

# SEGMENTASI METODE PARTICLE SWARM OPTIMIZATION PADA PENDETEKSI CITRA GESTUR TANGAN

*by Rohmat Syamsul Huda*

---

**Submission date:** 13-Aug-2022 11:35PM (UTC-0700)

**Submission ID:** 1882230111

**File name:** SKRIPSI\_SHINTA\_turnitin.pdf (950.93K)

**Word count:** 5875

**Character count:** 35766

**PENDAHULUAN****A. Latar Belakang**

Selama ini perkembangan teknologi telah berkembang pesat. Salah satu perkembangan teknologi yaitu <sup>21</sup> Computer Vision. Computer Vision adalah sebuah kemampuan komputer yang ke desain agar mampu melihat sebuah objek sehingga mampu menampilkan objek digital dan bisa mengoleksi data secara visual (Purno & Wibowo, 2016).

Perkembangan destinasi wisata di Indonesia tidak lepas dari perkembangan teknologi (Hartini, 2019). Pengunjung wisata yang membludak mengakibatkan keramaian (BPS, 2020). Umumnya keramaian yang tidak terkendali dapat memicu banyak faktor yang mengakibatkan situasi bahaya seperti pencurian, tindakan kekerasan dan sebagainya. Untuk menghindari situasi tersebut dapat dilakukan monitoring CCTV. Namun hal tersebut kurang efektif karena masih menggunakan peran manusia dalam monitoring CCTV tersebut.

<sup>29</sup> Closed Circuit Television (CCTV) pun sangatlah penting dalam kehidupan manusia saat ini. CCTV adalah perangkat video kamera digital yang mengirim sinyal disuatu ruangan untuk memantau suatu kondisi pada tempat tersebut. CCTV banyak digunakan di salah satu tempat seperti sekolah, perusahaan dll untuk memantau adanya keamanan atau adanya tindakan tawuran CCTV ini belum dapat mendeteksi pergerakan obyek yang

maksimal. (Hasanah, 2020)

Penelitian sebelumnya yang berjudul *Seleksi Fitur Dengan Particle Swarm Optimization Untuk Pengenalan Pola Wajah Menggunakan Naïve Bayes*. Hasil pada penelitian ini adalah Sistem presensi mahasiswa pada Fakultas Ilmu Komputer. Universitas Brawijaya masih menggunakan sistem manual yang sangat rawan disalahgunakan oleh mahasiswa itu sendiri seperti menitipkan absen pada temannya. Metode optimasi merupakan metode untuk pencarian solusi yang lebih cepat. Sedangkan metode klasifikasi adalah metode yang sangat erat kaitannya dengan hipotesis probabilitas (Hakim, 2017).

Pada penelitian selanjutnya oleh Nurul Lailatul Khikmah tahun 2021 menghasilkan sistem perbaikan citra gambar menggunakan metode *Particle Swarm Optimization*. Sistem ini digunakan untuk memperbaiki citra tangan hingga tampilan lebih jelas namun kurangnya mendeteksi citra tangan pada CCTV (Khikmah, 2021).

Berdasarkan permasalahan diatas peneliti tertarik untuk membuat sistem deteksi Segmentasi Metode *Particle Swarm Optimization* pada pendeteksi citra gestur tangan. Penelitian ini penting untuk menjaga situasi wisata agar tetap kondusif, sehingga pengunjung aman.

## **B. Identifikasi Masalah**

Dari penjelasan latar belakang diatas memiliki identifikasi masalah sebagai berikut:

1. Keramaian cenderung menimbulkan ancaman tindak bahaya.
2. Monitoring *CCTV* masih melibatkan peran manusia sehingga penggunaan teknologi kurang efektif.

### C. Rumusan Masalah

Dari uraian latar belakang dan identifikasi masalah tersebut dapat diperoleh rumusan masalah sebagai berikut:

1. Bagaimana mengimplementasikan metode *Particle Swarm Optimization* dari citra gambar untuk untuk kode deteksi adanya tindakan bahaya?
2. Bagaimana cara metode *Particle Swarm Optimization* mendeteksi citra tangan agar hasil lebih jelas?

### D. Batasan Masalah

Batasan-batasan masalah yang didapat dari rumusan masalah diatas adalah:

1. Gestur tangan yang diamati hanya bagian telapak tangan terbuka dan tertutup.
2. Penelitian ini menggunakan metode *PSO (Particle Swarm Optimization)*.
3. Menggunakan Bahasa pemrograman *Python*.
4. Pengguna harus berada ditempat yang cenderung terang untuk dapat menggunakan sistem ini.
5. Uji coba dilakukan menggunakan kamera *webcam*

### <sup>30</sup> E. Tujuan Penelitian

Dari penjelasan rumusan masalah diatas, maka tujuan penelitian diperoleh sebagai berikut:

1. Untuk mengimplementasikan metode *Partcile Swarm Optimization* terhadap citra gestur tangan untuk mendeteksi adanya Tindakan bahaya.
2. Untuk mengetahui hasil yang lebih jelas oleh sistem dalam mengidentifikasi citra tangan dengan metode *Partcile Swarm Optimization*.

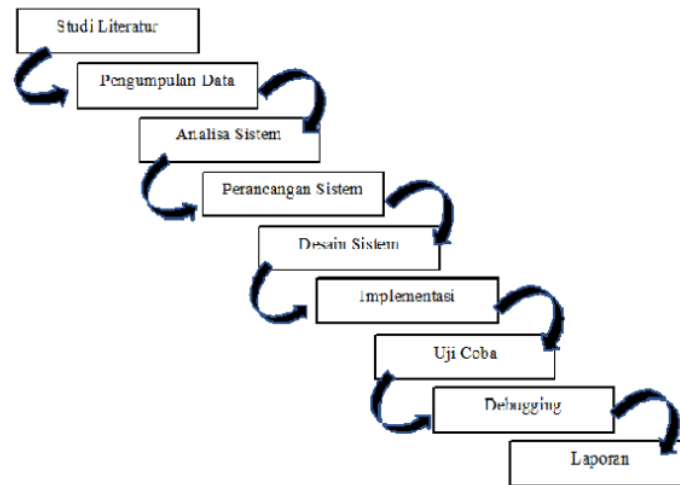
### F. Manfaat Dan Kegunaan Penelitian

Berikut adalah manfaat dan juga kegunaan dari penelitian yang diperoleh oleh penulis:

1. Pada penelitian ini bertujuan dapat membantu instansi atau perusahaan yang sedang memantau apapun kejadian yang berlangsung pada suatu pergerakan obyek pada *CCTV*.
2. Pada saat penangkapan sebuah obyek pada *CCTV* kurang maksimal hasil yang diperoleh, maka pada penelitian kali ini dibuatlah sistem pengolahan citra pada gesture tangan dengan menggunakan metode *Particle Swarm Optimization*.

### <sup>22</sup> G. Metode Penelitian

Dalam prosedur penelitian ini penulis melakukan beberapa metode untuk diterapkan dalam penelitian ini diantaranya sebagai berikut :



Gambar 1.1 Diagram Waterfall

Berdasarkan gambar diatas dapat dipaparkan penjelasan sebagai berikut :

#### 1. Studi Literatur

Penulisan ini dimulai langkah mencari referensi jurnal atau artikel yang berhubungan dengan citra dan metode apa yang dipakai pada penelitian ini. Kemudian dari jurnal-jurnal tersebut dibuatlah *review* jurnal perbandingan.

#### 2. Pengumpulan Data

Pengumpulan data ini dilakukan dan diperoleh dari pembelajaran studi literatur menggunakan metode PSO. Pada penelitian ini mengambil data gambar tangan mengepal dan terbuka untuk bahan penelitian. Data input adalah sebuah perintah yang mengisi memori komputer untuk diproses menuju processor. Pada penelitian ini menggunakan citra tangan membuka dan menutup untuk bahan

penelitian. Data citra tangan diperoleh melalui proses foto dari tangan peneliti dan juga beberapa orang disekitar peneliti.

### 3. Perancangan Sistem

Perancangan sistem ini sesuai dengan studi literatur dan dibuat alur yang rinci kemudian menentukan algoritma yang cocok untuk penelitian ini.

### 4. Desain Sistem

Desain Sistem ini dimulai dengan membuat proses Training dan Testing. Kemudian perancangan yang dibuat akan diimplementasikan sesuai desain sistem yang dibuat.

### 5. Implementasi

Hasil dari perancangan sistem sebelum dilakukan pengujian akan diimplementasikan melalui sebuah kode berupa bahasa pemrograman *Python*.

### 6. Uji Coba

Pada proses uji coba lebih ditekankan pada logika sistem bawa semua hal yang akan diuji telah dilakukan. Setelah selesai melewati tahap implementasi, maka pada tahap ini akan dilakukan uji coba terhadap sistem. Proses uji coba berfokus pada logika internal software dan eksternal fungsional.

### 7. Debugging

Apabila pada tahap uji coba dijumpai kesalahan pada program, maka akan dilakukan perbaikan pada program tersebut dengan cara

menghilangkan *bug* atau kesalahan yang terdapat dalam program tersebut.

