yang terdapat pada histogram *gray-level*. Jika input gambar *grayscale*  $I$  dianggap sebagai sekumpulan pixel  $A$ , maka multilevel thresholding dapat didefinisikan sebagai metode untuk membagi kumpulan pixel  $A$  menjadi kelompok-kelompok kecil ( $A_0, A_1, \dots, A_n$ ) dengan kumpulan threshold ( $t_0, t_1, \dots, t_{(n-1)}$ ) dengan definisi berikut:

$$\begin{aligned} A_0 &= \{x: 0 \leq f(x) < t_0\}, \\ A_1 &= \{x: t_0 \leq f(x) < t_1\}, \\ &\vdots \\ A_n &= \{x: t_{n-1} \leq f(x) \leq L - 1\}. \end{aligned} \quad (1)$$

C. Pickerling, Departemen Informatika, Institut Sains dan Teknologi Terpadu Surabaya, Jawa Timur, Indonesia (e-mail: pickerling@sts.edu)

Hendrawan Armanto, Departemen Informatika, Institut Sains dan Teknologi Terpadu Surabaya, Jawa Timur, Indonesia (e-mail: hendrawan@sts.edu)

Stefanus Kurniawan B., Departemen Informatika, Informatika, Institut Sains dan Teknologi Terpadu Surabaya, Surabaya, Jawa Timur, Indonesia.

$x = (x_1, x_2)$  adalah pixel yang didefinisikan dengan koordinat  $x_1$  dan  $x_2$  pada sistem koordinat Kartesius,  $f(x)$  melambangkan nilai gray-level untuk pixel  $x$ , dimana nilai  $f(x)$  berada di rentang  $[0, \dots, 255]$ . Tujuan dari multilevel thresholding adalah menghitung nilai optimal untuk kumpulan threshold  $(t_0, t_1, \dots, t_{(n-1)})$ . Kelompok-kelompok kecil  $(A_0, A_1, \dots, A_n)$  melambangkan bagian berbeda pada gambar I. Berlaku  $A_i \cap A_j = \emptyset$  untuk setiap kumpulan pixel  $A$ , dan gabungan seluruh kelompok kecil adalah keseluruhan input gambar grayscale I.

Pemilihan threshold optimal untuk image thresholding semula tidak memiliki banyak perhitungan, lain halnya dengan multilevel image thresholding yang memiliki banyak perhitungan dan membutuhkan waktu yang lama seiring bertambahnya threshold untuk dioptimalkan. Kumpulan threshold optimal untuk multilevel image thresholding ditentukan dengan mengoptimalkan kriteria sesuai histogram gray-level dari input gambar grayscale I. [2]

Metode Otsu dalam memaksimalkan variansi between-class adalah metode populer untuk image thresholding. Algoritma untuk metode ini dijelaskan sebagai berikut. Diasumsikan gambar grayscale I memiliki kelas sebanyak  $m+1$   $[C_0, C_1, \dots, C_m]$ , di mana  $m$  adalah inputan dimensi dan probabilitas setiap kelas dengan definisi berikut: [3]

$$\begin{aligned} \omega_0 &= \sum_{i=0}^{t_0-1} p_i, \\ \omega_1 &= \sum_{i=t_0}^{t_1-1} p_i, \\ &\vdots \\ \omega_m &= \sum_{i=t_{m-1}}^{L-1} p_i, \end{aligned} \quad (2)$$

di mana  $t_{\text{number}}$  adalah threshold yang memisahkan gambar I menjadi beberapa kelas. Untuk threshold sebanyak  $m$  yang membagi gambar I menjadi  $(m+1)$  kelas, dihitung nilai fungsi objektif dengan memaksimalkan variansi between-class seperti berikut:

$$\begin{aligned} \text{Maximize } J(t) &= \sigma_0 + \sigma_1 + \dots + \sigma_m, \\ \sigma_0 &= \omega_0(\mu_0 - \mu_T)^2, \\ \sigma_1 &= \omega_1(\mu_1 - \mu_T)^2, \\ &\vdots \\ \sigma_m &= \omega_m(\mu_m - \mu_T)^2. \end{aligned}$$

### B. Firefly Algorithm [4]

Firefly Algorithm (FA) adalah algoritma metaheuristic dengan inspirasi perilaku kunang-kunang dalam memancarkan cahaya. Pada Firefly Algorithm, terdapat tiga macam aturan yang sudah dianggap ideal: (1) semua kunang-kunang dianggap unisex, jadi seekor kunang-kunang dapat tertarik dengan kunang-kunang yang lain tanpa mempedulikan jenis kelamin kunang-kunang lain; (2) ketertarikan kunang-kunang pada kumpulan berbanding lurus dengan tingkat terangnya cahaya. Untuk dua kunang-kunang yang saling memancarkan cahayanya, kunang-kunang dengan cahaya yang lebih redup akan bergerak ke

arah kunang-kunang yang lebih terang. Apabila jarak mereka semakin jauh, ketertarikan mereka akan semakin berkurang. Seekor kunang-kunang akan bergerak secara random apabila tidak ada yang lebih terang daripada kunang-kunang tersebut; (3) terangnya kunang-kunang dipengaruhi atau ditentukan oleh bagaimana fungsi objektifnya.

Pergerakan kunang-kunang  $i$  yang tertarik oleh kunang-kunang  $j$  dengan pancaran cahaya lebih terang ditentukan oleh:

$$\begin{aligned} x_i &= x_i + \beta(x_j - x_i) + \alpha \left( \text{rand} - \frac{1}{2} \right), \\ \beta(d_{ij}) &= \beta_0 e^{-\gamma d_{ij}^2}, \end{aligned} \quad (4)$$

dengan  $x_i$  adalah posisi kunang-kunang  $i$  dan  $x_j$  adalah posisi kunang-kunang  $j$ .  $\beta(d_{ij})$  melambangkan ketertarikan kunang-kunang  $i$  terhadap kunang-kunang  $j$ , di mana  $d_{ij}$  adalah jarak dari kunang-kunang  $i$  dan kunang-kunang  $j$ .  $\beta_0$  adalah ketertarikan pada  $d_{ij}=0$ , dan  $\gamma$  adalah koefisien penyerapan cahaya.  $\alpha$  adalah parameter yang bernilai random, dan  $\text{rand}$  adalah nilai random antara  $[0, 1]$  yang dirandom setiap kali dipakai.

Langkah utama Firefly Algorithm dapat dituliskan sebagai berikut: (1) menghitung nilai fungsi objektif setiap kunang-kunang; (2) kunang-kunang dengan nilai fungsi objektif yang kurang optimal bergerak menuju kunang-kunang yang lebih optimal; (3) menentukan kunang-kunang yang memiliki nilai fungsi objektif paling optimal atau terbaik dan mencatatnya; (4) mengulangi iterasi hingga batas iterasi maksimum; (5) penggunaan kunang-kunang terbaik pada akhir iterasi maksimum.

### C. Improve Bat Algorithm [5]

Bat Algorithm dipilih untuk menyelesaikan masalah multilevel image thresholding karena mudah untuk diimplementasikan dan menghasilkan solusi yang baik untuk threshold sebanyak  $m$  dimensi, di mana  $m$  adalah angka yang kecil. Akan tetapi, Bat Algorithm dianggap kurang berhasil untuk threshold berdimensi besar, sehingga perlu ada perubahan pada algoritma ini [7]. Improved Bat Algorithm (IBA) dibuat untuk mengatasi kekurangan dari Bat Algorithm.

Improved Bat Algorithm menggabungkan dua perhitungan untuk pencarian solusi dari Bat Algorithm dan algoritma Differential Evolution (DE). Improved Bat Algorithm mengambil operator mutation dan crossover dari algoritma Differential Evolution dengan tujuan mempercepat pencarian solusi secara konvergen dan mencapai keseimbangan yang baik antara peningkatan dan penggolongan. Operator mutation dan crossover digunakan untuk meningkatkan perhitungan solusi lain ( $x_{\text{new}}$ ) dari Bat Algorithm semula, sehingga Improved Bat Algorithm dapat mengeksplorasi dan mengeksploitasi area pencarian baru dan dapat menghindari jebakan local optima.

Pada Bat Algorithm, eksplorasi dan eksploitasi diatur oleh besarnya pulse emission rate. Dengan menganalisa perhitungan ini, algoritma semakin kehilangan kemampuan eksplorasinya seiring berjalannya iterasi. Pada Improved Bat

Algorithm, bagian ini dimodifikasi agar dapat bergantian dalam eksplorasi dan eksploitasi lebih baik. Kemampuan eksplorasi dari Bat Algorithm dimodifikasi dengan menambahkan operator crossover dan mutation.

Meskipun modifikasi yang telah dijelaskan di atas dapat meningkatkan performa banyak solusi, beberapa solusi akan tetap terjebak di dalam local optima. Untuk memperbaiki kekurangan pada modifikasi sebelumnya, ditambahkan modifikasi kedua yang mendapatkan inspirasi dari lebah pengintai pada fase mengintai milik algoritma Artificial Bee Colony (ABC). Ketika beberapa solusi terjebak pada local optima setelah beberapa iterasi, akan ada perhitungan yang memantau apakah jumlahnya melebihi batas. Ketika batas tersebut dilalui, solusi akan mencari area baru agar dapat terlepas dari local optima.

Dengan adanya penggunaan echolocation dari microbats, dibuatlah Bat Algorithm yang terinspirasi dari perilaku kelelawar tersebut. Improved Bat Algorithm memiliki peraturan yang sama dengan Bat Algorithm. Untuk menjalankan Improved Bat Algorithm, berikut ini adalah peraturan yang sudah dianggap ideal: (1) semua kelelawar menggunakan echolocation untuk memperkirakan jarak, dan mereka dianggap sudah mengetahui sekitarnya; (2) kelelawar terbang secara random dengan kecepatan  $v_i$  pada posisi  $x_i$  dengan frekuensi tetap  $f_{min}$ , panjang gelombang  $\lambda$  yang bervariasi, dan kebisingan  $A_0$  untuk mencari mangsa. Mereka dapat menyesuaikan panjang gelombang yang dipancarkan secara otomatis dan mengatur pulse emission rate  $r$  [0, 1] tergantung dekatnya mereka dengan mangsa; (3) meskipun kebisingan dapat bervariasi dengan banyak cara, diasumsikan bahwa kebisingan bervariasi dari nilai positif  $A_0$  hingga nilai minimum  $A_{min}$  yang konstan.

Pada Improved Bat Algorithm, populasi kelelawar melambangkan keseluruhan solusi. Detail langkah-langkah Improved Bat Algorithm dijelaskan sebagai berikut:

*Langkah 1 (inisialisasi populasi kelelawar).* Improved Bat Algorithm memunculkan populasi awal secara random dengan ukuran populasi kelelawar  $N$  ( $i=1, 2, \dots, N$ ).  $N$  juga disebut sebagai banyaknya solusi, di mana setiap solusi memiliki  $m$  dimensi. Pada tahapan inisialisasi ini, dibuat parameter yang memberikan batasan kelelawar agar tidak terjebak pada local optima yang berkaitan dengan langkah 4. Selain itu perlu adanya input awal berupa nilai kebisingan awal ( $A_i^0$ ), pulse emission rate awal ( $r_i^0$ ), termasuk tambahan parameter Differential Evolution berupa bobot differential ( $F$ ) dan kemungkinan crossover ( $C_r$ ). Fungsi objektif untuk semua solusi dievaluasi dan iterasi untuk langkah berikutnya dapat dimulai. Improved Bat Algorithm mendeteksi solusi terbaik sebagai  $x_{best}$  sebelum iterasi dapat dimulai.

*Langkah 2 (menghitung populasi baru).* Solusi baru dihitung agar kelelawar dapat bergerak. Untuk iterasi saat ini ( $t$ ), perhitungan solusi baru ( $x_i^t$ ) dilakukan dengan menggerakkan solusi lama dari iterasi sebelumnya ( $x_i^{t-1}$ ) dengan kecepatan ( $v_i^t$ ) dan frekuensi ( $f_i^t$ ) seperti berikut:

$$\begin{aligned} x_i^t &= x_i^{t-1} + v_i^t, \\ v_i^t &= v_i^{t-1} + (x_i^t - x_{best}) * f_i^t, \\ f_i^t &= f_{min} + (f_{max} - f_{min}) * \beta. \end{aligned} \quad (5)$$

Pada langkah perhitungan ini, Improved Bat Algorithm memantau kondisi perbatasan dengan solusi baru ( $x_i^t$ ). Apabila nilai suatu variabel keluar dari batasan, maka nilai variabel tersebut perlu dirubah.

*Langkah 3 (meningkatkan solusi terbaik saat ini dengan operator DE).* Untuk setiap solusi baru ( $x_i^t$ ), dibuatlah solusi lain ( $x_{new}$ ) seperti berikut:

$$\begin{aligned} x_{new} &= \begin{cases} x_{dif}^t, & \text{if } rand_1 > r_i^t, \\ x_{loc}^t, & \text{else,} \end{cases} \\ r_i^t &= r_i^0 (1 - e^{-\beta t}), \end{aligned} \quad (6)$$

di mana  $rand_1$  adalah nilai random dari rentang [0, 1],  $r_i^t$  adalah pulse emission rate.  $x_{dif}^t$  adalah tambahan modifikasi berprinsip Differential Evolution (DE) dengan adanya operator untuk mutation dan crossover, sedangkan  $x_{loc}^t$  adalah operator pencarian lokal yang ada dari Bat Algorithm. Operator mutation dan crossover dituliskan sebagai berikut:

$$x_{dif,j}^t = \begin{cases} x_{c,j}^t + F(x_{a,j}^t - x_{b,j}^t), & \text{if } (rand_2 < C_r \text{ or } j = j_r), \\ x_{i,j}^t, & \text{else,} \end{cases} \quad (7)$$

di mana  $x_{a,j}^t$ ,  $x_{b,j}^t$ ,  $x_{c,j}^t$  adalah nilai random dari [0, N-1] yang berada pada iterasi kini ( $t$ ) di posisi dimensi  $j$ ,  $F$  adalah bobot differential yang mengatur besarnya modifikasi,  $C_r$  adalah kemungkinan crossover,  $j_r$  adalah dimensi bernilai random, dan  $rand_2$  merupakan nilai random dari rentang [0, 1]. Penambahan operator differential  $x_{dif,j}^t$  berguna untuk meningkatkan keberagaman posisi kelelawar dan mencapai akurasi serta efisiensi pencarian. Operator pencarian lokal dituliskan sebagai berikut:

$$\begin{aligned} x_{loc,j}^t &= \begin{cases} x_{lbest,j}^t, & \text{if } (f(x_{lbest,j}^t) > f(x_{i,j}^t)), \\ x_{i,j}^t, & \text{else,} \end{cases} \\ x_{lbest,j}^t &= x_{best,j}^{t-1} + \epsilon A_{i,j}^{t-1}. \end{aligned} \quad (8)$$

*Langkah 4 (mengecek solusi baru).* Pada langkah ini, solusi lain ( $x_{new}$ ) dapat diterima sebagai solusi baru dan  $f(x_{new})$  sebagai nilai fungsi objektif dengan detail:

$$\begin{aligned} &(x_i^t, f(x_i^t)) \\ &= \begin{cases} (x_{new}, f(x_{new})), \\ \text{if } (rand_3 < A_i^t \text{ and } f(x_{new}) > f(x_{i,old}^{t-1})), \\ (x_i^{t-1}, f(x_i^{t-1})) \end{cases} \text{tr}_i = \text{tr}_i + 1, \text{ else,} \\ &A_i^t = \alpha A_i^{t-1}, \end{aligned} \quad (9)$$

di mana  $rand_3$  adalah nilai random dari rentang [0, 1].  $\text{tr}_i$  adalah parameter tambahan agar solusi dapat keluar dari local optima dengan mengubah nilai  $x_i^t$  dengan nilai random. Apabila  $\text{tr}_i$  melebihi batas yang telah ditentukan, nilai  $x_i^t$  digantikan nilai random dan  $\text{tr}_i$  kembali ke nilai 0.

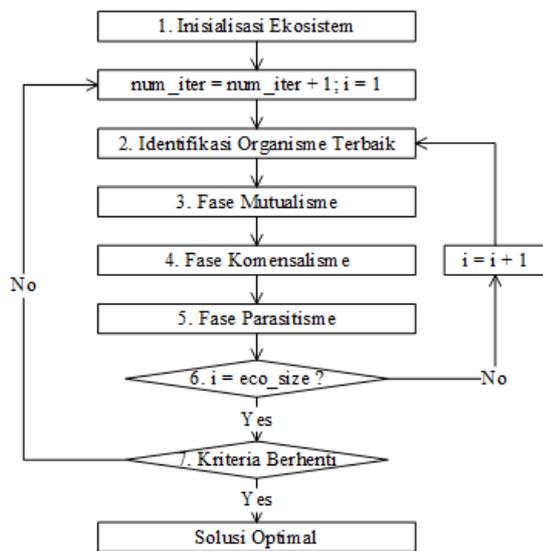
*Langkah 5 (mengingat solusi terbaik).* Mencatat solusi

terbaik saat ini ( $X_{best}$ ) yang merupakan solusi dengan nilai fungsi objektif paling optimal.

*Langkah 6 (mengecek iterasi).* Apabila iterasi saat ini sama dengan iterasi maksimum, algoritma berhenti. Selain itu, nilai iterasi ditambahkan satu dan langkah 2 diulang kembali.

**D. Symbiotic Organisms Search [6]**

Mirip dengan algoritma metaheuristic lain yang berbasis populasi, Symbiotic Organisms Search menggunakan sebuah populasi yang terdiri dari kandidat solusi di dalam area pencarian untuk mendapatkan solusi terbaik secara keseluruhan. Symbiotic Organisms Search dimulai dengan inisialisasi populasi yang disebut dengan ekosistem. Pada ekosistem awal, kumpulan organisme dimunculkan dengan posisi acak di area pencarian. Setiap organisme merepresentasikan satu kandidat solusi untuk masalah yang ingin diselesaikan. Setiap organisme di dalam ekosistem memiliki nilai fungsi objektif yang menandakan derajat adaptasi pada objektif yang diinginkan.



Gambar 1. Flowchart Symbiotic Organisms Search

Karakter pada setiap interaksi menentukan prinsip utama untuk setiap fase di dalam algoritma ini. Interaksi yang menguntungkan dua belah pihak berada di dalam fase mutualisme, menguntungkan satu pihak saja dan tidak mempengaruhi pihak lain berada di dalam fase komensalisme, menguntungkan satu pihak dan merugikan pihak lain berada di dalam fase parasitisme. Gambar 1 adalah flowchart dari penjelasan yang telah diberikan.

*Fase Mutualisme.* Contoh dari mutualisme yang menguntungkan dua belah pihak organisme adalah hubungan antara lebah dan tanaman berbunga. Lebah mendapatkan nektar dan tanaman berbunga mendapatkan penyerbukan. Fase Symbiotic Organisms Search ini menirukan hubungan saling menguntungkan tersebut. Di dalam Symbiotic Organisms Search,  $X_i$  adalah organisme yang merupakan anggota ke- $i$  dari ekosistem. Organisme lain  $X_j$  dipilih secara random dari ekosistem agar dapat

berinteraksi dengan  $X_i$ . Dua organisme menjalin hubungan saling menguntungkan dengan tujuan meningkatkan faktor bertahan hidup di dalam ekosistem. Kandidat solusi baru untuk  $X_i$  dan  $X_j$  dihitung berdasarkan hubungan saling menguntungkan di antara keduanya seperti berikut:

$$\begin{aligned}
 X_{i\_new} &= X_i + rand * (X_{best} - MV * BF_1), \\
 X_{j\_new} &= X_j + rand * (X_{best} - MV * BF_2), \quad (10) \\
 MV &= \frac{X_i + X_j}{2}.
 \end{aligned}$$

Variabel rand adalah nilai random berkisar [0, 1]. Peran dari parameter  $BF_1$  dan  $BF_2$  dijelaskan sebagai berikut. Pada hubungan mutualisme di dalam alam, satu organisme mungkin lebih diuntungkan daripada pihak organisme yang lain. Faktor keuntungan atau Benefit Factors ( $BF_1$  dan  $BF_2$ ) ditentukan secara random apakah bernilai 1 atau 2. Faktor ini merepresentasikan seberapa besar organisme diuntungkan. Mutual\_Vector (MV) merupakan representasi karakteristik hubungan organisme  $X_i$  dan  $X_j$ . Bagian perhitungan ( $X_{best} - MV * BF$ ) merefleksikan usaha bersama mereka untuk meraih keuntungan dalam bertahan hidup. Solusi baru dengan nilai yang lebih baik akan menggantikan solusi lama, selain itu solusi lama tetap bertahan.

*Fase Komensalisme.* Contoh dari komensalisme adalah hubungan ikan remora dan hiu. Ikan remora menempelkan bagian atas tubuhnya pada hiu. Ikan remora memakan sisa makanan yang ada, sehingga ikan ini mendapatkan keuntungan. Hiu tidak terpengaruh dengan kegiatan ikan remora yang menempel padanya. Mirip dengan fase mutualisme, organisme  $X_j$  dipilih secara random dari ekosistem untuk berinteraksi dengan organisme  $X_i$ . Pada keadaan ini, organisme  $X_i$  berusaha untuk mendapatkan keuntungan dari interaksi. Akan tetapi, organisme  $X_j$  tidak mendapatkan keuntungan dan tidak mendapat kerugian dari hubungan ini. Kandidat solusi baru didapatkan dari modifikasi organisme  $X_i$  yang dihitung berdasarkan simbiosis komensalisme antara organisme  $X_i$  dan  $X_j$  seperti berikut:

$$X_{i\_new} = X_i + rand * (X_{best} - X_j), \quad (11)$$

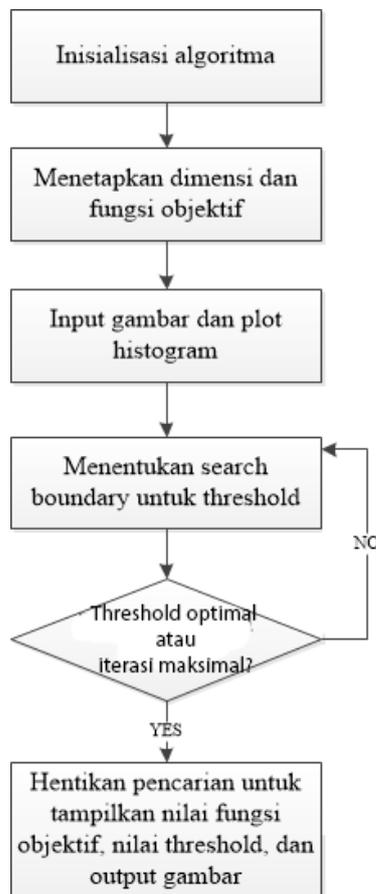
dengan bagian ( $X_{best} - X_j$ ) yang merupakan perhitungan tentang berapa besar keuntungan yang disediakan oleh  $X_j$  untuk membantu  $X_i$  dalam meningkatkan kesuksesan bertahan hidup di dalam ekosistem menuju adaptasi terbaik yang dimiliki organisme  $X_{best}$ . Solusi baru dengan nilai yang lebih baik akan menggantikan solusi lama, selain itu solusi lama tetap bertahan.

*Fase Parasitisme.* Contoh dari parasitisme adalah parasit plasmodium yang biasanya menggunakan perantara nyamuk anopheles untuk berpindah-pindah ke tubuh manusia yang lain sebagai inang. Ketika parasit plasmodium tiba, tubuh manusia akan terserang penyakit malaria dan dapat mengakibatkan kematian sebagai akibatnya. Hal ini tentu merugikan manusia dan hanya menguntungkan parasit plasmodium yang memanfaatkan tubuh manusia. Pada Symbiotic Organisms Search, organisme  $X_i$  memiliki peran yang serupa dengan nyamuk anopheles sebagai media

penyebaran parasit bernama Parasite\_Vector. Parasite\_Vector dibentuk di dalam area pencarian solusi dengan menduplikasi organisme  $X_i$ , kemudian memodifikasi hasil duplikasi organisme  $X_i$ . Modifikasi ini mengganti solusi yang dimiliki induk pada dimensi yang random dengan angka random. Dimensi yang dipilih dapat bervariasi dari hanya satu dimensi saja hingga keseluruhan dimensi. Organisme  $X_j$  adalah organisme selain  $X_i$  yang dipilih secara random dari ekosistem dan memiliki peran serupa tubuh manusia yang dijadikan inang oleh parasit plasmodium. Organisme  $X_j$  terserang oleh Parasite\_Vector dan Parasite\_Vector dapat menggantikan keberadaan organisme  $X_j$  di dalam ekosistem. Apabila Parasite\_Vector lebih baik daripada organisme  $X_j$ , maka Parasite\_Vector akan membunuh organisme  $X_j$  dan diterima sebagai organisme pengganti di dalam ekosistem. Sedangkan organisme  $X_j$  akan bertahan jika  $X_j$  lebih baik daripada Parasite\_Vector karena  $X_j$  memiliki imunitas pertahanan diri yang cukup dan Parasite\_Vector tidak dapat bertahan hidup.

### III. METODE DAN INTI PENELITIAN

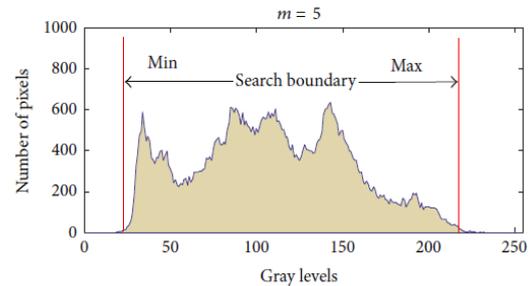
Permasalahan multilevel image thresholding berurusan dengan mencari nilai beberapa threshold yang memaksimalkan  $J(t)$ . Dimensi ( $m$ ) pencarian untuk masalah optimisasi ditentukan berdasarkan banyaknya threshold. Gambar 2 merupakan flowchart bagaimana penelitian ini bekerja.



Gambar. 2. Flowchart Alur Penelitian

FA [7,8], SOS, dan IBA [9] digunakan untuk mencari kumpulan nilai threshold. Pada metode optimisasi dengan

algoritma metaheuristic, adanya search boundary dapat mempercepat program dalam menemukan nilai optimal. Ketika banyaknya dimensi ( $m$ ) ditingkatkan, maka kerumitan untuk masalah optimisasi meningkat pula. Penggunaan search boundary terdapat pada Gambar 3.



Gambar. 3. Histogram Gray Level

Gambar 3 menunjukkan histogram gray-level menggunakan search boundary. Area pencarian dengan search boundary menjadi [20, 220] daripada area pencarian awal [0, 255]. Adanya search boundary memungkinkan algoritma menemukan nilai optimal pada iterasi yang lebih kecil.

### IV. UJI COBA

Percobaan dilakukan dengan berbagai macam gambar grayscale. Semua ukuran gambar yang dipakai disesuaikan menjadi 256x256 pixel agar dapat dibuat perbandingannya dengan seimbang, terutama dalam perhitungan waktu. Percobaan multilevel image thresholding menggunakan metode Otsu untuk mencari nilai optimal secara manual atau tanpa menggunakan algoritma metaheuristic tidak dilakukan karena beberapa penelitian telah mencobanya dan waktu perhitungan akan meningkat secara eksponensial sesuai dengan banyaknya dimensi ( $m$ ), sehingga setiap algoritma akan selesai ketika iterasi maksimal.

Besarnya dimensi ( $m$ ) yang dipakai adalah 2, 3, 4, dan 5. Karena algoritma multilevel image thresholding memiliki pengaruh nilai random, setiap percobaan diulang sebanyak 10 kali untuk gambar dan parameter yang sama. Uji coba dilakukan terlebih dahulu dengan membandingkan parameter besarnya populasi. Setelah itu, besarnya iterasi juga dibandingkan untuk mengetahui batasan iterasi ideal yang digunakan. Pada tahap akhir, uji coba menggunakan kombinasi besarnya populasi dan iterasi yang terbaik untuk menguji 20 gambar grayscale berbeda. Semua uji coba menggunakan Microsoft Visual Studio dengan bahasa C#.

**Uji Coba 1 (parameter populasi).** Pada tiga algoritma yang telah dibandingkan, banyak populasi ( $P_0$ ) ditentukan terlebih dahulu sebelum menentukan banyak iterasi ( $I_{tr}$ ) dengan uji coba perbandingan parameter populasi. Pada uji coba ini, banyak iterasi ditetapkan bernilai 10 dengan dimensi  $m$  bernilai 5 (dimensi tertinggi untuk uji coba) dan hanya menggunakan satu gambar saja (*Barbara*) untuk melihat bagaimana pengaruh perubahan parameter populasi terhadap hasilnya. Uji coba ini diawali dengan parameter populasi sebesar 3 untuk masing-masing algoritma metaheuristic yang digunakan dan terus ditingkatkan agar

dapat memantau pengaruhnya.

TABEL I  
PERFORMA FIREFLY ALGORITHM (POPULASI)

Po	Itr	Best	Mean	Std	Std / Mean (%)	Avg Time (ms)
3	10	2554.59	2150.98	376.05	17.48	12.80
5	10	2654.63	2382.36	259.41	10.89	13.20
10	10	2668.18	2481.18	168.18	6.78	29.20
15	10	2693.00	2504.42	155.80	6.22	52.80
20	10	2681.24	2594.20	68.48	2.64	86.70
40	10	2651.69	2587.09	63.07	2.44	310.20
60	10	2678.53	2597.45	81.00	3.12	675.00

TABEL II  
PERFORMA SYMBIOTIC ORGANISMS SEARCH (POPULASI)

Po	Itr	Best	Mean	Std	Std / Mean (%)	Avg Time (ms)
3	10	2687.59	2641.26	39.62	1.50	13.30
5	10	2700.91	2645.35	73.93	2.79	16.00
10	10	2699.55	2691.34	10.12	0.38	22.60
15	10	2700.75	2691.56	13.95	0.52	31.10
20	10	2702.41	2699.93	3.18	0.12	39.30
40	10	2702.43	2699.66	4.63	0.17	71.30
60	10	2702.38	2701.92	0.74	0.03	102.80

TABEL III  
PERFORMA IMPROVED BAT ALGORITHM (POPULASI)

Po	Itr	Best	Mean	Std	Std / Mean (%)	Avg Time (ms)
3	10	2696.27	2641.28	69.39	2.63	14.60
5	10	2700.16	2690.97	9.17	0.34	15.80
10	10	2701.18	2697.99	3.58	0.13	20.00
15	10	2702.46	2699.80	2.10	0.08	27.30
20	10	2702.34	2698.32	3.42	0.13	32.00
40	10	2702.41	2699.85	2.49	0.09	54.60
60	10	2701.90	2699.39	3.40	0.13	84.20

Tabel 1, 2, dan 3 mencantumkan solusi terbaik (Best) untuk beberapa macam nilai populasi. Akan tetapi, dibutuhkan perhitungan nilai rata-rata (Mean) dan standar deviasi atau besarnya penyimpangan (Std) untuk menentukan kestabilan algoritma yang digunakan. Pada uji coba ini, populasi bernilai 20 dianggap cukup stabil untuk digunakan.

**Uji Coba 2 (parameter iterasi).** Pada tiga algoritma yang telah dibandingkan, banyak populasi ditentukan bernilai 20 untuk uji coba perbandingan parameter iterasi. Pada uji coba ini, dimensi m bernilai 5 (dimensi tertinggi untuk uji coba) dan hanya menggunakan satu gambar saja (Barbara) untuk melihat bagaimana pengaruh perubahan parameter iterasi terhadap hasilnya. Uji coba ini diawali dengan parameter iterasi sebesar 3 untuk masing-masing algoritma metaheuristic yang digunakan dan terus ditingkatkan agar dapat memantau pengaruhnya.

TABEL IV  
PERFORMA FIREFLY ALGORITHM (ITERASI)

Po	Itr	Best	Mean	Std	Std / Mean (%)	Avg Time (ms)
20	3	2633.73	2516.19	116.4	4.63	26.60
20	5	2675.34	2549.90	94.74	3.72	43.60
20	10	2639.93	2594.70	45.81	1.77	90.10
20	20	2666.49	2576.26	91.15	3.54	208.70
20	40	2669.85	2588.24	57.72	2.23	417.50
20	60	2658.11	2612.18	33.75	1.29	498.20
20	80	2663.68	2628.17	32.48	1.24	837.10
20	100	2676.62	2637.25	31.57	1.20	1030.40
20	120	2690.76	2659.34	22.67	0.85	1262.90
20	200	2699.00	2689.96	6.10	0.23	2018.40

TABEL V  
PERFORMA SYMBIOTIC ORGANISMS SEARCH (ITERASI)

Po	Itr	Best	Mean	Std	Std / Mean (%)	Avg Time (ms)
20	3	2696.21	2673.92	12.98	0.49	12.60
20	5	2699.64	2687.35	8.93	0.33	26.40
20	10	2701.80	2697.74	8.93	0.33	49.80
20	20	2702.60	2699.97	3.44	0.13	100.10
20	40	2702.63	2701.16	2.51	0.09	203.10
20	60	2702.63	2701.19	3.43	0.13	291.30
20	80	2702.63	2699.94	6.89	0.26	405.80

TABEL VI  
PERFORMA IMPROVED BAT ALGORITHM (ITERASI)

Po	Itr	Best	Mean	Std	Std / Mean (%)	Avg Time (ms)
20	3	2696.28	2678.11	22.00	0.82	8.70
20	5	2700.31	2691.79	11.04	0.41	15.00
20	10	2702.20	2699.02	2.59	0.10	29.40
20	20	2702.45	2701.45	1.15	0.04	57.20
20	40	2702.63	2702.02	1.09	0.04	124.80
20	60	2702.60	2702.09	0.73	0.03	175.90
20	80	2702.63	2702.56	0.10	0.00	234.00

Tabel 4, 5, dan 6 mencantumkan solusi terbaik (Best) beserta kestabilan untuk beberapa macam nilai iterasi. Nilai iterasi sebesar 60 ditentukan cukup stabil pada kasus ini dan akan digunakan untuk tahapan uji coba berikutnya.

**Uji Coba 3 (berbagai gambar dan dimensi).** Uji coba bagian ini menggunakan 20 gambar grayscale berbeda. Semua gambar yang digunakan memiliki histogram gray-level yang bersifat unik. Sebagian besar gambar dianggap sulit untuk disegmentasi karena memiliki histogram yang bersifat multimodal atau memiliki banyak local optima. Untuk uji coba ini, setiap gambar diujikan dengan bermacam dimensi (m). Dimensi yang digunakan bernilai antara 2 hingga 5 dan nilai dimensi tersebut menandakan banyaknya threshold yang digunakan untuk memproses

gambar grayscale. Parameter populasi yang digunakan bernilai 20 dan parameter iterasi yang digunakan bernilai 60. Uji coba dilakukan sebanyak 10 kali untuk parameter dan gambar yang sama. Tabel 7 menunjukkan hasil uji coba menggunakan 7 dari keseluruhan 20 gambar uji coba (warna hijau menandakan hasil terbaik, kuning menandakan hasil terbaik kedua, dan merah menandakan hasil terburuk).

