

BAB III METODE PENELITIAN

3.1 Pendekatan Penelitian

Penelitian ini menggunakan pendekatan kuantitatif dengan metode eksperimen dalam bidang pengolahan citra digital dan machine learning. Pendekatan kuantitatif dipilih karena penelitian berfokus pada pengolahan data numerik berupa fitur citra digital, parameter model klasifikasi, serta hasil evaluasi performa sistem yang diukur menggunakan metrik statistik seperti accuracy, precision, recall, dan f1-score.

Metode eksperimen digunakan untuk menguji kemampuan algoritma klasifikasi dalam membedakan gambar wajah asli dan gambar wajah palsu berdasarkan karakteristik tekstur dan warna citra digital. Penelitian ini mengimplementasikan metode ekstraksi fitur Gray Level Co-occurrence Matrix (GLCM) dan Hue Saturation Value (HSV), kemudian menggunakan algoritma Support Vector Machine (SVM) sebagai model klasifikasi utama.

Selain itu, penelitian ini juga menerapkan algoritma Particle Swarm Optimization (PSO) untuk melakukan optimasi parameter Support Vector Machine (SVM) sehingga diperoleh kombinasi parameter terbaik yang mampu meningkatkan akurasi deteksi gambar wajah palsu.

Pendekatan penelitian dilakukan secara sistematis mulai dari tahap pengumpulan dataset, preprocessing citra, ekstraksi fitur, proses klasifikasi, optimasi parameter, hingga evaluasi performa model. Dengan pendekatan tersebut diharapkan sistem yang dihasilkan mampu mendeteksi fake image secara otomatis dengan tingkat akurasi yang tinggi

3.2 Desain Penelitian

Desain penelitian yang digunakan adalah desain penelitian eksperimental berbasis pengembangan sistem klasifikasi citra digital. Penelitian dirancang untuk membangun sistem deteksi gambar wajah palsu menggunakan kombinasi fitur tekstur dan warna dengan metode klasifikasi Support Vector Machine (SVM) yang dioptimasi menggunakan Particle Swarm Optimization (PSO).

Tahapan penelitian dimulai dengan pengumpulan dataset gambar wajah asli dan gambar wajah palsu. Dataset kemudian diproses melalui tahap preprocessing untuk menyeragamkan ukuran citra dan mempersiapkan data sebelum proses ekstraksi fitur.

Tahap preprocessing meliputi:

1. Resize citra menjadi ukuran yang seragam.
2. Konversi citra RGB ke grayscale untuk ekstraksi fitur tekstur.
3. Konversi ruang warna RGB ke HSV untuk ekstraksi fitur warna.

Selanjutnya dilakukan ekstraksi fitur menggunakan:

1. Metode Gray Level Co-occurrence Matrix (GLCM) untuk memperoleh karakteristik tekstur citra.
2. Metode Hue Saturation Value (HSV) untuk memperoleh informasi distribusi warna citra.

Fitur-fitur hasil ekstraksi kemudian digabungkan menjadi feature vector yang digunakan sebagai data input pada algoritma Support Vector Machine (SVM).

Pada tahap optimasi, algoritma Particle Swarm Optimization (PSO) digunakan untuk mencari parameter optimal SVM berupa parameter C dan gamma (γ) agar menghasilkan performa klasifikasi terbaik.

Tahap akhir penelitian dilakukan dengan mengevaluasi performa sistem menggunakan confusion matrix berdasarkan nilai:

1. Accuracy
2. Precision
3. Recall
4. F1-score

Hasil evaluasi model SVM standar kemudian dibandingkan dengan model SVM-PSO untuk mengetahui peningkatan performa yang dihasilkan oleh optimasi parameter.

3.3 Subjek Penelitian

Subjek penelitian dalam penelitian ini adalah citra digital wajah manusia yang terdiri dari gambar wajah asli (real image) dan gambar wajah palsu (fake

image). Dataset yang digunakan berasal dari sumber dataset publik yang menyediakan data gambar hasil manipulasi digital maupun sintesis Artificial Intelligence (AI).

Jumlah dataset yang digunakan dalam penelitian ini adalah sebanyak **4.656 gambar**, yang terdiri dari:

1. Gambar wajah asli.
2. Gambar wajah palsu hasil manipulasi atau sintesis AI.

Dataset digunakan sebagai data utama dalam proses pelatihan dan pengujian model klasifikasi machine learning. Seluruh data kemudian dibagi menjadi data training dan data testing menggunakan skenario pembagian data sebesar:

- 80% data training
- 20% data testing

Dengan pembagian tersebut diperoleh:

- Data training sebanyak 3.724 gambar.
- Data testing sebanyak 932 gambar.