#### 8. Laporan

Dalam penyusunan laporan hasil analisis yang diperoleh dari pengumpulan data, merancang sistem, desain sistem dan implementasi disertai dengan kesimpulan.

### H. Jadwal Penelitian

Penelitian dan juga perancangan proyek akhir memiliki deskripsi jadwal yang digunakan sebagai berikut:

16  
Tabel 1.1 Jadwal Penelitian

Jenis Kegiatan	Bulan Ke-																							
	1				2				3				4				5				6			
	1	2	3	4	1	2	3	4	1	2	3	4	1	2	3	4	1	2	3	4	1	2	3	4
Study Literatur	■	■	■	■																				
Pengumpulan Data		■	■	■	■	■	■	■																
Perancangan Sistem					■	■	■	■	■	■	■	■												
Desain Sistem									■	■	■	■	■	■	■	■								
Implementasi													■	■	■	■	■	■	■	■				
Laporan																	■	■	■	■	■	■	■	■



## I. <sup>47</sup> **Sistematika Penulisan Laporan**

Skripsi ini meliputi 5 bab dengan isi tiap bab <sup>17</sup> sebagai berikut:

### **BAB I PENDAHULUAN**

Bab ini berisi latar belakang masalah, identifikasi masalah, rumusan masalah, batasan masalah, tujuan penelitian, manfaat dan kegunaan, metode penelitian, jadwal penelitian, dan sistematika penulisan.

### **BAB II TINJAUAN PUSTAKA**

Bab ini menjelaskan tentang landasan teori, kajian pustaka, dan desain sistem yang mana menguraikan dasar-dasar teori yang terkait dalam penelitian serta desain rancangan <sup>31</sup> sistem yang akan dibuat.

### **BAB III ANALISA DAN DESAIN SISTEM**

Bab ini berisi Analisa Sistem yang mana adalah mengidentifikasi masalah, <sup>37</sup> dan desain sistem yang akan dibuat.

### **BAB IV IMPLEMENTASI**

Bab ini berisi tentang Implementasi Lembar Kerja, Keterkaitan Lembar Kerja, Implementasi Program, Pengujian Sistem, Hasil, dan Evaluasi Hasil.

### <sup>26</sup> **BAB V PENUTUP**

Bab ini merupakan bab terakhir yang mana berisi tentang Kesimpulan atau ringkasan tentang penelitian yang telah dilakukan, dan Saran yang berisi harapan-harapan untuk kesempurnaan <sup>45</sup> penelitian.

## **BAB II**

### **TINJAUAN PUSTAKA**

#### **A. Landasan Teori**

##### 1. <sup>44</sup> *Closed Circuit Television (CCTV)*

Menurut Tantoni (2021), *Closed Circuit Television (CCTV)*

dapat didefinisikan sebagai berikut:

<sup>6</sup> *Closed Circuit Television (CCTV)* yaitu pertumbuhan mobilitas masyarakat saat ini semakin pesat, terutama di kota-kota besar. Bagi mereka yang beroperasi banyak dan memiliki banyak bisnis di banyak tempat, mungkin sulit untuk melacak karyawan dari setiap bisnis yang mereka miliki, pada umumnya *CCTV* sering digunakan untuk memantau bank, hotel, perusahaan, pabrik dll.

Menurut Hasanah (2020), *Closed Circuit Television* atau

disingkat *CCTV* dapat didefinisikan sebagai berikut:

<sup>6</sup> *Closed Circuit Television* atau disingkat *CCTV* adalah sebuah perangkat kamera video digital yang digunakan untuk mengirim sinyal ke layar monitor ditempat tertentu dengan tujuan untuk memantau situasi dan kondisi suatu tempat, sehingga dapat mencegah terjadinya kejahatan. Jika sebuah perusahaan tidak diawasi dan dipantau, maka tidak menutup kemungkinan karyawan akan bekerja diluar standaryang ditargetkan, sehingga dapat menimbulkan kerugian diperusahaannya (Hasanah, 2020).

##### 2. Citra

Menurut Hestiningsih (2008), Citra dapat didefinisikan sebagai

berikut :

<sup>8</sup> Citra adalah suatu representasi, kemiripan, atau imitasi dari suatu objek atau benda misal foto seseorang mewakili entitas diri orang tersebut di depan kamera, foto sinar-X thorax mewakili keadaan bagian dalam tubuh seseorang, data dalam suatu file BMP mewakili apa yang digambarkannya

Menurut Utari (2016), Citra dapat didefinisikan seperti berikut

yaitu:

<sup>4</sup> Citra adalah suatu representasi (gambaran), kemiripan, atau imitasi dari suatu objek. Citra sebagai keluaran atau sistem perekaman data dapat bersifat optik berupa foto, bersifat analog berupa sinyal – sinyal video seperti gambar pada monitor televisi, atau bersifat digital yang dapat langsung disimpan pada suatu media penyimpanan. Citra dapat dikelompokkan menjadi dua bagian yaitu citra diam (still images) dan citra bergerak (moving images). Citra diam adalah citra tunggal yang tidak bergerak. Sedangkan citra bergerak adalah rangkaian citra diam yang ditampilkan secara beruntun (sekuensial) sehingga memberi kesan pada mata sebagai gambar yang bergerak. Setiap citra didalam rangkaian itu disebut frame.

### 3. Citra Digital

Menurut Guojun (1999), Citra digital dapat didefinisikan seperti

berikut:

<sup>8</sup> Citra digital adalah citra yang disimpan dalam format digital (dalam bentuk file). Citra digital dapat didefinisikan sebagai fungsi  $f(x,y)$  dimana  $x$  dan  $y$  adalah koordinat spasial dan nilai  $f(x,y)$  adalah intensitas citra pada koordinat tersebut.

Menurut Utari (2016), Citra digital dapat didefinisikan seperti

berikut yaitu:

Citra bisa dikatakan menjadi gambaran digital apabila gambaran tadi disimpan pada format digital (pada bentuk file). Citra digital didapatkan melalui proses digitalisasi terhadap gambaran kontinyu. Citra merupakan nomor (*image is just a number*), berdasarkan segi estetika, gambaran atau gambar merupakan gugusan rona yg mampu terlihat indah, mempunyai pola, berbentuk tak berbentuk & lain sebagainya. Citra digital bisa berupa foto udara, penampang lintang (*cross section*) berdasarkan suatu benda, gambar wajah, output tomografi otak & lain sebagainya. Di personal komputer rona bisa dinyatakan, contohnya menjadi nomor pada bentuk skala RGB. Lantaran gambaran merupakan nomor, maka gambaran bisa diproses secara digital.

#### 4. Gestur Tangan

Menurut Harjoko (2014), Gestur Tangan dapat didefinisikan sebagai berikut:

Gestur dapat didefinisikan sebagai gerakan fisik dari tangan, lengan, atau badan yang memberikan pesan ekspresif, dan sistem pengenalan gestur digunakan untuk menterjemahkan dan menjelaskan gerakan sebagai perintah yang berarti.

Semakin ke depan manusia membutuhkan antar muka interaksi dengan komputer yang lebih natural dan mudah digunakan. Antar muka komputer personal telah berkembang dari command line berbasis tek ke bentuk grafis dengan keyboard dan mouse sebagai media input.. Penggunaan gestur tangan memberikan sebuah alternatif alat antar muka Interaksi Manusia dan Komputer (IMK).

Menurut Makahaube (2021), Gesture dapat didefinisikan sebagai berikut:

Gesture adalah bentuk komunikasi non-verbal dengan aksi tubuh yang mengkomunikasikan pesan-pesan tertentu, sebagai pengganti ucapan. Sistem pengenalan gesture tangan terus berkembang, terutama untuk menggantikan peran perangkat masukan seperti mouse, keyboard dan joystick yang digunakan pada interaksi manusia dengan komputer, namun perangkat ini memiliki kelemahan diantaranya tidak alami bagi manusia dan jarak pengontrolan masih cukup dekat. Sistem pengenalan gesture tangan mengimplementasikan cara kerja sistem penglihatan manusia sebagai sensor yang merekam objek berupa gesture tangan dan kemudian mengirim informasi ke otak untuk diproses sehingga menghasilkan sebuah *output*.

#### 5. *Particle Swarm Optimization* (PSO)

Menurut Khikmah (2021), *Particle Swarm Optimization* (PSO) dapat didefinisikan sebagai berikut:

*Particle Swarm Optimization* merupakan metode algoritma pencarian yang menggunakan banyak individu, atau partikel, dikelompokkan dalam segerombolan. Masing-masing partikel ini mewakili solusi kandidat untuk optimasi masalah. Secara konsep, penggunaan *pbest* dan *gbest* pada *Particle Swarm Optimization* mirip dengan operasi *crossover* pada *Genetic Algorithm* (GA). *Particle Swarm Optimization* (PSO) juga menggunakan konsep nilai fitness, sesuai dengan paradigma komputasi evolusioner.