TABEL VII  
PERBANDINGAN PERFORMA ALGORITMA

Pic	m	Std/Mean (%)			Average Time (ms)		
		FA	SOS	IBA	FA	SOS	IBA
Aerial	2	2.80	0.00	0.03	472.7	215.3	130.1
	3	2.03	0.00	0.30	478.8	209.0	149.9
	4	2.10	0.03	0.36	481.7	218.3	177.0
	5	2.00	0.14	0.13	496.2	224.4	204.2
	Barbara	2	1.52	0.00	0.03	468.4	202.8
	3	2.71	0.00	0.04	485.5	205.9	161.3
	4	1.75	0.03	0.02	493.4	213.5	172.2
	5	1.92	0.17	0.02	498.2	219.7	193.8
Boats	2	1.00	0.00	0.03	474.5	198.1	139.4
	3	1.82	0.00	0.06	480.5	206.3	157.8
	4	1.32	0.04	0.06	499.5	213.6	173.2
	5	1.08	0.07	0.02	498.3	221.2	189.5
	Goldhill	2	5.16	0.00	0.09	466.8	202.9
3		3.94	0.00	0.06	480.8	209.0	164.3
4		2.10	0.00	0.08	491.9	229.1	177.3
5		2.15	0.15	0.10	502.9	224.6	190.1
Jet		2	0.96	0.00	0.28	460.5	201.3
	3	0.78	0.00	0.25	488.7	209.0	162.0
	4	1.48	0.01	0.30	496.7	215.1	185.6
	5	1.91	0.09	0.06	502.8	225.9	195.6
	Lake	2	0.44	0.00	0.07	478.9	209.0
3		1.43	0.00	0.01	487.2	212.8	169.9
4		1.78	0.05	0.01	502.8	221.5	185.7
5		1.41	0.01	0.03	493.4	224.4	195.9
Mandrill		2	2.23	0.00	0.01	464.5	194.9
	3	3.66	0.00	0.23	501.2	209.0	155.4
	4	1.49	0.08	0.10	496.6	210.5	157.4
	5	1.99	0.06	0.07	493.4	218.1	172.4

Berikut adalah salah satu contoh hasil threshold dan gambar hasil (gambar 4) dari tabel 7 untuk gambar Barbara dengan algoritma SOS:

1.  $m = 2; J(t) = 2427.38; t = [81.69, 144.69]$
2.  $m = 3; J(t) = 2594.87; t = [76.44, 127.8, 175.13]$
3.  $m = 4; J(t) = 2666.67; t = [66.87, 106.76, 141.59, 181.43]$
4.  $m = 5; J(t) = 2702.63; t = [56.29, 87.28, 117.14, 146.5, 182.8]$



Gambar. 4. Hasil Threshold Gambar Barbara.  
(A) – Versi Greylevel  
(B) – Versi Berwarna

### V. KESIMPULAN

Kesimpulan yang diperoleh melalui penelitian ini adalah sebagai berikut:

1. Pertambahan populasi menyebabkan bertambahnya waktu perhitungan pada FA secara lebih signifikan daripada pertambahan waktu pada SOS dan IBA.
2. Pertambahan nilai dimensi cenderung menyebabkan bertambahnya waktu perhitungan. Pertambahan waktu pada FA dan SOS dapat dikatakan kurang signifikan apabila dibandingkan dengan pertambahan waktu pada IBA, karena IBA memiliki modifikasi operator crossover dan mutation pada setiap dimensinya.
3. Nilai terbaik yang didapatkan dengan SOS dan IBA lebih optimal daripada nilai terbaik yang didapatkan dengan FA.
4. Kestabilan SOS dan IBA lebih baik daripada FA secara signifikan, walaupun SOS memiliki keunggulan terhadap IBA pada dimensi yang kecil.
5. Waktu rata-rata perhitungan IBA paling cepat dan FA paling lama. Bila diurutkan waktunya,  $IBA < SOS < FA$ .

### DAFTAR PUSTAKA

[1] Nobuyuki Otsu. 1979. *A Threshold Selection Method from Graylevel Histogram*. IEEE Transaction on Systems, Man, and Cybernetics, vol. SMC-27, no. 1.

- [2] Jun Zhang, Jinglu Hu. 2009. *Renal Biopsy Image Segmentation Based on 2-D Otsu Method with Histogram Analysis*. Medical Imaging Technology, vol. 27, no. 3.
- [3] Ping-Sung Liao, Tse-Sheng Chen, Pau-Choo Chung. 2001. *A Fast Algorithm for Multilevel Thresholding*. Journal of Information Science and Engineering 17.
- [4] X.-S. Yang. 2009. *Firefly Algorithms for Multimodal Optimization in Stochastic Algorithms: Foundations and Applications*, vol. 5792 of Lecture Notes in Computer Science, pp. 167-178, Springer, Berlin, Germany.
- [5] X.-S. Yang. 2010. *A New Metaheuristic Bat-inspired Algorithm, Studies in Computational Intelligence*, vol. 284, pp. 65-74, Springer, Berlin, Germany.
- [6] Min-Yuan Cheng, Doddy Prayogo. 2014. *Symbiotic Organisms Search: A new metaheuristic optimization algorithm*. Science Direct.
- [7] N. Sri Madhava Raja, V. Rajinikanth, K. Lantha. 2014. *Otsu Based Optimal Multilevel Image THresholding using Firefly Algorithm*. Hindawi Publishing Corporation.
- [8] Kai Chen, Yifan Zhou, Zhisheng Zhang, Min Dai, Yuan Chao, Jinfei Shi. 2016. *Multilevel Image Segmentation Based on an Improved Firefly Algorithm*. Hindawi Publishing Corporation.
- [9] Adis Alihodzic, Milan Tuba. 2014. *Improved Bat Algorithm Applied to Multilevel Image Thresholding*. Hindawi Publishing Corporation.

**C. Pickerling** lahir di Surabaya, Indonesia, pada tahun 1986. Menyelesaikan studi S1 di program studi Teknik Informatika STTS pada tahun 2008. Menyelesaikan studi masternya pada program studi Teknologi Informasi STTS pada tahun 2016. Minat penelitian adalah pada bidang ilmu software engineering dan evolutionary algorithm.

**Hendrawan Armanto** lahir di Surabaya, Indonesia, pada tahun 1986. Menyelesaikan studi S1 di program studi Teknik Informatika STTS pada tahun 2008. Menyelesaikan studi masternya pada program studi Teknologi Informasi STTS. Minat penelitian adalah bidang Artificial Intelligent, Evolutionary Algorithm, dan Game Development.

# Augmented Reality Marker Based Tracking Visualisasi Drawing 2D ke dalam Bentuk 3D dengan Metode FAST Corner Detection

Nanang Wahyudi, *Teknologi Informasi ISTTS*, Reddy Alexandro Harianto., S.Kom., M.Kom., *Informatika ISTTS*, Endang Setyati, Dr., Ir., Hj., M.T., *Teknologi Informasi ISTTS*

**Abstrak**— *Augmented Reality (AR)* memungkinkan pengguna dapat melihat objek virtual pada lingkungan nyata. Berbeda dengan *Virtual Reality (VR)* yang membawa pengguna masuk pada lingkungan virtual sepenuhnya tanpa melihat dunia nyata. Teknologi AR menggunakan marker sebagai target untuk memunculkan objek virtual. Penelitian ini menggunakan *Drawing 2D* sebagai *Marker Based Tracking* dalam mendeteksi target untuk memunculkan objek 3D virtual. Gambar 2D atau *Drawing 2D* merupakan alat untuk menyampaikan maksud dan informasi dari *drafter* kepada teknisi. Siswa lulusan Sekolah Menengah Kejuruan (SMK) harus mampu memahami *Drawing 2D* dan memvisualisasikan kedalam bentuk 3D. Kemampuan spasial dalam memvisualisasi ini harus dimiliki karena menyangkut masa depan setiap siswa. Penyampaian materi proyeksi *Drawing 2D* memerlukan teknis khusus agar mampu di pahami oleh siswa. Aplikasi AR ini menggunakan metode *Features from Accelerated Segment Test Corner Detection (FCD)* dalam proses *tracking* yang memiliki tingkat komputasi deteksi marker yang tinggi untuk memunculkan objek virtual 3D. Uji coba penelitian menggunakan 50 marker. Kriteria uji coba *tracking* dengan posisi marker antara lain : tegak lurus, miring 30°, 45°, 60° dan 75° serta dengan jarak deteksi 20 cm, 30 cm, 40 cm, 50 cm, dan 60 cm terhadap kamera. Dari hasil uji coba untuk mendeteksi 50 marker dapat disimpulkan bahwa marker dapat terdeteksi dengan posisi tegak lurus terhadap kamera maksimum jarak 50 cm, posisi marker miring 30° terdeteksi maksimum pada jarak 40 cm, dan posisi marker miring 45° terdeteksi maksimum pada jarak 30 cm. Untuk kemiringan marker 60° dan 75° rata-rata perangkat sudah tidak dapat mendeteksi marker. Proses deteksi marker dapat dipengaruhi oleh spesifikasi perangkat yang digunakan dalam ujicoba, pencahayaan, ukuran marker serta ketebalan garis yang digunakan.

**Kata Kunci**— *Augmented Reality, FAST Corner Detection, Gambar Teknik, Kemampuan Spasial*

## I. PENDAHULUAN

**A**ugmented Reality merupakan teknologi yang dapat menggabungkan objek virtual dengan lingkungan nyata secara real time. Azuma telah mengembangkan penelitian terkait teknologi augmented reality yang memungkinkan

pengguna untuk melihat dunia nyata bersama dengan objek virtual [1]. Augmented Reality (AR) berbeda dengan Virtual Reality (VR) ataupun Mixed Reality (MR). AR dan MR memungkinkan pengguna dapat melihat dan berinteraksi dengan objek virtual 3D pada lingkungan nyata. Sedangkan VR membawa pengguna pada dunia virtual tanpa melihat atau berinteraksi dengan dunia nyata.

Teknologi AR memerlukan target yang digunakan sebagai kunci dalam memunculkan objek 3D virtual yang disebut dengan marker [2]. Marker dapat menggunakan barcode bergaris, QR Code atau gambar bebas [3]. Marker terbagi menjadi 2 jenis yaitu : marker based tracking dan markerless.

Dengan perkembangan teknologi yang semakin cepat kini teknologi AR dapat digunakan secara mobile dan di manfaatkan pada berbagai bidang seperti bidang pendidikan, bidang kesehatan dan bidang militer [4]. Pada bidang pendidikan khususnya untuk pembelajaran proyeksi menggambar 2D memerlukan kemampuan spasial dalam memvisualisasi ke dalam bentuk objek 3D atau sebaliknya. Dalam memberikan pemahaman terhadap materi menggambar 2D dan 3D memerlukan teknik khusus serta alat dalam menyampaikan materi yang nantinya mampu di fahami oleh siswa. Pembelajaran dengan waktu yang relatif singkat juga menjadi salah satu faktor kendala penyampaian pemahaman siswa dalam memahami gambar 2D.

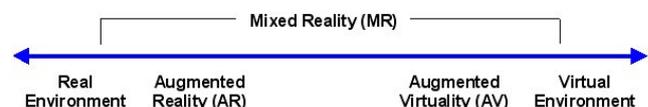
Gambar 2D atau disebut juga dengan *Drawing 2D* merupakan alat bantu dalam menyampaikan informasi untuk membuat benda kerja atau pekerjaan teknik [5]. Seorang siswa lulusan Sekolah Menengah Kejuruan (SMK) terutama pada jurusan teknik pemesian dan teknik otomotif harus mampu memahami *drawing 2D* dan memvisualisasikan kedalam bentuk 3D. Kemampuan spasial dalam memvisualisasi ini merupakan skill atau keahlian yang harus dimiliki oleh setiap siswa karena menyangkut masa depan siswa di kemudian hari dalam bekerja.

Penelitian pengembangan teknologi AR dalam menggambar 2D dan penelitian terkait pengembangan

Nanang Wahyudi, Departemen Teknologi Informasi, Institut Sains dan Teknologi Terpadu Surabaya, Surabaya, Jawa Timur, Indonesia (e-mail: [nanang\\_wahyudie@yahoo.co.id](mailto:nanang_wahyudie@yahoo.co.id))

Reddy Alexandro Harianto, S.Kom., M.Kom., Departemen Informatika, Institut Sains dan Teknologi Terpadu Surabaya, Surabaya, Jawa Timur, Indonesia (e-mail: [reddy@stts.edu](mailto:reddy@stts.edu))

Endang Setyati, Dr., Ir., Hj., M.T., Departemen Teknologi Informasi, Institut Sains dan Teknologi Terpadu Surabaya, Surabaya, Jawa Timur, Indonesia (e-mail: [endang@stts.edu](mailto:endang@stts.edu))



Gambar 1. Satuan Reality-Virtuality dengan Milgram

kemampuan spasial yang telah dilakukan oleh beberapa peneliti. Jorge Martin G. membuktikan bahwa kemampuan spasial dalam pemahaman gambar teknik dapat ditingkatkan dengan bantuan teknologi AR [6]. Heen Chen mengembangkan teknologi AR dalam visualisasi 3 jenis proyeksi gambar teknik 2D ke dalam objek virtual 3D [7]. Untuk proses deteksi marker (*tracking*) metode FAST (*Features from Accelerated Segment Test*) *Corner Detection* memiliki tingkat komputasi yang tinggi jika dibandingkan dengan metode Harris *Corner Detection* dan *Edge Based Corner Detection* [8].

Berdasarkan permasalahan tersebut maka penelitian ini memberikan alternatif teknologi Augmented Reality untuk memvisualisasikan drawing 2D kedalam bentuk objek 3D virtual. Proses deteksi marker sebagai target dalam memunculkan objek 3D menggunakan metode FAST (*Features from Accelerated Segment Test*) *Corner Detection* (FCD).

## II. STUDI LITERATUR

### A. Augmented Reality

*Augmented Reality* (AR) adalah penggabungan antara objek virtual dengan lingkungan nyata. AR menurut Paul Milgram merupakan penggabungan dunia nyata dan virtual, bersifat interaktif secara real time dan berbentuk 3D [9].

Augmented Reality memiliki hubungan yang erat dengan Virtual Reality dan Mixed Reality dimana sama-sama menambahkan objek virtual secara nyata. Namun untuk memahami hubungan tersebut dapat dilihat pada Gambar 1.

*Augmented Reality* (AR) didominasi dengan penambahan objek virtual di dunia nyata dengan informasi yang dihasilkan oleh komputer. AR dapat memberikan informasi tambahan berupa visual tambahan yang ditambahkan di dunia nyata dengan 6 derajat kebebasan (*6 Degree Of Freedom* (6DOF)).

*Virtual Reality* (VR) mengacu pada sistem yang sebagian besar berupa objek dan lingkungannya buatan. VR membawa seseorang ke dalam dunia komputer atau virtual tanpa bisa melihat dunia nyata dengan 3 derajat kebebasan (*3 Degree Of Freedom* (3DOF)).

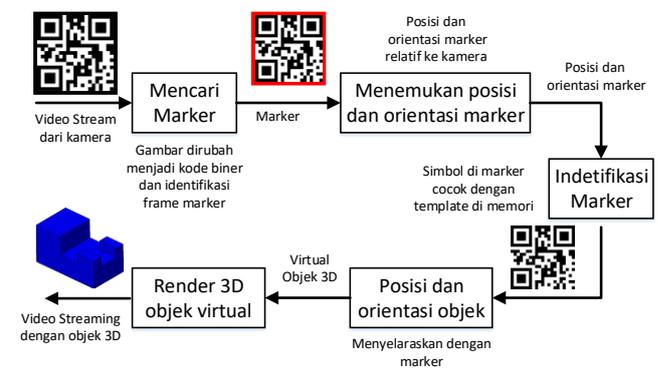


Gambar 2. Perbandingan Augmented Reality, Virtual Reality, dan Mixed Reality

*Mixed Reality* (MR) termasuk dalam dunia simulasi yang mengacu pada kombinasi atau penggabungan lingkungan

virtual bersama dengan lingkungan nyata, dimana keduanya dapat hidup berdampingan. *Mixed Reality* juga disebut dengan "*Hybrid Reality*". Perbedaan AR, VR dan MR dapat dilihat pada Gambar 2.

Saat ini AR dan VR dapat dimanfaatkan untuk pelatihan, pendidikan dan hiburan. Dalam militer AR dapat digunakan sebagai alat untuk latihan berperang atau mekanik pesawat terbang. Di dunia kesehatan dan pendidikan AR dapat digunakan sebagai alat bantu dalam proses pembelajaran virtual. Hasil review dari beberapa riset terkait AR dalam dunia pendidikan menyimpulkan bahwa dengan perantara Augmented Reality mampu memberikan manfaat dan pengalaman belajar yang efektif dan dampak yang positif bagi siswa. Hasilnya telah ditunjukkan dalam bentuk motivasi, kinerja pembelajaran, dan peluang baru untuk menciptakan pengalaman belajar yang interaktif.



Gambar 3. Alur proses Augmented Reality

Pada Gambar 3 merupakan alur proses pada AR. Proses dimulai dari pengambilan gambar marker dengan kamera. Kemudian masuk ke dalam proses pencarian dan orientasi marker, indentifikasi marker, perhitungan posisi dan orientasi marker terhadap objek dan render objek. Proses render menggabungkan gambar asli dan komponen virtual menggunakan perhitungan pose dan orientasi marker. Hasil keluaran pelacakan marker ditampilkan pada layar komputer atau smartphone.

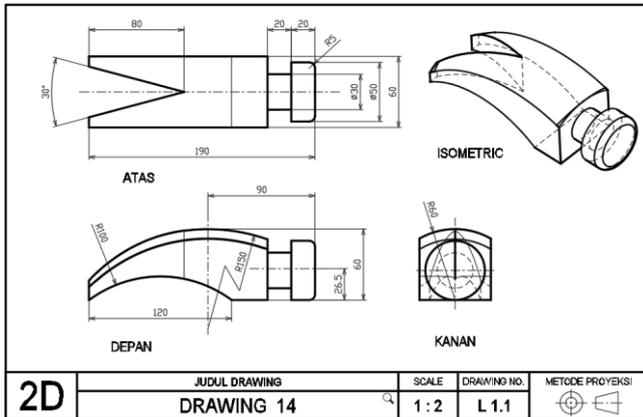
### B. Marker

Marker merupakan sebuah penanda khusus yang memiliki pola tertentu yang akan dikenali oleh kamera, sehingga saat kamera mendeteksi marker maka objek 3D dapat ditampilkan. Marker *Augmented Reality* dapat dikategorikan menjadi dua yaitu deteksi menggunakan (*Marker Based Tracking*) dan tanpa marker (*markerless*). *Marker based tracking* komputer akan mengenali posisi dan orientasi marker dan menciptakan dunia virtual 3D [2][10]. Contoh marker yang digunakan pada penelitian ini seperti pada Gambar 4.

Marker atau bisa disebut dengan Image target memiliki kriteria untuk dapat dilacak oleh sistem AR yang meliputi :

- Fitur gambarnya kaya (polanya rumit), misalnya gambar pemandangan, gambar sekumpulan orang, kolase dan lain-lain.
- Kontrasnya bagus, gelap dan terangnya jelas.
- Tidak ada pengulangan pola, misal lapangan rumput, kotak-kotak.

- d. Grafik warnanya 8 atau 24 bit format PNG atau JPG, ukurannya kurang dari 2MB, JPGs haruslah RGB atau grayscale (bukan CMYK).

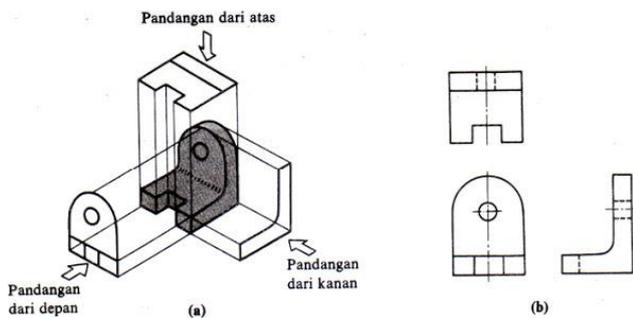


Gambar 4. Contoh marker drawing 2D

C. Gambar Kerja / Drawing

Gambar kerja atau *drawing* merupakan sebuah alat dalam memberikan informasi dari *drafter* atau seorang desainer kepada teknisi atau pembuat benda kerja [5].

Drawing 2D pada umumnya memiliki 2 jenis proyeksi gambar yaitu proyeksi sistem Amerika (*Third Angle Projection*) dan proyeksi sistem Eropa (*First Angle Projection*). Pada penelitian ini menggunakan drawing 2D dengan sistem amerika seperti pada Gambar 5.



Gambar 5. Bentuk Proyeksi Pandangan Amerika

D. FAST (Features from Accelerated Segment Test) Corner Detection (FCD)

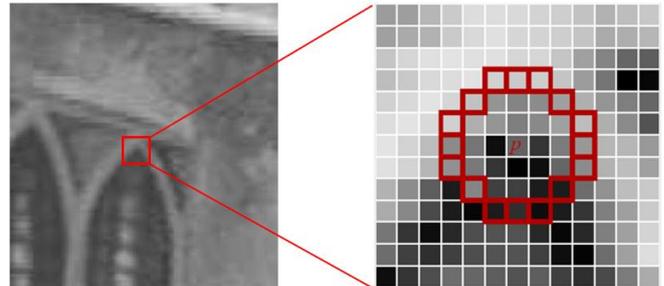
Algoritma FAST (*Features from Accelerated Segment Test*) Corner Detection (FCD) merupakan penentuan *corner point* atau deteksi sudut untuk mendeteksi sudut-sudut dari suatu objek yang dikembangkan oleh Edward Rosten, Reid Porter, dan Tom Drummond [11]. Algoritma FCD ini dibuat dengan tujuan mempercepat waktu komputasi secara *real-time* dengan konsekuensi menurunkan tingkat akurasi pendeteksian sudut [12].

Pada algoritma FCD proses penentuan *corner point* adalah dengan cara merubah gambar menjadi warna hitam dan putih. Algoritma ini menentukan *corner point* dengan sebuah titik yakni  $p$  dari input gambar yang memeriksa keliling 16 pixel dari titik  $p$ .

Metode FAST *Corner Detection* ini menggunakan jenis *high-speed test* FAST *Corner Detection* yang digunakan untuk mengecualikan jumlah besar dari *non-corners* (bukan sudut), dimana penerapan algoritma dengan mengambil

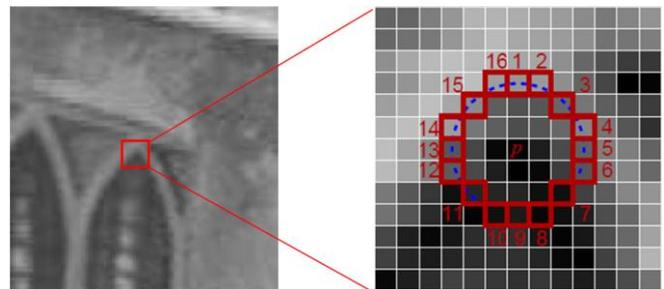
4 titik dari 16 pixel kemudian membandingkan intensitas keempat pixel dengan pixel titik  $p$  (titik pusat). Jika nilai intensitas di titik  $p$  bernilai lebih besar atau lebih kecil daripada intensitas sedikitnya tiga titik disekitarnya ditambah dengan intensitas batas ambang (*threshold*), maka didapatkan titik  $p$  merupakan titik sudut (*corner*). Setelah itu titik  $p$  akan digeser ke posisi  $x_{p+1}, y_p$  dan melakukan perbandingan intensitas sampai semua titik pada citra. Tahapan proses dari algoritma FCD sebagai berikut :

1. Tentukan sebuah titik  $p$  pada citra dengan posisi awal  $(x,y)$  dan nilai *threshold* seperti pada Gambar 6



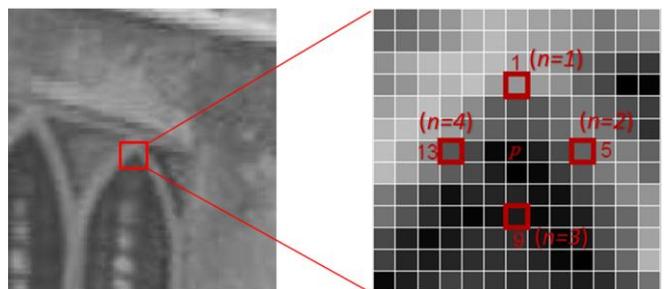
Gambar 6. Menentukan titik awal (titik P)

2. Tentukan 16 titik pixel dengan radius 3 pixel dari titik  $p$  seperti pada Gambar 7.



Gambar 7. Menentukan titik awal (titik P)

3. Tentukan lokasi 4 titik dari 16 pixel. Titik pertama ( $n=1$ ) koordinat  $(x_p, y_{p+3})$ , titik kedua ( $n=2$ ) koordinat  $(x_{p+3}, y_p)$ , titik ketiga ( $n=3$ ) koordinat  $(x_p, y_{p-3})$ , titik keempat ( $n=2$ ) koordinat  $(x_{p-3}, y_p)$  seperti pada Gambar 8.



Gambar 8. Titik p pada koordinat  $n=1, n=2, n=3, n=4$

4. Bandingkan intensitas titik pusat  $p$  dengan keempat titik disekitar. Titik pusat  $p$  merupakan titik sudut atau *corner* seperti pada Gambar 9 apabila terdapat paling sedikit 3 titik yang memenuhi 3 kategori yang ditetapkan dalam algoritma FCD yaitu :

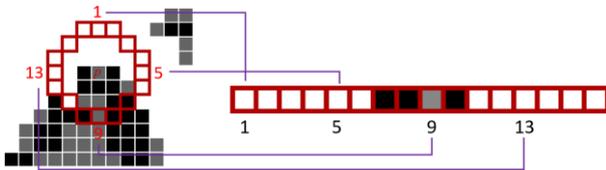
$$S_{p \rightarrow x} = \begin{cases} d, & I_{p \rightarrow x} \leq I_p - t \quad (\text{Gelap}) \\ s, & I_p - t < I_{p \rightarrow x} < I_p + t \quad (\text{Normal}) \\ b, & I_p + t \leq I_{p \rightarrow x} \quad (\text{Cerah}) \end{cases}$$

Keterangan :

$S_{p \rightarrow x}$  : Intensitas titik pusat (titik p)

$I_{p \rightarrow x}$  : Intensitas pixel x (titik intensitas tetangga ke-n)

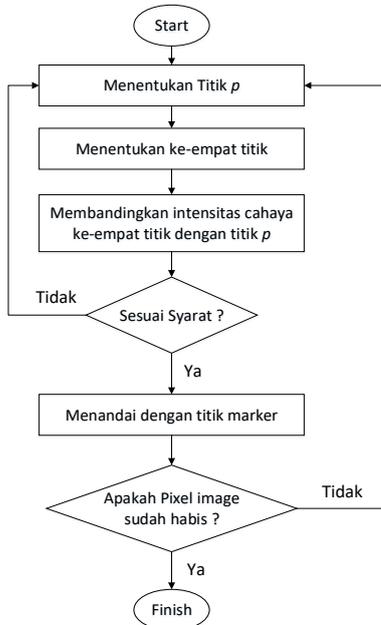
t : threshold



Gambar 9. Perbandingan intensitas pada 16 pixel dari titik p

5. Ulangi proses sampai seluruh titik pada citra sudah dibandingkan intensitasnya.

Penerapan algoritma FCD dapat digambarkan pada diagram alir Gambar 10.



Gambar 10. Diagram alir algoritma FAST Corner Detection (FCD)

### E. Penelitian Terdahulu

#### 1. Penelitian AR menggunakan Marker based Tracking

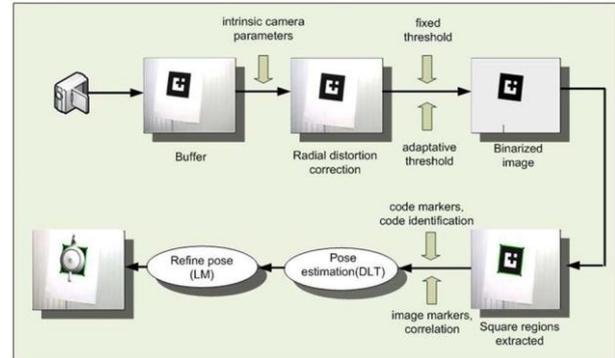
##### 1.1. AR-Dehaes

Jorge Martin G. membuat aplikasi pembelajaran gambar teknik sederhana bernama buku AR-Dehaes [13]. Metode AR menggunakan marker based tracking seperti pada Gambar 11. Metode sistem dapat dilihat pada blok diagram Gambar 12 dan deteksi marker (tracking) menggunakan 4 tahapan sebagai berikut :

1. Camera calibration
2. Marker detection
3. Calculation of marker position and orientation (pose estimation)
4. Augmentation of virtual object



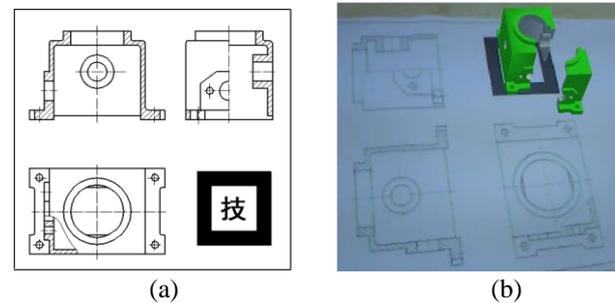
Gambar 11. Marker AR-Dehaes



Gambar 12. Blok Diagram AR-Dehaes

#### 1.2. AR in Engineering Graphics Education

Heen Chen mengembangkan teknologi AR dalam pembelajaran gambar teknik proyeksi 2D untuk divisualisasikan ke dalam objek virtual 3D [7]. Metode yang digunakan berupa proyeksi gambar teknik 2D yang diberikan marker sebagai acuan memunculkan objek 3D. Contoh gambar 2D seperti pada Gambar 13.



Gambar 13. (a). Gambar 2D beserta marker, (b) Visualisasi objek 3D virtual

### 2. Penelitian perbandingan tracking algoritma FCD

#### 2.1. Perbandingan Tracking algoritma Harris Corner Detection, Edge Based Corner Detection dan Fast Corner Detection dalam Deteksi Senyum Pada wajah Manusia

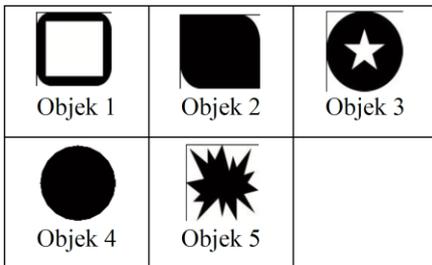
Eduard Royce Siswanto melakukan perbandingan metode algoritma dalam deteksi senyum pada manusia [8]. Hasil dari penelitian menyatakan bahwa metode Fast Corner Detection memiliki tingkat komputasi tertinggi atau waktu yang pendek dalam mendeteksi citra dibandingkan dengan metode Harris dan Edge Based Detection. Hasil penelitian dapat dilihat pada Tabel 1.

Tabel 1  
Akurasi dan Waktu Komputasi Sistem Pendeteksi Senyum

Metode	Akurasi (%)	Waktu Komputasi (mili detik)
Harris Corner Detection	29	70.6
Edge Based Corner Detection	42	50.9
FAST Corner Detection	59.5	13.1

2.2. Perbandingan Kinerja Algoritma Untuk Mendeteksi Objek Dua Dimensi Pada Augmented Reality

Rujianto Eko Saputro melakukan perbandingan metode FAST dan ORB dalam mendeteksi objek 2D [14]. Contoh objek 2D yang diuji pada Gambar 14. Hasil dari penelitian dapat dilihat pada Tabel 2. Hasil penelitian didapatkan bahwa metode FAST memiliki performa paling baik dibandingkan dengan algoritma ORB, pengujian algoritma FAST mampu menampilkan interest point antara 0 sampai dengan 2 milli seconds, sedangkan algoritma ORB mencapai 0 sampai dengan 15 milli seconds.



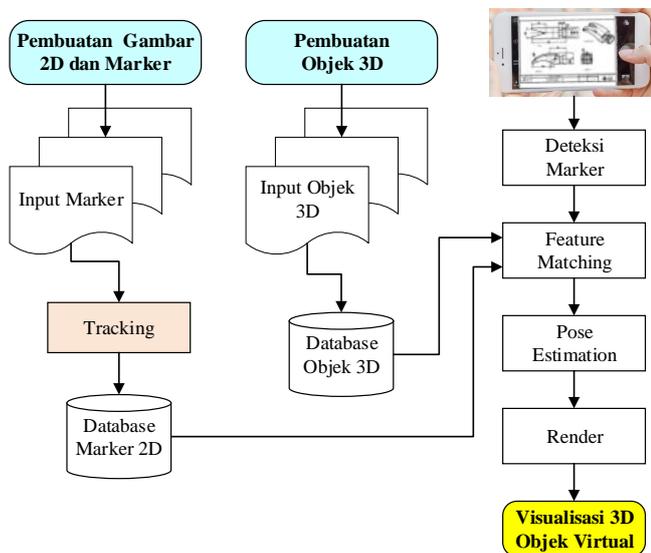
Gambar 14. Objek 2 dimensi yang diuji

Tabel 2  
Kecepatan deteksi metode FAST dan ORB

Objek	Kecepatan Deteksi ( ms )	
	FAST	ORB
Objek 1	0 – 1	0 - 13
Objek 2	0	0 – 11
Objek 3	0 – 1	1 – 15
Objek 4	0 – 1	1 – 11
Objek 5	0 – 1	1 – 14

III. METODOLOGI PENELITIAN

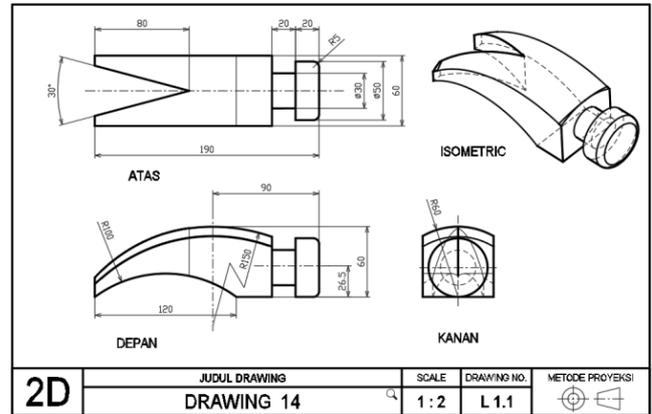
Proses pembuatan aplikasi dimulai dari pembuatan gambar 2D yang sekaligus digunakan sebagai marker. Kemudian membuat objek virtual 3D dan pembuatan aplikasi. Rancangan blok diagram proses pada Gambar 15.



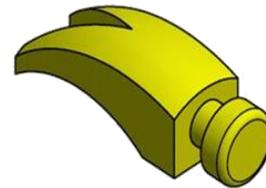
Gambar 15. Blok Diagram Rancangan Aplikasi AR

A. Drawing 2D, Marker dan Objek 3D

Input dari aplikasi menggunakan Drawing 2D yang sekaligus sebagai marker dibuat menggunakan autocad. Sedangkan untuk objek 3D virtual dibuat menggunakan aplikasi blender. Contoh drawing 2D dan objek 3D dapat dilihat pada gambar Gambar 16 dan Gambar 17.



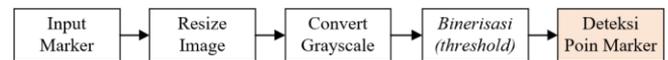
Gambar 16. Contoh Drawing 2D dan Marker



Gambar 17. Gambar 3D objek virtual

B. Proses Tracking

Aplikasi harus mengenali target atau marker untuk memunculkan objek 3D. Proses mengenali marker disebut dengan proses *tracking*. Tahapan dari proses tracking seperti pada Gambar 18.