Langkah-langkah algoritma *Particle Swarm Optimization* (PSO)

1 dalam penyelesaian suatu masalah adalah sebagai berikut:

- a. Menentukan jumlah partikel yang akan digunakan.
- b. Menentukan posisi dan kecepatan partikel secara random.
- c. Evaluasi nilai fitness menggunakan rumus yang telah ditentukan sebelumnya dari masing-masing partikel berdasarkan posisinya.
- d. Tentukan partikel dengan fitness terbaik untuk dijadikan sebagai *Gbest*.
- e. *Pbest* awal sama dengan posisi awal.
- f. Menggunakan *Pbest* dan *Gbest* yang ada, perbarui kecepatan partikel dengan rumus berikut :

$$V_t(t) = V_t(t-1)C1R1(XL - X_i(t-1)) + 2R2(XG - X_i(t-1)) \dots\dots\dots(1)$$

Keterangan:

- V = Kecepatan partikel
- VI = Kecepatan partikel pada suatu indeks
- T = Iterasi ke-t
- I = Indeks artikel
- X = Posisi Partikel
- R1 dan R2 = Nilai random dengan range antara 0 sampai 1
- C1 dan C2 = Konstanta yang bernilai positif yang biasanya disebut dengan learning factor
- XL = Local best dari suatu partikel
- XG = Global best dari seluruh kawan

## 6. Bahasa Pemrograman *Python*

Menurut Software Foundation (2016), *Python* dapat didefinisikan sebagai berikut:

Python adalah bahasa pemrograman dinamis yang ditafsirkan, berorientasi objek. Python memiliki struktur data tingkat tinggi, pengetikan dinamis. Python memiliki sintaks yang sederhana dan mudah dipelajari yang menekankan keterbacaan dan mengurangi biaya perbaikan program. Python mendukung modul dan paket untuk mendorong penggunaan kembali program dan kode secara modular. Penerjemah Python dan pustaka standar tersedia secara bebas untuk semua platform dan dapat didistribusikan secara bebas.

Menurut Syahrudin (2018), Python dapat didefinisikan sebagai berikut:

Python adalah bahasa pemrograman interpretatif multiguna dengan filosofi perancangan yang berfokus pada tingkat keterbacaan kode. Python diklaim sebagai bahasa yang menggabungkan kapabilitas, kemampuan, dengan sintaksis kode yang sangat jelas, dan dilengkapi dengan fungsionalitas pustaka standar yang besar serta komprehensif. Python juga didukung oleh komunitas yang besar.

#### **2** **Search Metode:**

- a. *Input* (Memasukan Data) Setiap program yang kita ciptakan nantinya akan berinteraksi dengan user yang mana program akan meminta data yang diperlukan oleh user misalnya dalam menghitung luas persegi panjang tentu diperlukan data berupa panjang dan lebar dari persegi panjang tersebut.
- b. *Output* (Menampilkan Data) Mencetak atau menampilkan informasi data yang dihasilkan oleh algoritma, misalnya dalam algoritma penghitung luas persegi panjang, hasil akhir yang diinginkan adalah luas persegi panjang dengan kata lain algoritma tersebut memiliki satu output yaitu luas persegi panjang.
- c. Penggunaan *input* dan *output* pada *python Input* :

```
Nama_variabel = Input("masukan data :")
```

Penjelasannya, data yang kita masukan nantinya akan tersimpan dalam Nama\_Variabel, namun apabila data yang diinput membutuhkan proses pengolahan kembali atau pengevaluasian maka perintah yang ditulis sebagai berikut:

```
Nama_variabel = eval(input("masukan data
```

Namun ada juga data yang diproses secara otomatis tanpa meminta user untuk menginputkan, yaitu data yang ditulis langsung etika membuat program, penulisannya sebagai berikut:

```
Nama_variabel2 = eval(input("masukan data:"))
```

```
Nama_variabel = isi data
```

Penjelasannya, "Nama\_variabel = isi data" ini artinya ketika program dijalankan maka sistem tidak akan menampilkan perintah tersebut, namun data akan otomatis terbaca oleh sistem. *Output* untuk menampilkan hasil program pada *python*, menggunakan perintah "*print*", penulisan programnya sebagai berikut:

```
Nama_variabel = Input("masukan data :")
```

```
Print("hasil akhir nya adalah :  
",Nama_Variabel)
```

Penjelasannya, perintah "print" akan memanggil data yang tersimpan pada Nama\_variabel (Syahrudin, 2018).

## B. Kajian Pustaka

Pada penelitian ini menggunakan 5 studi literatur yang mendukung penelitian yang akan diteliti, yang meliputi sebagai berikut:

1. Penelitian oleh Emilia Ayu Wijayanti, Volume 5 Nomor 2 Juli 2021 dengan Judul *Perbandingan Algoritma SVM Dan SVM Berbasis Particle Swarm Optimization Pada Klasifikasi Beras Mekongga*.

Hasil pada penelitian ini adalah beras merupakan bahan makanan pokok paling penting di Indonesia. Di karawang jenis beras Mekongga paling banyak diminati dan unggul dibandingkan dengan yang lainnya. Mengklasifikasikan varietas jenis beras dapat dilakukan guna mengidentifikasi jenis dari beras tersebut. Dari hasil penelitian klasifikasi beras mekongga dengan Support Vector Machine dihasilkan nilai akurasi sebesar 46.67% dengan AUC 0.475.

Perbandingan penelitian yang dilakukan adalah jika pada penelitian sebelumnya mengarah pada klasifikasi beras mekongga, maka penelitian ini adalah untuk segmentasi pada obyek tangan.

2. Penelitian oleh Yuni Eka Achyani, Volume 5 Nomor 1 April 2018 dengan judul *Penerapan Metode Particle Swarm Optimization Pada Optimasi Prediksi Pemasaran Langsung*.

Hasil pada penelitian ini adalah Peningkatan akurasi prediksi pemasaran langsung dapat dilakukan dengan cara melakukan seleksi terhadap atribut, karena seleksi atribut mengurangi dimensi dari data sehingga operasi algoritma data mining dapat berjalan lebih efektif dan



lebih cepat. Kemudian dilakukan seleksi atribut dengan menggunakan *Particle Swarm Optimization* dimana atribut yang semula berjumlah 16 variabel prediktor terpilih 12 atribut yang digunakan.

Perbandingan penelitian yang dilakukan adalah apabila penelitian sebelumnya fokus pada optimasi prediksi pemasaran, maka penelitian ini adalah untuk mendeteksi obyek citra pada tangan.

3. Penelitian oleh Satria Habiburrahman Fathul Hakim, Volume 1 Nomor 10 Oktober 2017 dengan judul Seleksi Fitur Dengan *Particle Swarm Optimization* Untuk Pengenalan Pola Wajah Menggunakan Naïve Bayes (Studi Kasus Pada Mahasiswa Universitas Brawijaya Fakultas Ilmu Computer Gedung A).

Hasil pada penelitian ini adalah Sistem presensi mahasiswa pada Fakultas Ilmu Komputer, Universitas Brawijaya masih menggunakan sistem manual yang sangat rawan disalahgunakan oleh mahasiswa itu sendiri seperti menitipkan absen pada temannya. Metode optimasi merupakan metode untuk pencarian solusi yang lebih cepat. Sedangkan metode klasifikasi adalah metode yang sangat erat kaitannya dengan hipotesis probabilitas.

Perbedaan penelitian yang dilakukan adalah Perbandingan pada penelitian sebelumnya fokus pada seleksi fitur pengenalan pola wajah, maka pada penelitian ini untuk obyek gerakan pada tangan.

4. Penelitian oleh Nurul Lailatul Khikmah Juli 2021 dengan judul Perbaikan Citra Gambar Tangan Menggunakan *Particle Swarm*

### *Optimization.*

Hasil pada penelitian ini adalah <sup>1</sup>Salah satu perkembangan teknologi yang sekarang ini cukup pesat adalah pengolahan citra digital. <sup>1</sup>Namun, pada permasalahan umumnya hasil citra dari tangkapan sebuah *Closed Circuit Television* tampak kurang tajam, jelas dan sulit untuk dipahami oleh manusia ataupun petugas yang memantau *CCTV*. Untuk identifikasi objek citra gambar tangan.

Perbandingan penelitian yang dilakukan adalah apabila penelitian sebelumnya fokus pada perbaikan citra gambar, maka pada penelitian ini adalah untuk segmentasi obyek tangan.

5. Penelitian oleh Kurniati Volume 1 Nomor 3 Juli 2020 dengan judul <sup>2</sup>Penerapan Algoritma *Particle Swarm Optimization* Pada Segmentasi Citra Pengenalan Aksara Bugis.

Hasil pada penelitian ini adalah <sup>2</sup>Aksara Bugis merupakan warisan berharga leluhur Suku Bugis yang perlu dilestarikan keberadaannya. <sup>2</sup>Bugis terutama bagi kalangan remaja yang kini tidak banyak mengetahuinya.

Perbandingan penelitian yang dilakukan adalah Jika penelitian sebelumnya menerapkan pada segmentasi pengenalan aksara bugis, maka penelitian ini fokus pada segmentasi obyek pada tangan.

**ANALISIS DAN DESAIN SISTEM****A. Analisis Sistem**

20

**1. Analisa Sistem yang Diusulkan**

Pada subbab ini akan dijelaskan mengenai gambaran umum dari sistem yang dibangun berdasarkan:

**a. Analisa Kebutuhan Fungsi**

Kebutuhan fungsi yang terdapat dalam program yang dikembangkan antara lain:

- 1) Sistem mampu melakukan segmentasi citra tangan pada background polos.
- 2) Proses detektif cenderung lama karena adanya metode yang telah diproses pada citra tangan.
- 3) Sistem mampu menampilkan citra tangan membuka dan menutup yang akan mengeluarkan peringatan bahaya atau terdeteksi.

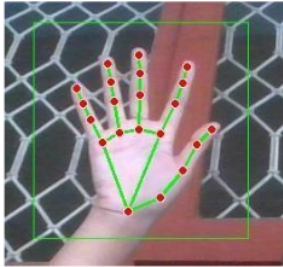
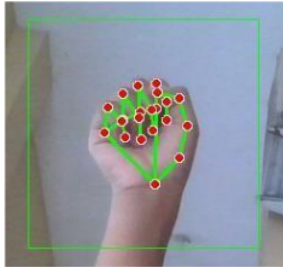
**b. Analisa Kebutuhan Data****1) Data *Input***

Data input adalah sebuah perintah yang mengisi memori komputer untuk diproses menuju processor. Pada penelitian ini menggunakan citra tangan membuka dan menutup untuk bahan penelitian. Data citra tangan diperoleh melalui

18

proses foto dari tangan peneliti dan juga beberapa orang disekitar peneliti.

Tabel 3.1 Data Input

Citra tangan terbuka	Citra tangan tertutup
 <p>A photograph of an open hand with a green skeletal overlay. The skeleton consists of red dots at the joints and green lines connecting them. A green rectangular bounding box is drawn around the hand.</p>	 <p>A photograph of a closed hand with a green skeletal overlay. The skeleton consists of red dots at the joints and green lines connecting them. A green rectangular bounding box is drawn around the hand.</p>

## 2) Gambaran Proses

Langkah mendeteksi citra tangan dengan obyek tangan membuka dan menutup dilakukan dengan menggunakan metode *Particle Swarm Optimization*. Metode ini menggunakan nilai fitness (Khikmah, 2021). Adapun tahapan yang dilakukan pada preprocessing data yaitu :

### a) Gambaran Preprocessing data

Pada pengujian ini menyiapkan data sebelum melakukan segmentasi yaitu preprocessing pada data citra. Adapun tahap preprocessing yang dilakukkan pada penelitian ini yaitu :

#### a. Pertama Input citra tangan

Proses pertama yaitu memasukkan citra tangan asli kode

membuka ke dalam program yang sudah dikerjakan.

b. Deteksi gestur tangan

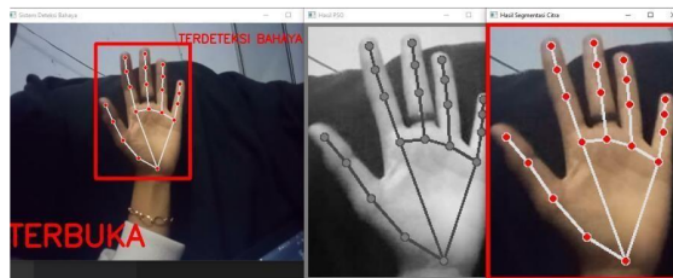
Pada proses kedua ini yaitu citra tangan membuka akan terdeteksi pada frame dan muncul tulisan “TERBUKA”.

c. Citra tangan terbuka “TERDETEKSI”

Pada proses ketiga ini merupakan citra tangan membuka yang telah tampil tulisan “TERBUKA”. Setelah memasukkan ke dalam frame lalu mendapatkan teks “TERDETEKSI” pada layar.

- d. Segmentasi ke dalam <sup>1</sup>metode *Particle Swarm Optimization* Mendeteksi citra tangan membuka yang telah di segmentasi menggunakan metode *Particle Swarm Optimization*. Proses yang dilakukan dari citra asli yang diubah ke grayscale, kemudian dimasukkan kedalam frame untuk mendeteksi. <sup>1</sup>Sistem ini dapat menampilkan citra gambar tangan lebih tajam.

3) Data Output



Gambar 3.1 Tampilan Deteksi Citra Tangan

Pada gambar 3.1 menampilkan tampilan hasil deteksi citra tangan yang telah disegmentasi menggunakan metode *Particle Swarm Optimization*. Dan gerakan ke – 5 berhasil menyimpan pada array yang telah dibuat.

## 2. Analisa Kebutuhan Perangkat

### a. Kebutuhan Perangkat Kerat

- 1) Laptop Asus processor processor Intel CPU;Intel® Core™ i3-6006U.
- 2) Hardisk 1TB
- 3) RAM 4GB
- 4) Akses Internet

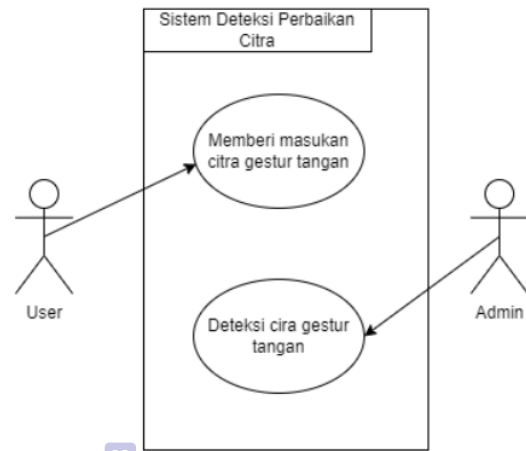
### b. Kebutuhan Perangkat Lunak

- 1) Windows 10 *Home Single Language*
- 2) Python 3.10.0
- 3) *Visual Studio Code*

## B. Desain Sistem (Arsitektur)

### 1. *Use Case Diagram*

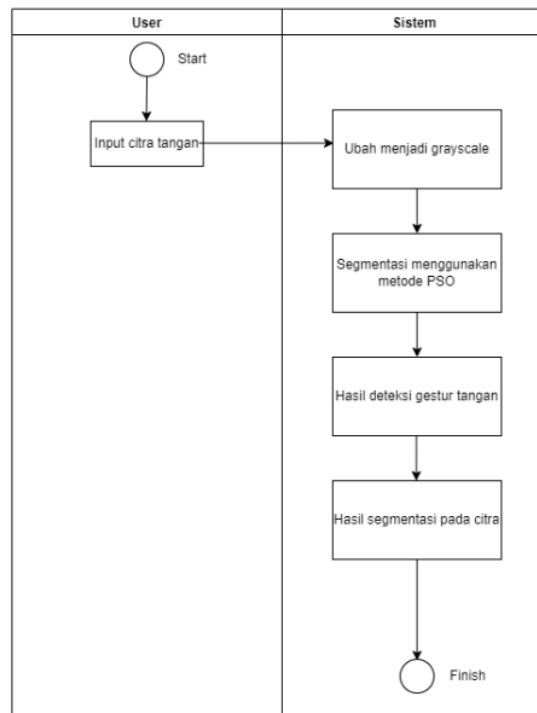
Pada gambar 3.2 dapat dijelaskan bahwa perancangan sistem pada pendeteksi perbaikan citra terdapat user yang dapat memasukkan citra gestur tangan. Kemudian admin dapat memantau deteksi citra gestur tangan dari monitor.



23

Gambar 3.2 Use Case Diagram

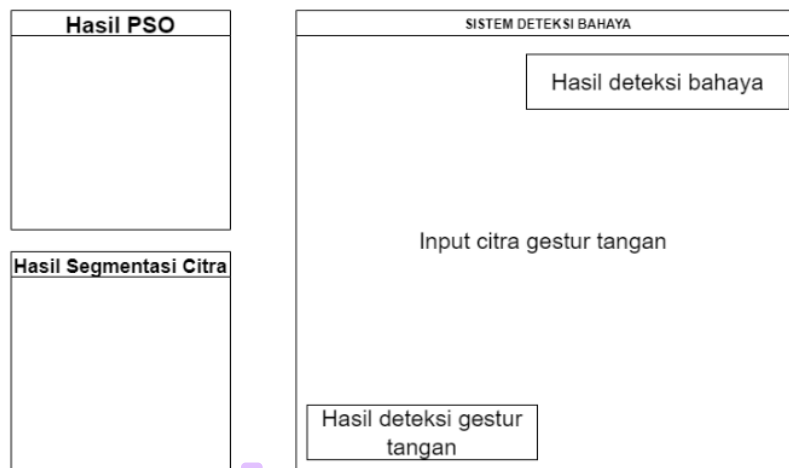
## 2. Activity Diagram



Gambar 3.3 Activity Diagram

Pada Gambar 3.3 menjelaskan bahwa user memasukkan citra tangan membuka dan menutup. Kemudian sistem akan melakukan proses mengubah citra menjadi bentuk grayscale. Selanjutnya citra tersebut akan disegmentasi melalui metode Particle Swarm Optimization. Lalu setelah disegmentasi sistem akan menampilkan hasil deteksi gestur tangan yang telah diproses. Citra yang berhasil disegmentasi akan tampil pada layar monitor yang diketahui oleh pengguna.