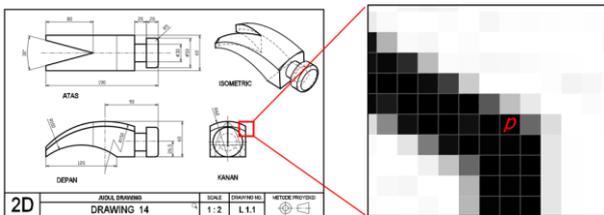
Gambar 18. Tahapan proses tracking marker

- a. Input Marker  
Mempersiapkan marker sebagai target untuk memunculkan objek 3D
- b. Resize Image  
Proses ini bertujuan untuk menyamakan ukuran gambar dan memperkecil jumlah pixel sehingga proses perhitungan untuk mendeteksi marker semakin cepat.
- c. Convert Grayscale  
Proses ini merubah gambar menjadi grayscale atau hitam putih dengan nilai intensitas 0 sampai 255. Tujuan dari proses ini adalah untuk memudahkan sistem agar dapat mengenali pola pada gambar yang diterima.
- d. Binerisasi (*threshold*)  
Proses binerisasi dilakukan dengan cara mengkonversi citra grayscale ke dalam biner dengan nilai *threshold* default adalah 128, jika pixel yang nilai intensitasnya dibawah 128 maka akan diubah menjadi warna hitam (nilai intensitas 0) dan pixel yang nilainya 128 maka akan diubah menjadi warna putih

e. Deteksi Poin Marker

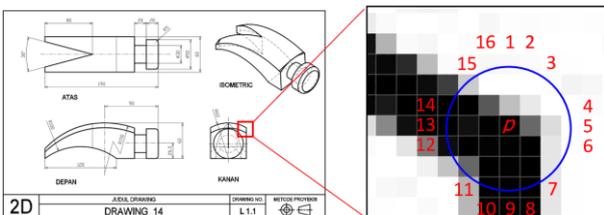
Proses deteksi poin marker menggunakan metode *Features from Accelerated Segment Test (FAST) Corner Detection* atau lebih dikenal dengan metode FCD. Metode FCD digunakan untuk menentukan titik objek dan pencocokan pola pada marker. Output dari metode FCD adalah penentuan jumlah titik atau poin pada marker. Proses perhitungan algoritma ini dimulai dari penentuan titik  $p$  pada koordinat yang didefinisikan oleh koordinat  $(x_p, y_p)$ . Tahap proses algoritma FCD sebagai berikut :

1. Tentukan sebuah titik  $p$  pada citra dengan posisi awal  $(x,y)$  dan nilai threshold seperti pada Gambar 19.



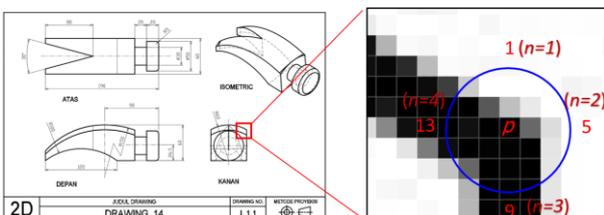
Gambar 19. Menentukan titik awal (titik P)

2. Tentukan 16 titik pixel dengan radius 3 pixel dari titik  $p$  seperti pada Gambar 20.



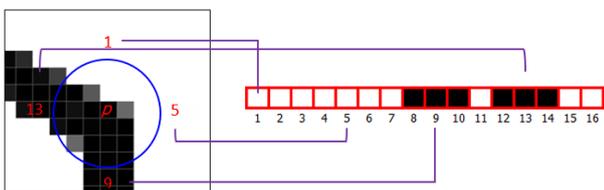
Gambar 20. Menentukan titik awal (titik P)

3. Tentukan empat titik dari 16 pixel. Titik pertama ( $n=1$ ) koordinat  $(x_p, y_{p+3})$ , titik kedua ( $n=2$ ) koordinat  $(x_{p+3}, y_p)$ , titik ketiga ( $n=3$ ) koordinat  $(x_p, y_{p-3})$ , titik keempat ( $n=4$ ) koordinat  $(x_{p-3}, y_p)$  seperti pada Gambar 21.



Gambar 21. Titik  $p$  pada koordinat  $n=1, n=2, n=3, n=4$

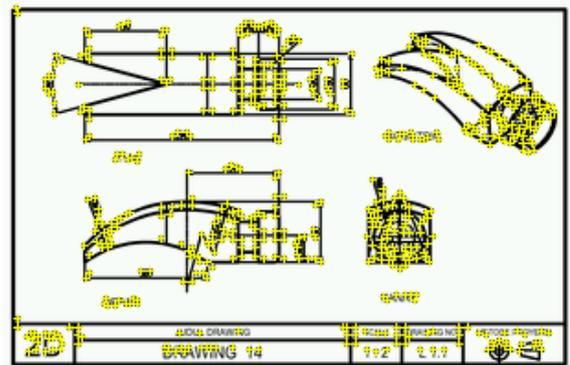
4. Bandingkan intensitas titik pusat  $p$  dengan keempat titik disekitar. Jika terdapat paling sedikit 3 titik yang memenuhi syarat, maka titik pusat  $p$  merupakan titik sudut seperti pada Gambar 22.



Gambar 22. Perbandingan intensitas pada 16 pixel dari titik  $p$

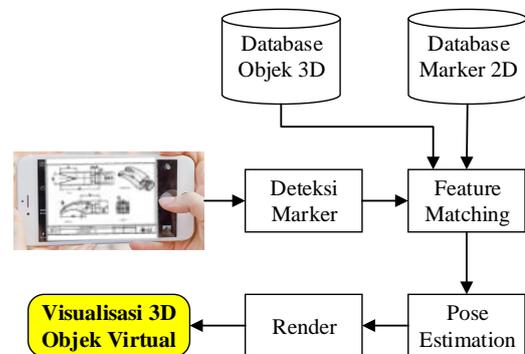
5. Ulangi proses sampai seluruh titik pada citra sudah dibandingkan intensitasnya.

Hasil dari proses deteksi poin marker dapat dilihat pada Gambar 23.



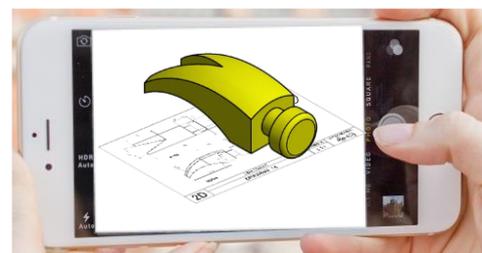
Gambar 23. Hasil proses deteksi poin metode FCD

C. Visualisasi Objek 3D Virtual



Gambar 24. Tahapan Visualisasi Objek 3D

Visualisasi objek 3D pada aplikasi dilakukan dengan terlebih dahulu mendeteksi drawing 2D atau marker. Kemudian melakukan proses *matching* antara marker yang telah di proses *tracking* dengan drawing 2D pada dunia nyata. Jika hasilnya sama maka dilakukan input objek 3D virtual dan proses render. Tahapan visualisasi seperti pada Gambar 24. Ilustrasi visualisasi aplikasi seperti pada Gambar 25.



Gambar 25. Ilustrasi render objek pada unity

D. User Interface

Aplikasi *Augmented Reality* pembelajaran drawing 2D yang akan diujikan dibuat menggunakan editor unity seperti pada Gambar 26.

Aplikasi terdapat menu button utama yaitu button pandangan/view objek yang dapat menampilkan jenis pandangan depan, belakang, samping kiri, samping kanan,

atas dan bawah. Button auto rotasi dapat memutar objek 3D secara continue terhadap sumbu X, Y dan Z. Untuk menu Zoom in dan out atau skala objek serta rotasi objek secara manual dapat menggunakan jari.



Gambar 26. User interface aplikasi AR drawing 2D

#### IV. UJI COBA DAN HASIL

##### A. Kriteria Pengujian

Uji coba aplikasi *augmented reality* menggunakan perangkat *smartphone* dengan spesifikasi seperti pada Tabel 3.

Tabel 3  
Spesifikasi perangkat uji coba aplikasi AR

Perangkat	OS	CPU	Resolusi	Kamera
Samsung Grand 2	Android 4.42	Quad-core 1.2 GHz	720 x 1280 pixels	2 MP dan 8 MP
Vivo Y35	Android 5.02	Snapdragon 410 Quad-core 1.2 GHz	720 x 1280 pixels	5 MP dan 13 MP

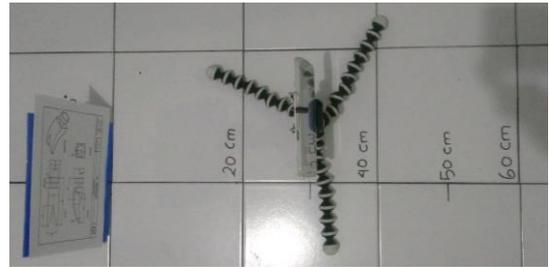
Kriteria untuk uji coba deteksi marker atau drawing 2D sebagai berikut :

1. Maksimum deteksi marker : 1 detik
2. Ukuran marker : 18 x 11 cm.
3. Jumlah marker uji coba : 50 drawing 2D.
4. Tempat uji coba dengan ruangan 3m x 3m.
5. Penerangan menggunakan lampu philip 20 watt.
6. Jarak deteksi marker dengan kamera : 20 cm, 30 cm, 40 cm, 50 cm, 60 cm
7. Posisi marker terhadap kamera :
  - Marker tegak lurus terhadap kamera.
  - Marker dimiringkan 30°, 45°, 60° dan 75° terhadap kamera.

Visualisasi untuk pengujian deteksi marker seperti pada Gambar 27, Gambar 28 dan Gambar 29.



Gambar 27. Visualisasi cara deteksi marker



Gambar 28. Cara uji coba deteksi dengan marker tegak lurus terhadap kamera



Gambar 29. Cara uji coba deteksi dengan marker dimiringkan 30°, 45°, 60°, dan 75° terhadap kamera

##### B. Hasil uji coba

###### 1. Uji Coba Deteksi Marker

Hasil dari uji coba deteksi marker dengan lama waktu deteksi maksimum 1 detik sebagai berikut :

Tabel 4  
Uji coba deteksi dengan marker tegak lurus terhadap kamera

Jarak Deteksi (Cm)	Jumlah Marker Terdeteksi (Pcs)		Jumlah Marker Tidak Terdeteksi (Pcs)	
	Samsung Grand 2	Vivo Y35	Samsung Grand 2	Vivo Y35
20	50	50	-	-
30	50	50	-	-
40	50	50	-	-
50	38	50	12	-
60	-	50	50	-

Untuk tambahan hasil pengujian dengan dengan marker tegak lurus terhadap kamera pada Tabel 4 sebagai berikut :

- Perangkat Vivo Y35  
Pada jarak 70 cm masih dapat mendeteksi marker dengan waktu 2-3 detik. Sedangkan pada jarak 80 cm sudah tidak dapat mendeteksi marker.

Tabel 5  
Uji coba deteksi dengan marker miring 30° terhadap kamera

Jarak Deteksi (Cm)	Jumlah Marker Terdeteksi (Pcs)		Jumlah Marker Tidak Terdeteksi (Pcs)	
	Samsung Grand 2	Vivo Y35	Samsung Grand 2	Vivo Y35
20	50	50	-	-
30	50	50	-	-
40	50	50	-	-
50	-	31	50	19
60	-	29	50	21

Untuk tambahan hasil pengujian dengan kemiringan marker 30° pada Tabel 5 sebagai berikut :

- Perangkat Samsung Grand 2  
Pada jarak 50 cm masih dapat mendeteksi 17 marker dengan waktu 2-3 detik, 33 marker lainnya tidak dapat terdeteksi.  
Untuk jarak 60 cm sudah tidak dapat mendeteksi marker.

- Perangkat Vivo Y35  
Pada jarak 50 cm masih dapat mendeteksi 14 marker dengan waktu 2-3 detik, 5 marker lainnya tidak dapat terdeteksi.  
Pada jarak 60 cm masih dapat mendeteksi 12 marker dengan waktu 3-6 detik, 9 marker lainnya tidak dapat terdeteksi.

Tabel 6

Uji coba deteksi dengan marker miring 45° terhadap kamera

Jarak Deteksi (Cm)	Jumlah Marker Terdeteksi (Pcs)		Jumlah Marker Tidak Terdeteksi (Pcs)	
	Samsung Grand 2	Vivo Y35	Samsung Grand 2	Vivo Y35
20	28	50	22	-
30	50	50	-	-
40	-	-	50	50
50	-	-	50	50
60	-	-	50	50

Untuk tambahan hasil pengujian dengan kemiringan marker 45° pada Tabel 6 sebagai berikut :

- Perangkat Samsung Grand 2  
Pada jarak 20 cm masih dapat mendeteksi 8 marker dengan waktu 3-6 detik, 14 marker lainnya tidak dapat terdeteksi  
Pada jarak 40 cm masih dapat mendeteksi 18 marker dengan waktu 3-6 detik, 32 marker lainnya tidak dapat terdeteksi
- Perangkat Vivo Y35  
Pada jarak 40 cm masih dapat mendeteksi 20 marker dengan waktu 3-6 detik, 30 marker lainnya tidak dapat terdeteksi.

Tabel 7

Uji coba deteksi dengan marker miring 60° terhadap kamera

Jarak Deteksi (Cm)	Jumlah Marker Terdeteksi (Pcs)		Jumlah Marker Tidak Terdeteksi (Pcs)	
	Samsung Grand 2	Vivo Y35	Samsung Grand 2	Vivo Y35
20	-	16	50	34
30	-	-	50	50
40	-	-	50	50
50	-	-	50	50
60	-	-	50	50

Untuk tambahan hasil pengujian dengan kemiringan marker 60° pada Tabel 7 sebagai berikut :

- Perangkat Vivo Y35  
Pada jarak 20 cm dapat mendeteksi 21 marker dengan waktu 2-4 detik, 13 marker lainnya tidak dapat terdeteksi.

Tabel 8

Uji coba deteksi dengan marker miring 75° terhadap kamera

Jarak Deteksi (Cm)	Jumlah Marker Terdeteksi (Pcs)		Jumlah Marker Tidak Terdeteksi (Pcs)	
	Samsung Grand 2	Vivo Y35	Samsung Grand 2	Vivo Y35
20	-	9	50	41
30	-	-	50	50
40	-	-	50	50
50	-	-	50	50
60	-	-	50	50

Tambahan hasil pengujian dengan kemiringan marker 75° pada Tabel 8 untuk perangkat Vivo Y35 berhasil mendeteksi 9 marker dengan kriteria drawing 2D yang kompleks sehingga hasil deteksi poin lebih banyak daripada gambar yang sederhana.

## 2. Uji Coba Tool Pada User Interface

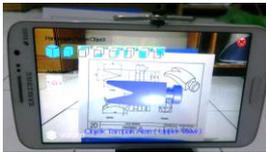
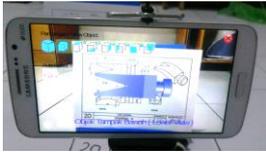
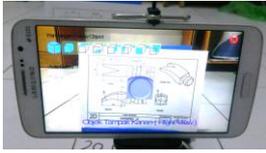
Tool pada user interface terdiri dari button view atau pandangan dan button auto rotasi. Untuk button pandangan/view objek dapat menunjukkan jenis pandangan objek yang terdiri dari pandangan isometrik, pandangan atas, pandangan bawah, pandangan samping kiri, pandangan samping kanan, pandangan depan dan pandangan belakang. Sedangkan untuk button auto rotasi dapat merotasi objek terhadap sumbu X, sumbu Y dan sumbu Z.

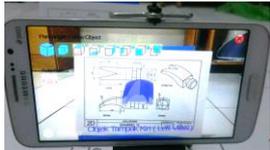
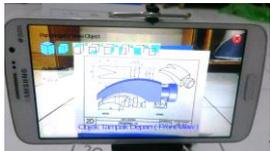
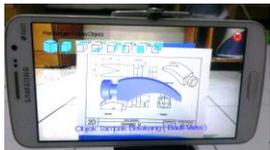
Untuk proses zoom in dan zoom out objek dapat menggunakan interaksi dua jari. Sedangkan untuk proses rotasi objek secara manual dapat menggunakan satu jari.

Hasil ujicoba untuk jenis pandangan objek dengan proyeksi amerika terdapat pada Tabel 9.

Tabel 9

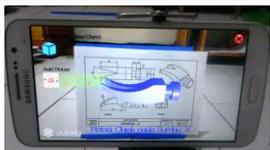
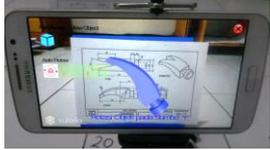
Hasil ujicoba jenis pandangan objek proyeksi amerika

Judul Drawing	Jenis Proyeksi Amerika	Pandangan Objek 3D
Drawing 14	Pandangan Isometrik	
Drawing 14	Pandangan Atas	
Drawing 14	Pandangan Bawah	
Drawing 14	Pandangan Samping Kanan	

Drawing 14	Pandangan Samping Kiri	
Drawing 14	Pandangan Depan	
Drawing 14	Pandangan Belakang	

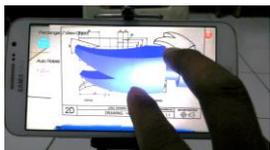
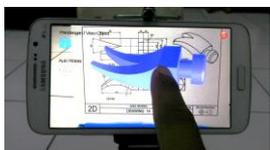
Hasil ujicoba untuk rotasi objek terhadap sumbu X, sumbu Y dan sumbu Z terdapat pada Tabel 10.

Tabel 10  
Hasil ujicoba rotasi pada sumbu X, Y dan Z

Judul Drawing	Jenis Rotasi Objek	Pandangan Objek 3D
Drawing 2	Rotasi objek sumbu X	
Drawing 2	Rotasi objek sumbu Y	
Drawing 2	Rotasi objek sumbu Z	

Sedangkan hasil ujicoba untuk interaksi objek menggunakan jari pada Tabel 11.

Tabel 11  
Hasil ujicoba interaksi menggunakan jari

Judul Drawing	Jenis Interaksi pada Aplikasi	Pandangan Objek 3D
Drawing 2	Zoom In dan Zoom Out	
Drawing 2	Rotasi objek manual	

### V. KESIMPULAN

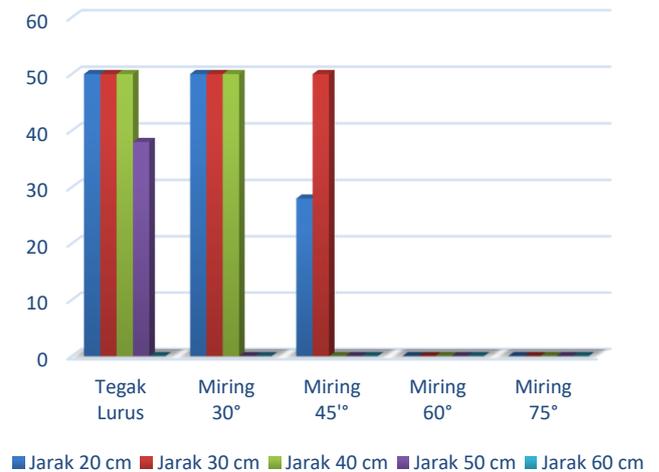
Pada penelitian yang dilakukan, telah dikembangkan aplikasi *Augmented Reality* untuk memvisualisasikan drawing 2D ke dalam bentuk objek 3D dengan marker based tracking menggunakan algoritma *Features from Accelerated Segment Test (FAST) Corner Detection*.

Resume dari hasil deteksi marker menggunakan perangkat Samsung Grand 2 dapat dilihat pada Tabel 12 dan Gambar 30.

Tabel 12. Resume marker terdeteksi dengan perangkat Samsung Grand 2

Jarak (Cm)	Jumlah Marker Terdeteksi dengan Posisi Marker Terhadap Kamera (Pcs)				
	Tegak Lurus	Miring 30°	Miring 45°	Miring 60°	Miring 75°
20	50	50	28	-	-
30	50	50	50	-	-
40	50	50	-	-	-
50	38	-	-	-	-
60	-	-	-	-	-

Jumlah Marker Terdeteksi dengan Perangkat Samsung Grand 2

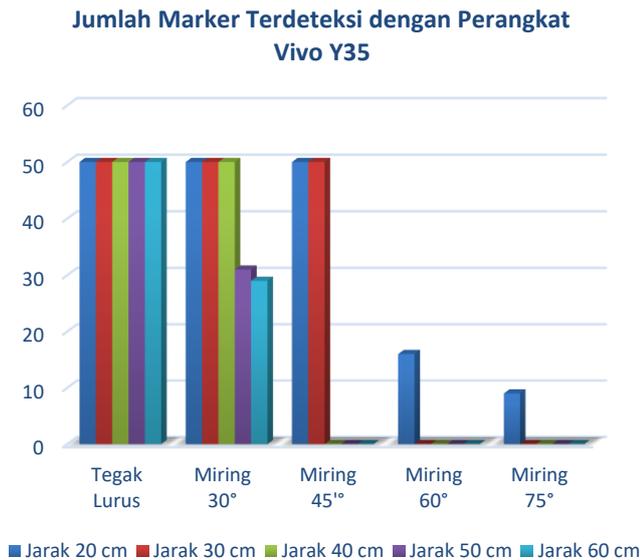


Gambar 30. Jumlah Marker Terdeteksi Pada Perangkat Samsung Grand 2

Resume dari hasil deteksi marker menggunakan perangkat Vivo Y35 dapat dilihat pada Tabel 13 dan Gambar 31.

Tabel 13  
Resume marker terdeteksi dengan perangkat Vivo Y35

Jarak (Cm)	Jumlah Marker Terdeteksi dengan Posisi Marker Terhadap Kamera (Pcs)				
	Tegak Lurus	Miring 30°	Miring 45°	Miring 60°	Miring 75°
20	50	50	50	16	9
30	50	50	50	-	-
40	50	50	-	-	-
50	50	31	-	-	-
60	50	29	-	-	-



Gambar 31. Jumlah Marker Terdeteksi Pada Perangkat Vivo Y35

Dari Gambar 30 dan Gambar 31 disimpulkan bahwa rata-rata marker dapat terdeteksi dengan posisi marker tegak lurus terhadap kamera maksimum pada jarak 50 cm. Sedangkan untuk posisi marker dimiringkan 30° terhadap kamera marker dapat terdeteksi rata-rata pada jarak maksimum 40 cm. Uji coba posisi marker dimiringkan 45° terhadap kamera marker dapat terdeteksi rata-rata pada jarak maksimum 30 cm. Untuk posisi marker dengan kemiringan 60° dan 75° terhadap kamera rata-rata perangkat tidak dapat mendeteksi marker.

Untuk *user interface, tool* yang digunakan untuk membantu proses belajar dapat berjalan sesuai dengan perintah. Proses deteksi marker dapat dipengaruhi oleh spesifikasi perangkat untuk melakukan pengujian. Semakin tinggi spesifikasi perangkat terutama besarnya pixel kamera mempengaruhi tingkat kecepatan dalam proses deteksi marker. Kemudian pencahayaan lampu dapat berpengaruh terhadap auto fokus kamera. Semakin gelap pencahayaan maka semakin lama proses deteksi marker. Selain itu besar kecilnya ukuran marker serta ketebalan garis yang digunakan juga mempengaruhi proses deteksi. Semakin kecil marker dan garis yang tipis maka semakin sulit untuk melacak poin marker dan auto fokus dari kamera.

### DAFTAR PUSTAKA

[1] R. T. Azuma, "A Survey of Augmented Reality," vol. 4, pp. 355–385, 1997.

[2] Y. Gao, H. Wang, and X. Bian, "MARKER TRACKING FOR VIDEO-BASED AUGMENTED REALITY," pp. 10–13, 2016.

[3] T. Kan, C. Teng, and M. Y. Chen, "QR Code Based Augmented Reality Applications," 2011.

[4] G. Kipper, *Augmented Reality An Emerging Technologies Guide to AR*. Elsevier, 2013.

[5] G. T. Sato and N. S. Hartanto, *Menggambar mesin menurut standard ISO*. 1986.

[6] J. Martín-gutiérrez, M. Contero, and M. Alcañiz, "Augmented Reality to Training Spatial Skills," *Procedia - Procedia Comput. Sci.*, vol. 77, pp. 33–39, 2015.

[7] H. Chen, K. Feng, C. Mo, S. Cheng, Z. Guo, and Y. Huang, "Application of Augmented Reality in Engineering Graphics

Education."

[8] E. R. S. Siswanto, "PERBANDINGAN METODE HARRIS CORNER DETECTION , EDGE BASED CORNER DETECTION DAN FAST CORNER DETECTION DALAM APLIKASI PENDETEKSI SENYUM PADA WAJAH MANUSIA Oleh Eduard Royce Siswanto NIM : 612009003 Skripsi Untuk melengkapi syarat-syarat memperoleh Ijasah Sar," 2013.

[9] P. Milgram, "Mixed Reality ( MR ) Reality-Virtuality ( RV ) Continuum," vol. 2351, pp. 282–292, 1994.

[10] H. Kato, K. Tachibana, M. Tanabe, T. Nakajima, and Y. Fukuda, "MagicCup : A Tangible Interface for Virtual Objects Manipulation in Table-Top Augmented Reality," pp. 4–5.

[11] E. Rosten and T. Drummond, "Machine learning for high-speed corner detection," pp. 1–14.

[12] E. Rosten and T. Drummond, "Fusing Points and Lines for High Performance Tracking."

[13] J. Martu, M. Ortega, and D. C. Pe, "Design and validation of an augmented book for spatial abilities development in engineering students," vol. 34, pp. 77–91, 2010.

[14] R. E. Saputro, "Perbandingan Kinerja Algoritma Untuk Mendeteksi Objek Dua Dimensi Pada Augmented Reality," vol. 6, no. 2, pp. 57–74, 2013.

**Nanang Wahyudi** lahir di Sidoarjo, Jawa Timur, Indonesia, pada tahun 1987. Dia menyelesaikan studi S1 di program studi Teknik Informatika Universitas Muhammadiyah Sidoarjo pada tahun 2011, dan menyelesaikan studi masternya pada jurusan Teknologi Informasi di STTS Surabaya. Minat penelitiannya adalah bidang *Artificial Intelligence*.

**Reddy Alexandro Harianto** lahir di Surabaya, Jawa Timur, Indonesia, pada tahun 1989. Dia menyelesaikan studi S1 di program studi Teknik Informatika Sekolah Tinggi Teknik Surabaya pada tahun 2011, dan menyelesaikan studi masternya pada jurusan Teknologi Informatika di Institut Teknologi Sepuluh Nopember Surabaya pada tahun 2015. Minat penelitiannya adalah bidang *Artificial Intelligence, Computer Vision, Data Mining*.

# Klasifikasi SMS Center RSUD SMART Berdasarkan Jenis Keluhan Pelayanan Menggunakan TS Smoothing

Sholeh Rachmatullah, *Teknik Informatika Universitas Madura*, Esther Irawati Setiawan, *Teknologi Informasi Institut Sains dan Teknologi Terpadu Surabaya*, Reddy Alexandro Harianto, *Informatika Institut Sains dan Teknologi Terpadu Surabaya*

**Abstrak** – RSUD SMART memiliki layanan *SMS Center* yang digunakan untuk berinteraksi dengan masyarakat dalam menerima pertanyaan, masukan, saran dan kritik maupun pengaduan. Informasi yang diterima dalam bentuk SMS hanya disimpan dan tidak dikelompokkan berdasarkan unit atau layanan yang dituju sehingga pihak manajemen rumah sakit tidak bisa mengukur tingkat pelayanan di tiap unit. Penelitian ini melakukan klasifikasi terhadap data SMS dan saran responden dari masyarakat yang ditujukan kepada Direktur Rumah Sakit. Data SMS yang diklasifikasi berasal dari *database* aplikasi SMS Center RSUD SMART Pamekasan. Data SMS dan saran responden diklasifikasikan dalam 10 kelas yaitu Rawat Inap, Rawat jalan, Laboratorium, Farmasi, BPJS, Humas, Loket Pembayaran, Sarana dan Prasarana, Profesi dan tidak Terklasifikasi, serta melakukan *scoring SMS*. Sebelum melakukan proses klasifikasi terlebih dahulu dilakukan *pre-processing* seperti penyamaan karakter, penghapusan tanda baca, mengembalikan singkatan, terjemah bahasa daerah (Bahasa Madura), penghapusan angka, penghapusan kata yang tidak penting dalam SMS, dan stemming untuk mengubah kata menjadi kata dasar. Penelitian ini menggunakan algoritma *Naive Bayes* dengan *Two Stage (TS) Smoothing*. Dalam beberapa uji coba yang telah dilakukan terhadap 2292 data dengan presentase data training sebesar 20%, 30%, 40% dan 50% mendapatkan rata-rata akurasi sebesar 82,97% dengan nilai  $\lambda=0.2$ ,  $\mu=2000$  dan  $\text{threshold}=3$ . Bahkan dalam salah satu uji coba klasifikasi dengan *threshold* statis mencapai akurasi 86,73% sedangkan akurasi terendah dengan *threshold* dinamis mencapai 74,28%. Pengaturan *threshold* statis terbukti meningkatkan akurasi klasifikasi sebesar 6,14%.

**Kata Kunci:** *SMS Center*, klasifikasi, *Naive Bayes*, *TS Smoothing*

## I. PENDAHULUAN

**S**MMS Center merupakan layanan SMS *gateway* yang digunakan oleh RSUD dr. H. Slamet Martodirdjo (SMART) kabupaten Pamekasan untuk berinteraksi dengan

masyarakat menggunakan fasilitas SMS berupa pertanyaan, masukan, saran, kritik maupun pengaduan dari masyarakat, dan digunakan sebagai media untuk menyampaikan SMS di lingkup penanggung jawab unit. Informasi berupa SMS yang diterima dalam dua tahun hanya disimpan dalam data base dalam jumlah besar dan tidak terklasifikasi secara khusus.

Hal ini menyebabkan petugas kesulitan menghitung statistik keluhan masyarakat berdasarkan isi pesan singkat di setiap tahunnya. Penulis berusaha untuk mengklasifikasikan data SMS tersebut ke dalam *multi class* [1] berdasarkan ruang lingkungannya [2] dan hasilnya dapat dipergunakan oleh pihak manajemen untuk melakukan evaluasi di setiap unit serta proyeksi pengembangan Sumber Daya Manusia. Salah satu keluaran dari hasil klasifikasi misalnya dengan memberikan surat rekomendasi berupa surat Evaluasi Internal RSUD SMART terhadap unit pelayanan yang mendapatkan skor tertinggi dari laporan yang diterima melalui SMS Center [3].

Terdapat 4 jenis teknik *smoothing* [4], [5] di dalam penggunaan metode *Naive Bayes* untuk proses klasifikasi teks. Teknik tersebut adalah Jelinek-Mercer (JM) *smoothing*, Dirichlet (Dir) *smoothing*, *Absolute Discounting (AD) smoothing*, dan *Two-Stage (TS) smoothing*. Dari hasil penelitian tersebut diperlihatkan bahwa metode *smoothing* mampu meningkatkan tingkat akurasi dari *Naive Bayes* untuk proses klasifikasi teks.

Untuk memaksimalkan performa dari *Naive Bayes Classification* [6] dalam penelitian ini penulis menggunakan *Two-Stage Smoothing*. *Two-Stage Smoothing* digunakan karena dikenal dengan akurasi dan kecepatan dalam proses klasifikasi teks dibandingkan dengan tanpa menggunakan *smoothing* atau hanya mengandalkan *Naive Bayes* untuk mengklasifikasikan SMS [3]. Metode ini digunakan untuk menyelesaikan permasalahan *multi labeling* [7], [8], [9] dan multi klasifikasi yang disesuaikan dengan pola atau model data input yang ada dalam sistem sebelumnya.

## II. TINJAUAN PUSTAKA

*Text Classification* harus melalui beberapa tahapan yang harus dilakukan secara berurutan mulai dari awal atau input sampai dengan outputnya. Hal tersebut disebabkan karena masing-masing tahapan memiliki keterkaitan antara satu

13 Juni 2019, RSUD SMART

Sholeh Rachmatullah, Teknik Informatika, Universitas Madura, Pamekasan, Jawa Timur, Indonesia (e-mail: sholeh@unira.ac.id)

Esther Irawati Setiawan, Teknologi Informasi, Institut Sains dan Teknologi Terpadu Surabaya, Surabaya, Jawa Timur, Indonesia (e-mail: esther@stts.edu)

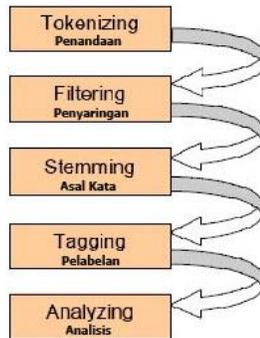
Reddy Alexandro Harianto, Informatika, Institut Sains dan Teknologi Terpadu Surabaya, Surabaya, Jawa Timur, Indonesia (e-mail: reddy@stts.edu)

dengan yang lainnya. Tahapan-tahapan dalam *Klasifikasi teks* tersebut memiliki tujuan untuk mengetahui pola dokumen yang diolah.

**A. Tahapan Klasifikasi**

Tahapan-tahapan dalam klasifikasi teks tersebut adalah sebagai berikut:

1. Tahapan *Tokenizing*. Pada tahap ini *string input* dipotong menurut setiap kata yang menyusunnya.
2. Tahapan *Filtering*. Tahapan ini merupakan kelanjutan tahapan *tokenizing* dimana kata-kata penting yang dihasilkan pada tahapan *tokenizing* diambil untuk disaring menggunakan dua metode yaitu *stop list* yang artinya membuang kata yang kurang penting, ataupun dengan metode *word list* yang artinya menyimpan kata penting. Kedua metode ini berfungsi untuk menyimpan kata yang dianggap penting. Tahapan ini sangat penting mengingat sebuah dokumen sering menggunakan kata penghubung.
3. Tahapan *Stemming*. Pada tahapan ini kata dasar atau akar kata yang berasal dari hasil *filtering* dicari, adalah tahap dimana mencari akar kata dari tiap kata hasil *filtering*, misalkan dengan membuang imbuhan yang ada pada kata tersebut. Proses ini menggunakan algoritma ECS [2], pengembangan dari *Confix Stemmer* [10].
4. Tahapan *Tagging*. Tahapan ini berfungsi untuk mencari bentuk awal dari tiap kata lampau dari hasil *stemming*. Bedanya tahapan ini tentu saja tidak dilakukan untuk teks Bahasa Indonesia yang tidak mengenal istilah kata lampau.
5. Tahap *Analyzing*. Tahapan ini merupakan tahap penentuan seberapa jauh keterhubungan antar kata dengan dokumen yang ada. Tahapan klasifikasi lihat Gambar 1.



Gambar 1. Tahapan *Text Classification*.

**B. Algoritma Naive Bayes dan Smoothing**

Algoritma *Naive Bayes* [6] merupakan pengklasifikasian dengan metode probabilitas dan statistik yang dikemukakan oleh ilmuwan Inggris Thomas Bayes, yaitu memprediksi peluang di masa depan berdasarkan pengalaman di masa sebelumnya sehingga dikenal sebagai teorema Bayes. *Naive Bayes* sendiri menggunakan kemiripan fitur antara data training dan data testing dimana nantinya akan diambil class yang paling mirip dari data training tersebut. Berdasarkan beberapa penilaian, algoritma ini dikenal sebagai algoritma yang sederhana, cepat dan berakurasi tinggi. *Naive Bayes* diasumsikan setiap atribut saling bebas, maka persamaan yang didapatkan adalah sebagai berikut:

$$P(c|d) = \frac{P(d|c)}{P(c)+P(d)} \quad (1)$$

Dimana *c* adalah sebuah kelas, *d* adalah sebuah dokumen. Setiap peluang pada masing-masing kelas akan dikalikan dan akan menghasilkan nilai *Naive Bayes* pada masing-masing rumus tersebut. Nilai tertinggi pada klasifikasi ini akan menjadi hasil klasifikasi dari *Naive Bayes* tersebut. Berikut adalah rumusnya:

$$C_{map} = \max_{c \in C} \hat{P}(c|d) = \max_{c \in C} \hat{P}(c) \prod_{1 \leq k \leq nd} P(t_k|c) \quad (2)$$

$$\hat{P}(c) = \frac{N(c)}{N} \quad (3)$$

Dimana  $\hat{P}(c|d)$  adalah estimasi probabilitas Kategori *c* terhadap dokumen *d*,  $\hat{P}(c)$  adalah estimasi probabilitas *prior* dari dokumen muncul di kategori *c*,  $P(t_k|c)$  adalah probabilitas bersyarat dari term *t<sub>k</sub>* yang muncul di kategori *c*.