### C. Desain Antarmuka



Gambar 3.4 Desain Antarmuka

Pada gambar 3.3 merupakan tampilan sistem yang digunakan untuk segmentasi citra gestur tangan menggunakan metode *Particle Swarm Optimization*. Tahapan pada tampilan yang digunakan sebagai berikut :

1. Input citra gestur tangan yang akan dideteksi ini menampilkan tampilan utama untuk memasukkan citra tangan yang akan diproses.



2. Hasil citra tangan terbuka/tertutup merupakan hasil setelah input citra pada kamera, yang nantinya akan menampilkan sebuah teks “Terbuka/Tertutup”.
3. Hasil tampilan terdeteksi bahaya yaitu proses dari citra tangan yang diinputkan. Ketika gestur tangan membuka dan menutup akan menampilkan kode terdeteksi bahaya.
4. Hasil bagian citra tangan yang dideteksi ini merupakan hanya bagian telapak tangan terbuka dan tertutup yang akan diproses segmentasi menggunakan metode *Particle Swarm Optimization*.
5. Hasil segmentasi menggunakan metode *Particle Swarm Optimization* merupakan hasil dari citra tangan yang telah diinput dan diproses kedalam segmentasi menggunakan metode yang telah dibuat.

**IMPLEMENTASI DAN HASIL****A. Implementasi Lembar Kerja**

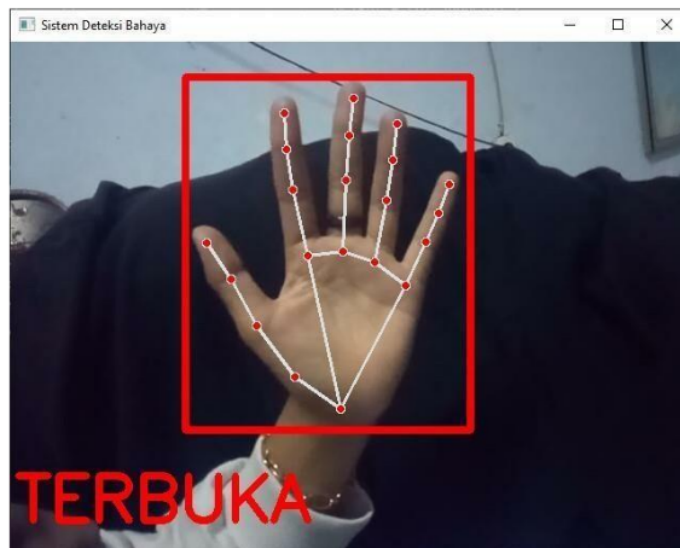
28

Implementasi Program adalah hasil implementasi dari analisa dan desain sistem yang telah dibuat sebelumnya dan dituangkan ke dalam kode program.

3

**1. Data Masukan**

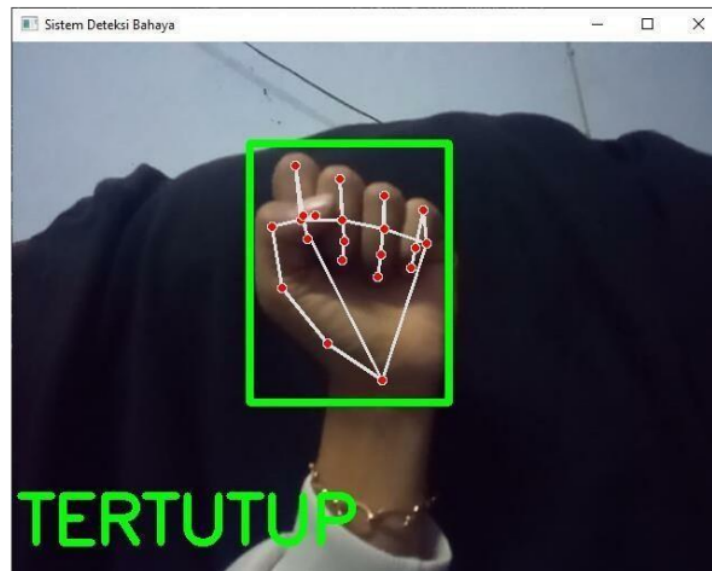
Data masukan adalah data yang diberikan oleh pengguna kepada kamera sebagai pendeteksi adanya indikator bahaya atau tolong menggunakan gestur tangan.

**a. Citra Tangan Terbuka**

Gambar 4.1 Citra Tangan Terbuka

Pada gambar 4.1 ini merupakan tampilan citra tangan terbuka, input citra menampilkan teks “TERBUKA” sesuai Gerakan tangan. Citra tangan yang dimasukkan menyatakan belum terdeteksi kode apapun.

b. Citra Tangan Tertutup



Gambar 4.2 Citra Tangan Tertutup

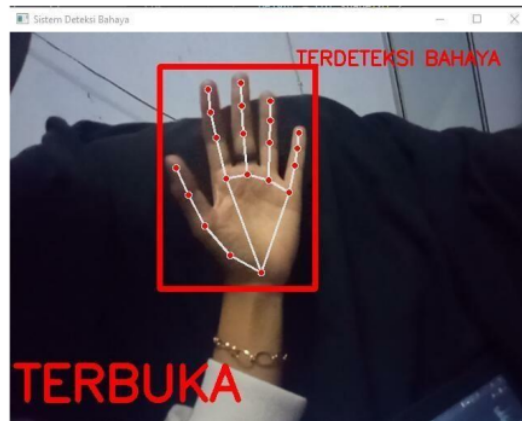
Pada gambar 4.2 merupakan tampilan citra tertutup, yang mengartikan bahwa salah satu syarat pendeteksi adanya tindakan bahaya.

2. Data Keluaran

Data Keluaran adalah hasil dari data input yang diberikan oleh pengguna dan diolah oleh sistem sehingga menampilkan hasil terdeteksi. Sistem akan menampilkan peringatan terdeteksi berupa teks pada

tampilan sistem.

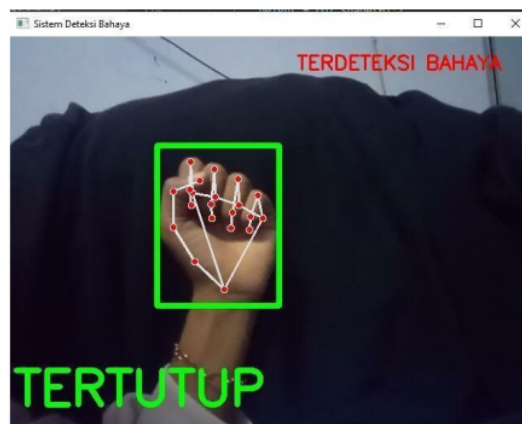
a. Citra Tangan Terbuka TERDETEKSI



Gambar 4.3 Citra Tangan Terbuka Terdeteksi

Pada gambar 4.3 merupakan tampilan dari citra tangan setelah melakukan gerakan 3 kali. Setelah input citra tangan menampilkan hasil "TERDETEKSI" pada tangan terbuka.

b. Citra Tangan Tertutup TERDETEKSI



Gambar 4.4 Citra Tangan Tertutup Terdeteksi

Pada gambar 4.4 merupakan tampilan dari citra tangan setelah melakukan gerakan 3 kali. Setelah input citra tangan menampilkan hasil “TERDETEKSI” pada tangan tertutup.

## B. Implementasi Program (*Development*)

Pada alur program ini menjelaskan setiap fungsi atau kegunaan. Adapun alur diantaranya yaitu sebagai berikut:

### 1. Memulai kamera video

```
a. cap = cv2.VideoCapture(0)
```

Pada program diatas digunakan untuk menampilkan kamera sebagai awal mula sebagai pendeteksi citra.

### 2. Deteksi tangan dengan mediapipe

```
a. result = hands.process(frame_rgb)
```

```
b. hand_landmarks = result.multi_hand_landmarks
```

```
c. if hand_landmarks:
```

```
d. bawah = [hand_landmarks[0].landmark[0].x,
            hand_landmarks[0].landmark[0].y]
```

```
e. tengah = [hand_landmarks[0].landmark[5].x,
             hand_landmarks[0].landmark[5].y]
```

```
f. atas = [hand_landmarks[0].landmark[8].x,
           hand_landmarks[0].landmark[8].y]
```

```
g. angle = calculate_angle(bawah, tengah, atas)
```

```
h. # cv2.putText(frame, str(angle),
```

```

tuple(np.multiply(tengah, [640,
480])).astype(int)),
cv2.FONT_HERSHEY_SIMPLEX, 0.5, (255, 255, 255),
2, cv2.LINE_AA)
i. # Curl counter logic
j. if angle > 175:
k. stage = "buka"
l. if angle < 50 and stage == 'buka':
m. stage="tutup"
n. counter +=1

```

Pada program diatas sebagai kerangka node dan tepi atau landmark, mereka melacak titik - titik kunci pada bagian telapak tangan.