*Smoothing* untuk NBC ini mengacu pada penyesuaian estimator maksimum *likelihood* untuk model bahasa sehingga akan lebih akurat. *Smoothing* sendiri tidak diperlukan untuk menetapkan nilai nol untuk kata yang tak terlihat. *Smoothing* memainkan dua peran penting yaitu meningkatkan akurasi model bahasa, serta mengakomodasi generasi pada umumnya dan kata noninformatif. Generator kemungkinan maksimum umumnya di bawah perkiraan probabilitas dari kata yang tak terlihat. Jadi tujuan utama *smoothing* adalah untuk memberikan probabilitas tidak nol untuk kata-kata yang tak terlihat dan meningkatkan akurasi probabilitas estimator.

Untuk menghindari nilai 0 pada hasil perhitungan probabilitas bersyarat dari *term* tertentu yang muncul di sebuah kategori diperlukan metode *smoothing* [5]. Beberapa metode *smoothing* yang dapat dilakukan, tetapi dari penelitian sebelumnya terdapat beberapa metode *smoothing* yang sudah diujicobakan pada *Naive Bayes* meliputi [4] :

1. Laplace Smoothing.

$$p(w|c_i) = \frac{1+c(w,c_i)}{|V|+\sum_{w' \in V} c(w',c_i)} \quad (4)$$

2. Two-Stage (TS) Smoothing, merupakan kombinasi antara Dirichlet Smoothing dengan metode interpolation.

$$p_{\lambda,\mu}(w|c_i) = (1 - \lambda) \frac{c(w,c_i)+\mu p(w|C)}{\sum_{w' \in V} c(w',c_i)+\mu} + \lambda p(w|C) \quad (5)$$

Modifikasi ini didasarkan pada penggunaan persamaan rumus laplace dan Two Stage Smoothing menjadi seperti dibawah ini:

$$p_{\lambda,\mu}(w|c_i) = (1 - \lambda) \frac{1+c(w,c_i)+\mu p(w|C)}{\sum_{w' \in V} c(w',c_i)+\mu} + \lambda p(w|C) \quad (6)$$

Modifikasi dilakukan jika pada saat perhitungan nilai probabilitas dengan menggunakan persamaan rumus Two-Stage Smoothing sudah ditentukan dan kata unik hasil *stemming* tidak ada di dalam semua basis data training maupun di dalam basis data training kelas itu sendiri.

**C. Multi Klasifikasi Teks**

Kategorisasi dokumen teks secara multi-label adalah masalah yang berperan penting dalam sistem temu kembali informasi. Klasifikasi multi-label berbeda dengan klasifikasi single label. Klasifikasi *single label* akan mengklasifikasikan suatu dokumen ke dalam satu kategori dokumen saja. Sedangkan klasifikasi multi-label dapat mengelompokkan suatu dokumen ke lebih dari satu kategori. Sebagai contoh, sebuah artikel ilmiah yang di dalamnya membahas secara rinci

mengenai deteksi dini kanker ovarium dengan pendekatan bioinformatika dapat memiliki lebih dari satu label dokumen yaitu: kesehatan, bioinformatika dan wanita.

Untuk mengklasifikasikan artikel ke dalam *multi label* [7], dilakukan pendekatan dengan mentransformasikan klasifikasi multi label menjadi beberapa kasus klasifikasi *single label* [6]. Dengan melakukan transformasi *single label* maka tiap-tiap data dilakukan pengecekan terhadap semua label yang ada. Pengecekan yang dilakukan akan menentukan apakah data tersebut “termasuk” atau “tidak termasuk” pada suatu label. Hal ini kurang efisien karena memerlukan waktu komputasi yang cukup lama untuk mengecek suatu dokumen ke semua label. Metode ini tidak memperhitungkan korelasi antara label yang satu dengan label yang lain.

Penentuan multi klasifikasi dilakukan dengan dua cara yaitu dengan memberikan nilai threshold statis dan nilai threshold dinamis. Cara kerja multi klasifikasi mengambil 2 nilai probabilitas tertinggi dari kelas tertentu, hal ini dilakukan untuk menentukan apakah dokumen tersebut hanya tergolong *single classification* atau *multi classification*. Pada penentuan multi klasifikasi dengan nilai threshold statis, sistem akan menetapkan nilai threshold yang diinput secara statis, kemudian melakukan perbandingan selisih antara nilai threshold yang sudah ditentukan dengan nilai probabilitas terbesar kedua. Jika nilai threshold lebih besar dari nilai probabilitas terbesar kedua maka program akan menjadikan probabilitas kelas tersebut sebagai dokumen multi kelas. Ketika nilai threshold lebih kecil atau sama dengan nilai probabilitas terbesar kedua, maka dokumen tersebut hanya memiliki *single classification*.

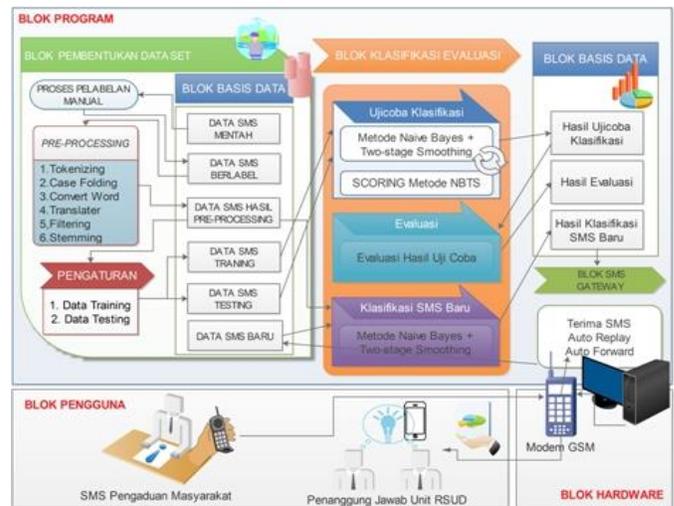
Pada penentuan *multi classification* [9] dengan nilai threshold dinamis, maka setiap dokumen yang diuji coba akan memiliki nilai threshold yang berubah-ubah tergantung pada jarak antar probabilitas kelas yang satu dengan probabilitas kelas lainnya. Cara kerja dari konsep ini adalah jika jarak nilai probabilitas kedua terbesar dengan nilai probabilitas terbesar pertama lebih dominan dari pada nilai probabilitas kelas lainnya terhadap nilai probabilitas pertama, maka probabilitas kelas tersebut merupakan bagian dari kelas probabilitas terbesar pertama yang berarti dokumen tersebut terdeteksi sebagai multi klasifikasi. Namun jika jaraknya tidak lebih dominan dari pada kelas lainnya maka dokumen diklasifikasikan ke dalam satu kelas saja [11].

### III. METODOLOGI PENELITIAN

Penelitian ini memiliki input utama yaitu data SMS yang masuk atau diterima pada database aplikasi SMS Center RSUD dr. H. Slamet Martodirdjo Kabupaten Pamekasan dan form isian komentar responden hasil survei kepuasan masyarakat selama 4 bulan yang dilakukan dalam kurun waktu bulan Mei dan Juni tahun 2015 untuk tahap pertama, kemudian pengambilan data responden tahap ke dua pada bulan Oktober dan Nopember tahun 2015 oleh LP3M Akademi Keperawatan Pemerintah Kabupaten Pamekasan dengan total keseluruhan yaitu 2292 data dan 415 data yang termasuk multi label klasifikasi. Sistem ini bersifat *multi-class* dimana setiap SMS yang diterima akan masuk ke salah satu atau lebih kelas yang selanjutnya akan diteruskan ke

penanggungjawab unit terkait yang disesuaikan dengan hasil klasifikasi oleh aplikasi.

Beberapa data SMS yang tersimpan dalam database Aplikasi SMS RSUD Pamekasan terdiri dari nomor urut, nomor pengirim, waktu diterimanya SMS (jam, tanggal, bulan dan tahun) teks SMS, isi pesan SMS dan jenis teks. Dalam proses klasifikasi data yang digunakan hanya teks SMS sehingga diperlukan dilakukan proses cleaning dengan mengambil data teks SMS saja, sedangkan nomor pengirim, waktu diterimanya SMS boleh diabaikan hal ini karena tidak berpengaruh terhadap proses selanjutnya. Sedangkan form isian komentar responden hasil survei kepuasan masyarakat selama 4 bulan pada bulan Mei, Juni, Oktober dan Nopember tahun 2015 oleh LP3M Akademi Keperawatan Pemerintah Kabupaten Pamekasan, ini merupakan hasil kerjasama dengan pihak RSUD yang dilakukan hanya dalam 1 tahun saja atau tidak rutin. Formulir yang berasal dari masyarakat berisikan form isian no responden, lokasi ruangan, nama, alamat asal, jenis kelamin, pendidikan, pekerjaan, usia, tanggal, status pemeriksaan, hari rawat ke, jawaban pilihan ganda dan saran responden. Data responden yang akan di masukkan ke dalam sistem aplikasi ini adalah hanya data saran responden dan proses pelabelan setiap isian data saran dari responden.



Gambar 2. Arsitektur Sistem

Dalam penelitian ini data SMS dan saran responden diklasifikasikan ke dalam dalam sepuluh kelas antara lain Humas, Sarana dan Prasarana, Laboratorium, Farmasi, Rawat Inap, Rawat Jalan, Loker Pembayaran, BPJS, Profesi dan Tidak Terklasifikasi. Kelas Tidak Terklasifikasikan diperuntukkan untuk SMS yang sifatnya tidak spesifik mengarah ke sembilan kelas yang sudah disebutkan di atas, misalnya SMS Spam yang bersal dari orang yang tidak bertanggungjawab, [12] SMS Iklan yang berisikan berbagai promosi layanan atau penjualan dari orang lain, dan SMS Operator merupakan informasi yang dikirimkan oleh operator penyedia jaringan kartu selular, biasanya SMS yang diterima dari operator memiliki nomor khusus empat sampai enam digit nomor pengirim.

Klasifikasi tersebut dimaksudkan untuk mendistribusikan SMS ke masing-masing unit agar ditangani dan dijadikan bahan monitoring dan evaluasi unit tersebut. Setiap SMS yang masuk diproses untuk mengetahui kecocokan data dengan salah satu atau lebih kelas yang tersedia. SMS baru yang diterima oleh SMS Gateway akan proses sistem, sistem mengolah data SMS dengan melakukan *pre-processing* dan melakukan proses klasifikasi sehingga menghasilkan multi kelas, kemudian aplikasi akan meneruskan pesan SMS tersebut kepada unit yang dituju berdasarkan hasil klasifikasi. Selain unit, pihak manajemen rumah sakit juga dapat memonitoring dan mengevaluasi kinerja pelayanan dengan lebih mudah. Dari hasil klasifikasi setiap periode tertentu, sistem akan memberikan rekomendasi berupa surat pemberitahuan terhadap unit yang memiliki skoring tertinggi berdasarkan laporan dari data *SMS Center*. Model klasifikasi yang dibangun akan dijabarkan secara menyeluruh dalam suatu sistem yang terstruktur yang disajikan pada Gambar 2.

Membangun arsitektur sistem untuk mempermudah kerangka kerja sistem yang akan dikerjakan, dalam hal ini dilakukan pemisahan blok untuk program, hardware dan pengguna. Pada Gambar 2 arsitektur sistem bagian BLOK PROGRAM merupakan bagian dari sistem perangkat lunak yang berfungsi untuk melakukan pengolahan dan pemrosesan data. BLOK PROGRAM memiliki 4 sub blok diantaranya Blok Pembentukan Data Set, Blok Basis Data, Blok Klasifikasi Evaluasi, dan Blok *SMS Gateway*.

Proses awal ditunjukkan dalam Blok Pembentukan Data Set, ada 3 proses yang dilakukan yaitu proses pelabelan secara manual pada data SMS mentah, *Pre-Processing*, dan pengaturan data *testing* serta *training*. Pada proses *Pre-Processing* ada tahapan *stemming* yang menggunakan algoritma *Enhanced Confix Stripping (ECS) Stemmer* [13]. Pada Blok Basis Data mempunyai 9 jenis data yaitu data SMS mentah, SMS berlabel, SMS hasil *pre-processing*, SMS *trainig*, SMS *testing*, hasil uji coba klasifikasi, hasil evaluasi, SMS baru, dan hasil klasifikasi SMS baru.

Blok Klasifikasi Evaluasi terdiri dari 3 proses yaitu proses pertama yaitu Uji Coba Klasifikasi berfungsi untuk melakukan perhitungan menggunakan metode *Naive Bayes* dengan *Two-Stage Smoothing (NBTS)* [4] dan *Scoring*. *Scoring* menggunakan metode NBTS untuk menghitung skor masing masing dokumen agar diketahui tingkat keluhan yang ada. Tingkat keluhan dibagi menjadi tiga yaitu ringan dengan skor 1, sedang dengan skor 3, dan tinggi dengan skor 5. Nilai rata rata skor diperoleh dari jumlah dokumen dengan total skor dibagi 2. Selanjutnya proses kedua Evaluasi berfungsi menganalisa hasil uji coba klasifikasi agar sesuai harapan penelitian, dan proses ketiga Klasifikasi SMS Baru berfungsi untuk mengklasifikasikan data SMS baru yang diterima dari blok *SMS gateway*. BLOK HARDWARE berfungsi sebagai SMS gateway yang bertugas untuk menerima SMS, dan *auto respon SMS*.

$$Score\ Total = \frac{\sum Dokumen + \sum Score\ dokumen}{2} \quad (8)$$

Fungsi dari proses perhitungan NBTS dalam melakukan proses klasifikasi teks yang bersifat multi kelas dibedakan berdasarkan penentuan nilai *threshold* statis dan nilai *threshold* dinamis. Proses perhitungan fungsi *Naive Bayes*

dengan *Two Stage Smoothing* terhadap setiap kelasnya akan menghasilkan nilai probabilitas NBTS setaiap kelas. Setelah itu carilah 2 angka nilai maksimum dari keseluruhan kelas, kemudian ambillah 2 nilai terbesar dari nilai probabilitas NBTS, langkah selanjutnya adalah menentukan nilai *threshold* dengan nilai statis atau dinamis.

Nilai maksimum terhadap suatu fungsi akan dipilih sebagai hasil keputusan. Hasil keputusan bisa menunjukkan data SMS tersebut tergolong dalam satu kelas atau maksimal dua kelas. Dalam menentukan hasil klasifikasi bersifat multi kelas atau tidak maka dibuatlah pendekatan penyelesaian dengan menggunakan batasan atau *threshold*. *Threshold* berfungsi untuk mengukur apakah nilai *Naive Bayes Classification Two Stage (NBCTS)* maksimum ke 2 merupakan bagian dari nilai *Naive Bayes* tertinggi ke 1. Jika nilai *Naive Bayes* lebih besar dari nilai *threshold* yang ditentukan maka hasil klasifikasinya hanya memiliki satu kelas saja, jika nilai *Naive Bayes* terbesar ke 2 lebih kecil atau sama dengan nilai *threshold* maka hasil klasifikasinya memiliki dua kelas atau multi kelas.

Pengaturan nilai *threshold* ada dua cara, yang pertama adalah *threshold* statis artinya penentuan nilai *threshold* ditentukan oleh user dan bersifat tetap atau tidak berubah ketika diuji cobakan terhadap setiap teks, sedangkan yang ke dua adalah *threshold* dinamis dengan penentuan nilai *threshold* bergantung pada rata-rata nilai probabilitas kelas di setiap proses klasifikasinya sehingga sifat *threshold* ini selalu berubah atau selalu berbeda disetiap teks yang di klasifikasikan.

Algoritma 1. Klasifikasi NBTS *Threshold* Statis

[Fungsi : menghitung nilai probabilitas teks]

[input : hasil *stemming* teks data *training* dan *testing*]

[output : klasifikasi kelas]

- 1: Lakukan proses *pre-processing* [*tokenizing, casefolding, convert word, translater, filtering, stemming*]
- 2: Tentukan koefisien kontrol  $\mu$ (miu),  $\lambda$ (lamda), *threshold* dan modifikasi rumus.
- 3: Hitung *prior probability naive bayes* semua kelas.
- 4: Menghitung nilai probabilitas *Two Stage Smoothing* setiap kata terhadap semua kelas.
- 5: Hitung nilai probabilitas setiap kelas.
- 6: Lakukan normalisasi.
- 7: *Sorting* nilai probabilitas kelas.
- 8: Ambil 2 nilai probabilitas tertinggi.
- 9: Periksa nilai probabilitas tertinggi ke 2. Jika nilainya lebih besar dari *threshold* maka output satu kelas, tetapi jika nilainya lebih kecil dari *threshold* maka outputnya multi kelas.

Contoh SMS dengan menggunakan algoritma 1, dengan teks awal “Kami keluarga pasien rsud pamekasan, kami merasa kurang puas dengan pelayanan pengambilan obat di depo farmasi 1 karena kurangnya pegawai, sehingga kami harus nunggu+- 1 jam lebih. Kami mohon pelayanan yg cepet dan memuaskan,ttd ach muchlis. ”. Kemudian lakukan proses *pre-processing*, teks berubah menjadi “keluarga pasien puas layan ambil obat depo farmasi kurang pegawai nunggu jam layan cepet puas muchlis ”. Menentukan koefisien kontrol  $\mu = 0,1$ ,  $\lambda = 0,3$ , *threshold* dan modifikasi rumus. Menghitung nilai

prior probability naive bayes rumus no 1 terhadap semua kelas, diperoleh hasil  $P(\text{Class}=\text{loket pembayaran}) = 68/1574 = 0.0432$ .

Menghitung nilai probabilitas *Two Stage Smoothing* setiap kata terhadap semua kelas, maka hasil yang diperoleh adalah  $P_{\lambda\mu}(\text{word}=\text{keluarga}|\text{class}=\text{tidak terklasifikasi}) = 0.0019$ . Kemudian menghitung nilai probabilitas setiap kelas, maka diperoleh hasil  $P(D|\text{rawat inap}) = 3,7107067445934E-50$ . Lakukan normalisasi hasilnya  $P(\text{Class}=\text{tidak terklasifikasi}) = 1.5870950299765E-14$ . Urutkan nilai probabilitas kelas dari tertinggi hingga terendah dan ambil 2 nilai probabilitas tertinggi, diperoleh kelas Farmasi=54455863,6174 dan Rawat Jalan= 1,6049. Periksalah nilai probabilitas tertinggi ke 2 untuk menentukan hasil klasifikasi.

Proses disinilah yang membedakan hasil output klasifikasinya. Hal ini dilakukan untuk mengetahui apakah outputnya memiliki satu atau dua kelas. Jika nilai probabilitas lebih besar dari threshold maka output single kelas, tetapi jika nilainya lebih kecil dari dari *threshold* maka outputnya multi kelas. Dari contoh diatas nilai probabilitas rawat jalan adalah 1,6049045186192E-8 sedangkan thresholdnya 0,6 maka hasilnya satu kelas, artinya Rawat Jalan bukan bagian dari kelas pertama. Kesimpulan dari hasil klasifikasi teks diatas adalah Farmasi.

Aplikasi yang dirancang menunjukkan hasil perhitungan dengan menggunakan *threshold* statis dan *threshold* dinamis. *Threshold* statis bersifat tidak berubah untuk setiap teks yang diklasifikasikan berdasarkan metode NBTS. Sedangkan *threshold* dinamis [13], [14] bersifat selalu berubah secara otomatis untuk setiap teks yang diklasifikasikan berdasarkan nilai NBTS dengan menghitung jarak rata-rata yang paling dominan terhadap nilai NB tertinggi pertama.

Algoritma 2. Multi Klasifikasi *Threshold* Dinamis

[Fungsi : klasifikasi multi kelas dinamis]

[input : nilai probabilitas semua kelas ]

[output : hasil klasifikasi]

- 1: Hitung rata2 nilai probabilitas NB+Smoothing setiap teks.
- 2: Ambil nilai rata2 sebagai nilai *threshold* dinamis.
- 3: Ambil 2 nilai probabilitas tertinggi.
- 4: Bandingkan nilai probabilitas tertinggi ke 2 dengan nilai *threshold*.
- 5: Tampilkan hasil klasifikasi nilai probabilitas ke-1 dan ke-2 jika ada.

Contoh SMS dengan menggunakan algoritma 2, dengan teks awal “perawat cuek, kurang ramah pelayanan lebih diperbaiki lagi khususnya untuk perawat harus lebih perhatikan kondisi dan kebutuhan pasien.”. Kemudian lakukan proses *pre-processing*, teks berubah menjadi “perawat cuek ramah layan baik awat hati kondisi pasien”. Menentukan Koefisien control nilai  $\mu=0,2$  dan  $\lambda =2000$ . Mencari nilai probabilitas *naive bayes* setiap kelasnya, diperoleh  $P(\text{Class}=\text{profesi}) = 0.1910$ . Menghitung nilai probabilitas

setiap kata hasil *stemming* terhadap masing-masing kelas misalnya kata “ramah”, maka diperoleh  $P_{\lambda\mu}(\text{word}=\text{ramah} | \text{class}=\text{tidak terklasifikasi}) = 0.0042$ . Menghitung nilai probalitas seluruh kelas terhadap setiap kata misalnya klasifikasi Profesi,  $P(D|\text{profesi}) = 2,0220 E-20$ . Normalisasi menjadi  $P(\text{Class}=\text{profesi}) = 1.5870$ . Urutkan nilai probabilitas kelas dari tertinggi hingga terendah dan ambil 2 nilai probabilitas tertinggi, maka hasilnya kelas Rawat Inap=1,4361 dan profesi=0,4647. Periksalah nilai probabilitas tertinggi ke 2 untuk menentukan hasil klasifikasi.

Proses disinilah yang membedakan hasil output klasifikasinya. Nilai *threshold* disetiap data uji nilainya akan selalu berubah tergantung pada nilai probabilitasnya, maka rata-rata nilai probabilitas=0,4995. Jadikan nilai nilai rata-rata sebagai nilai *threshold* dinamis. Ambil 2 nilai probabilitas tertinggi, maka diperoleh klasifikasi 1 adalah Rawat Inap dengan nilai 1,4361 dan klasifikasi ke 2 adalah Profesi dengan nilai 0,4647. Bandingkan nilai probabilitas tertinggi ke 2 dengan *threshold*. Karena nilai probabilitas kelas Profesi lebih kecil dari nilai *threshold* maka kelas tersebut bagian dari kelas pertama atau output dari hasil klasifikasi ini adalah multi kelas. Berdasarkan hasil perhitungan sebelumnya, disimpulkan hasil klasifikasinya adalah Rawat Inap dan Profesi.

Proses awal aplikasi ini dimulai dari melakukan pelabelan manual terhadap data SMS mentah sesuai dengan daftar kelas yang telah ditentukan dan dilanjutkan dengan proses *pre-processing* dari data SMS yang sudah berlabel. Hasil dari tahap *pre-processing* kemudian lakukan pengaturan data training dan data testingnya dipilah berdasarkan persentase 20-50% untuk dilakukan proses uji coba klasifikasi menggunakan NBTS. Hasilnya dievaluasi untuk mengetahui dan membandingkan tingkat akurasi pada proses klasifikasi. Setelah proses pengujian dan evaluasi selesai maka dihitung jumlah kelas terbesar akan mendapatkan rekomendasi berupa surat teguran mengenai pelayanan yang diberikan terhadap masyarakat. Setiap SMS baru yang dikirimkan oleh masyarakat akan dilakukan *pre-processing* dan dilakukan pengklasifikasian dengan menggunakan data training yang didapatkan dari seluruh data SMS hasil *pre-processing* dengan menggunakan *Naive Bayes* dan *TS Smoothing* [9]. Hasil klasifikasi SMS baru digukan untuk meneruskan pesan secara otomatis dari masyarakat kepada penanggung jawab setiap unit yang berkaitan berdasarkan isi SMS yang telah diklasifikasi.

IV. HASIL EKSPERIMEN

Sebelum melakukan uji coba klasifikasi, yang harus dilakukan adalah memilah data SMS dan responden menjadi data *training* dan data *testing*. Pemilihan ini dilakukan 4 variasi data *testing* berdasarkan presentase 20%, 30%, 40%, dan 50% dari keseluruhan data. Pemilihan data testing ini dilakukan secara acak berdasarkan indeknya (kode teks). Untuk setiap pemilihan tersebut dilakukan uji coba klasifikasi menggunakan NBTS. Pemilahan data testing dan data training dalam sistem ditunjukkan pada Tabel 1.



#	Isi Pesan	Hasil Preprocessing	Class	Hasil Class	Skor
46	perawat kalau berkomunikasi harus lebih halus., dokter saja halus kok perawatnya agak kasar	perawat komunikasi halus dokter halus perawat kasar	profesi	profesi	1
47	kalau malam perawat jangan suka tidur, apalgi yang punya tugas malam	malam perawat tidur punya tugas malam	rawat inap, profesi	rawat inap	1
48	perawat harus tanggap terhadap pasien yang lagi sakit di rawat inap	perawat tanggap pasien sakit rawat inap	rawat inap, profesi	profesi, rawat inap	1

Gambar 4. Hasil Klasifikasi Data Testing

TABEL I  
DATA TESTING DAN DATA TRAINING

No	Persentase Data Testing (%)	Jumlah Data	Data Testing	Data Training
1	20	2292	458	1834
2	30	2292	682	1610
3	40	2292	912	1380
4	50	2292	1143	1149

Pada Tabel 1 pemilahan data testing dan data training dilakukan secara dinamis dengan memilih kode teks dalam *database*. Hal ini dilakukan agar dimungkinkan untuk menambah variasi pemilihan data *testing* dan data *training*, dengan ketentuan lebih banyak data *training* dari pada *testing*.

TABEL II  
UJI COBA KLASIFIKASI THRESHOLD STATIS

No	Data uji	$\lambda$	$\mu$	Threshold	Akurasi	Rata2
1	458	0,2	2000	3	85,22%	
2	458	0,2	0,5	3	81,25%	
3	458	0,3	1000	2	84,56%	
4	458	0,2	0,5	2	80,14%	82,79%
5	682	0,2	2000	3	85,48%	
6	682	0,2	0,5	3	82,11%	
7	682	0,3	1000	2	83,28%	
8	682	0,2	0,5	2	81,23%	83,03%
9	912	0,2	2000	3	86,73%	
10	912	0,2	0,5	3	82,02%	
11	912	0,3	1000	2	85,09%	
12	912	0,2	0,5	2	80,70%	83,64%
13	1143	0,2	2000	3	85,91%	
14	1143	0,2	0,5	3	80,66%	
15	1143	0,3	1000	2	83,55%	
16	1143	0,2	0,5	2	79,70%	82,46%

Pada Tabel 2 menunjukkan hasil uji coba klasifikasi menggunakan metode NBTS dengan pengaturan *threshold* statis. Koefisien lamda, miu dan *threshold* disesuaikan nilainya untuk mendapatkan nilai akurasi yang tinggi, pada beberapa percobaan akan diambil akurasi yang tertinggi di setiap presentase data testing, dalam hal ini no uji coba ke 1, 5, 9 dan 13 adalah hasil akurasi tertinggi.

Pada Tabel 3 menunjukkan hasil uji coba klasifikasi menggunakan metode NBTS dengan pengaturan *threshold* dinamis. Pada Tabel 2 kolom *threshold* bernilai 0 artinya nilai

ini memiliki nilai *threshold* yang selalu berubah tergantung nilai probabilitas masing-masing teks yang akan diklasifikasikan.

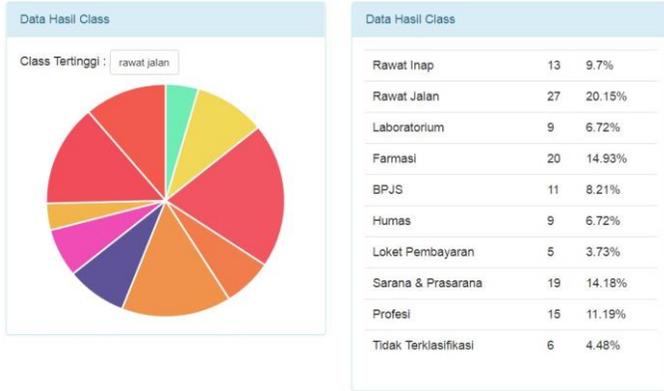
TABEL III  
UJI COBA KLASIFIKASI THRESHOLD DINAMIS

No	Data uji	$\lambda$	$\mu$	Threshold	Akurasi	Rata2
1	458	0,1	3000	0	77,49%	
2	458	0,2	900	0	76,61%	
3	458	0,1	0,3	0	74,83%	
4	458	0,2	0,3	0	75,07%	76,00%
5	682	0,1	3000	0	80,06%	
6	682	0,2	900	0	76,98%	
7	682	0,1	0,3	0	76,25%	
8	682	0,2	0,3	0	76,10%	77,35%
9	912	0,1	3000	0	80,59%	
10	912	0,2	900	0	78,95%	
11	912	0,1	0,3	0	76,21%	
12	912	0,2	0,3	0	76,21%	77,99%
13	1143	0,1	3000	0	77,87%	
14	1143	0,2	900	0	77,17%	
15	1143	0,1	0,3	0	74,28%	
16	1143	0,2	0,3	0	74,72%	76,01%

Pada Gambar 4 hasil klasifikasi data *testing* menunjukkan isi Pesan merupakan field untuk menampung isi pesan, hasil dari proses pre-processing di tampilkan pada kolom hasil preprocessing, proses pelabelan yang dilakukan oleh pakar hasilnya ditunjukkan pada kolom *Class* sedangkan hasil proses dari perhitungan klasifikasi multi kelas ditunjukkan pada kolom Hasil *Class*, kemudian hasil proses *scoring* ditunjukkan pada Skor. Pada pojok kanan atas juga disediakan filter yang berfungsi untuk menampilkan hasil klasifikasi berdasarkan seluruh data atau berdasarkan multi label-nya atau berdasarkan hasil multi klasifikasi-nya. Ada dua tombol proses yaitu tombol a untuk melihat detail perhitungan proses klasifikasi NBTS dan tombol b untuk melihat hasil detail perhitungan proses *SCORING NBTS*.

Pada Gambar 5 disajikan contoh SMS baru yang telah diklasifikasikan menggunakan *NBTS*. Fitur ini akan digunakan setelah dilakukan sinkronisasi antara aplikasi ini dengan Aplikasi SMS Center RSUD yang sudah ada sebelumnya ditampilkan sebelah kanan berupa tabel statistik hasil klasifikasi dari data SMS berupa presentase masing-masing kelas dan tampak sebelah kiri ada grafik *pie* yang

menggambarkan luasan persentase kelas yang ada. Terdapat skoring tertinggi yang mendapatkan SMS terbanyak dan skor berdasarkan tingkat keluhan, sehingga dalam hal ini bisa direkomendasikan mendapatkan surat peringatan terhadap penanggung jawab Rawat Jalan.



Gambar 5. Grafik hasil Klasifikasi SMS baru

Informasi kelas tertinggi yang terletak di atas gambar grafik, juga berfungsi untuk mencetak surat peringatan terhadap penanggung jawab unit berdasarkan hasil skoring terbesar yang diperoleh dari rata rata jumlah klasifikasi ditambah total skor masing-masing SMS.

TABEL IV  
ANALISA AKURASI UJI COBA KLASIFIKASI

No	Threshold	1	2	3	4	Rerata (%)
1	Manual	82,78%	83,03%	83,64%	82,46%	82,97%
2	Dinamis	75,99%	77,35%	77,99%	76,01%	76,84%
				Selisih		6,14%

Pada Tabel 4 dapat diketahui rata-rata akurasi uji coba klasifikasi menggunakan *Naïve Bayes* dengan *TS Smoothing* menggunakan *threshold* Statis(Manual) 82,97%, sedangkan rata-rata akurasi uji coba klasifikasi menggunakan *Naïve Bayes* dengan *TS-Smoothing* menggunakan *threshold* Dinamis sebesar 76,84%. Terbukti klasifikasi menggunakan *NBTS threshold* Statis dapat meningkatkan akurasi klasifikasi sebesar 6,14% apabila dibandingkan dengan klasifikasi *NBTS* menggunakan *threshold* Dinamis. Besar kecilnya akurasi klasifikasi sangat dipengaruhi oleh terjadinya kesalahan klasifikasi.

TABEL V  
STATISTIK DATA UJI COBA KLASIFIKASI

No	Class	Jumlah Data Testing
1	Rawat Inap	122
2	Rawat Jalan	74
3	Laboratorium	8
4	Farmasi	35
5	BPJS	36
6	Humas	67
7	Loket Pembayaran	25
8	Sarana dan Prasarana	172
9	Profesi	116
10	Tidak Terklasifikasi	27

TABEL VI  
STATISTIK DATA UJI COBA KLASIFIKASI

No	Medode NBTS	Klasifikasi Benar	Klasifikasi Salah
1	Threshold statis	583	99
2	Threshold dinamis	546	136

Uji coba klasifikasi dilakukan dengan data latih 30% dari total data sms dan responden sebesar 682 data testing pada Tabel 1 nomor 2. Pemilahan data testing dilakukan secara dinamis oleh sistem dengan cara mengambil kode teks secara acak pada masing-masing kelas sebesar 30%. Statistik jumlah data testing bisa dilihat pada Tabel 5. Selisih Jumlah data testing antar kelas sangat besar atau tidak seimbang, sehingga berpengaruh pada hasil uji coba klasifikasi lihat Tabel 6, hasil klasifikasi salah semakin besar.

Pada Tabel 7 bagian ini akan dipaparkan hasil uji metode dari semua *smoothing* yang ada. Tahap pengujian ini menggunakan data testing baru sebesar 20% dari keseluruhan data.

TABEL VII  
AKURASI SEMUA SMOOTHING

No	Smoothing	Koefisien Kontrol	Threshold	Akurasi (%)
1	Laplace	-	2	85,65
2	Jelinek-Mercer(JM)	$\lambda=0.1$	2	86,75
3	Dirichlet (DIR)	$\mu=0.1$	2	88,30
4	Absolute Discounting (DS)	$\delta=0.1$	2	83,22
5	Two Stage (TS)	$\lambda=0.1, \mu=3000$	0	88,52

Berdasarkan hasil uji coba bisa disimpulkan Laplace smoothing, Dir Smoothing, JM Smoothing dan AD smoothing akurasi masih dibawah TS-Smoothing.

V. KESIMPULAN

Adapun beberapa kesimpulan yang didapatkan dalam penelitian ini, antara lain:

1. Penelitian ini telah berhasil mengklasifikasikan dokumen dalam bentuk teks dengan baik menggunakan metode *naive bayes* dengan *TS-Smoothing*.
2. Pada penelitian ini rata-rata akurasi klasifikasi dengan menggunakan *Naïve Bayes TS-Smoothing* dan *threshold Statis* sebesar 82,97%, sedangkan rata-rata akurasi klasifikasi dengan *threshold Dinamis* sebesar 76,84%. Maka *Naïve Bayes* dengan *TS-Smoothing* mampu untuk memecahkan permasalahan pada multi klasifikasi.
3. Metode *Two Stage Smoothing* dengan *threshold statis* terbukti meningkatkan akurasi klasifikasi sebesar 6,14%.
4. Pada uji coba klasifikasi *threshold statis* diperoleh pengujian akurasi terendah sebesar 80,13% dan akurasi tertinggi sebesar 86,73%.
5. Pada uji coba klasifikasi *threshold dinamis* diperoleh pengujian akurasi terendah sebesar 74,28% dan akurasi tertinggi sebesar 80,59%.
6. Rekomendasi surat evaluasi internal diberikan kepada unit dengan skoring terbesar, tidak hanya berdasarkan jumlah SMS yang diterima di setiap unit.

7. Kesalahan klasifikasi sering disebabkan oleh tidak seimbangnya jumlah SMS di setiap kelas pada data training.
  8. TS-Smoothing lebih baik dari segi akurasi dibandingkan dengan metode yang lainnya.
- Minat penelitiannya adalah bidang *Artificial Intelligence, Computer Vision, Data Mining*.

#### DAFTAR PUSTAKA

- [1] G. Malik, M. Tarique, "On Machine Learning Techniques For Multi-class Classification", *International Journal of Advancements in Research & Technology*, vol.3, no.2, Feb 2014.
- [2] R. Adipratama, R.A. Hendrawan, R.P. Kusumawardani, "Klasifikasi SMS Pada Sistem Support Ticket Pengaduan Pelanggan Dengan Menggunakan Metode Naïve Bayes", *Undergraduate Thesis of Information System: ITS Library*, 2011.
- [3] Badar Said, Yuliana Melita P, "Klasifikasi Data SMS Center Bupati Pamekasan Menggunakan Naïve Bayes Dengan Mad Smoothing", *In IdeaTech*, pp.92-99, 2015.
- [4] Q Yuan, G Cong, N.M. Thalmann, "Enhancing naive bayes with various smoothing methods for short text classification," in *WWW 12 Companion*, ed. New York: ACM, 2012, pp.645-646.
- [5] Shruti Aggarwal, Devinder Kaur, "Naïve Bayes Classifier with Various Smoothing Techniques for Text Documents", *IJCTT*, 2013.
- [6] D.D. Arifin, Shaufiah, M.A. Bijaksana, "SMS Filtering Menggunakan Naive Bayes Classifier dan FP-Growth Algorithm Frequent Itemset", *e-Proceeding of Engineering*, vol.2, no.2, Agustus 2015.
- [7] Afrianto, Kurniawati, "Kategorisasi Dokumen Teks secara Multi Label Menggunakan Fuzzy C-Means dan K-Nearest Neighbors pada Artikel Berbahasa Indonesia", *JUTI*, 2013.
- [8] Tsoumakas, G., & Katakis, I., "Multilabel classification: An overview", *International Journal of Data Warehousing and Mining*, vol.3, no.3, pp. 1-13, 2007.
- [9] Zhihua Wei, Hongyun Zhang, Zhifei Zhang, Wen Li, Duoqian Miao. "A Naive Bayesian Multi-label Classification Algorithm With Application to Visualize Text Search Results". *International Journal of Advanced Intelligence*, vol.3, No.2, pp. 173-188, July 2011.
- [10] R. K. Hapsari, Y. J. Santoso, "Stemming Artikel Berbahasa Indonesia dengan Pendekatan *Confix-stripping*", in *Prosiding Seminar Nasional Manajemen Teknologi XXII*, ed. Surabaya: Program Studi MMT-ITS, pp.251-258, Jan 2015.
- [11] M.K. Dalal, M.A. Zaveri, "Automatic Text Classification: A Technical Review", *International Journal of Computer Applications*, vol.28, pp.0975-8887, Agst. 2011.
- [12] Gilang Jalu Selo W.T, Budi Susanto, Rosa Delima, "Implementasi Naïve Bayesian Classifier Untuk Kasus Filtering SMS Spam", *Informatika*, 2013.
- [13] Arifin, A.Z., I.P.A.K. Mahendra dan H.T. Ciptaningtyas, "Enhanced Confix Stripping Stemmer and Ants Algorithm for Classifying News Document in Indonesian Language". *ICTS*, 2009.
- [14] Karl-Michael Schneider, *Techniques for Improving the Performance of Naive Bayes for Text Classification*, citeseerx, 2005.

**Sholeh Rachmatullah** lahir di kota Pamekasan, Jawa Timur, pada tahun 1985. Menyelesaikan studi S1 di program studi Teknik Informatika Universitas Trunojo pada tahun 2007. Sholeh menyelesaikan studi masternya pada jurusan Pasca Sarjana Teknologi Informasi STTS pada tahun 2017. Sekarang sebagai Dosen program studi Teknik Informatika Universitas Madura. Minat penelitian adalah data mining, rekayasa perangkat lunak dan sistem informasi.

**Reddy Alexandro Harianto** lahir di Surabaya, Jawa Timur, Indonesia, pada tahun 1989. Dia menyelesaikan studi S1 di program studi Teknik Informatika Sekolah Tinggi Teknik Surabaya pada tahun 2011, dan menyelesaikan studi masternya pada jurusan Teknologi Informatika di Institut Teknologi Sepuluh Nopember Surabaya pada tahun 2015.

# Klasifikasi Citra Daun Memanfaatkan Angular Partition, Edge Detection dan Neural Network

Elkana Lewi Santoso, *STT Cahaya Surya Kediri*, Endang Setyati, *Teknologi Informasi iSTTS*, dan Yosi Kristian, *Informatika iSTTS*

**Abstrak**— Penelitian yang dilakukan ini adalah untuk mengklasifikasikan tanaman berdasarkan bentuk daunnya. Terdapat 1907 citra daun berasal dari 32 jenis tanaman yang berbeda. Citra daun tersebut diambil dari website flavia. Proses klasifikasi daun-daun dilakukan dengan menggunakan Neural Network dengan metode *Back Propagation Neural Network* (BPNN). Tahap awal adalah mengubah ukuran gambar yang didapat dari flavia 1600x1200 piksel menjadi berukuran 300x225 piksel, terdiri dari tahap membuat gambar dengan Sobel untuk mendeteksi tepi, membuat gambar dengan Sobel dan menambah kecerahan gambar, sehingga tepi dan serat daun dapat lebih jelas. Citra daun di potong-potong dengan *angular partition* (sudut pembagi)  $5^\circ$ ,  $11.25^\circ$ ,  $22.5^\circ$ ,  $30^\circ$ ,  $45^\circ$  dan  $90^\circ$ . Pembuatan data set dengan delapan ekstrasi fitur daun yang terdiri dari menghitung: jumlah piksel-piksel pada bagian daun, jumlah piksel pada keliling daun, jumlah sudut pada daun, jumlah piksel pada keliling dan serat daun serta menghitung jarak tiap-tiap piksel tersebut dengan *center of gravity* (titik berat). Pada penelitian ini dilakukan 100 macam variasi nilai hidden layer untuk tiap-tiap sudut pembagi tersebut. Hasil dari penelitian ini adalah: semakin kecil sudut pembagi ( $5^\circ$ ) dan semakin besar sudut pembagi ( $90^\circ$ ) tidak menghasilkan akurasi yang semakin bagus. Hasil akurasi tertinggi yang diraih sebesar 96,7488%. Hasil tersebut didapat dengan membagi daun menjadi sudut  $11,25^\circ$  tiap bagian (16 bagian), Susunan BPNN yang digunakan terdiri dari 1 input, 1 output, 3 hidden layer, tiap hidden layer berisi 128, 120, 112 neurons, dan 32 neurons untuk output layer.

**Kata Kunci**—Angular Partition, Back Propagation, Center of Gravity, Neural Network.

## I. PENDAHULUAN

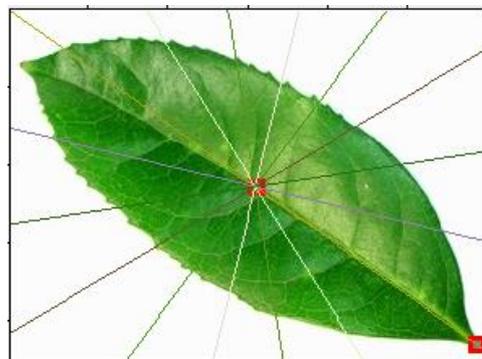
Tanaman adalah makhluk hidup yang sangat berperan penting dalam ekosistem. Tanpa tanaman susunan ekosistem pasti akan terganggu dan besar kemungkinan akan musnah. Hal ini dikarenakan tanaman berfungsi sebagai bahan makanan, obat-obatan, bio energy, dan juga sebagai penyeimbang alam karena tanaman juga menghasilkan gas  $O_2$  dan  $CO_2$  selain itu tanaman juga berfungsi sebagai penahan resapan air didalam tanah. Fungsi tanaman sangat penting, tanpa tanaman maka tidak akan ada kehidupan. Ada sekitar 350.000 jenis tanaman di bumi yang telah dikenal. Indonesia terletak didaerah khatulistiwa sehingga mendapat

sinar matahari sepanjang tahun dan curah hujan yang tinggi hal ini sangat menunjang untuk kelangsungan hidup flora dan fauna.

Dengan semakin berkembangnya teknologi dibidang komputer, sehingga komputer mampu mengerjakan tugas-tugas secara cepat dan simultan. Salah satu perkembangan computer dewasa ini adalah dalam bidang pengenalan obyek (*object Recognition*). Dalam penelitian ini akan diterapkan teknologi pengenalan terhadap obyek daun. Daun adalah salah satu unsur yang penting untuk mengenali species tanaman[1]. Banyaknya jenis tumbuhan dapat dilihat dengan bentuk daun yang berbeda sebagai ciri yang mewakili jenis tanaman. Sehingga tidak mungkin untuk dihafalkan, oleh karena itu diperlukan suatu aplikasi cerdas yang dapat mengenali berbagai berbagai macam daun. Sehingga diharapkan dapat membantu para ahli biologi, pelajar/mahasiswa dan peneliti lain untuk dapat mengklasifikasikan tanaman berdasarkan bentuk daun.

Dalam penelitian ini, digunakan metode Anggular Partition[2] untuk membentuk ekstraksi karakteristik daun dan *Back Propagation Neural Network* (BPNN) untuk mengklasifikasikan daun berdasarkan karakteristik bentuk daun.

Angular partition yang digunakan pada penelitian ini sebesar  $5^\circ$ ,  $11.25^\circ$ ,  $22.5^\circ$ ,  $30^\circ$ ,  $45^\circ$  dan  $90^\circ$ . Hal ini dilakukan untuk mengetahui berapa besar sudut yang sesuai dan menghasilkan akurasi yang paling besar.



Gambar 1. Daun dengan Angular Partition  $22.5^\circ$

## II. PENELITIAN PENUNJANG

Bab ini membahas tentang teori-teori penunjang yang dipergunakan dalam penyelesaian penelitian. Teori penunjang ini mencakup penjelasan dari beberapa jurnal

Elkana Lewi Santoso, *Pengajar di STT Cahaya Surya Kediri*, Jawa Timur, Indonesia (e-mail: elkanaliu@cahayasurya.ac.id)

Endang Setyati, *Teknologi Informasi Institut Sains dan Teknologi Terpadu Surabaya*, Surabaya, Jawa Timur, Indonesia (e-mail: endang@stts.edu)

Yosi Kristian, *Informatika Institut Sains dan Teknologi Terpadu Surabaya*, Surabaya, Jawa Timur, Indonesia (email: yosi@stts.edu)

ilmiah yang diambil untuk membandingkan dan menunjang penelitian ini beserta teori umum seperti biner, grayscale, sobel, angular partition, dan BPNN

Pada tahun 2004 Abdolah Chalechale[3], Golshah Naghdy, Alfred Mertins telah meneliti metode untuk angular partition pada sebuah image abstrak, Data set yang mereka gunakan dalam percobaan yaitu sebanyak 365 image full color dengan ukuran yang bervariasi dan 180 dari image sketsa hitam putih sebagian untuk dijadikan data uji dengan dilakukan berbagai rotasi  $90^0$ ,  $180^0$ , dan  $270^0$ . Data tersebut mereka dapatkan dari World Art Kiosk at California State University.

Pada tahun 2007 Stephen Gang Wu, Forrest Sheng Bao, Eric You Xu, Yu-Xuan Wang, Yi-Fan Chang, Qiao-Liang Xiang menemukan cara mengklasifikasikan daun dengan metode Probabilitas Neural Network (PNN). Penelitian ini dimuat juga pada laman Flavia yang menyediakan kumpulan gambar daun berjumlah 1907 daun yang terdiri dari 32 klas[1]. Dari gambar daun tersebut dilakukan penelitian ini.

Pada tahun 2007 Johann Misterio, Krshna Ravindra, Rene D Rivero, Henry McCloud, Leville Burr-Alexander, Nuggehalli Ravindra meneliti tentang dimensi fraktal pada daun [4] yaitu suatu bentuk yang sama yang bisa berulang pada suatu dimensi pada daun. Dari penelitian ini maka dapat disimpulkan bahwa data perbandingan dari jumlah piksel tiap potongan dengan total semua piksel dari daun yang disebut normalisasi bisa dilakukan pada data set.

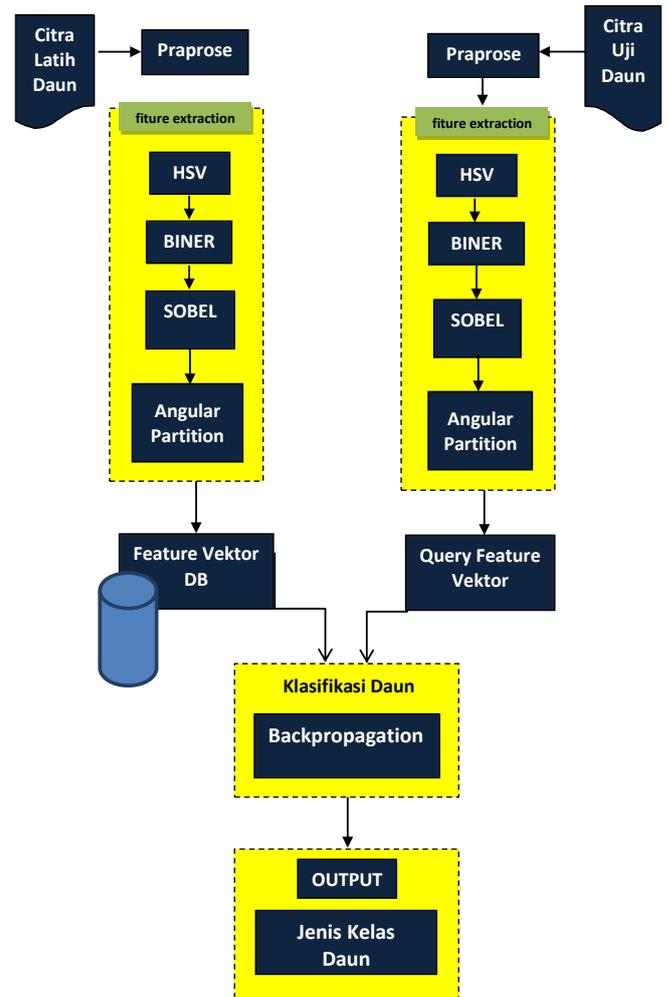
Pada tahun 2011 Dimpy Adira Ratu [11], meneliti tentang pengenalan tanaman obat-obatan dengan menggunakan box clustering dan *Probabilistic Neural Network* (PNN)[1][10][11], Box Clustering digunakan untuk pembentukan data set. sedangkan dalam penelitian ini menggunakan Angular Partition.

Pada tahun 2014 sebuah penelitian yang diterbitkan semantic Scholar yang ditulis oleh Kesari Verma, Ligendra Kumar Verma, dan Priyanka Tripathi menemukan BPA (*Back Propagation Algorithm*) adalah salah satu teknik kategorisasi image yang paling sesuai, karena semua vektor fitur tersedia dalam bentuk numerik. eksperimen yang telah dilakukan menemukan bahwa ketepatan klasifikasi juga bergantung pada hidden layer dan epoches [12].

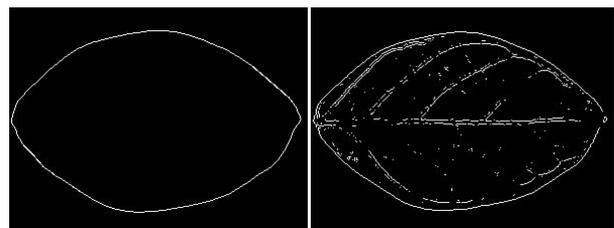
Dari penelitian-penelitian tersebut maka pada penelitian ini dilakukan penelitian klasifikasi citra daun dengan pembagian sudut pada struktur permukaan citra daun dan metode backpropagation untuk proses pengklasifikasian citra daun tersebut. Citra daun berasal dari flavia web yang berbentuk jpg dan berukuran 1600(p) x 1200 (L) terdiri dari 3 layer RGB. Citra tersebut dikecilkan menjadi berukuran 300x225 tetap berupa RGB. Dari citra hasil resize tersebut dilakukan preprocessing untuk membentuk citra dengan tampak tepi luar daun dan citra dengan tampak tepi luar dan serat daun seperti yang terlihat pada Gambar 3.

Pada Gambar 2 adalah gambar dari rancangan sistem yang digunakan. Hasil dari ekstraksi fitur dari citra daun disimpan dalam bentuk feature vektor untuk kemudian dilakukan proses klasifikasi dengan persentasi data 70% untuk training, 15% untuk testing dan 15% untuk validasi data. Validasi data digunakan untuk memvalidasi bahwa jaringan itu berjalan normal dan untuk menghentikan pelatihan sebelum terjadi overfitting [5]. Dari pembelajaran klasifikasi citra daun pada neural network, didapat simulasi

neural network yang dapat digunakan untuk mengklasifikasikan citra daun lain sebagai citra uji.



Gambar 2. Rancangan Sistem



Gambar 3. Citra Tepi Luar dan Serat Daun.

### III. METODOLOGI

#### A. DATA SET

Data citra daun diambil dari situs flavia sebanyak 1907 citra yang terdiri dari 32 jenis tanaman seperti yang terdapat pada Tabel I dan akan diresize menjadi 300x225 piksel, Pada semua citra daun tersebut dilakukan proses untuk membuat data set:

1. HSV[9] untuk membuat deteksi tepi agar lebih terlihat jelas.
2. Biner adalah proses untuk menghasilkan citra hitam dan putih.
3. Sobel untuk mendeteksi tepi daun dan serat daun.
4. Angular partition sebesar  $5^{\circ}$ ,  $11.25^{\circ}$ ,  $22.5^{\circ}$ ,  $30^{\circ}$ ,  $45^{\circ}$ ,  $90^{\circ}$ .

TABEL II  
DAFTAR DAUN dari 32 JENIS TANAMAN

No.	Common Name(s)	Jumlah
1	pubescent bamboo	59
2	Chinese horse chestnut	63
3	Chinese redbud	72
4	true indigo	73
5	Japanese maple	56
6	Nanmu	62
7	castor aralia	52
8	goldenrain tree	59
9	Chinese cinnamon	55
10	Anhui Barberry	65
11	Big-fruited Holly	50
12	Japanese cheesewood	63
13	wintersweet	52
14	camphortree	65
15	Japan Arrowwood	60
16	sweet osmanthus	56
17	deodar	77
18	ginkgo, maidenhair tree	62
19	Crape myrtle, Crepe myrtle	61
20	oleander	66
21	yew plum pine	60
22	Japanese Flowering Cherry	55
23	Glossy Privet	55
24	Chinese Toon	65
25	peach	54
26	Ford Woodlotus	52
27	trident maple	53
28	Beale's barberry	55
29	southern magnolia	57
30	Canadian poplar	64
31	Chinese tulip tree	53
32	tangerine	56

Pada penelitian ini digunakan 8 *shape feature extraction* (ekstraksi feature permukaan) citra daun yaitu:

1. Jumlah titik piksel pada satu area partition.
2. jarak tiap titik pada area partition dengan titik imbang.
3. Jumlah titik piksel pada dektesi tepi disuatu area.
4. Jarak tiap titik piksel pada dektesi tepi dengan titik berat di satu area.
5. Jumlah titik piksel pada dektesi tepi dan serat daun di satu area partisi.
6. jarak tiap titik piksel pada dektesi tepi dan serat daun dengan titik imbang.
7. Jumlah titik corner/sudut pada satu area partisi.
8. Jarak titik corner/ sudut dengan titik imbang di satu partisi.

Feature vektor extractio tersebut disimpan kedalam suatu file untuk proses training dan testing BPNN.

### B. ANGULAR PARTITION (AP)

Angular partition adalah membagi citra daun dengan sudut tertentu, titik pusat yang dipakai adalah titik berat dari citra daun seperti pada Gambar 1.

Pada penelitian ini digunakan 8 ekstraksi fiture yaitu menghitung jumlah piksel pada luas, keliling, serat, jumlah gerigi serta jarak semua piksel-piksel tersebut dengan titik berat citra daun. Data set dan feature vektor yang dihasilkan dengan 8 ekstraksi feature tersebut tampak pada Tabel II.

$$y_{total} = \frac{\sum_{i=1}^{225} (\sum_{k=1}^{300} \text{jumlah titik pada } (i,k)) * i}{total \text{ isi}} \quad (1)$$

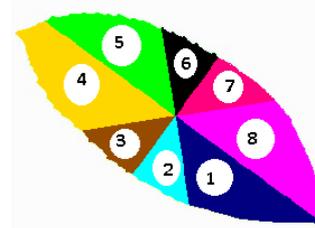
$$x_{total} = \frac{\sum_{i=1}^{300} (\sum_{k=1}^{225} \text{jumlah titik pada } (k,i)) * i}{total \text{ isi}} \quad (2)$$

Pada persamaan (1) dan (2)  $x_{total}$  adalah posisi titik berat pada sumbu X,  $y_{total}$  adalah posisi titik berat pada sumbu Y, sedangkan totalisi adalah jumlah piksel pada citra daun. Persamaan untuk mencari jarak diantara dua titik:

$$r = \sqrt{(x_{total} - x_2)^2 + (y_{total} - y_2)^2} \quad (3)$$

Pada persamaan (3) r adalah jarak antara ( $x_{total}, y_{total}$ ) dengan titik ( $x_2, y_2$ ),  $x_2$  adalah nilai sumbu X pada titik yang dicari,  $y_2$  adalah nilai sumbu Y pada titik yang dicari,  $x_{total}$  dan  $y_{total}$  adalah titik dari titik imbang.

Potongan pertama adalah potongan yang berasal dari garis yang dibuat dengan mencari titik terjauh citra daun dengan titik berat dari citra dan berputar searah jarum jam terlihat pada Gambar 4.



Gambar 4. Urutan Potongan Citra Daun dengan Angular Partition 45°

Ekstraksi fiture daun dengan menggunakan angular partition ini memiliki kelebihan diantaranya adalah citra daun yang hendak dipakai untuk pengujian dapat di flip, diresize dan dirotasi, masih dapat diklasifikasikan dengan benar.

Pada penelitian ini hanya menggunakan shape feature extraction, tidak menambah dengan texture dan color extraction. Sehingga proses learning machine untuk mengklasifikasi citra daun yang dilakukan oleh komputer dirasa cepat dan ringan.

TABEL II  
PEMBAGIAN DATA SET & FEATURE VECTOR

SUDUT PEMBAGI	DATA SET (8 EKSTRAKSI FEATURE)	FEATURE VEKTOR (1907 citra)
5	72*8=576	1.098.432
11.5	32*8=256	488.192
22.5	16*8=128	244.096
30	12*8=96	183.072
45	8*8=64	122.048
90	4*8=32	61.024

### C. BACKPROPAGATION NEURAL NETWORK

BackPropagation Neural Network adalah meliputi proses forward pass dan backward pass. Sebuah input akan dimasukkan pada layer neural network secara forward untuk menghasilkan suatu target yang telah ditetapkan, hal ini memunculkan suatu nilai error loss jika target tersebut tidak terpenuhi, dilakukan back pass kedalam layer neural network kembali dengan merubah gradien untuk menghasilkan error lost terkecil. Metode pelatihan yang diawasi (*supervised learning*), dalam pengertian pelatihan ini mempunyai target yang akan dihasilkan. Ciri dari Backpropagation adalah meminimalkan error pada output yang dihasilkan oleh jaringan dengan cara mengubah



Gambar 6. Susunan Hidden Layer: (a) untuk AP 5°, (b) untuk AP 11.25°, (c) untuk AP 22.5°, (d) untuk AP 30°, (e) untuk AP 45°, (f) untuk AP 90°

gradien. Pada metode backpropagation, seringkali mempunyai susunan dengan jaringan multilayer.[7]

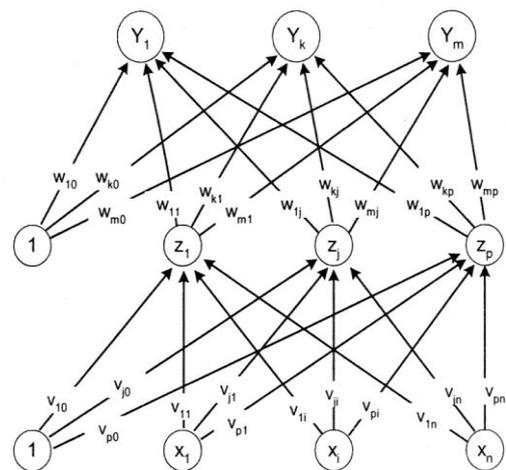
Untuk mencari nilai akurasi tertinggi tidak ada rumusan tentang berapa jumlah hidden layer serta jumlah neuron yang terdapat pada tiap multilayer, sehingga pada dilakukan eksperimen 100 variasi susunan layer neural network untuk setiap angular partition.

Training yang digunakan adalah SCG (Scaled Conjugate Gradient) yang berfungsi untuk memperbarui bobot dan meminimalkan *error* yang terjadi. Inisialisasi scg ini berupa 400 epoch, 287 max-fail, minimum gradien 1e-3.

**D. Perbandingan Waktu Training Data.**

Penelitian ini menghasilkan data feature vector dari 1907 citra daun dan dilanjutkan kedalam proses pembelajaran untuk mengklasifikasikan citra daun dengan BPNN maka didapat akurasi tertinggi untuk angular partition (AP) 11.25° dan 22.5° adalah sama, sebesar 96,75%.

Dalam pengujian waktu tercepat yang dimulai saat program dijalankan didapat bahwa untuk sudut 11.25° (±86 detik) adalah yang terbaik dan tercepat dibandingkan menggunakan angular partition 22.5° (±100 detik) seperti yang terlihat pada Tabel III.



Gambar 5. BackPropagation Neural Network

Perhitungan waktu tersebut untuk mengetahui waktu training mana yang tercepat, waktu yang dihasilkan sangat tergantung dari spesifikasi hardware, software, besar AP yg ditrainningkan, dan jumlah neuron didalam NN yang berjalan saat proses.

TABEL III  
AKURASI dan WAKTU

SUDUT PEMBAGI	AKURASI	WAKTU (DETIK)
5°	96,49%	±131
11.25°	96,75%	±86
22.5°	96,75%	±100
30°	95,75%	±73
45°	94,76%	±74
90°	82,28	±81.2

Perhitungan akurasi didapat dengan persamaan 4, TP (true positif) adalah jumlah citra yang diklasifikasikan dengan benar,

$$AKURASI = \frac{TP}{\text{jumlah citra daun}} * 100\% \quad (4)$$

Pada Gambar 6, tampak jumlah hidden layer dan jumlah Susunan Neural Network yang digunakan untuk training, validating dan testing beserta jumlah neuron didalamnya untuk tiap angular partition

TABEL IV  
JUMLAH CITRA DAUN YANG DIKLASIFIKASI DENGAN BENAR

No.	Common Name(s)	Jumlah Daun	Angular Partitition yang digunakan					
			5	11.25	22,5	30	45	90
1	pubescent bamboo	59	59	59	59	58	57	52
2	Chinese horse chestnut	63	61	60	59	59	59	51
3	Chinese redbud	72	71	71	71	69	71	60
4	true indigo	73	71	73	73	72	73	71
5	Japanese maple	56	52	52	55	53	54	48
6	Nanmu	62	58	57	56	60	59	51
7	castor aralia	52	51	50	48	52	49	43
8	goldenrain tree	59	59	59	58	58	58	54
9	Chinese cinnamon	55	54	49	49	49	46	40
10	Anhui Barberry	65	62	65	63	62	61	53
11	Big-fruited Holly	50	48	46	49	46	42	39
12	Japanese cheesewood	63	61	63	61	63	56	54
13	wintersweet	52	48	50	47	49	51	43
14	camphortree	65	63	62	63	63	57	46
15	Japan Arrowwood	60	53	58	57	54	57	38
16	sweet osmanthus	56	56	52	55	55	52	39
17	deodar	77	77	77	77	77	76	69
18	ginkgo, maidenhair tree	62	60	61	62	60	61	53
19	Crape myrtle	61	61	59	60	58	60	47
20	oleander	66	64	65	66	64	63	55
21	yew plum pine	60	59	60	58	59	59	51
22	Japanese Flowering Cherry	55	53	52	55	52	54	51
23	Glossy Privet	55	51	54	54	53	53	41
24	Chinese Toon	65	64	63	63	63	64	58
25	peach	54	48	51	51	45	44	34
26	Ford Woodlotus	52	52	50	48	47	47	32
27	trident maple	53	51	50	50	52	49	48
28	Beale's barberry	55	50	52	55	53	55	51
29	southern magnolia	57	56	56	56	54	53	52
30	Canadian poplar	64	62	64	62	63	63	58
31	Chinese tulip tree	53	50	50	51	48	52	42
32	tangerine	56	55	55	54	56	52	45

IV. PENGUJIAN

Dalam bagian ini dilakukan beberapa macam pengujian terhadap hasil training.

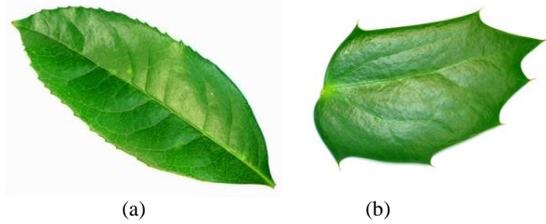
Pada Tabel 4 terdapat data jumlah daun yang dapat dikenali untuk tiap jenis beserta angular partition yang digunakan untuk membentuk ekstraksi fitur.

Pada proses pengujian ini dilakukan beberapa modifikasi citra daun. citra daun yang dimodifikasi adalah citra daun yang awalnya dapat diklasifikasikan dengan benar.

Modifikasi citra daun tersebut meliputi horisontal flip,

vertikal flip, dan rotasi image.

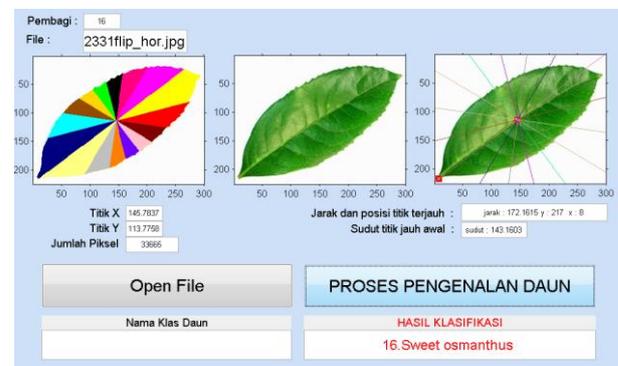
Citra daun yang diambil untuk pengujian terdapat pada Gambar 7.



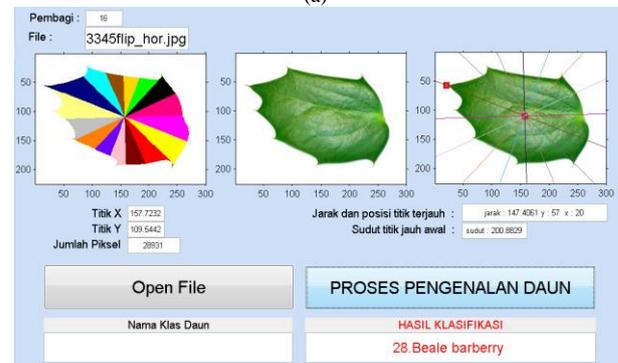
Gambar 7. Citra Awal  
(a) 2331 Sweet Osmanthus, (b) 3345 Beale Barberry.

A. Horisontal Flip.

Gambar 7(a) dan 7(b) direfleksikan pada sumbu Y, dengan cara mengubah mengubah setiap koordinat piksel citra f(x,y) menjadi f(-x,y).



(a)



(b)

Gambar 8. Hasil Klasifikasi BPNN Citra Daun di Flip Horisontal  
(a) 2331 Sweet Osmanthus, (b)3345 Beale Barberry.

Gambar 8(a) adalah pengujian untuk gambar 7(a) yang telah dilakukan proses flip secara horisontal, hasil pengujian tampak bahwa Citra daun 2331 Sweet Osmanthus yang telah di flip horisontal masih dapat diklasifikasikan dengan benar.

Gambar 8(b) adalah pengujian untuk gambar 7(b) yang telah dilakukan proses flip secara horisontal, hasil pengujian tampak bahwa Citra daun 3345 Beale Barberry yang telah di flip horisontal masih dapat diklasifikasikan dengan benar.

B. Vertikal Flip

Gambar 7(a) dan 7(b) direfleksikan pada sumbu X, dengan cara mengubah koordinat setiap piksel pada citra f(x,y) menjadi f(x,-y).

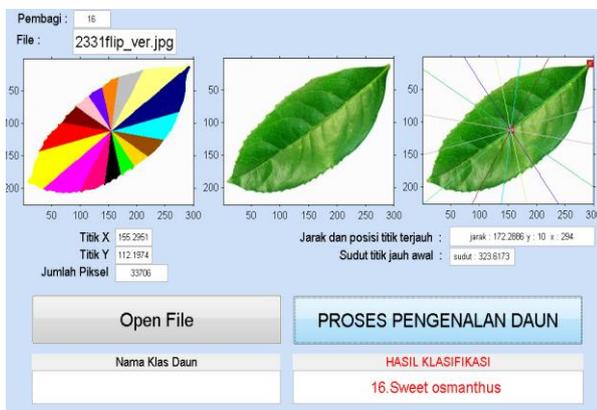
Gambar 9(a) adalah pengujian untuk gambar 7(a) yang telah dilakukan proses flip secara vertikal, hasil pengujian tampak bahwa Citra daun 2331 Sweet Osmanthus yang telah di flip vertikal masih dapat diklasifikasikan dengan benar.

Gambar 9(b) adalah pengujian untuk gambar 7(b) yang telah dilakukan proses flip secara vertikal, hasil pengujian tampak bahwa Citra daun 3345 Beale Barberry yang telah di flip vertikal masih dapat diklasifikasikan dengan benar.

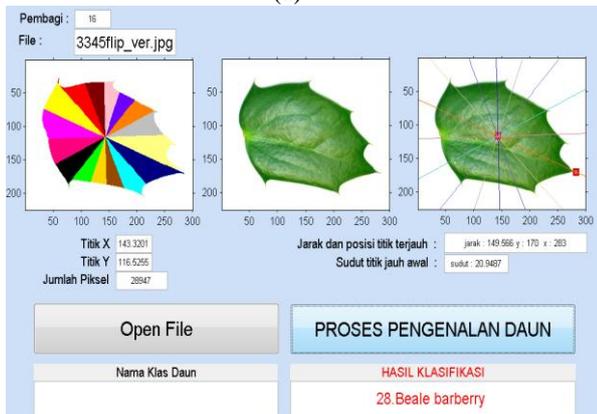
**C. Rotasi 90° searah jarum jam (kekanan)**

Gambar 7(a) dan 7(b) dirotasikan 90° kekanan dengan ms paint, lalu dilakukan proses pengklasifikasian dengan BPNN.

Gambar 10(a) adalah pengujian untuk gambar 7(a) yang telah dilakukan proses rotasi kekanan 90°, hasil pengujian tampak bahwa Citra daun 2331 Sweet Osmanthus yang telah di rotasi 90° kekanan masih dapat diklasifikasikan dengan benar.



(a)



(b)

Gambar 9. Hasil Klasifikasi Citra Daun di Flip Vertikal (a) 2331 Sweet Osmanthus, (b)3345 Beale Barberry

Gambar 10(b) adalah pengujian untuk gambar 7(b) yang telah dilakukan proses rotasi 90° kekanan, hasil pengujian tampak bahwa Citra daun 3345 Beale Barberry yang telah di rotasi 90° kekanan masih dapat diklasifikasikan dengan benar.