### 3. Menghitung prediksi gerakan gestur tangan

```

a. if total>=3:
b. print('TERBUKA')
c. cv2.rectangle(frame, (x_min-20, y_min-20),
(x_max+20, y_max+20), (0, 0, 255), 5)
d. cv2.putText(frame, "TERBUKA", (5, 450),
cv2.FONT_HERSHEY_SIMPLEX, 2, (0, 0, 255), 5)
e. else:
f. print('TERTUTUP')
g. cv2.rectangle(frame, (x_min-20, y_min-20),
(x_max+20, y_max+20), (0, 255, 0), 5)

```

```
h. cv2.putText(frame, "TERTUTUP", (5,
    450), cv2.FONT_HERSHEY_SIMPLEX, 2, (0, 255, 0),
    5)
```

Pada program diatas sebagai prediksi gerakan gestur tangan. Jika tangan membuka lebih sama dengan 3 kali gerakan, maka akan menampilkan teks berupa “TERBUKA” dan “TERTUTUP” yang kemudian akan menghasilkan notif terdeteksi bahaya.

#### 4. Proses cropping gambar

```
a. # crop gambar
b. roi = frame[y_min-20: y_max+20, x_min-20:
    x_max+20]
c. width = roi.shape[1]*2
d. height = roi.shape[0]*2
e. dim = (width, height)
f. # perbesar
g. resized = cv2.resize(roi, dim, interpolation =
    cv2.INTER_AREA)
h. cv2.imwrite(dir_result+"gray1.png", resized)
i. gray1 = cv2.imread(dir_result+"gray1.png")
j. cv2.imshow("Hasil Segmentasi Citra",gray1)
```

Pada program diatas digunakan untuk cropping segmentasi citra tangan yang dideteksi, bagian telapak tangan akan dicrop dan diperbesar. Maka diperoleh dan menampilkan tampilan “Hasil Segmentasi Citra”.

5. Menampilkan metode *Particle Swarm Optimization*

- a. # PSO
- b. `restore_img = RESTORE(gray1)`
- c. `cv2.imwrite(dir_result+"citra_perbaikan.png",  
restore_img)`
- d. `gray2=cv2.cvtColor(restore_img,cv2.COLOR_BGR2GRAY)`
- e. `cv2.imwrite(dir_result+"gray2.png",gray2)`
- f. ## Hasil PSO
- g. `img2 = cv.imread(dir_result+'gray2.png')`
- h. `cv2.imshow('Hasil PSO', img2)`
- i. `cv2.waitKey(0)`

Pada program diatas digunakan untuk mendefinisikan melalui metode *Particle Swarm Optimization*. Program tersebut sebagai perbaikan citra tangan yang tambak jelas setelah dideteksi.

6. Menampilkan frame deteksi bahaya

- a. `cv2.imshow("Sistem Deteksi Bahaya", frame)`

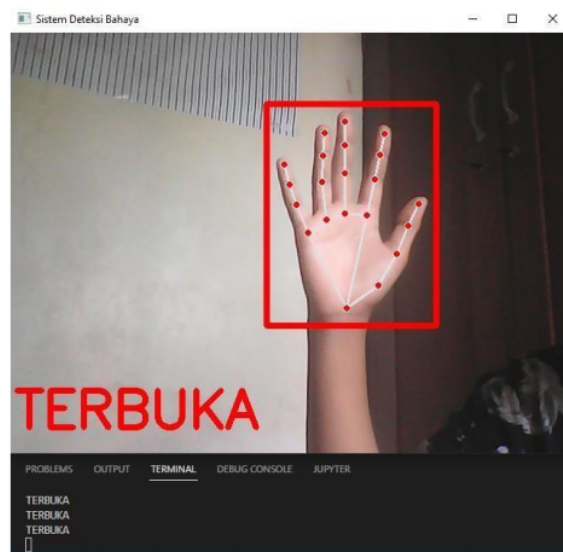
Pada program diatas mendefinisikan kode program untuk menampilkan hasil dari pendeteksi. Program yang telah dibuat merupakan hasil akhir dari citra tangan yang terdeteksi bahaya menggunakan citra gestur tangan.



### C. Pengujian Sistem

Pada pengujian program yang dilakukan dengan aturan 3 kali tangan terbuka - tertutup - terbuka - tertutup - terbuka. Tampilan program seperti berikut:

1. Pengguna memasukkan citra gestur tangan dengan gerakan terbuka ke -1

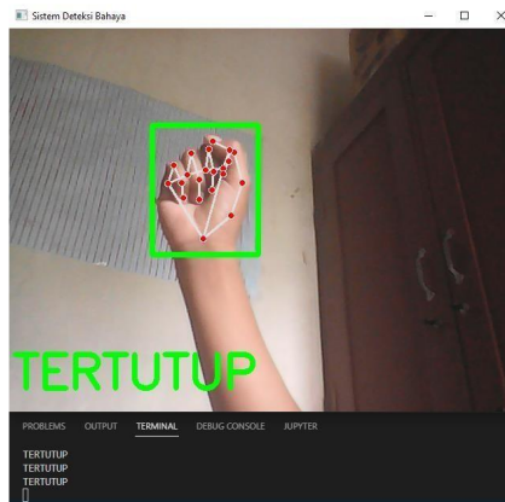


Gambar 4.5 Tampilan Gestur Tangan Ke-1

Pada gambar 4.5 merupakan tampilan gestur tangan terbuka ke – 1 pada gerakan pertama berhasil menyimpan pada array yang telah dibuat.

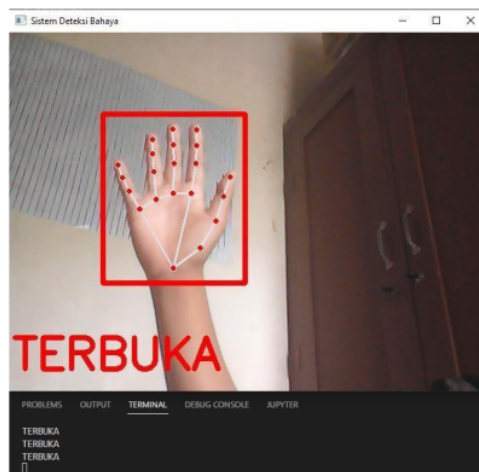
2. Pengguna memasukkan citra gestur tangan dengan gerakan tertutup.

Pada gambar 4.6 merupakan tampilan gestur tangan terbuka ke- 2 pada gerakan pertama berhasil menyimpan pada array yang telah dibuat.



Gambar 4.6 Tampilan Gestur Tangan Ke-2

3. Pengguna memasukkan citra gestur tangan dengan gerakan terbuka.

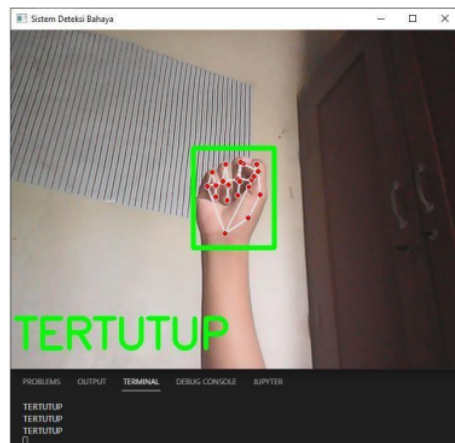


Gambar 4.7 Tampilan Gestur Tangan Ke-3

Pada gambar 4.7 merupakan tampilan gestur tangan terbuka ke-3 pada gerakan pertama berhasil menyimpan pada array yang telah dibuat.

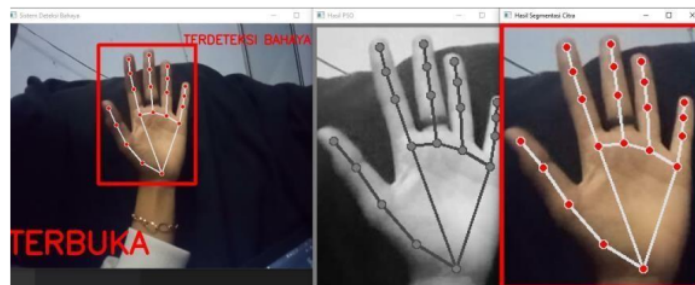
4. Pengguna memasukkan citra gestur tangan dengan gerakan tertutup.

Pada gambar 4.8 merupakan tampilan gestur tangan tertutup ke-4 pada gerakan pertama berhasil menyimpan pada array yang telah dibuat



Gambar 4.8 Tampilan Gestur Tangan Tertutup

5. Proses kelima adalah pengguna memberikan input citra gestur tangan kepada system. Menampilkan sebagai hasil deteksi sebagai berikut:



Gambar 4.9 Tampilan Deteksi Citra Tangan

Pada gambar 4.9 menampilkan tampilan hasil deteksi citra tangan yang telah disegmentasi menggunakan metode *Particle Swarm Optimization*. Dan gerakan ke – 5 berhasil menyimpan pada array yang telah dibuat

Tahap pengujian sistem adalah tahapan yang dilakukan untuk mengetahui bahwa sistem yang dibuat sudah sesuai dengan analisa dan kebutuhan program. Berikut adalah hasil dari pengujian sistem yang telah dilakukan dibagi menjadi 4 skenario uji coba :

1. Skenario Uji Coba ke-1

Skenario Uji Coba ke-1 dilakukan dengan menggunakan cahaya ruangan redup dengan *background* polos dan jarak yang berbeda.

Tabel 4.1 Skenario Uji Coba ke-1

NO	JARAK	HASIL
1.	1 Meter	Berhasil Terdeteksi dengan Baik
2.	2 Meter	Berhasil Terdeteksi dengan Baik
3.	3 Meter	Berhasil Terdeteksi dengan Baik
4.	4 Meter	Kadang bisa Terdeteksi dan Tidak
5.	5 Meter	Tidak Bisa Terdeteksi

Dari hasil Skenario Uji Coba ke-1 didapatkan hasil bahwa pengujian yang dilakukan di ruangan redup background polos hasil terbaiknya dan dapat berhasil mendeteksi hanya sampai jarak 3 meter, jika sudah mencapai jarak 4 meter maka sistem terkadang bisa mendeteksi dan kadang tidak bisa mendeteksi. Sedangkan jika jaraknya lebih dari 4 meter maka sistem tidak bisa mendeteksi citra gestur tangan.

2. Skenario Uji Coba ke-2

Skenario Uji Coba ke-2 dilakukan dengan menggunakan cahaya ruangan redup dengan *background* bercorak dan jarak yang berbeda.

Tabel 4.2 Skenario Uji Coba Ke-2

NO	JARAK	HASIL
1.	1 Meter	Berhasil Terdeteksi dengan Baik
2.	2 Meter	Berhasil Terdeteksi dengan Baik
3.	3 Meter	Kadang bisa Terdeteksi dan Tidak
4.	4 Meter	Tidak Bisa Terdeteksi
5.	5 Meter	Tidak Bisa Terdeteksi

Dari hasil Skenario Uji Coba ke-2 didapatkan hasil bahwa pengujian yang dilakukan di ruangan redup *background* bercorak hasil terbaiknya dan dapat berhasil mendeteksi hanya sampai jarak 2 meter, maka sistem terkadang bisa mendeteksi dan kadang tidak bisa mendeteksi. Sedangkan jika jaraknya lebih dari 4 meter maka sistem tidak bisa mendeteksi citra gestur tangan.

### 3. Skenario Uji Coba ke-3

Skenario Uji Coba ke-3 dilakukan dengan menggunakan cahaya terang ruangan dengan *background* polos dan jarak yang berbeda.

Tabel 4.3 Skenario Uji Coba Ke-3

NO	JARAK	HASIL
1.	1 Meter	Berhasil Terdeteksi dengan Baik
2.	2 Meter	Berhasil Terdeteksi dengan Baik
3.	3 Meter	Berhasil Terdeteksi dengan Baik
4.	4 Meter	Tidak Bisa Terdeteksi
5.	5 Meter	Tidak Bisa Terdeteksi

Dari hasil Skenario Uji Coba ke-3 didapatkan hasil bahwa pengujian yang dilakukan di cahaya terang ruangan *background* polos

hasil terbaiknya dan dapat berhasil mendeteksi hanya sampai jarak 3 meter, jika sudah mencapai jarak 4 meter maka sistem terkadang bisa mendeteksi dan kadang tidak bisa mendeteksi. Sedangkan jika jaraknya lebih dari 4 meter maka sistem tidak bisa mendeteksi citra gestur tangan.

#### 4. Skenario Uji Coba ke-4

Skenario Uji Coba ke-4 dilakukan dengan menggunakan cahaya terang ruangan dengan *background* bercorak dan jarak yang berbeda.

Tabel 4 4 Skenario Uji Coba Ke-4

NO	JARAK	HASIL
1.	1 Meter	Berhasil Terdeteksi dengan Baik
2.	2 Meter	Berhasil Terdeteksi dengan Baik
3.	3 Meter	Tidak Bisa Terdeteksi
4.	4 Meter	Tidak Bisa Terdeteksi
5.	5 Meter	Tidak Bisa Terdeteksi

Dari hasil Skenario Uji Coba ke-4 didapatkan hasil bahwa pengujian yang dilakukan di cahaya normal ruangan *background* bercorak hasil terbaiknya dan dapat berhasil mendeteksi hanya sampai jarak 2 meter, jika sudah mencapai jarak 3 meter maka sistem tidak bisa mendeteksi citra gestur tangan.

Tahap pengujian sistem adalah tahapan yang dilakukan untuk mengetahui bahwa sistem yang dibuat sudah sesuai. Berikut adalah hasil dari pengujian sistem tanpa metode *PSO* yang telah dilakukan dibagi menjadi 4 skenario uji coba :

### 1. Skenario Uji Coba ke-1

Skenario Uji Coba ke-1 dilakukan dengan menggunakan cahaya ruangan redup dengan *background* polos dan jarak yang berbeda.

Tabel 4.5 Skenario Uji Coba Ke-1

NO	JARAK	HASIL
1.	1 Meter	Berhasil Terdeteksi dengan Baik
2.	2 Meter	Berhasil Terdeteksi dengan Baik
3.	3 Meter	Berhasil Terdeteksi dengan Baik
4.	4 Meter	Tidak Bisa Terdeteksi
5.	5 Meter	Tidak Bisa Terdeteksi

Dari hasil Skenario Uji Coba ke-1 didapatkan hasil bahwa pengujian yang dilakukan di ruangan redup *background* polos hasil terbaiknya dan dapat berhasil mendeteksi hanya sampai jarak 3 meter, jika sudah mencapai jarak 4 meter maka sistem tidak bisa mendeteksi citra gestur tangan.

### 2. Skenario Uji Coba ke-2

Skenario Uji Coba ke-2 dilakukan dengan menggunakan cahaya ruangan redup dengan *background* bercorak dan jarak yang berbeda.

Tabel 4.6 Skenario Uji Coba ke-2

NO	JARAK	HASIL
1.	1 Meter	Berhasil Terdeteksi dengan Baik
2.	2 Meter	Kadang bisa Terdeteksi dan Tidak
3.	3 Meter	Tidak Bisa Terdeteksi
4.	4 Meter	Tidak Bisa Terdeteksi
5.	5 Meter	Tidak Bisa Terdeteksi

Dari hasil Skenario Uji Coba ke-2 didapatkan hasil bahwa pengujian yang dilakukan di ruangan redup *background* bercorak hasil terbaiknya dan dapat berhasil mendeteksi hasil jarak 1 meter, jarak 2 meter sistem terkadang bisa mendeteksi dan kadang tidak bisa mendeteksi. Sedangkan jika jaraknya lebih dari 3 meter maka sistem tidak bisa mendeteksi citra gestur tangan.

### 3. Skenario Uji Coba ke-3

Skenario Uji Coba ke-3 dilakukan dengan menggunakan cahaya terang ruangan dengan *background* polos dan jarak yang berbeda.

Tabel 4.7 Skenario Uji Coba Ke-3

NO	JARAK	HASIL
1.	1 Meter	Berhasil Terdeteksi dengan Baik
2.	2 Meter	Kadang bisa Terdeteksi dan Tidak
3.	3 Meter	Tidak Bisa Terdeteksi
4.	4 Meter	Tidak Bisa Terdeteksi
5.	5 Meter	Tidak Bisa Terdeteksi

Dari hasil Skenario Uji Coba ke-3 didapatkan hasil bahwa pengujian yang dilakukan di cahaya terang ruangan *background* polos hasil terbaiknya dan dapat berhasil mendeteksi hanya jarak 1 meter. Jika jarak 2 meter sistem terkadang bisa mendeteksi dan kadang tidak bisa mendeteksi, jika sudah mencapai jarak 3 meter maka tidak bisa mendeteksi citra gestur tangan.



#### 4. Skenario Uji Coba ke-4

Skenario Uji Coba ke-4 dilakukan dengan menggunakan cahaya terang ruangan dengan *background* bercorak dan jarak yang berbeda.

Tabel 4.8 Skenario Uji Coba Ke-4

NO	JARAK	HASIL
1.	1 Meter	Berhasil Terdeteksi dengan Baik
2.	2 Meter	Tidak Bisa Terdeteksi
3.	3 Meter	Tidak Bisa Terdeteksi
4.	4 Meter	Tidak Bisa Terdeteksi
5.	5 Meter	Tidak Bisa Terdeteksi

Dari hasil Skenario Uji Coba ke-4 didapatkan hasil bahwa pengujian yang dilakukan di cahaya normal ruangan *background* bercorak hasil terbaiknya dan dapat berhasil mendeteksi hanya jarak 1 meter, jika sudah mencapai jarak 2 meter maka sistem tidak bisa mendeteksi citra gestur tangan

#### D. Hasil

Metode *Particle Swarm Optimization* ini diterapkan dalam penelitian yang digunakan untuk mendeteksi citra tangan dan diproses kedalam segmentasi. Penelitian ini dapat menampilkan pendeteksian citra tangan yang maksimal. Dalam proses ini pengguna akan melakukan gestur tangan sebanyak 3 kali Gerakan, dimana Gerakan berupa telapak tangan terbuka dan tertutup. Kemudian Gerakan tersebut diproses kedalam sistem yang nantinya akan menampilkan kode terdeteksi. Setelah citra tangan telah terdeteksi maka

proses selanjutnya adalah melakukan segmentasi dengan menggunakan metode *Particle Swarm Optimization*. Sistem ini memudahkan bagi pengguna untuk menganalisis gambar dari tangan yang terdeteksi dengan jelas dan akurat. Oleh karena itu sistem yang dibangun telah sesuai oleh rancangan yang dibuat.