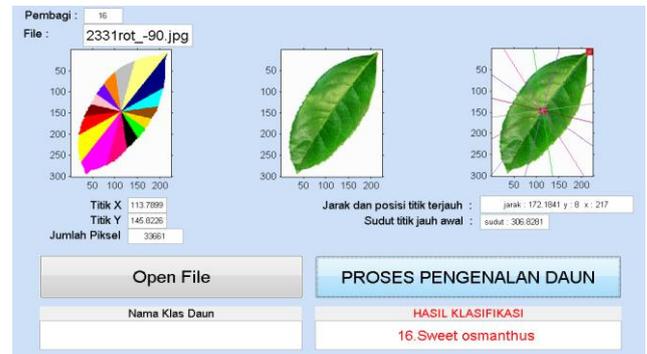
**D. Rotasi 90° berlawanan arah jarum jam (kekiri)**

Gambar 7(a) dan 7(b) dirotasikan 90° kekiri dengan ms paint, lalu dilakukan proses pengklasifikasian dengan BPNN.

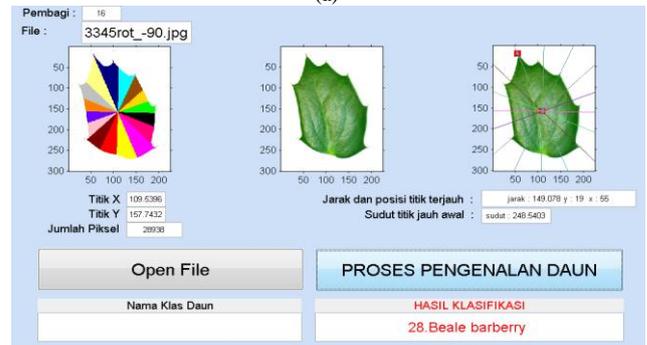
Gambar 11(a) adalah pengujian untuk gambar 7(a) yang

telah dilakukan proses rotasi kekiri 90°, hasil pengujian tampak bahwa Citra daun 2331 Sweet Osmanthus yang telah di rotasi 90° kekiri masih dapat diklasifikasikan dengan benar.

Gambar 11(b) adalah pengujian untuk gambar 7(b) yang telah dilakukan proses rotasi 90° kekiri, hasil pengujian tampak bahwa Citra daun 3345 Beale Barberry yang telah di rotasi 90° kekiri masih dapat diklasifikasikan dengan benar.



(a)



(b)

Gambar 11. Hasil Klasifikasi BPNN Citra Daun di Rotasi 90° Kekiri (a) 2331 Sweet Osmanthus, (b)3345 Beale Barberry

**E. Pengujian dengan Meresize Citra Daun**

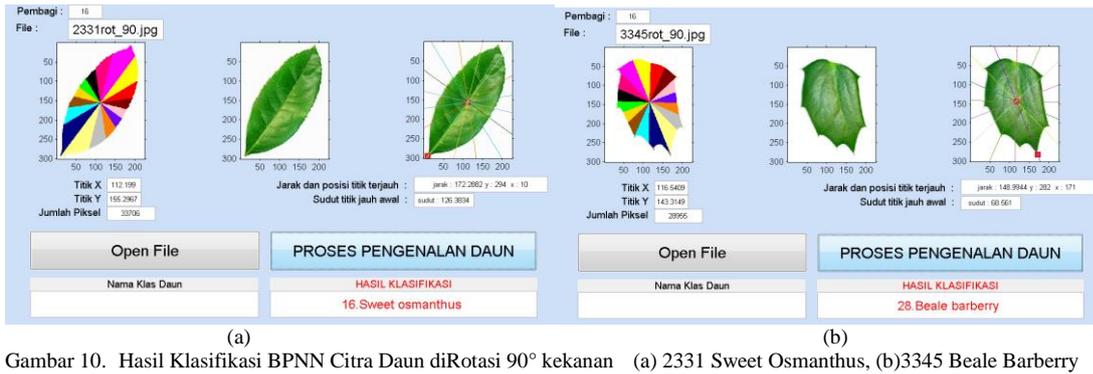
Pengujian ini dilakukan dengan mengecilkan citra daun menjadi 50% dan membesarkan citra daun sebesar 150% dari citra asal. Pada Gambar 12(a), (b),(c) dan (d) terlihat bahwa dengan angular partition untuk membuat dataset, citra daun masih dapat diklasifikasikan dengan benar.

TABEL V  
PERBANDINGAN AKURASI

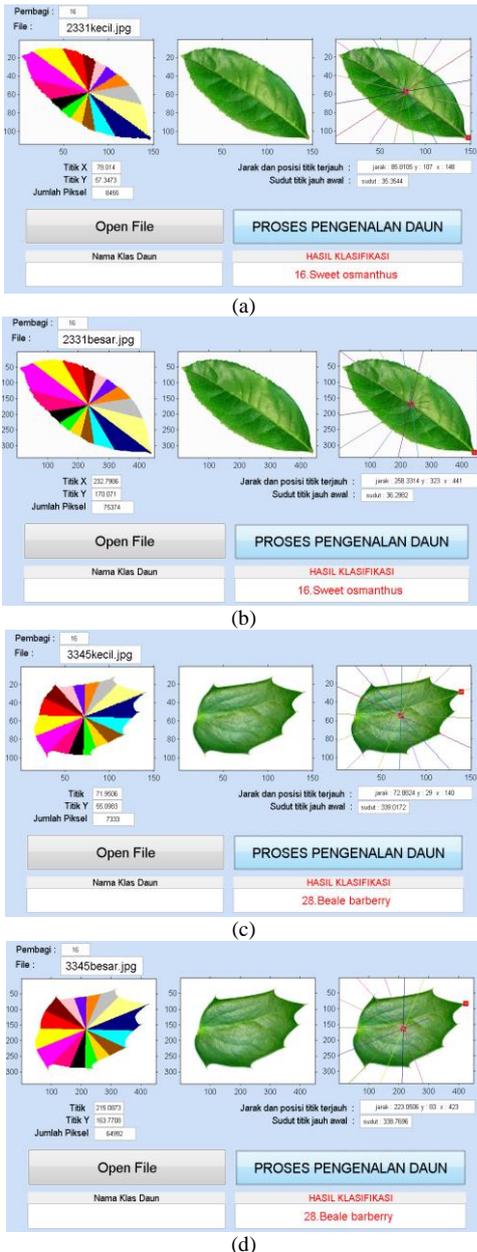
METODE	AKURASI
PNN [1]	90%
PNN [10]	93.75%
BPNN [13] (swedish leaf dataset)	96.53%
PNN [11] (600 citra tumbuhan obat)	86.19%
Feed Forward NN [6]	71.4%
Penelitian ini	96,75%

TABEL VI  
AKURASI ANTAR AP

SUDUT	AKURASI
5	96,49%
11.25	96,75%
22.5	96,75%
30	95,75%
45	94,76%
90	82,28



Gambar 10. Hasil Klasifikasi BPNN Citra Daun diRotasi 90° kekanan (a) 2331 Sweet Osmanthus, (b)3345 Beale Barberry



Gambar 12. Hasil Klasifikasi BPNN Citra Daun diresize. (a) 2331 dikecilkan 50%, (b) 2331 diperbesar 150% (c)3345 dikecilkan 50%, (d) 3345 diperbesar 150%

### V. KESIMPULAN

Kesimpulan yang didapat dari penelitian ini adalah

- Metode BackPropagation Neural Network dapat dipakai untuk mengklasifikasikan citra daun dengan tingkat

akurasi yang tinggi yaitu sebesar 96,75% dengan Angular Partition sebesar 11,25° dan 22,5°.

- Ekstraksi feature dengan Angular Partition memiliki kelebihan yaitu citra dapat di flip secara horisontal maupun vertikal, dapat dirotasi kekanan 90 maupun ke kiri 90, serta dapat diresize baik dikecilkan 50% maupun diperbesar 150%. Seperti yang tampak pada Gambar 8,9,10,11, dan 12
- Pada Tabel 5 tampak bahwa semakin besar angular partition menyebabkan ciri-ciri dari citra daun menjadi bias, sedangkan semakin kecil angular partition, ciri-ciri dari citra daun semakin tidak terlihat.

### DAFTAR PUSTAKA

[1]S. G. Wu, F. S. Bao, E. Y. Xu, Y. X. Wang, Y. F. Chang, and Q. L. Xiang, "A leaf recognition algorithm for plant classification using probabilistic neural network," *ISSPIT 2007 - 2007 IEEE Int. Symp. Signal Process. Inf. Technol.*, pp. 11–16, 2007

[2]A. Chalechale, G. Naghdy, and A. Mertins, "Sketch-based image matching using angular partitioning," *IEEE Trans. Syst. Man, Cybern. Part A Systems Humans.*, vol. 35, no. 1, pp. 28–41, 2005.

[3]N. (New J. I. of T. Ravindra, L. (New J. I. of T. Buss-Alexander, H. (New J. I. of T. McCloud, R. D. (New J. I. of T. Rivero, K. (Johns H. U. Ravindra, and J. (William D. H. S. Misterio, "Variation of Fractal Dimension of Leaves Based on Stem Position," p. 12, 2007.

[4] Boran Şekeröğlü, Yücel İnan, "Leaf Recognition System Using Neural Network, *Procedia Computer Science*, vol. 10, pp. 578-582, August, 2016

[5] Y. LeCun *et al.*, "Backpropagation Applied to Handwritten Zip Code Recognition," *Neural Computation*, Vol. 1, No. 4, hal. 541–551, 1989.

[6] Y. Ye, C. Chen, C.-T. Li, H. Fu, and Z. Chi, "A computerized plant species recognition system," in *Proceedings of 2004 International Symposium on Intelligent Multimedia, Video and Speech Processing*, Hong Kong, October 2004.

[7] Aurélien Géron, "Hands-On Machine Learning With Scikit-Learn & TensorFlow", 1<sup>st</sup> Edition, O'reilly, US, 2017

[8] Simon Haykin, "Neural Network A Comprehensive Foundation", 2<sup>nd</sup> Edition, Prentice Hall, 2001.

[9] B. S. Bama, S. M. Valli, S. Raju, and V. A. Kumar, "Content based leaf image retrieval using shape, color and texture features", *Indian Journal of Computer Science and Engineering*, vol. 2, no. 2, 2011, pp. 202-211

[10] A.Kadir,Lukito E.N, Adhi N, "Leaf Classification Using Shape, Color, and Texture Features", *International Journal of Computer Trends and Technology*, July to Aug, 2011.

[11] D. A. Ratu, "Ekstraksi Daun Menggunakan Dimensi Fraktal untuk Identifikasi Tumbuhan Obat di Indonesia." *Dep. ILMU Komput. Fak. Mat. DAN ILMU Pengetah. ALAM Inst. Pertan. BOGOR*, 2011.

[12] K. Verma, L. K. Verma, and P. Tripathi, "Image Classification using Backpropagation Algorithm," *Journal of Computer Science*, July 2014.

[13] I. Yahiaoui and A. Verroust-blondet, "A shape-based approach for leaf classification using multiscale triangular representation To cite this version: A shape-based approach for leaf classification using multiscale triangular representation.," 2013.

Elkana Lewi Santoso ([elkanaliu@gmail.com](mailto:elkanaliu@gmail.com)) lahir di Surabaya tahun 1973, Jawa Timur, Indonesia. Lulus S1 di Sekolah Tinggi Teknologi Cahaya Surya Kediri tahun 2000. Aktif Mengajar di STT Cahaya Surya Kediri tahun 2000-sekarang. Fokus pada bidang Networking, Operating System, Neural Network.

# Game Battle of Artificial Intelligence Berbasis Android

Riandika Lumaris, *Informatika Institut Sains dan Teknologi Terpadu Surabaya*,  
Hendrawan Armanto, *Informatika Institut Sains dan Teknologi Terpadu Surabaya*,  
dan Willy Adrianus, *Informatika Institut Sains dan Teknologi Terpadu Surabaya*

**Abstrak** — Dengan berkembangnya permainan dan artificial intelligence saat ini, maka dibuat sebuah permainan Battle of Artificial Intelligence. Game ini merupakan game berbasis Android dan merupakan permainan pertarungan antar robot, tetapi robot yang dijalankan berdasarkan pada sistem artificial intelligence yang dibuat oleh pemain. Dalam konsep ini pemain dapat membuat sebuah sistem Artificial Intelligence yang digunakan pada sebuah robot, dan robot tersebut akan digunakan untuk bertarung dengan robot musuh. Permainan ini merupakan permainan strategi dimana pemain diajak untuk memikirkan cara memenangkan pertempuran, yaitu dengan memilih prioritas dan membuat sistem AI yang berbeda. Pemain dapat mengembangkan, mengubah sebuah sistem Artificial Intelligence sesuai dengan yang pemain inginkan. Tidak hanya itu, pengguna juga dapat mengembangkan sistem AI yang dibuat dengan cara melakukan pertempuran dengan pemain lain secara online. Pemain dapat mengetahui sejauh mana sistem Artificial Intelligence yang dibuat mampu bertahan. Dari 22 pengguna yang sudah melakukan uji coba permainan dan mengisi kuisioner diperoleh bahwa konsep permainan ini menarik. Berdasarkan pemahaman tutorial, pengguna dapat menambah pengetahuan pengguna mengenai bagaimana artificial intelligence bekerja. Hampir 60% pengguna menyukai interface yang dibuat di dalam game. Pengguna juga dapat menambah pengetahuan mengenai pembuatan sistem artificial intelligence pada permainan ini berdasarkan hasil pemahaman sistem artificial intelligence. Dengan adanya multiplayer, pengguna termotivasi untuk mengembangkan sistem artificial intelligence dengan bertanding dengan pemain lain secara online. Rating rata-rata yang diperoleh dari 22 pengguna yang telah melakukan uji coba adalah 3.45.

**Kata Kunci** — Android, Artificial Intelligence, Artificial Intelligence for Games, Game, Strategy

## I. PENDAHULUAN

Tactical video game adalah genre game yang mengutamakan perencanaan dan pemikiran yang mahir untuk dapat memenangkan permainan. Biasanya tactical video game ini menggunakan lawan bermain komputer yaitu Artificial Intelligence yang disediakan oleh game. Pada permainan ini, pemain juga membuat artificial intelligence

yang digunakan dalam pertarungan Artificial Intelligence. Artificial Intelligence for Games merupakan bagian atau kunci dari permainan ini.

Dengan adanya permainan ini maka pemain dapat mengetahui bagaimana Artificial Intelligence dapat bekerja dan bagaimana membuat sistem Artificial Intelligence itu sendiri dalam bentuk game. Permainan ini adalah permainan berbasis dari aplikasi game pada mobile device yang menggunakan sistem operasi Android dan merupakan permainan pertarungan antar robot dalam suatu arena yaitu antara robot pemain dan robot musuh. Pemain membuat sistem AI untuk robot yang mereka pertarungkan. Sistem AI yang dibuat oleh pemain dapat menjalankan robot. Tidak hanya itu pemain dapat membuat lebih dari satu sistem AI dan dapat diubah sesuai kebutuhan pemain. Fitur-fitur yang tersedia dalam permainan ini yaitu Sistem AI, Storymode, AI Editor, Tutorial, Battle Controls, Ally and Enemy, Difficulty Level, Achievement, Interaktif UI AI Editor, Multiplayer and Ranking Battle.

## II. TINJAUAN PUSTAKA

Penelitian ini menggunakan beberapa tinjauan pustaka sebagai dasar agar penelitian dapat berjalan sesuai arah yang tepat. Tinjauan pustaka yang digunakan antara lain:

### A. Game

Game merupakan suatu metode bermain yang digunakan untuk kesenangan ataupun sebagai alat pembelajaran (pendidikan) [1]. Seiring perkembangan teknologi, tantangan dari game terus ditingkatkan dengan memberikan gameplay baru ataupun fitur baru. Dan pada saat yang sama, game tersebut harus tetap menarik untuk dimainkan dan tidak memberi tantangan yang mustahil untuk dimainkan. Karena itu seiring dengan perkembangan teknologi, variasi game menjadi luas dan dikategorikan. Salah satu perkembangan game yang hingga saat ini terus berkembang adalah video games.

### B. Game Strategy

Game strategy merupakan game yang berfokus pada pikiran dan rencana untuk mencapai kemenangan. Game strategy menekankan pada strategi yang dipakai, tactic dan logika. Pada game jenis ini, pemain harus merencanakan tindakan yang akan dilakukan berikutnya terhadap lawan atau tujuan untuk mencapai kemenangan pada game tersebut. Pada sebagian besar permainan genre strategy, pemain diberikan pandangan seperti ahli strategi yang mengontrol semua unit di permainan dan dikendalikan langsung oleh pemain.

Riandika Lumaris, Departemen Informatika, *Institut Sains dan Teknologi Terpadu Surabaya*, Jawa Timur, Indonesia (e-mail: riandika@stts.edu)

Hendrawan Armanto, Departemen Informatika, *Institut Sains dan Teknologi Terpadu Surabaya*, Jawa Timur, Indonesia (e-mail: hendrawan@stts.edu)

Willy Adrianus, Departemen Informatika, *Institut Sains dan Teknologi Terpadu Surabaya*, Surabaya, Jawa Timur, Indonesia.

C. Artificial Intelligence

Artificial Intelligence (Kecerdasan Buatan) biasanya didefinisikan sebagai Intelligence Agents dimana mesin yang merasakan lingkungan dan mengambil tindakan yang mempunyai peluang besar untuk berhasil mencapai tujuannya [2]. Hingga saat ini, Artificial Intelligence tetap berkembang hingga mesin tersebut dapat meniru pemikiran manusia seperti belajar dan melakukan sebuah pemecahan masalah. Tujuan Artificial Intelligence dapat sederhana ataupun kompleks. Artificial Intelligence sering dipakai pada penggunaan algoritma. Banyak algoritma Artificial Intelligence mampu belajar dari data, Artificial Intelligence dapat meningkatkan kemampuannya dengan mempelajari strategi dan aturan yang telah bekerja atau menggabungkan beberapa algoritma yang berbeda.

Artificial Intelligence banyak digunakan pada permainan video, dimana pemain dapat bertanding melawan musuh yang biasanya disebut Computer atau AI. Computer atau AI tersebut juga dibagi berdasarkan tingkat kesulitannya. Setiap tingkatan pada Computer tersebut memiliki pemikiran yang berbeda terhadap langkah yang akan diambil berikutnya. Model Artificial Intelligence pada permainan dibagi menjadi tiga bagian yaitu Strategy, Decision making, dan Movement.

Behavior Tree merupakan model matematika dari rencana eksekusi yang digunakan dalam ilmu komputer, robotika, sistem kontrol dan permainan video. Secara tampilan, behavior tree direpresentasikan sebagai pohon yang memiliki nodes berbeda-beda. Perjalanan behavior tree dimulai dari root yang mengirimkan tanda-tanda dengan frekuensi tertentu ke child dari root ini. Tanda merupakan sinyal yang memungkinkan untuk menjalankan atau mengeksekusi child.

III. METODE DAN INTI PENELITIAN

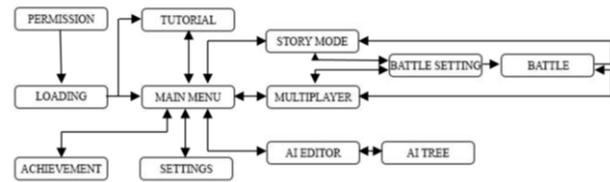
Berikut adalah langkah-langkah yang dilakukan dalam melakukan penelitian ini:

1. Pencarian sumber informasi terkait tactical game
2. Pencarian sumber informasi terkait AI for Games
3. Perancangan sistem Game
4. Pencarian resource pendukung
5. Perancangan sistem Artificial Intelligence
6. Melakukan ujicoba sistem Artificial Intelligence
7. Menggabungkan sistem game dan sistem artificial intelligence
8. Melakukan ujicoba sistem (alfa dan beta testing)
9. Publikasi game ke playstore

A. Arsitektural Sistem

Alur permainan yang dibuat pada permainan ini dimulai dari permintaan persetujuan pemain untuk mengakses perangkat pemain. Kemudian dilanjutkan pada loading dari permainan. Apabila pemain baru pertama kali menginstall aplikasi maka pemain wajib menyelesaikan tutorial dan lanjut ke tampilan main menu. Pada permainan ini terdapat enam menu yaitu story mode, multiplayer, tutorial, AI editor, settings, dan achievement. Pada story mode, pemain dapat memilih arena yang dimainkan dan setelah itu mengatur sistem AI pada robot yang digunakan untuk bertarung. Pada multiplayer terdapat tiga menu yaitu find match, history dan leaderboard. Find match digunakan untuk bertanding dengan pemain lain secara online. History

digunakan untuk melihat pertandingan yang sudah selesai. Leaderboard digunakan untuk melihat ranking semua pemain yang pernah bermain.



Gambar. 1. Alur Permainan

Pada tutorial berisi tiga tahap. Tahap pertama tutorial berisi pengenalan terhadap unit dan gameplay pada permainan ini. Tahap kedua tutorial berisi pengenalan terhadap sistem AI mulai dari cara kerja AI dan cara membuat sebuah node pada sistem AI. Tahap ketiga tutorial berisi pengenalan mengatur sistem AI pada robot yang ada pada arena. Pada AI editor berisi sistem AI yang telah dibuat oleh pemain. Pemain dapat membuat, mengubah, menghapus, mengganti nama dan duplicate sistem AI pada tampilan AI editor. Pada achievement, pemain dapat melihat achieve apa saja yang sudah didapat. Pada settings, pemain dapat mengatur suara background music dan sound effect permainan.

B. Desain Interface

Bagian ini akan menjelaskan beberapa tampilan yang ada pada permainan. Pada gambar 2 merupakan tampilan main menu yang berisi menu story mode, multiplayer, tutorial, ai editor, achievement dan settings.



Gambar. 2. Tampilan Main Menu

Pada Gambar 3 merupakan tampilan battle dimana terdapat map arena yang digunakan untuk melihat jalannya pertempuran. Terdapat objective atau goal, waktu, skor dan battle controls. Pada battle controls terdapat tiga tombol yaitu tombol pause, tombol fast forward dan tombol skip. Ketika pertempuran berakhir, ditampilkan tombol replay dan tombol back to story mode.



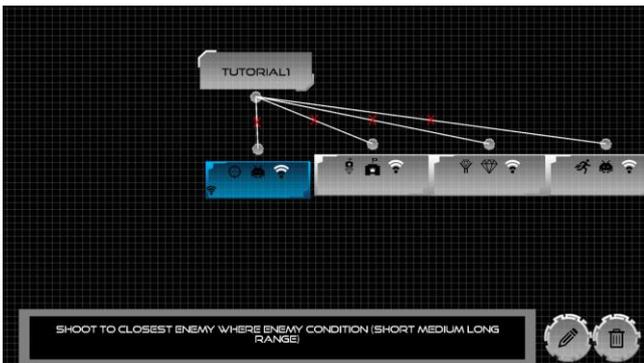
Gambar. 3. Tampilan Battle

Pada Gambar 4 merupakan tampilan tutorial pendahuluan dimana tutorial ini mengenalkan pemain mengenai unit dan cara bermain permainan ini.



Gambar. 4. Tampilan Tutorial

Pada Gambar 5 merupakan tampilan AI Tree dimana ditampilkan sistem AI yang telah dibuat. Pada tampilan ini, pemain dapat mengubah sistem AI dengan menambah, mengubah atau menghapus node. Pemain juga dapat melihat info node yang telah dibuat.



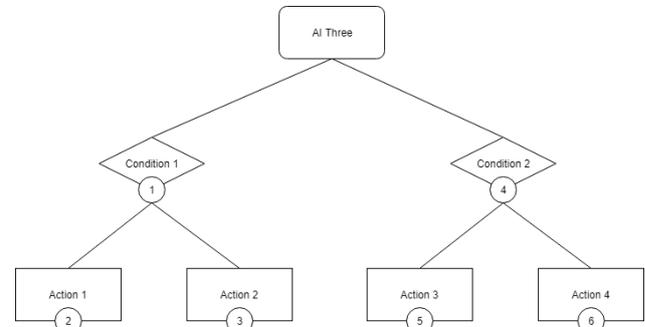
Gambar. 5. Tampilan AI Tree

### C. Sistem Artificial Intelligence

Bagian ini akan membahas mengenai sistem artificial intelligence yang merupakan bagian utama dari permainan. Konsep sistem artificial intelligence yang dibuat berdasarkan konsep behavior tree. Dimana konsep behavior tree merupakan salah satu model yang banyak digunakan untuk membuat sebuah artificial intelligence dan sangat dikenal di kalangan game developer. Konsep sistem artificial intelligence ini secara grafik diimplementasikan dalam bentuk pohon dimana pohon tersebut mempunyai akar dan cabang ke segala arah yang nantinya disebut node. Konsep sistem artificial intelligence yang digunakan diubah sesuai dengan permainan yang dimainkan.

Untuk awal pembacaan konsep sistem artificial intelligence dimulai dari root dimana pada konsep sistem artificial intelligence (gambar 6) menggunakan judul artificial intelligence sebagai root. Root merupakan awal dari pencarian sebuah aksi yang akan dijalankan. Pada konsep sistem artificial intelligence tersebut, root berbentuk sebuah persegi panjang dengan pinggiran bulat dan pada permainan Battle Artificial intelligence akan disebut Root Node. Dari root akan dimulai mencari sebuah aksi yang dapat dijalankan dimulai dengan mencari pada bagian child dari root. Jika child yang dicari adalah sebuah aksi yang

dapat dijalankan maka pencarian akan berhenti dan aksi tersebut akan dijalankan. Prioritas sebuah child pada konsep sistem artificial intelligence ini adalah posisi. Dimana posisi sebuah child diurutkan dari kiri ke kanan. Jika child pertama diperiksa dan hasilnya tidak dapat dijalankan maka sistem artificial intelligence akan memeriksa child berikutnya yaitu child kedua.



Gambar. 6. Contoh Konsep Sistem AI

Konsep sistem artificial intelligence ini tidak membatasi jumlah child atau aksi yang akan dibuat bahkan jika child dari root memiliki child, tidak ada batasan jumlah yang harus dimiliki. Pencarian ini akan berulang hingga aksi yang dapat dijalankan ditemukan atau seluruh child telah diperiksa dan tidak ada yang dapat dijalankan maka pencarian akan berhenti. Jika ditemukan aksi yang dapat dijalankan maka aksi tersebut dijalankan. Jika tidak ditemukan aksi yang dapat dijalankan maka sistem artificial intelligence tidak akan menjalankan aksi apapun (idle). Pencarian dilakukan ulang setiap tick pada sistem artificial intelligence berjalan dan dimulai kembali dari root. Konsep sistem artificial intelligence ini berulang terus menerus selama sistem artificial intelligence masih berjalan. Pada konsep sistem artificial intelligence ini juga dapat dimanfaatkan dengan Logic Gate (Gerbang Logika). Dimana sistem artificial intelligence ini dapat melakukan kondisi OR dan AND di dalam satu sistem artificial intelligence.

Node sistem artificial intelligence dibagi menjadi empat bagian yaitu action node, decision node, connector node dan root node. Action node digunakan untuk menjalankan aksi yang dipilih pada robot. Action node berisi aksi, target, prioritas dan filter yang akan dijalankan. Aksi yang tersedia pada action node yaitu idle, move, flee, shoot, catch, drop dan secure. Target yang tersedia pada action node yaitu enemy, ally, our base, enemy base dan resource. Prioritas yang tersedia pada action node yaitu closest range, furthest range, lowest health, highest health, lowest shield dan highest shield. Filter yang tersedia pada action node yaitu carrying resource, short range, medium range, long range, out of range, 0-25% health, 25-50% health, 50-75% health, 75-100% health, 0-25% shield, 25-50% shield, 50-75% shield, 75-100% shield. Decision node digunakan untuk mencari apakah sebuah target ada atau tidak ada pada arena. Decision node juga digunakan untuk menentukan apakah node berikutnya dijalankan atau tidak. Connector node berisi target, prioritas, filter dan kondisi. Target yang tersedia pada decision node yaitu self, enemy, ally, resource,

ally base, enemy base. Prioritas dan filter yang tersedia pada decision node seperti prioritas dan filter yang tersedia pada action node. Kondisi yang tersedia pada decision node yaitu exist or not exist. Connector node digunakan sebagai penghubung antar node. Root node digunakan sebagai awal dari proses pencarian sebuah aksi yang dijalankan atau penghubung dengan sistem artificial intelligence lainnya.

Sistem artificial intelligence pada permainan ini didukung dengan penggunaan nama untuk setiap sistem artificial intelligence yang berbeda serta dapat diubah setiap saat. Sistem artificial intelligence pada permainan ini juga didukung dengan fitur duplicate dimana sebuah sistem artificial intelligence dapat dicopy menjadi lebih dari satu. Dengan adanya fitur duplicate ini, pemain dapat menggunakannya sebagai fungsi backup and restore ataupun sistem version untuk mengembangkan sistem artificial intelligence yang pemain telah buat. Sistem artificial intelligence pada permainan ini tidak memiliki batasan berapa banyak sistem artificial intelligence yang dapat dibuat.

Pada permainan ini, pemain perlu memikirkan strategi yang dilakukan dalam memenangkan pertempuran. Pada permainan ini, terdapat 10 arena dengan tingkat kesulitan yang berbeda. Setiap arena memiliki jumlah yang berbeda atau posisi bot yang berbeda. Dengan posisi masing-masing bot yang berbeda di setiap arena, pemain dapat membuat sistem AI yang berbeda dengan memprioritaskan aksi yang berbeda. Dengan strategi, pemain dapat memenangkan pertempuran yang sebelumnya belum dimenangkan.

**Segmen Program 1 Membaca Sistem AI**

```

1. boolean findAction = false; // Set Find Action
2. AITree aiTree = null;
3. if (team.equals("TEAM_1") && botColor.equals("BLUE"))
   aiTree = SystemAI.playerAITrees.get(aiTreeIndex); //
   Player System AI
4. else aiTree = SystemAI.getEnemySystem("ENEMY_SYSTEM_"
   + arena.getIndex()); // Enemy or Ally System AI
5. Target oldTarget = target;
6. if (aiTree != null) findAction =
   recursiveFindAction(aiTree.root, findAction); // FIND
   ACTION
7. if (!findAction){
8.   doIdle(); // IF NO ACTION.. DO IDLE
9.   if (oldTarget != null) updateTarget("TARGET",
   null);
10. }
11. else {
12.   if (!(oldTarget == null && target == null)){
13.     if (oldTarget != null && target != null) {
14.       if (!(oldTarget.type.equals(target.type) &&
   oldTarget.index == target.index))
   updateTarget("TARGET", target);
15.     }
16.     else updateTarget("TARGET", target);
17.   }
18. }

```

Segmen program 1 merupakan program untuk pembacaan sistem Artificial Intelligence. Segmen program ini dijalankan ketika pertandingan telah dimulai dan setiap bot akan membaca sistem AI masing-masing. Sedangkan segmen program 2 mengenai pencarian aksi pada sistem Artificial Intelligent. Segmen program ini dijalankan ketika membaca sistem AI.

**Segmen Program 2 Pencarian Aksi Pada Sistem AI**

```

1. public boolean recursiveFindAction(Node current,
   boolean findAction){
2.   for (Node n : current.children) {
3.     if (n.type.equals("ROOT")) {
4.       AITree aiTree =
   SystemAI.getPlayerSystem(((NodeRoot) n).aiTreeName);
5. Segmen Program 5.15 Pencarian Aksi Pada Sistem AI
   (Lanjutan)
6.       if (aiTree != null){
7.         findAction = recursiveFindAction(aiTree.root,
   findAction);
8.         if (findAction) break;
9.       }
10.    }
11.    else if (n.type.equals("CONNECTOR")){
12.      findAction = recursiveFindAction(n,
   findAction);
13.      if (findAction) break;
14.    }
15.    else if (n.type.equals("DECISION")) {
16.      if (checkDecision(((NodeDecision)
   n).target.name, ((NodeDecision) n).priority.name,
   ((NodeDecision) n).getFilters(), ((NodeDecision)
   n).count.name)) {
17.        findAction = recursiveFindAction(n,
   findAction);
18.        if (findAction) break;
19.      }
20.    }
21.    else if (n.type.equals("ACTION")) {
22.      // Action IDLE and DROP doesnt need Target
23.      if (!((NodeAction)
   n).action.name.equals("IDLE") && !((NodeAction)
   n).action.name.equals("DROP")) target =
   findTarget(((NodeAction) n).target.name,
   ((NodeAction) n).priority.name, ((NodeAction)
   n).getFilters());
24.      // Do Action.. If have Target
25.      if (target != null || ((NodeAction)
   n).action.name.equals("IDLE") || ((NodeAction)
   n).action.name.equals("DROP")) {
26.        findAction = doAction(((NodeAction)
   n).action.name);
27.        if (findAction) break;
28.      }
29.    }
30.  }
31. }
32. return findAction;
33. }

```

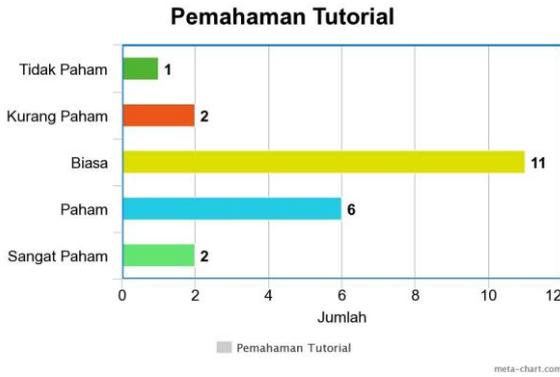
**IV. UJI COBA**

Bagian ini akan menjelaskan hasil uji coba kuisisioner yang diberikan kepada pengguna setelah mencoba permainan yang telah dibuat. Pertanyaan yang ditanyakan mulai dari pemahaman tutorial, konsep dan gameplay permainan, serta rating keseluruhan permainan.



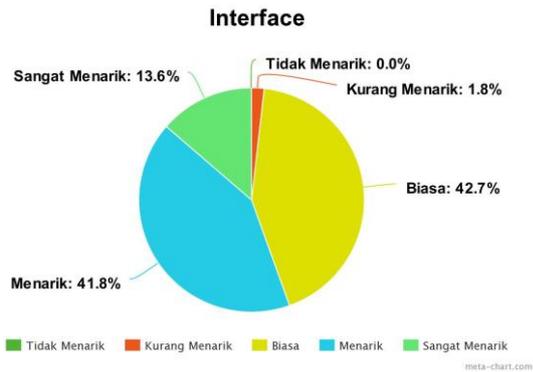
Gambar. 7. Konsep Permainan

Pada Gambar 7 adalah pertanyaan mengenai nilai konsep (gameplay) dari permainan ini. Hasil yang didapatkan berada pada tingkatan biasa, menarik, dan sangat menarik dengan jumlah yang sama 7 dari 22 orang yang telah mengisi kuisioner. Sedangkan pada Gambar 8 adalah pertanyaan mengenai pemahaman cara bermain melalui tutorial yang disediakan pada permainan ini. Hasil yang didapatkan berada pada tingkatan biasa dengan jumlah 11 dari 22 orang yang telah mengisi kuisioner.



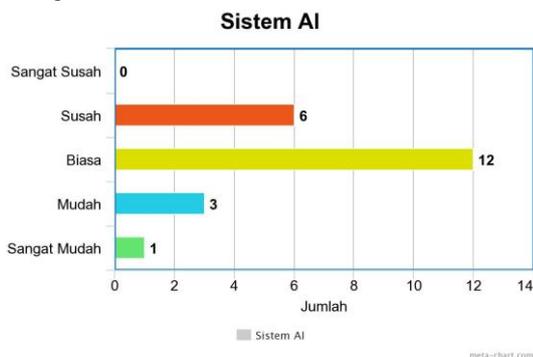
Gambar. 8. Pemahaman Tutorial

Pada Gambar 9 adalah pertanyaan mengenai tampilan-tampilan yang terdapat pada permainan ini. Hasil tampilan yang didapatkan berada pada tingkatan biasa dengan 42.7% dari penilaian 22 orang yang telah mengisi kuisioner.



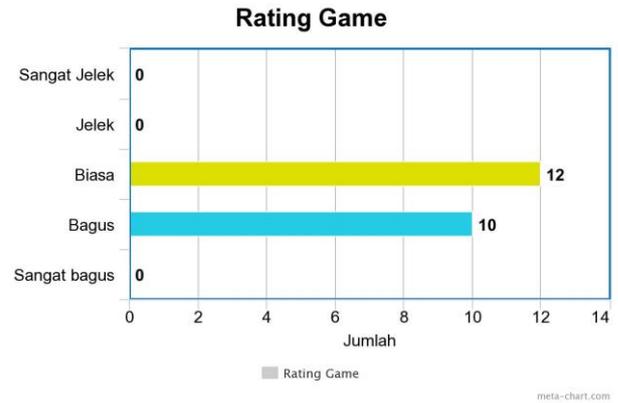
Gambar. 9. Interface

Pada Gambar 10 adalah pertanyaan mengenai konsep, rancangan, dan pembuatan sistem artificial intelligence yang terdapat pada permainan ini. Hasil yang didapatkan berada pada tingkatan biasa dengan jumlah 12 dari 22 orang yang telah mengisi kuisioner.



Gambar. 10. Sistem Artificial Intelligence

Pada Gambar 11 adalah pertanyaan mengenai rating secara keseluruhan permainan ini. Rating yang didapatkan yaitu 3.45 dari 22 orang yang telah mengisi kuisioner.



Gambar. 11. Rating Game

## V. KESIMPULAN

Kesimpulan yang diperoleh melalui penelitian ini adalah sebagai berikut:

1. Dengan penggabungan konsep strategi dan artificial intelligence, pengguna dapat memahami bagaimana musuh yang menggunakan artificial intelligence sebagai metode permainan memperoleh sebuah kemenangan di dalam permainan.
2. Game ini dapat menambah pengetahuan kepada pengguna terhadap rancangan Artificial Intelligence, cara sebuah Artificial Intelligence bekerja, dan pengguna mampu membuat sebuah sistem Artificial Intelligence bekerja melalui permainan atau program ini yang dibuktikan dengan hasil kuisioner pada tingkat pemahaman tutorial dan pemahaman sistem Artificial Intelligence.
3. Game ini dapat memberikan motivasi kepada pengguna untuk bersaing mengembangkan sebuah sistem Artificial Intelligence dengan bertarung dengan pengguna lain secara online.

## DAFTAR PUSTAKA

- [1] Horton, John. 2015. *Android Game Programming*. Packt Publishing.
- [2] Millington, Ian. & Funge, John. 2009. *Artificial Intelligence for Games*. United States of America: Morgan Kaufman Publishing.
- [3] Horton, John. 2015. *Learning Java by Building Android Games*. Packt Publishing.
- [4] *Getting Started in Android [Online]*. Available at: <https://developer.android.com/training/index.html> [Accessed 18 Mei 2017]
- [5] Cho, James. 2014. *The Beginner's Guide to Android Game Development*. Glasnevin Publishing.
- [6] Horton, John. 2015. *Android Programming for Beginners*. Packt Publishing.
- [7] *Info AI Decision Tree [Online]*. Available at: [https://www.aihorizon.com/essays/generalai/decision\\_trees.htm](https://www.aihorizon.com/essays/generalai/decision_trees.htm) [Accessed 18 Mei 2017]
- [8] Buckland, Mat. 2005. *Programming Game AI by Example*. Wordware Publishing.
- [9] Norvig, Peter. & Russel, Stuart J. 2016. *Artificial Intelligence: A Modern Approach*. Pearson Publishing.
- [10] Bourg, David. & Seemann, Glenn. 2014. *AI for Game Developers*. O'Reilly Publishing.

**Riandika Lumaris** lahir di Surabaya, Indonesia, pada tahun 1992. Menyelesaikan studi S1 di program studi Teknik Informatika STTS pada tahun 2014. Menyelesaikan studi masternya pada program studi Teknologi Informasi STTS pada tahun 2016. Minat penelitian adalah pada bidang Web Technologies, Cloud Computing, dan E-Commerce.

# Analisis Peramalan Stok Barang dengan Metode Weight Moving Average dan Double Exponential Smoothing pada Jovita Ms Glow Lamongan

Azza Wirdhatul Zannah, Teknik Informatika Universitas Islam Lamongan, Nur Nafi'iyah, Teknik Informatika Universitas Islam Lamongan

**Abstrak**— Jovita MS Glow Lamongan merupakan agen yang menjual produk kecantikan dari brand MS Glow, produk yang dijual di antaranya perawatan wajah, tubuh, kosmetik dengan perkembangan penjualan dari bulan ke bulan semakin meningkat maka dibutuhkan perhitungan perkiraan jumlah barang yang akan dibeli untuk meramalkan persediaan barang bulan berikutnya. Persediaan barang yang tidak tepat dapat menimbulkan kerugian maka perlu adanya sistem peramalan. Oleh karena itu penelitian menggunakan metode *Weight Moving Average* dan *Double Exponential Smoothing* untuk menentukan nilai *error* yang lebih kecil. Data yang digunakan pada penelitian ini mulai bulan Januari 2015 sampai bulan Desember 2016. Metode *Weight Moving Average* yaitu metode yang memberikan bobot yang berbeda untuk setiap historis sedangkan Metode *Double Exponential Smoothing* merupakan metode yang memiliki nilai pemulusan dua kali pada waktu sebelum data sebenarnya. Hasil peramalan kedua metode ini menghasilkan nilai *error Weight Moving Average* yaitu 698.7180 dan *Double Exponential Smoothing* yaitu 1.429.1015, sehingga *Weight Moving Average* adalah metode yang tepat digunakan untuk meramalkan persediaan barang karena memiliki nilai *error* yang lebih kecil.

**Kata Kunci**— Peramalan, Stok Barang, *Weight Moving Average*, *Double Exponential Smoothing*.

## I. PENDAHULUAN

Jovita MS Glow Lamongan merupakan agen yang menjual produk kecantikan. Dengan perkembangan penjualan dari bulan ke bulan yang semakin bertambah, maka kami ingin mengetahui berapa jumlah barang yang akan terjual bulan berikutnya dengan mengacu pada penjualan bulan sebelumnya. Serta untuk mengetahui berapa jumlah barang yang harus disediakan untuk periode penjualan berikutnya.

Perhitungan persediaan barang sebelumnya di hitung secara manual, dengan cara menghitung berapa jumlah barang yang tersisa, cara tersebut tidak efisien karena pemilik hanya mengira jumlah barang yang akan dibeli dan sering terjadi kesalahan dalam perhitungan peramalan sehingga tidak sesuai dengan kebutuhan konsumen.

Azza Wirdhatul Zannah, Mahasiswa Teknik Informatika Universitas Islam Lamongan, email: azzawirdha83@gmail.com.  
 Nur Nafi'iyah, Dosen Teknik Informatika Universitas Islam Lamongan, email: mynaff26@gmail.com.

Dari permasalahan di atas timbul gagasan untuk membuat sistem peramalan dengan metode *weight moving average* dan *double exponential smoothing*. Tujuan penelitian ini adalah untuk mengetahui metode yang tepat dalam menghitung peramalan stok barang dengan tingkat nilai *error* yang lebih kecil.

Peramalan (*Forecasting*) adalah suatu usaha untuk meramalkan keadaan dimasa mendatang melalui pengujian keadaan masa lalu peramalan bertujuan mendapatkan hasil yang mampu meminimumkan kesalahan dalam peramalan (*forecast error*).

Metode *weight moving average* adalah metode rata-rata bergerak yang banyak digunakan untuk menentukan trend dari suatu deret waktu. Metode ini digunakan untuk data yang perubahannya tidak cepat. Metode *double exponential smoothing* adalah metode penghalusan *exponential* adalah teknik peramalan rata-rata bergerak dengan pembobotandimana data diberi bobot oleh sebuah fungsi *exponential*. Penghalusan *exponential* merupakan metode peramalan rata-rata bergerak dengan pembobotan canggih, namun masih mudah digunakan.

## II. METODOLOGI PENELITIAN

Dalam pembuatan sistem peramalan persediaan barang menggunakan metode *weight moving average* dan *double exponential smoothing* bisa diartikan sebagai metode yang berfungsi untuk mencari nilai *error* terkecil meramalkan jumlah persediaan barang yang nantinya hasil outputnya berupa perbandingan dua metode dengan jumlah nilai *error* terkecil. Persamaan 1 perhitungan menggunakan metode *weight moving average*:

$$WMA = \frac{\sum(Dt \cdot \text{bobot})}{\sum(\text{bobot})} \dots\dots\dots(1)$$

$$WMA = ((30 \times 3) + (30 \times 2) + (20 \times 1)) / 3 + 2 + 1$$

$$= 90 + 60 + 20 / 6 = 28.33333333$$

$$ERROR = 30 - 28.33333333 = 1.66666667$$

Tabel 1 adalah data perhitungan metode *weight moving average*.

TABEL I  
 DATA PERHITUNGAN WMA

Bulan	Tahun	Nilai	Hitung WMA	ERROR
Januari	2015	20	0	
Februari	2015	30	0	

Bulan	Tahun	Nilai	Hitung WMA	ERROR
Maret	2015	30	28.333333	1.6666667
April	2015	40	35	5
Mei	2015	50	43.333333	6.6666667
Juni	2015	50	48.333333	1.6666667
Juli	2015	30	40	10
Agustus	2015	20	28.333333	8.3333333
September	2015	25	24.166667	0.8333333
Oktober	2015	40	31.666667	8.3333333
November	2015	35	35	0
Desember	2015	40	38.333333	1.6666667
Januari	2016	53	45.666667	7.3333333
Februari	2016	66	57.333333	8.6666667
Maret	2016	60	60.833333	0.8333333
April	2016	65	63.5	
Mei	2016	70	66.666667	1.5333333 3
Juni	2016	55	61.666667	6.6666667
Juli	2016	75	67.5	7.5
Agustus	2016	60	64.166666	7 4.1666667
September	2016	68	66.5	1.5
Oktober	2016	80	72.666667	7.3333333
November	2016	90	83	7
Desember	2016	85	85.833333	0.8333333

Persamaan 2 perhitungan menggunakan metode *double exponential smoothing*:

1. Menghitung nilai Ft

$$F_t = \frac{\text{bulan sekarang} + \text{bulan lalu}}{2} \dots\dots\dots(2)$$

2. Menghitung prediksi single exponential smoothing

$$F_{t+1} = F_t + \alpha (Y_t - F_t) \dots\dots\dots(3)$$

3. Menghitung prediksi double exponential smoothing

$$F'_{t+1} = F_{t+1} + \alpha (Y_t - F_{t+1}) \dots\dots(4)$$

$$\text{Single Exponential} = (0.4 * (40-35)) + 35 = 37$$

$$\text{Double Exponential} = (0.4 * (40-37)) + 37 = 38.2$$

$$\text{ERROR} = 38.2 - 50 = 11.8$$

Tabel 2 adalah data perhitungan metode *double exponential smoothing*.

Pada kamus data ini terdapat lima tabel yaitu admin, persediaan barang, peramalan metode *weight moving average*, peramalan metode *double exponential smoothing* dan hasil akurasi. Dalam perancangan sistem ini database penyimpanan data adalah sebagai berikut:

Tabel persediaan barang berfungsi sebagai tempat menyimpan data persediaan barang seperti no, nama, nilai, bulan dan tahun, dengan no sebagai *primary key*. Proses tersebut dapat dijelaskan pada Tabel 3.

Tabel metode *weight moving average* berfungsi sebagai perhitungan peramalan persediaan barang seperti no, nama, bulan, tahun, nilai, has\_prediksi, eror, dengan no sebagai *primary key*. Proses tersebut dapat dijelaskan pada Tabel 4.

TABEL II  
DATA PERHITUNGAN DES

Bulan	Tahun	Nilai	PP	SE	DES	Error
Januari	2015	20	10	14	16.4	13.6
Februari	2015	30	25	27	28.2	1.8
Maret	2015	30	30	30	30	10
April	2015	40	35	37	38.2	11.8
Mei	2015	50	45	47	48.2	1.8
Juni	2015	50	50	50	50	20
Juli	2015	30	40	44	49.6	29.6
Agustus	2015	20	25	27	29.8	4.8
September	2015	25	22.5	23.5	24.1	15.9
Oktober	2015	40	32.5	35.5	37.3	2.3
November	2015	35	37.5	38.5	39.9	0.1
Desember	2015	40	37.5	38.5	39.1	13.9
Januari	2016	53	46.5	49.1	50.66	15.34
Februari	2016	66	59.5	62.1	63.66	3.66
Maret	2016	60	63	64.2	65.88	0.88
April	2016	65	62.5	63.5	64.1	5.9
Mei	2016	70	67.5	68.5	69.1	14.1
Juni	2016	55	62.5	65.5	69.7	5.3
Juli	2016	75	65	69	71.4	11.4
Agustus	2016	60	67.5	70.5	74.7	6.7
September	2016	68	64	65.6	66.56	13.44
Oktober	2016	80	74	76.4	77.84	12.16
November	2016	90	85	87	88.2	3.2
Desember	2016	85	87.5	88.5	89.9	

TABEL III  
TABEL DATA PERSEDIAAN BARANG

No	Nama Field	Tipe Data	Size	Keterangan
1	No	Integer	11	Primary key
2	Nama	Varchar	30	
3	Bulan	Varchar	15	
4	Tahun	Integer	4	
5	Nilai	Integer	11	

TABEL IV  
TABEL PERAMALAN METODE *WEIGHT MOVING AVERAGE*

No	Nama Field	Tipe Data	Size	Keterangan
1	No	Integer	11	Primary key
2	Nama	Varchar	30	
3	Bulan	Varchar	15	
4	Tahun	Integer	4	
5	nilai	Integer	11	
6	Has_prediksi	Float		
7	Eror	Float		

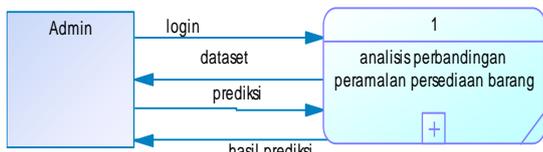
Tabel metode *double exponential smoothing* berfungsi sebagai perhitungan peramalan persediaan barang seperti no,

nama, bulan, tahun, nilai, has\_prediksi dan eror dengan no sebagai *primary key*. Proses tersebut dapat dijelaskan pada Tabel 5.

TABEL V  
TABEL METODE *DOUBLE EXPONENTIAL SMOOTHING*

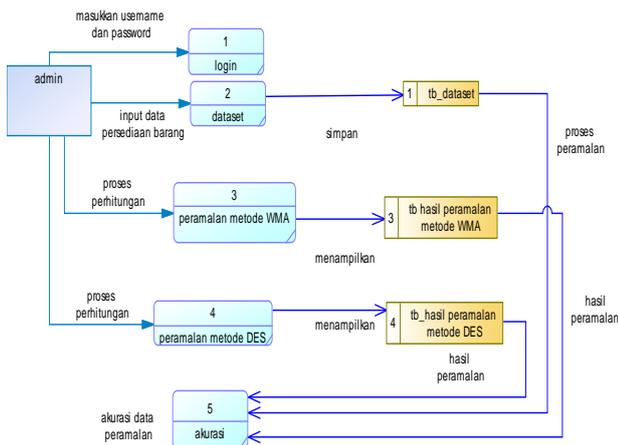
No	Nama Field	Tipe Data	Size	Keterangan
1	No	Integer	11	Primary key
2	Nama	Varchar	30	
3	Bulan	Varchar	15	
4	Tahun	Integer	4	
5	Nilai	Integer	11	
6	Has_prediksi	Float		
7	Error	Float		

DFD level 0 ini adalah perhitungan persediaan barang, admin melakukan login kemudian mengisi dataset, setelah dataset terisi maka admin melakukan proses prediksi persediaan barang, Kemudian proses tersebut berupa hasil prediksi. Proses tersebut di jelaskan pada Gambar 1.



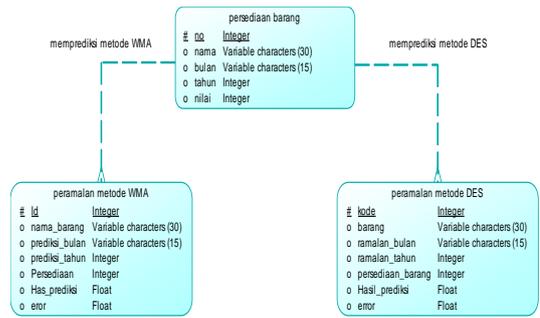
Gambar 1 DFD Level 0

Pada DFD level 1 admin melakukan input dataset dan dimasukkan pada tabel dataset kemudian di masukkan pada tabel akurasi. Setelah itu admin melakukan perhitungan metode WMA kemudian dimasukkan pada tabel hasil peramalan WMA, admin melakukan peramalan metode DES dan dimasukkan pada tabel hasil metode DES. Kemudian admin melakukan akurasi data. Proses tersebut dapat di jelaskan pada Gambar 2.



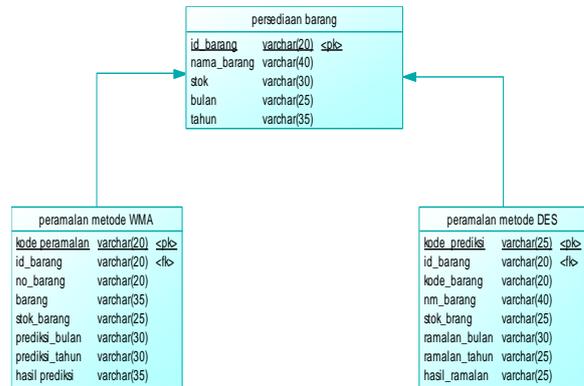
Gambar 2 DFD Level 1

Model konsep data atau *Conceptual Data Modeling* sebuah model diagram yang digunakan untuk menggambarkan basis data sebuah system. Pada CDM ini proses yang dilakukan adalah perhitungan persediaan barang dengan menggunakan prediksi metode WMA dan metode DES. Proses tersebut di jelaskan pada Gambar 3.



Gambar 3 *Conceptual Data Modeling*

Pemodelan data fisik adalah *generate* dari konsep pemodelan data yang telah dibuat sebelumnya pada gambar 3. Pada PDM ini proses yang dilakukan adalah perhitungan persediaan barang dengan menggunakan prediksi metode WMA dan metode DES proses tersebut di jelaskan pada Gambar 4.

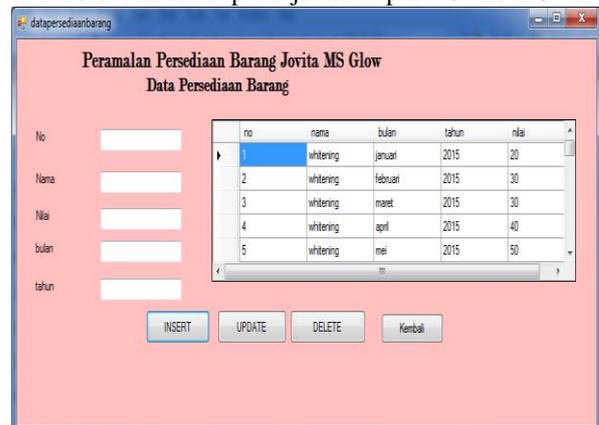


Gambar 4 *Physical Data Modeling*

### III. HASIL DAN PEMBAHASAN

Pembahasan fitur merupakan kajian yang dilakukan untuk memberikan gambaran utama aplikasi. Adapun komponen-komponen utama yang ada pada aplikasi sebagai berikut: Pada menu utama terdapat form data admin, form data persediaan barang, form peramalan metode *weight moving average*, form peramalan metode *double exponential smoothing* dan form hasil akurasi.

Pada form data persediaan barang memiliki tiga tombol yaitu *insert*, *update* dan *delete*. Tombol *insert* digunakan untuk menyimpan data yang sudah ditambahkan oleh admin. Tombol *update* digunakan untuk memperbarui data yang akan diubah. Tombol *delete* digunakan untuk menghapus data. Proses tersebut dapat dijelaskan pada Gambar 8.



Gambar 8 Form Data Persediaan Barang

Perhitungan persediaan barang dihitung menggunakan metode *weight moving average*. Proses tersebut dapat dijelaskan pada Gambar 9.

Perhitungan persediaan barang dihitung menggunakan metode *double exponential smoothing* yang menampilkan hasil prediksi dan nilai error. Proses tersebut dapat dijelaskan pada Gambar 10.

Pada form hasil akurasi terdapat nilai perhitungan metode *weight moving average* dan metode *double exponential smoothing*. Kemudian dibandingkan nilai eror terkecil antara dua metode tersebut dan dimasukkan pada hasil perbandingan.

nilai *error* terkecil dibandingkan metode Double Exponential Smoothing.

UCAPAN TERIMA KASIH / ACKNOLEDGMENT

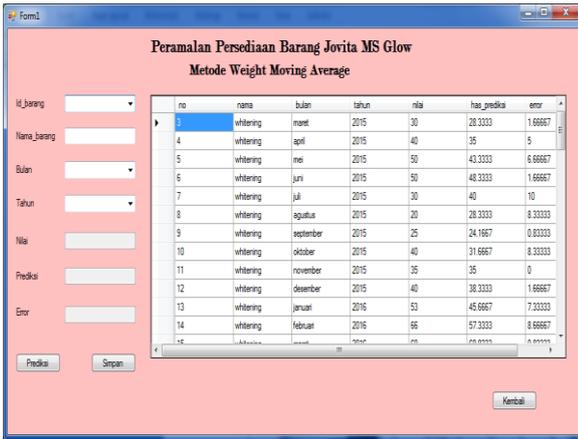
Terima kasih banyak kepada semua dosen Teknik Informatika yang telah memberikan pengetahuan dan ilmu kepada kami, terutama kepada Dosen pembimbing.

DAFTAR PUSTAKA

[1] Agus, Ristono. 2009. *Manajemen persediaan* edisi 1. Yogyakarta : Graha Ilmu.  
 [2] Lieberty, Annastasya. Et al. 2015. *Sistem Informasi Peramalan Penjualan Barang Dengan Metode Double Exponential Smoothing (Studi Kasus : PD. Padalarang Jaya)*. Bandung : Jurnal Teknik Informatika dan Sistem Informasi Volume 1 Nomor 1.  
 [3] Baktiar, Cindy, dkk. 2014 *Pembuatan Sistem Peramalan Penjualan Dengan Metode Weighted Moving Average dan Double Exponential Smoothing Pada UD Y. Universitas Kristen Petra*.  
 [4] Render, Heizer. 2005. *Penerapan Metode Single Moving Average dan Exponential Smoothing Dalam Peramalan Permintaan Produk Meubel Jenis Coffee Table Pada Java Furniture Klaten*.  
 [5] Sahli, Muchamad. 2013. *Penerapan Metode Exponential Smoothing Dalam Sistem Informasi Pengendalian Persediaan Bahan Baku (Studi Kasus Toko Tirta harum)*. Kudus : Jurnal SIMETRIS, Vol 3 No 1.

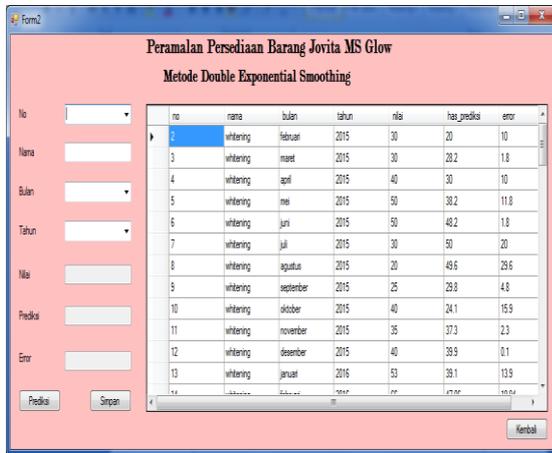
**Azza Wirdhatul Zannah**, alumni Teknik Informatika Universitas Islam Lamonga.

**Nur Nafi'iyah** lahir di Lamongan, 26 Januari 1987. Sekarang sebagai tenaga pengajar di Teknik Informatika Universitas Islam Lamongan. Alumni dari pascasarjana Teknologi Informasi Sekolah Tinggi Teknik Surabaya. Bidang penelitian kecerdasan buatan, pengolahan citra, data mining.



no	nama	bulan	tahun	nilai	has_prediksi	error
3	whitening	februari	2015	30	28.3333	1.66667
4	whitening	maret	2015	40	35	5
5	whitening	mei	2015	50	43.3333	6.66667
6	whitening	juni	2015	50	48.3333	1.66667
7	whitening	juli	2015	30	40	10
8	whitening	agustus	2015	20	28.3333	8.33333
9	whitening	september	2015	25	24.1667	0.833333
10	whitening	oktober	2015	40	31.6667	8.33333
11	whitening	november	2015	35	35	0
12	whitening	desember	2015	40	38.3333	1.66667
13	whitening	januari	2016	53	45.6667	7.33333
14	whitening	februari	2016	66	57.3333	8.66667

Gambar 9 Form Peramalan Metode WMA



no	nama	bulan	tahun	nilai	has_prediksi	error
1	whitening	februari	2015	30	20	10
3	whitening	maret	2015	30	28.2	1.8
4	whitening	april	2015	40	30	10
5	whitening	mei	2015	50	38.2	11.8
6	whitening	juni	2015	50	48.2	1.8
7	whitening	juli	2015	30	50	20
8	whitening	agustus	2015	20	45.6	25.6
9	whitening	september	2015	25	28.8	4.8
10	whitening	oktober	2015	40	24.1	15.9
11	whitening	november	2015	35	37.3	2.3
12	whitening	desember	2015	40	38.9	0.1
13	whitening	januari	2016	53	39.1	13.9

Gambar 10 Form Peramalan Metode DES

IV. KESIMPULAN

Berdasarkan uraian dan hasil analisa yang telah dilakukan selama pengembangan Aplikasi Analisis Perbandingan Peramalan Persediaan Barang Menggunakan Metode Weight Moving Average dan Double exponential Smoothing Pada Jovita MS Glow Lamongan, kesimpulan yang dapat diambil adalah: Untuk studi kasus, persediaan barang pada Jovita MS Glow Lamongan dapat diterapkan metode Weight Moving Average sedangkan untuk metode Double Exponential Smoothing mempunyai tingkat nilai *error* yang lebih besar. Metode Weight Moving Average dalam sistem peramalan persediaan barang menghasilkan

# Analisis Sentimen Pengguna Twitter Terhadap Polemik Persepakbolaan Indonesia Menggunakan Pembobotan TF-IDF dan K-Nearest Neighbor

Jeremy Andre Septian, Tresna Maulana Fahrudin, Aryo Nugroho  
Fakultas Ilmu Komputer, Universitas Narotama

**Abstrak**— Persepakbolaan Indonesia belakangan ini memiliki banyak polemik mulai dari kasus pengaturan skor, pergantian pelatih timnas senior hingga pergantian ketua umum Persatuan Sepak bola Seluruh Indonesia (PSSI). Polemik ini menimbulkan banyaknya opini maupun pendapat dari pengguna *twitter* terhadap persepakbolaan di Indonesia sehingga diperlukan sebuah sistem untuk memudahkan dalam mengetahui sentimen pada setiap kalimat. Tujuan dari penelitian ini adalah untuk menganalisis sentimen pada setiap kalimat dari pengguna *twitter* terhadap persepakbolaan Indonesia apakah memiliki sentimen negatif atau positif menggunakan *K-Nearest Neighbor*. Data yang digunakan dalam penelitian ini didapatkan dari hasil *crawling* dari media sosial *twitter* terkait persepakbolaan di Indonesia yang diambil dari akun *twitter* resmi PSSI. Setelah data dikumpulkan kemudian akan dilakukan beberapa tahapan yaitu *preprocessing* yang terdiri dari *cleansing*, *tokenizing*, *stopword removal*, dan *stemming*. Pembobotan kata menggunakan *Term Frequency-Invers Document Frequency (TF-IDF)*. Pada tahap validasi data dilakukan pengujian silang sebanyak 10 kali menggunakan *k-fold cross validation*, kemudian diklasifikasikan dengan metode *K-Nearest Neighbor* dapat menghasilkan akurasi yang cukup baik. Dari 2000 data *tweet* berbahasa Indonesia didapatkan hasil akurasi optimal pada nilai  $k=23$  sejumlah 79.99%.

**Kata Kunci**— Persepakbolaan Indonesia, Analisis sentimen, *Preprocessing*, TF-IDF, *K-Nearest Neighbor*

## I. PENDAHULUAN

Persepakbolaan di Indonesia dalam beberapa tahun terakhir ini sedang mengalami banyak polemik seperti contohnya pada kasus pengaturan skor, pergantian pelatih timnas senior, dan pergantian ketua umum PSSI. Dengan adanya polemik yang terjadi dalam beberapa bulan terakhir ini banyak menimbulkan opini maupun pendapat dari para pengguna media sosial salah satunya adalah *twitter*. *Twitter* merupakan layanan media sosial yang berkategori *microblogging* yang paling populer, pengguna dapat membaca dan berbagi pesan singkat dengan jumlah maksimal 280 karakter [1],[2]. Berdasarkan permasalahan

tersebut maka untuk memudahkan dalam mengetahui sentimen dari setiap opini maupun pendapat pada setiap kalimat perlu dibuatkan suatu sistem untuk menganalisis sentimen. Analisis sentimen adalah suatu proses untuk mengetahui pendapat atau opini pada sebuah kalimat maupun dokumen apakah memiliki sentimen negatif atau positif [3]. Dalam akun *twitter* resmi PSSI ini berisi banyak opini maupun pendapat dari pengguna *twitter* terkait tentang polemik yang terjadi pada persepakbolaan Indonesia sehingga dapat digunakan sebagai sumber data untuk melakukan penelitian terkait analisis sentimen [4]. Data yang digunakan dalam penelitian ini sejumlah 2000 data *tweet* berbahasa Indonesia yang akan dilakukan pembobotan kata dengan TF-IDF dan diklasifikasi menggunakan *K-Nearest Neighbor*. Kinerja *K-Nearest Neighbor* sebagai algoritma untuk klasifikasi data teks cukup bagus [5].

Penelitian terdahulu terkait analisis sentimen dengan *K-Nearest Neighbor* pernah dilakukan untuk menganalisis sentimen dari data komentar akun *facebook* jasa ekspedisi barang J&T pada tahun 2018 dan mendapatkan akurasi tertinggi sebesar 79.21% [6]. Penelitian selanjutnya *K-Nearest Neighbor* digunakan untuk mengklasifikasikan dokumen komentar pada situs *youtube* didapatkan akurasi tertinggi sebesar 80.6% [7].

Kontribusi pada penelitian ini adalah untuk dapat memberikan gambaran umum terkait dengan adanya polemik yang terjadi pada persepakbolaan Indonesia, dimana data *tweet* diambil sebagai sampel untuk mewakili saran dan opini yang disampaikan oleh masyarakat.

## II. TINJAUAN PUSTAKA

### A. Analisis Sentimen

Analisis sentimen atau penambangan opini adalah suatu bidang studi untuk menganalisis pendapat orang terhadap entitas seperti produk, layanan, organisasi, individu, masalah, peristiwa, dan topik. Analisis sentimen ini berfokus pada pendapat seseorang yang mengekspresikan atau menyiratkan sentimen positif atau negatif, kebanyakan analisis sentimen ini berkaitan dengan orang-orang di media sosial. Faktanya, analisis sentimen sekarang berada di pusat penelitian media sosial. Oleh karena itu, penelitian dalam analisis sentimen tidak hanya memiliki dampak penting pada NLP (*Natural Language Processing*), tetapi mungkin juga memiliki dampak mendalam pada ilmu manajemen, politik, ekonomi, dan ilmu sosial karena mereka semua dipengaruhi oleh pendapat orang. Memperoleh pendapat dari publik dan konsumen telah lama menjadi bisnis besar dalam bidang pemasaran, hubungan masyarakat, dan

Jeremy Andre Septian, Fakultas Ilmu Komputer, Universitas Narotama, Surabaya, Jawa Timur, Indonesia (e-mail: jeremyandreseptian@gmail.com)

Tresna Maulana Fahrudin, Fakultas Ilmu Komputer, Universitas Narotama, Surabaya, Jawa Timur, Indonesia (e-mail: tresna.maulana@narotama.ac.id)

Aryo Nugroho Fakultas Ilmu Komputer, Universitas Narotama, Surabaya, Jawa Timur, Indonesia (e-mail: aryo.nugroho@narotama.ac.id)

perusahaan kampanye politik. Besarnya pengaruh dan manfaat dari analisis sentimen menyebabkan penelitian ataupun aplikasi mengenai analisis sentimen berkembang pesat, bahkan di Amerika kurang lebih 20-30 perusahaan yang memfokuskan pada layanan analisis sentimen [8].

**B. Pembobotan TF-IDF (Term Frequency-Inverse Document Frequency)**

Pembobotan TF-IDF (Term Frequency-Inverse Document Frequency) adalah suatu proses untuk melakukan transformasi data dari data tekstual ke dalam data numerik untuk dilakukan pembobotan pada tiap kata atau fitur. TF-IDF ini adalah sebuah ukuran statistik yang digunakan untuk mengevaluasi seberapa penting sebuah kata di dalam sebuah dokumen. TF adalah frekuensi kemunculan kata pada di tiap dokumen yang diberikan menunjukkan seberapa penting kata itu di dalam tiap dokumen tersebut. DF adalah frekuensi dokumen yang mengandung kata tersebut menunjukkan seberapa umum kata tersebut. IDF adalah *inverse* dari nilai DF. Hasil dari pembobotan kata menggunakan TF-IDF ini adalah hasil perkalian dari TF dikalikan dengan IDF. Bobot kata semakin besar jika sering muncul dalam suatu dokumen dan semakin kecil jika muncul dalam banyak dokumen [9].

**C. K-Nearest Neighbor**

*K-Nearest Neighbor* (KNN) adalah algoritma klasifikasi *supervised learning* atau biasa dikenal dengan metode berbasis jarak. Metode ini bekerja dengan hanya menghafal semua contoh pelatihan yang tersedia selama fase pelatihan. Selanjutnya pada fase pengujian, dokumen yang akan diklasifikasi dibandingkan dengan contoh-contoh berdasarkan ukuran jarak yang ditentukan sebelumnya. Dokumen yang paling mirip disebut "tetangga terdekat" untuk jumlah tetangga terdekat dapat ditentukan dari berapa jumlah nilai k. KNN ini merupakan metode yang sederhana untuk pengklasifikasian tanpa harus melakukan perhitungan secara kompleks, oleh sebab itu KNN ini juga biasa disebut *lazy learning*. Untuk menghitung jarak antar fitur KNN ini menggunakan perhitungan *Euclidean Distance* [10].