#### E. Hasil Evaluasi

Evaluasi sistem yang dilakukan merupakan <sup>3</sup> hasil dari uji coba sistem yang telah dibuat. Hasil Analisa ini didapat dari kesimpulan semua uji coba yang dijalankan oleh sistem. Kemudian evaluasi hasil coba ini digunakan untuk mengevaluasi tahapan yang dilakukan.

##### 1. Kelebihan Sistem

Kelebihan yang dimiliki oleh sistem adalah :

- a. Sistem ini dapat mendeteksi suatu Gerakan dengan telapak tangan terbuka dan tertutup. Proses yang dilakukan dengan gestur tangan membuka dan menutup sebanyak 3 kali, maka akan menampilkan teks“TERDETEKSI”.
- b. Sistem mengolah data input dan dilakukan secara *real time*.

##### 2. Kekurangan Sistem

Kekurangan yang dimiliki sistem adalah :

- a. Sistem hanya bisa melakukan Gerakan tangan hanya berfokus pada telapak tangan manusia.
- b. Penggunaberada ditempat yang terang untuk dapat menggunakan

sistem ini dengan baik.

Dapat disimpulkan bahwa sistem yang dibuat sesuai dengan kebutuhan sebagai pendeteksi adanya bahaya menggunakan citra gesture tangan.

## PENUTUP

## A. Kesimpulan

Berdasarkan hasil dan implementasi dalam penelitian ini, maka diperoleh kesimpulan sebagai berikut:

1. Pembuatan kode program yang diimplementasikan telah sesuai perintah yang dibuat. Pada sistem segmentasi citra tangan ini menggunakan metode *Particle Swarm Optimization* yang mampu mendeteksi gestur tangan.
2. Segmentasi citra tangan ini sebagai pendeteksi adanya bahaya, dan mampu menampilkan teks berupa tangan terbuka dan tertutup sesuai gestur, serta tampil teks "TERDETEKSI".
3. Dengan menggunakan metode *Particle Swarm Optimization* proses pendeteksian tergantung pada intensitas cahaya, jarak dan background. Sistem ini mempermudah pengguna dalam menganalisis gambar.

## B. Saran

Sistem pendeteksi citra tangan dengan metode *Particle Swarm Optimization* masih jauh dari kata sempurna. Oleh karena itu, adapun saran yang dapat penulis berikan untuk penelitian selanjutnya yaitu sebagai berikut:

1. Bagi peneliti selanjutnya diharapkan dapat mengembangkan metode yang digunakan dalam pendeteksi citra tangan, sehingga dapat mendeteksi

dalam keadaan cahaya efektif.

2. Penelitian selanjutnya diharapkan dapat mendeteksi bukan hanya dari telapak tangan, tetapi juga dari bagian wajah atau lainnya.
3. Untuk penelitian selanjutnya bisa mengembangkan sistem ini melalui Gerakan kedua tangan manusia.

# SEGMENTASI METODE PARTICLE SWARM OPTIMIZATION PADA PENDETEKSI CITRA GESTUR TANGAN

## ORIGINALITY REPORT

34%

SIMILARITY INDEX

31%

INTERNET SOURCES

9%

PUBLICATIONS

10%

STUDENT PAPERS

## PRIMARY SOURCES

1

[proceeding.unpkediri.ac.id](http://proceeding.unpkediri.ac.id)

Internet Source

5%

2

[www.researchgate.net](http://www.researchgate.net)

Internet Source

4%

3

[repository.its.ac.id](http://repository.its.ac.id)

Internet Source

3%

4

[ejournal.stmik-time.ac.id](http://ejournal.stmik-time.ac.id)

Internet Source

2%

5

[j-ptiik.ub.ac.id](http://j-ptiik.ub.ac.id)

Internet Source

1%

6

[e-journal.stmiklombok.ac.id](http://e-journal.stmiklombok.ac.id)

Internet Source

1%

7

Herpendi Herpendi, Veri Julianto, Khairul Anwar Hafizd. "Perancangan Multicontrol Pada Lampu Berbasis Internet Of Think (IOT)", Jurnal SAINTEKOM, 2018

Publication

1%

8

[slideplayer.info](http://slideplayer.info)

Internet Source

		1 %
9	<a href="http://ojs.unpkediri.ac.id">ojs.unpkediri.ac.id</a> Internet Source	1 %
10	<a href="http://doaj.org">doaj.org</a> Internet Source	1 %
11	Submitted to Flinders University Student Paper	1 %
12	<a href="http://journal.uii.ac.id">journal.uii.ac.id</a> Internet Source	1 %
13	Submitted to Universitas Putera Batam Student Paper	1 %
14	<a href="http://www.mindopen.nl">www.mindopen.nl</a> Internet Source	1 %
15	<a href="http://www.coursehero.com">www.coursehero.com</a> Internet Source	1 %
16	<a href="http://123dok.com">123dok.com</a> Internet Source	1 %
17	<a href="http://adoc.pub">adoc.pub</a> Internet Source	1 %
18	<a href="http://jurnalteknik.unisla.ac.id">jurnalteknik.unisla.ac.id</a> Internet Source	1 %
19	Submitted to North West University Student Paper	<1 %

20	<a href="http://lp2m.unpkediri.ac.id">lp2m.unpkediri.ac.id</a> Internet Source	<1 %
21	<a href="http://ejournal.stikom-db.ac.id">ejournal.stikom-db.ac.id</a> Internet Source	<1 %
22	<a href="http://widuri.raharjo.info">widuri.raharjo.info</a> Internet Source	<1 %
23	<a href="http://eprints.dinus.ac.id">eprints.dinus.ac.id</a> Internet Source	<1 %
24	<a href="http://eprints.umpo.ac.id">eprints.umpo.ac.id</a> Internet Source	<1 %
25	<a href="http://jurnalmahasiswa.unesa.ac.id">jurnalmahasiswa.unesa.ac.id</a> Internet Source	<1 %
26	<a href="http://repository.uinjkt.ac.id">repository.uinjkt.ac.id</a> Internet Source	<1 %
27	Submitted to Universitas Brawijaya Student Paper	<1 %
28	<a href="http://docplayer.info">docplayer.info</a> Internet Source	<1 %
29	<a href="http://ejurnal.stmik-budidarma.ac.id">ejurnal.stmik-budidarma.ac.id</a> Internet Source	<1 %
30	<a href="http://eprints.perbanas.ac.id">eprints.perbanas.ac.id</a> Internet Source	<1 %
31	<a href="http://repository.binadarma.ac.id">repository.binadarma.ac.id</a> Internet Source	<1 %



32	<a href="https://repository.ub.ac.id">repository.ub.ac.id</a> Internet Source	<1 %
33	<a href="https://elibrary.unikom.ac.id">elibrary.unikom.ac.id</a> Internet Source	<1 %
34	<a href="https://eprints.umm.ac.id">eprints.umm.ac.id</a> Internet Source	<1 %
35	<a href="https://text-id.123dok.com">text-id.123dok.com</a> Internet Source	<1 %
36	<a href="https://gitlab.sliit.lk">gitlab.sliit.lk</a> Internet Source	<1 %
37	<a href="https://repository.usd.ac.id">repository.usd.ac.id</a> Internet Source	<1 %
38	<a href="https://etheses.uin-malang.ac.id">etheses.uin-malang.ac.id</a> Internet Source	<1 %
39	<a href="https://repository.atmaluhur.ac.id">repository.atmaluhur.ac.id</a> Internet Source	<1 %
40	<a href="https://repository.radenintan.ac.id">repository.radenintan.ac.id</a> Internet Source	<1 %
41	<a href="https://www.docstoc.com">www.docstoc.com</a> Internet Source	<1 %
42	Erika - Mutiara. "ALGORITMA KLASIFIKASI NAIVE BAYES BERBASIS PARTICLE SWARM OPTIMIZATION UNTUK PREDIKSI PENYAKIT TUBERCULOSIS (TB)", Swabumi, 2020 Publication	<1 %

43	<a href="http://e-journal.uajy.ac.id">e-journal.uajy.ac.id</a> Internet Source	<1 %
44	<a href="http://eprints.unm.ac.id">eprints.unm.ac.id</a> Internet Source	<1 %
45	<a href="http://repository.umj.ac.id">repository.umj.ac.id</a> Internet Source	<1 %
46	<a href="http://www.potensi-utama.ac.id">www.potensi-utama.ac.id</a> Internet Source	<1 %
47	<a href="http://www2.slideshare.net">www2.slideshare.net</a> Internet Source	<1 %

Exclude quotes On

Exclude matches Off

Exclude bibliography On