**III. METODE PENELITIAN**

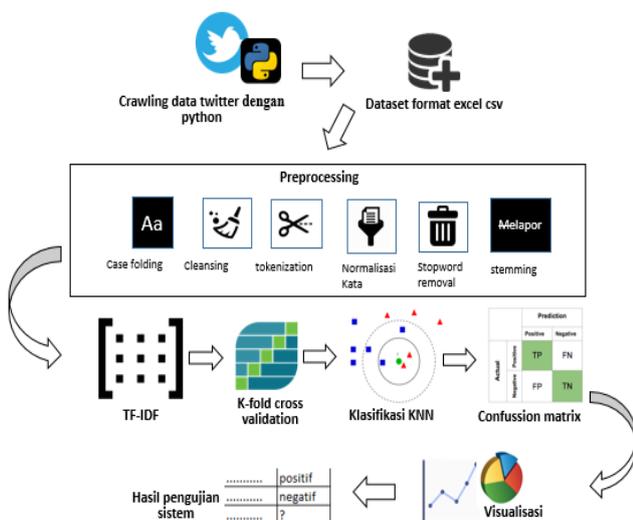
Dalam penelitian ini terdapat alur proses penelitian untuk dapat melakukan analisis sentimen yaitu dimulai dengan pengumpulan data, *preprocessing*, pembobotan kata dengan TF-IDF, validasi dengan *k-fold cross validation*, klasifikasi *K-Nearest Neighbor*, penghitungan akurasi dengan *confusion matrix*, visualisasi dan pengujian sistem. Seperti ditunjukkan pada Gambar 1.

**A. Pengumpulan data**

Pada penelitian ini data diambil dari media sosial *twitter* dengan program *crawling* menggunakan bahasa pemrograman *python* dengan *library tweepy* yang memanfaatkan *API Twitter*. Langkah awal dalam tahap pengumpulan data ini adalah membuat aplikasi *API Twitter* melalui akun *twitter* pribadi untuk mendapatkan *consumer key*, *consumer secret*, *access token*, dan *access token secret* yang digunakan untuk syarat kunci pengambilan data. Kemudian memasukan *query '@pssi'* nantinya pada *query* ini memuat semua *tweet* yang ditujukan kepada akun *twitter* PSSI dan menggunakan filter tambahan *no-retweet*. Periode pengambilan data pada *twitter* dalam penelitian adalah 21 februari – 04 Mei 2019, seluruh data hasil *crawling* disimpan kedalam format excel csv (*comma separated value*). Dataset yang digunakan dalam penelitian ini adalah *tweet* dan sentimen. 0 untuk merepresentasikan sentimen negatif dan 1 untuk sentimen positif. Untuk lebih jelas, dapat diperhatikan Tabel 1.

TABEL I  
DATA TWEET PENGGUNA TWITTER TERKAIT PERSEPAKBOLAAN INDONESIA

Tweet	Sentimen
@PSSI Semangat garudaku,,, jangan minder lihat lawan, jangan anggap remeh juga dengan lawan.. KAMU PASTI BISA...DOAKU UNTUK KALIAN	1
@PSSI Indonesia mesti menang lawan Mafia Bola Garuda Pancasila. Akulah pendukungmu#TimnasIndonesiaMaju	1
@KEMENPORA_RI @PSSI #AFFU22 https://t.co/9Wl0aWpfD5	1
Bila kelak Indonesia lolos ke final AFF U22 2019, itu tandanya kita sedang mulai menuju prestasi sepakbola yang kita dambakan. #TimnasIndonesiaMaju	1
@KEMENPORA_RI @PSSI	0
@PSSI bukan hari yang kurang baik tapi emnk Kualitas kalian cuman segitu!!! nyadar diri lah BGST!!!	0
@PSSI Berat utk lolos	0
@PSSI mencium aroma kekalahan	0
@PSSI Federasi BUSUK,timnas jeblok\n#busuk\n#busuk\n#busuk'	0
@PSSI Organisasinya induknya aja asal2an ya maen bola ya asal2an aja'	0
@PSSI timnas bobrok karena federasi korup n tidak becus'	0



Gambar. 1. Desain sistem analisis sentimen terhadap persepakbolaan Indonesia

**B. Preprocessing**

Data yang sudah dikumpulkan kemudian dilakukan *preprocessing* untuk menghindari data yang belum siap untuk diolah seperti terdapat gangguan (*noise*) dan data yang tidak konsisten [11]. Adapun tahapan pada *preprocessing* ini adalah sebagai berikut:

### 1. Case Folding dan Cleansing

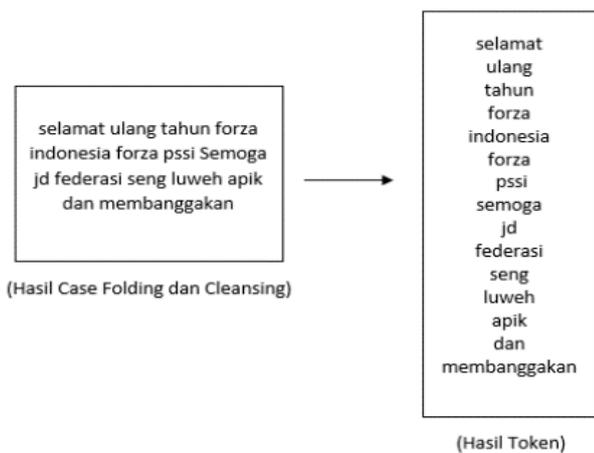
Pada tahapan ini dilakukan penyeragaman seluruh teks menjadi huruf kecil (*lowercase*) dan pembersihan atau penghapusan pada pada semua dokumen yang berisi angka, url (<http://>), *username* (@), tanda pagar (#), delimiter seperti koma (,) dan titik (.) dan juga tanda baca lainnya. Pada Gambar 2 diilustrasikan proses *case folding* dan *cleansing*.



Gambar. 2. Proses *Case Folding* dan *Cleansing*

### 2. Tokenizing

Pada tahapan ini dilakukan proses pemotongan pada sebuah dokumen ataupun kalimat menjadi kata atau biasa disebut token. Pada Gambar 3 diilustrasikan proses *tokenizing*.



Gambar. 3. Proses *Tokenizing*

### 3. Normalisasi Kata

Pada tahapan ini dilakukan proses untuk menormalisasi kata terhadap setiap kata-kata yang mengandung kata tidak baku maupun *noise* menjadi kata yang baku dan siap diolah. Kata tidak baku dan adanya *noise* yang dimaksud adalah kata yang mengandung unsur bahasa daerah yang tidak sesuai dengan Kamus Besar Bahasa Indonesia (KBBI) dan kata singkatan dalam media sosial. Seperti contoh kata singkatan yang ering muncul pada media sosial ini adalah kata “tdk” dengan proses normalisasi kata ini diubah menjadi kata “tidak”, kata yang mengandung unsur bahasa daerah seperti kata “apik” dengan proses normalisasi kata ini diubah menjadi “baik”, kata tidak baku seperti kata “endonesah” dengan proses normalisasi kata ini diubah menjadi kata “Indonesia”. Untuk lebih jelas dapat diperhatikan Gambar 4 ilustrasi proses Normalisasi kata.

### 4. Stopword Removal

*Stopword removal* adalah suatu proses untuk menghapus kata yang dianggap tidak penting seperti contoh kata ‘di’,



Gambar. 4. Proses Normalisasi Kata

‘yang’, ‘dan’, ‘ke’, dan semua kata yang terdapat dalam kamus *stopword* yang sudah dibuat. Tujuan dari proses ini adalah untuk mengurangi jumlah kata yang disimpan dalam daftar token yang nantinya akan dilakukan proses selanjutnya. Pada Gambar 5 diilustrasikan proses *stopword removal*.



Gambar. 5. Proses *Stopword Removal*

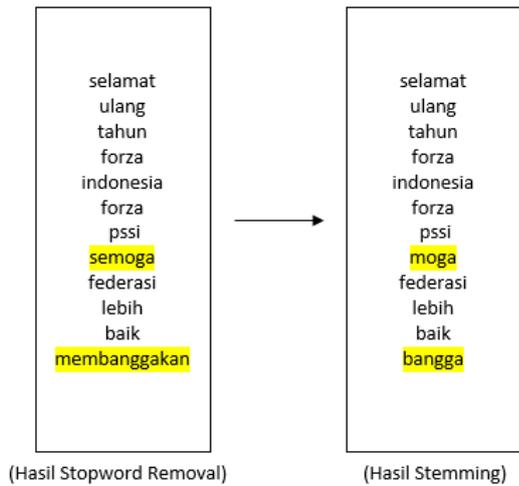
### 5. Stemming

Pada tahapan terakhir pada *preprocessing* ini dilakukan proses untuk mengubah semua kata-kata pada dokumen menjadi kata dasar dengan menghilangkan semua kata imbuhan. Kata imbuhan yang dihilangkan terdiri dari awalan (prefix), akhiran (suffix), sisipan (infix), dan gabungan awalan-akhiran (confix). Pada penelitian ini menggunakan *library python* sastrawi untuk proses *stemming*. Pada Gambar 6 diilustrasikan proses *stemming*.

### C. Pembobotan Kata dengan TF-IDF

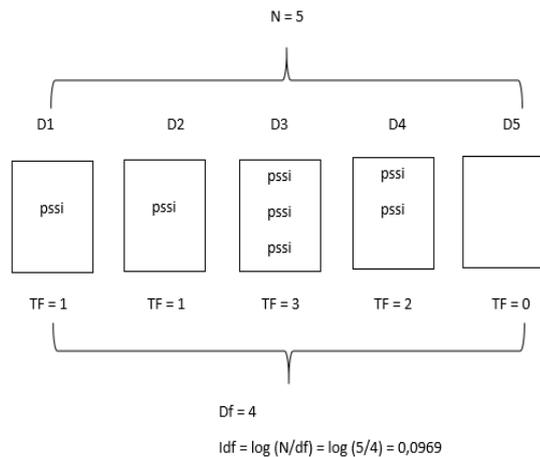
Data hasil *preprocessing* yang berupa kata akan diubah ke dalam bentuk angka dengan dilakukan proses pembobotan kata yang bertujuan untuk menghitung bobot pada masing-masing kata yang akan digunakan sebagai fitur, semakin banyak dokumen yang akan diproses maka semakin banyak fitur. Pada tahapan ini terdapat dua bagian proses yaitu TF (*Term Frequency*) dan IDF (*Inverse Document Frequency*), TF adalah jumlah kemunculan tiap

kata pada sebuah dokumen semakin banyak kata muncul pada tiap dokumen maka semakin besar nilai TF. IDF



Gambar. 6. Proses Stemming

adalah jumlah nilai dokumen pada tiap kata yang berbanding terbalik yaitu apabila suatu kata jarang muncul pada sebuah dokumen maka nilai IDF lebih besar daripada kata yang sering muncul [12],[13].



Gambar. 7. Ilustrasi Algoritma TF-IDF

Keterangan:

D1,...,D5 = Dokumen

TF = Jumlah kata pada tiap dokumen

N = Total dokumen

Df= Jumlah dokumen pada kata yang dicari

Adapun rumus dari pembobotan kata TD-IDF adalah:

$$W_{t,d} = tf_{t,d} \times idf_t = tf_{t,d} \times \log \frac{N}{df_t} \quad (1)$$

Keterangan (1):

$W_{t,d}$  = Bobot TF-IDF

$tf_{t,d}$  = Jumlah frekuensi kata

$idf_t$  = Jumlah *inverse* frekuensi dokumen tiap kata

$df_t$  = Jumlah frekuensi dokumen tiap kata

N = Jumlah total dokumen

Hasil dari pembobotan kata dengan TF-IDF ini adalah perkalian dari nilai TF dan IDF yang akan menghasilkan

bobot lebih kecil apabila kata tersebut sering muncul pada setiap dokumen dalam koleksi, sebaliknya bobot TF-IDF akan lebih besar apabila kata tersebut jarang muncul pada setiap dokumen dalam koleksi. Dalam penelitian ini pembobotan TF-IDF yang digunakan adalah TF-IDF tanpa normalisasi.

#### D. Validasi dengan k-fold cross validation

Data yang sudah dilakukan pembobotan kata dengan TF-IDF akan dilakukan validasi menggunakan *k-fold cross validation*. Cara kerjanya adalah dilakukan pengelompokan antara data latih dan data uji kemudian dilakukan pengujian yang diulang sebanyak jumlah k. Dalam penelitian ini k yang digunakan adalah *10-fold* yang berarti akan dilakukan 10 kali pengujian pada seluruh isi dokumen secara acak. Dimana nantinya hasil akurasi didapatkan dari rata-rata akurasi pada 10 kali hasil pengujian [14],[15].

iterasi ke-	10-fold cross validation									
1	1	2	3	4	5	6	7	8	9	10
2	1	2	3	4	5	6	7	8	9	10
3	1	2	3	4	5	6	7	8	9	10
4	1	2	3	4	5	6	7	8	9	10
5	1	2	3	4	5	6	7	8	9	10
6	1	2	3	4	5	6	7	8	9	10
7	1	2	3	4	5	6	7	8	9	10
8	1	2	3	4	5	6	7	8	9	10
9	1	2	3	4	5	6	7	8	9	10
10	1	2	3	4	5	6	7	8	9	10

■ data latih ■ data uji

Gambar. 8. Ilustrasi tabel 10-fold cross validation

#### E. Klasifikasi K-Nearest Neighbor

*K-Nearest Neighbor* (KNN) adalah salah satu algoritma klasifikasi *supervised learning* yang digunakan untuk mengklasifikasikan objek berdasarkan atribut kelas dan data *training* [16]. Cara kerja KNN ini adalah melakukan klasifikasi berdasarkan data latih yang jaraknya paling dekat dengan objek data uji. Setiap data pada data latih memiliki atribut kelas, selanjutnya adalah dengan menguji model dengan data uji yang tidak memiliki atribut kelas. Adapun tahapan proses pada KNN ini adalah sebagai berikut:

1. Tahapan awal pada KNN ini adalah menentukan nilai K, misal k=23 artinya 23 dokumen terdekat dengan dokumen uji yang akan diambil.
2. Menghitung jarak antara data baru di setiap label data (jarak *euclidean*) dengan jarak semua data *training*. Untuk menghitung tingkat kesamaan dalam dokumen ini menggunakan *Euclidean distance*. Cara perhitungan *euclidean distance* adalah dengan rumus berikut.

$$D(X, Y) = \sqrt{\sum_{k=1}^n (X_k - Y_k)^2} \quad (2)$$

Keterangan (2):

D = Jarak antara dua titik x dan y

X = Data uji

Y = Sampel data

n = Dimensi data

3. Kemudian urutkan hasil *euclidean distance* berdasarkan jarak K yang ditentukan jika K=3 artinya akan dipilih 3 jarak terkecil dari hasil *euclidean distance*.

- Selanjutnya gunakan mayoritas atribut kelas pada 3 tetangga terdekat yang sudah dipilih untuk menentukan prediksi kelas pada data baru tersebut. Misalkan pada 3 tetangga terdekat memiliki 2 atribut kelas positif dan 1 atribut kelas negatif maka kelas pada data baru tersebut adalah positif.

**F. Penghitungan akurasi menggunakan Confusion Matrix**

Setelah dilakukan proses klasifikasi dengan menggunakan *K-Nearest Neighbor* selanjutnya akan dilakukan proses penghitungan akurasi menggunakan *confusion matrix*. Langkah pengujian untuk menghitung akurasi dengan *confusion matrix* adalah sebagai berikut [6]:

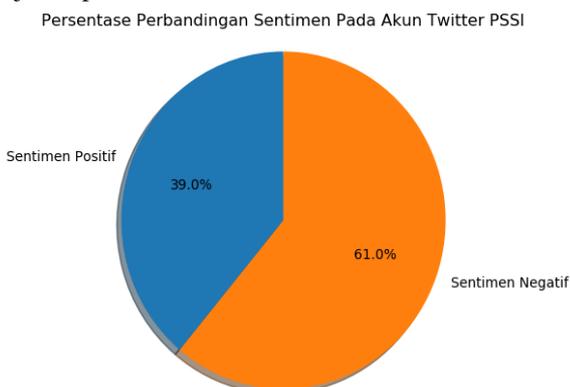
- Menghitung jumlah hasil data asli positif dan data klasifikasi positif (TP).
- Menghitung jumlah hasil data asli positif dan data klasifikasi negatif (FN).
- Menghitung jumlah hasil data asli negatif dan data klasifikasi negatif (TN).
- Menghitung jumlah hasil data asli negatif dan data klasifikasi positif (FP).
- Setelah diketahui jumlah pada masing-masing langkah 1-4, kemudian jumlahkan TP dan TN
- Selanjutnya jumlahkan semua TP, FN, TN dan FP.
- Lakukan pembagian dari langkah 5 dan 6 kemudian hasil bagi dikalikan 100.

Adapun rumus untuk perhitungan *confusion matrix* untuk mencari akurasi adalah sebagai berikut :

$$Akurasi = \frac{TP + TN}{TP + TN + FP + FN} \quad (3)$$

**IV. HASIL DAN PEMBAHASAN**

Data pada penelitian ini diambil dari media sosial *twitter* memanfaatkan *Twitter API* dengan menggunakan *query '@PSSI'*. Periode pengumpulan data dilakukan pada 21 februari – 04 Mei 2019 kemudian data dilakukan sorting untuk setiap *tweet* yang memiliki sentimen yang akan ditambahkan ke dalam dataset yang digunakan untuk penelitian analisis sentimen ini yaitu sejumlah 2000 *tweet*. Kemudian data tersebut diberi label sentimen negatif dan positif, data diharuskan mempunyai label dikarenakan dalam penelitian analisis sentimen dengan metode *K-Nearest Neighbor* ini merupakan *supervised learning*. Dari 2000 *tweet* terdapat 790 *tweet* positif dan 1210 *tweet* negatif , sehingga dapat disimpulkan bahwa dari data yang diambil dari akun *twitter* pssi ini cenderung negatif. Adapun proporsi data yang digunakan pada penelitian ini seperti ditunjukkan pada Gambar 9.



Gambar. 9. Diagram pie persentase sentimen positif dan negatif dari dataset

Berdasarkan Gambar 9, data dengan sentimen negatif lebih banyak dibandingkan data dengan sentimen positif dikarenakan pada periode pengumpulan data 21 februari – 04 Mei 2019 ditemukan lebih banyak opini dari pengguna *twitter* terhadap polemik yang terjadi pada persepakbolaan Indonesia. Setelah data dilakukan *preprocessing* dari 2000 *tweet* ditemukan 5843 kata yang siap untuk diolah. Setelah kata hasil dari *preprocessing* dikumpulkan selanjutnya akan dilakukan penghitungan untuk pembobotan kata dengan TF-IDF seperti pada Tabel 2.

TABEL II  
PEMBOBOTAN KATA DENGAN TF-IDF

D1 = selamat ulang tahun forza indonesia forza pssi moga federasi lebih baik bangga						
D2 = selamat pssi menang						
Term	D1	D2	DF	IDF	TFIDF	
					D1	D2
selamat	1	1	2	0,1761	0,1761	0,1761
ulang	1	0	1	0,4771	0,4771	0
tahun	1	0	1	0,4771	0,4771	0
forza	2	0	2	0,1761	0,3522	0
indonesia	1	0	1	0,4771	0,4771	0
pssi	1	1	2	0,1761	0,1761	0,1761
moga	1	0	1	0,4771	0,4771	0
federasi	1	0	1	0,4771	0,4771	0
lebih	1	0	1	0,4771	0,4771	0
baik	1	0	1	0,4771	0,4771	0
bangga	1	0	1	0,4771	0,4771	0
menang	0	1	1	0,4771	0	0,4771

Pada Tabel 2 dapat diperhatikan cara penghitungan untuk mencari bobot pada tiap dokumen, dengan menghitung jumlah kata yang muncul pada sebuah dokumen (TF) kemudian jumlah dokumen yang memiliki kata (DF). Setelah nilai DF didapatkan kemudian hitung nilai IDF dengan rumus  $\log=N/df$ , dimana N merupakan jumlah seluruh dokumen yang ada.

TABEL III  
AKURASI NILAI K

Nilai K	Akurasi %	Nilai K	Akurasi %
23	79.99	15	78.65
19	79.89	11	77.39
21	79.75	9	76.89
25	79.75	7	76.20
27	79.75	5	74.34
29	79.6	1	67.00
17	79.54	3	65.75
13	79.00		

Setelah pembobotan kata kemudian dilakukan validasi dengan *k-fold cross validation* menggunakan 10 *fold*, klasifikasi KNN, dan penghitungan akurasi dengan *confusion matrix*. Dilakukan pengujian untuk mencari akurasi terbaik pada nilai  $k=1, k=3, k=5, k=7, k=9, k=11, k=13, k=15, k=17, k=19, k=21, \text{ dan } k=23, k=25, k=27, k=29$ . Untuk lebih jelas, dapat diperhatikan Tabel 3.

Berdasarkan Tabel 3, akurasi nilai K didapatkan hasil optimal nilai K pada  $K=23$  dengan akurasi yang didapatkan adalah 79.99%. Dari hasil nilai akurasi mulai dari  $K=1$  hingga  $K=19$  semakin banyak nilai  $k$  maka semakin tinggi juga akurasinya, namun pada  $K=21$  hingga  $K=29$  tidak selalu semakin tinggi nilai K semakin tinggi akurasinya. Jumlah akurasi dan nilai K juga dipengaruhi dari banyaknya dokumen jika semakin banyak dokumen maka untuk mendapatkan nilai akurasi K optimal perlu semakin banyak nilai K nya. Hal tersebut dikarenakan semakin banyak dokumen akan semakin banyak pula fitur atau jumlah kata. Gambar 10 dapat diperhatikan untuk proses dari validasi menggunakan *10-fold cross validation* dan evaluasi *confusion matrix* pada nilai  $K=23$ .

K = 23			
	Accuraction	Error Rate	Confussion Matrix
0	0.635	0.365	[[34, 6], [67, 93]]
1	0.735	0.265	[[78, 16], [37, 69]]
2	0.775	0.225	[[109, 20], [25, 46]]
3	0.835	0.165	[[150, 19], [14, 17]]
4	0.835	0.165	[[156, 5], [28, 11]]
5	0.865	0.135	[[123, 6], [21, 50]]
6	0.880	0.120	[[158, 11], [13, 18]]
7	0.850	0.150	[[91, 8], [22, 79]]
8	0.795	0.205	[[76, 9], [32, 83]]
9	0.795	0.205	[[120, 15], [26, 39]]
Accuracy : 0.7999999999999999			
Average error rate : 0.2			

Gambar. 10. Hasil Validasi dan Evaluasi  $K=23$

Gambar 10 menunjukkan proses validasi menggunakan *10-fold cross validation* pada nilai  $K=23$  dimana dilakukan proses validasi silang dengan 10 kali iterasi. Setiap iterasi atau *fold* berisi 1800 *tweet* yang digunakan sebagai data latih dan 200 *tweet* yang digunakan sebagai data uji/validasi yang diklasifikasikan dengan KNN. Kemudian untuk menghitung hasil akurasi dari klasifikasi KNN maka akan dilakukan penghitungan pada tabel *confusion matrix*.

TABEL IV  
CONFUSION MATRIX

Aktual	Klasifikasi KNN	
	Positif	Negatif
Positif	1095	115
Negatif	285	505

Tabel 4 menunjukkan tabel *confusion matrix* dari nilai  $K=23$  yang sudah dilakukan validasi menggunakan *10-fold cross validation* dengan porsi setiap *fold* sebanyak 200 data sehingga menghasilkan 2000 data uji validasi untuk *10-fold*. Adapun uraian hasil proses pembentukan *confusion matrix* sebagai berikut:

1. Hasil aktual positif yang diprediksi positif oleh KNN (TP) sebanyak data 1095 data.

2. Hasil aktual positif yang diprediksi negatif oleh KNN (FN) sebanyak 115 data.
3. Hasil aktual negatif yang diprediksi positif oleh KNN (FP) sebanyak 285 data.
4. Hasil aktual negatif yang diprediksi negatif oleh KNN (TN) sebanyak 505 data.

Penghitungan dari tabel *Confussion Matrix* total dari *10-fold cross validation* untuk mencari akurasi dan *error rate* adalah.

$$\text{Akurasi} = \frac{TP + TN}{TP + TN + FP + FN} = \frac{1095 + 505}{1095 + 505 + 285 + 115} = 0,80$$

$$\text{Error Rate} = \frac{FP + FN}{TP + TN + FP + FN} = \frac{285 + 115}{1095 + 505 + 285 + 115} = 0,20$$

Akurasi adalah tingkat keberhasilan klasifikasi untuk memprediksi data aktual yang terdiri dari data dengan label positif dan negatif. *Error rate* adalah tingkat kesalahan klasifikasidalam memprediksi data aktual.

Dilakukan pengujian sistem dengan nilai  $K=23$ , pengujian ini diperlukan untuk membuktikan bahwa sistem dapat menganalisis sentimen dengan baik. Dengan dimasukkannya *tweet* baru hasil *crawling* dari *twitter* pada akaun *twitter* resmi PSSI terkait topik persepakbolaan Indonesia untuk dilakukan analisis sentimen secara otomatis adapun hasil pengujian dapat diperhatikan pada Tabel 5.

TABEL V  
HASIL PENGUJIAN SISTEM

No.	Tweet	Aktual	Prediksi
1	Congrats utk semua pemain,coach @indra_sjafri Dan staff dan utk @PSSI Juga kalian pantas di apresiasi <a href="https://t.co/rKCpfkxMuo">https://t.co/rKCpfkxMuo</a>	Positif	Positif
2	Semoga semakin Jaya dan Berprestasi.\n\n#PSSI <a href="https://t.co/OZTEIUdVv">https://t.co/OZTEIUdVv</a>	Positif	Positif
3	@PSSI Optimis Indonesia mendapat pelajaran berharga dari Pertandingan semalam	Positif	Positif
4	walau kalah kalian tetap membanggakan @PSSI	Positif	Negatif
5	@PSSI udahlah mau ada mafia mau enggak, emang pemaen timnas indo jelek jelek. percuma juga kalo ngarep prestasi cuma gara gara berantas mafia wkwkwk.	Negatif	Negatif
6	paling kalah lagi wkwk udah jadi tradisi tiap tahunnya kalah @PSSI	Negatif	Negatif
7	@PSSI Pengurus pssi bobrok prestasi timnas amburadul'	Negatif	Negatif
8	Banyak kasus, organisasi gajelas, gapunya uang. Sosoan pgn ngehibur. Nyadar ga sih masyarakat udah jijik liat kelakuan kalian? @PSSI <a href="https://t.co/ITbmPDLI8y">https://t.co/ITbmPDLI8y</a>	Negatif	Negatif
9	@PSSI kita lemahnya di finising,crosing,taktikal dll.kita hrs banyak latihan suting krn bnyk melebar, melambung ,crosing bnyk tdk tpt sasaran maka dari itu kt hrs latihan yg keras semuaitu pasti ada ilmu nya , knp tdk tpt,melambung	Negatif	Negatif
10	Clear bisa bersihin ketombe sama bisa bersihin orang" busuk di @PSSI gak??!!	Negatif	Negatif

Berdasarkan Tabel 5, dapat dibuktikan bahwa sistem terbukti dapat menganalisis sentimen dengan baik. Dari 10 *tweet* baru tanpa label yang dilakukan pengujian terdapat 1 *tweet* yang tidak sesuai prediksinya.

## V. KESIMPULAN DAN SARAN

Berdasarkan penelitian yang telah dilakukan, maka diperoleh kesimpulan bahwa:

1. Dari 2000 data *tweet* terkait polemik persepakbolaan Indonesia yang diambil dari akun *twitter* PSSI dengan proporsi 790 *tweet* positif dan 1210 *tweet* negatif, dilakukan percobaan untuk mencari model KNN dengan akurasi terbaik menggunakan range nilai  $k=1$  hingga  $k=30$  yang adalah bilangan ganjil, didapatkan akurasi optimal pada  $k=23$  dengan akurasi sebesar 79,99% dan error rate sebesar 20,01%.
2. Analisis sentimen pengguna *twitter* terhadap topik sepak bola Indonesia menggunakan pembobotan TF-IDF dan metode KNN berhasil dilakukan. Dengan dibuktikan saat pengujian model KNN dengan diberikan nilai  $K=23$ , dari 10 *tweet* baru yang dilakukan pengujian untuk mendapatkan sentimen hanya 1 *tweet* yang tidak sesuai prediksinya.

## DAFTAR PUSTAKA

- [1] R. Hidayatillah, M. Mirwan, M. Hakam, and A. Nugroho, "Levels of Political Participation Based on Naive Bayes Classifier," *IJCCS (Indonesian J. Comput. Cybern. Syst.*, vol. 13, no. 1, p. 73, 2019.
- [2] R. A. Setiawan and D. B. Setyohadi, "Analisis Komunikasi Sosial Media Twitter sebagai Saluran Layanan Pelanggan Provider Internet dan Seluler di Indonesia," *J. Inf. Syst. Eng. Bus. Intell.*, vol. 3, no. 1, p. 16, 2017.
- [3] B. J. M. Putra, A. Helen, and A. R. Barakbah, "Rule-based Sentiment Degree Measurement of Opinion Mining of Community Participatory in the Government of Surabaya," *Emit. Int. J. Eng. Technol.*, vol. 6, no. 2, p. 200, 2018.
- [4] C. R. Vinodhini, G., "Sentiment Analysis and Opinion Mining: A Survey," *Int. J. Adv. Res. Comput. Sci. Softw. Eng.*, vol. 2, no. 6, 2012.
- [5] R. R. Sani, J. Zeniarza, and A. Luthfiarta, "Pengembangan Aplikasi Penentuan Tema Tugas Akhir Berdasarkan Data Abstrak Menggunakan Algoritma K-nearest Neighbor," *Proceeding SENDI\_U*, no. 207, pp. 103–111, 2016.
- [6] A. Salam, J. Zeniarja, and R. S. U. Khasanah, "Analisis Sentimen Data Komentar Sosial Media Facebook Dengan K-Nearest Neighbor ( Studi Kasus Pada Akun Jasa," *Pros. SINTAK*, pp. 480–486, 2018.
- [7] N. MOH, "Klasifikasi Dokumen Komentar Pada Situs Youtube Menggunakan Algoritma K-Nearest Neighbor (K-Nn)," *Univ. Dian Nuswantoro*, no. 5, 2016.
- [8] B. Liu, "Sentiment Analysis and Opinion Mining Morgan & Claypool Publishers," *Lang. Arts Discip.*, no. May, p. 167, 2012.
- [9] V. Amrizal, "Penerapan Metode Term Frequency Inverse Document Frequency (Tf-Idf) Dan Cosine Similarity Pada Sistem Temu Kembali Informasi Untuk Mengetahui Syarah Hadits Berbasis Web (Studi Kasus: Hadits Shahih Bukhari-Muslim)," *J. Tek. Inform.*, vol. 11, no. 2, pp. 149–164, 2019.
- [10] A. Lüscho and C. Wartena, "Classifying medical literature using k-nearest-neighbours algorithm," *CEUR Workshop Proc.*, vol. 1937, no. Ddc, pp. 26–38, 2017.
- [11] I. Hemalatha, G. S. Varma, and A. Govardhan, "Preprocessing the Informal Text for Efficient Sentiment Analysis," *Int. J. Emerg. Trends Technol. Comput. Sci.*, vol. 1, no. 2, pp. 58–61, 2012.
- [12] T. Winarti, J. Kerami, and S. Arief, "Determining Term on Text Document Clustering using Algorithm of Enhanced Confix Stripping Stemming," *Int. J. Comput. Appl.*, vol. 157, no. 9, pp. 8–13, 2017.
- [13] M. Subhan, A. Sudarsono, and A. R. Barakbah, "Classification of Radical Web Content in Indonesia using Web Content Mining and k-Nearest Neighbor Algorithm," *Emit. Int. J. Eng. Technol.*, vol. 5, no. 2, p. 328, 2018.
- [14] S. E. Syahfitri Kartika Lidya, Opim Salim Sitompul, "Sentiment Analysis Pada Teks Bahasa Indonesia Menggunakan Support Vector Machine ( Svm )," *Semin. Nas. Apl. Teknol. Inf. 2015 (SNATI 2015)*, vol., no., pp. 1–8, 2015.
- [15] C. K. Park and D. G. Kim, "Historical background," *Curr. Futur. Manag. Brain Metastasis*, vol. 25, pp. 1–12, 2012.
- [16] S. Ernawati and R. Wati, "Penerapan Algoritma K-Nearest Neighbors Pada Analisis Sentimen Review Agen Travel," *J. Khatulistiwa Inform.*, vol. VI, no. 1, 2018.

**Jeremy Andre Septian.** Mahasiswa semester 8 jurusan Teknik Informatika, Universitas Narotama, Surabaya. Dia bekerja sebagai admin di salah satu perusahaan *trading* besi beton di Surabaya. Minat penelitian di bidang *text minning* dan *data minning*.

**Tresna Maulana Fahrudin.** Menyelesaikan jenjang D4 dan S2 di Politeknik Elektronika Negeri Surabaya di Program Studi Teknik Informatika. Saat ini berkarir sebagai Dosen Tetap Fakultas Ilmu Komputer, Universitas Narotama Surabaya. Minat bidang penelitiannya adalah *Data Mining*, *Machine Learning*, *Metaheuristic* dan *Text Mining*.

**Aryo Nugroho.** Menyelesaikan jenjang S1 dibidang Teknik Sipil dan Teknik Informatika kemudian menyelesaikan S2 di ITS. Berkarir sebagai Dosen Tetap di Fakultas Ilmu Komputer, Universitas Narotama Surabaya. Sejak 2014, telah mengambil Program Doktor di Pascasarjana Teknik Elektro ITS. Minat penelitiannya di bidang *Text Mining*, *IT Strategic*, *Artificial Intelligent*, dan *Business Intelligence*.

# INSYST

Journal of Intelligent System and Computation

Volume 01 Nomor 01 Oktober 2019

---

## Author Guidelines

- Manuscript should be written in Indonesia and be submitted online via journal website. Online Submission will be charged at no Cost
- Manuscript should not exceed 15 pages including embedded figures and tables, without any appendix, and the file should be in Microsoft Office (.doc/.docx). [download template](#)
- Title, Abstract and Keywords must be written in bilingual
- Title should be less than 15 words
- Abstracts consists of no more than 200 words, contains the essence of the article and includes a brief background, objectives, methods and results or findings of the study. Abstract is written in one paragraph.
- Keywords are written in Indonesia and English three to five words/phrases, separated with coma and consist of important words/phrases from the article.
- Author's name, affiliation, affiliation address and email. State clearly and include country's name on your affiliation address.
- The main text of the writing should be consists of: Introduction, Method, Result and Discussion, and Conclusion; followed by Acknowledgment and Reference
- Introduction State adequate background, issues and objectives, avoiding a detailed literature survey or a summary of the results. Explain how you addressed the problem and clearly state the aims of your study.
- Used method is the scientific in the form of study of literature, observation, surveys, interviews, Focus Group Discussion, system testing or simulation and other techniques commonly used in the world of research. It is also recommended to describe analysis techniques used briefly and clearly, so that the reader can easily understand.
- Results should be clear, concise and not in the form of raw data. Discussion should explore the significance of the results of the work, not repeat them. Avoid extensive citations and discussion of published literature. INSYST will do the final formatting of your paper.
- Conclusion should lead the reader to important matter of the paper. Authors are allowed to include suggestion or recommendation in this section. Write conclusion, suggestion and/or recommendation in narrative form (avoid of using bulleting and numbering)
- Acknowledgments. It is highly recommended to acknowledge a person and/or organizations helping author(s) in many ways. Sponsor and financial support acknowledgments should be included in this section. Should you have lots of parties

to be acknowledged, state your acknowledgments only in one paragraph. Avoid of using bulleting and numbering in this section

- The number of references are not less than 10 with at least 8 primary references. Primary references are include journal, thesis, dissertation and all kinds of research reports. All refferences must come from source published in last 7 years.
- Figure and table should be in black and white, and if it is made in color, it should be readable when it is later printed in black and white.
- Figure and table should be clearly readable and in a proportional measure to the overall page.

### **Tim Redaksi**

Journal of Intelligent System and Computation

Departement of Informatics

Institut Sains dan Teknologi Terpadu Surabaya

Jl. Ngagel Jaya Tengah 73-77 Surabaya

Email: [insyst@istts.ac.id](mailto:insyst@istts.ac.id)

Website: <https://jurnal.stts.edu/index.php/INSYST/index>