Objek penelitian difokuskan pada analisis karakteristik tekstur dan warna yang terdapat pada gambar wajah untuk membedakan pola visual antara gambar asli dan gambar palsu. Penelitian ini hanya berfokus pada citra wajah statis (static image) dan tidak membahas video deepfake.

3.4 Instrumen Penelitian

Instrumen penelitian yang digunakan dalam penelitian ini terdiri atas perangkat keras (*hardware*) dan perangkat lunak (*software*) yang digunakan untuk proses pengembangan sistem, pelatihan model, serta implementasi aplikasi deteksi citra wajah **real** dan **fake**.

3.4.1 Perangkat Keras (Hardware)

Perangkat keras (*hardware*) merupakan komponen fisik yang digunakan sebagai media untuk menjalankan seluruh proses komputasi dalam penelitian. Perangkat keras ini digunakan untuk proses *preprocessing* citra, ekstraksi fitur menggunakan metode HSV dan GLCM, optimasi parameter menggunakan Particle Swarm Optimization (PSO), pelatihan model Support Vector Machine (SVM), hingga implementasi aplikasi deteksi berbasis

Streamlit. Adapun spesifikasi perangkat keras yang digunakan pada penelitian ini ditunjukkan pada Tabel 3.1.

Tabel 3.1 Spesifikasi Perangkat Keras

No	Komponen	Spesifikasi
1	Laptop	Acer Aspire A314-22
2	Processor	AMD Ryzen 3 3250U with Radeon Graphics (2.60 GHz, 4 Logical Processor)
3	RAM	8GB
4	Graphics	AMD Radeon Graphics (Integrated GPU)
5	Media Penyimpanan	SSD 512GB
6	Sistem Operasi	Windows 11 Home Single Language 64-bit
7	DirectX	DirectX 12

Berdasarkan Tabel 3.1, perangkat utama yang digunakan dalam penelitian adalah laptop **Acer Aspire A314-22** dengan prosesor **AMD Ryzen 3 3250U** berkecepatan 2,60 GHz dan memiliki 4 *logical processor*. Perangkat tersebut didukung oleh **RAM sebesar 8 GB** yang digunakan untuk menjalankan proses pengolahan dataset, *preprocessing* citra, ekstraksi fitur HSV dan GLCM, serta proses pelatihan dan pengujian model SVM.

Pada aspek grafis, perangkat menggunakan **AMD Radeon Graphics** yang terintegrasi (*integrated GPU*). Sementara itu, media penyimpanan menggunakan **SSD berkapasitas 512 GB** yang digunakan untuk menyimpan dataset citra, hasil ekstraksi fitur, model hasil pelatihan, serta berbagai berkas yang diperlukan selama proses penelitian. Sistem operasi yang digunakan adalah **Windows 11 Home Single Language 64-bit** dengan **DirectX 12** sebagai bagian dari konfigurasi sistem perangkat.

Dengan spesifikasi tersebut, perangkat digunakan sebagai lingkungan komputasi untuk menjalankan keseluruhan tahapan penelitian, mulai dari pengolahan dataset hingga implementasi aplikasi deteksi citra wajah. **Tabel tersebut tidak dimaksudkan untuk menunjukkan bahwa setiap komponen secara khusus menjalankan satu algoritma tertentu**, tetapi untuk mendokumentasikan lingkungan perangkat keras yang digunakan selama penelitian sehingga proses penelitian dapat direplikasi dengan kondisi yang diketahui.

3.4.2 Perangkat Lunak (Software)

Perangkat lunak (*software*) merupakan sekumpulan program dan pustaka (*library*) yang digunakan untuk mendukung proses pengembangan sistem serta implementasi metode yang digunakan dalam penelitian. Perangkat lunak tersebut berfungsi untuk melakukan pengolahan citra digital, ekstraksi fitur, pelatihan dan optimasi model **Support Vector Machine (SVM)**, evaluasi performa model, hingga pembangunan antarmuka sistem berbasis web. Spesifikasi perangkat lunak yang digunakan dalam penelitian ini disajikan pada Tabel 3.2.

Tabel 3.2 Spesifikasi Perangkat Lunak

No	Perangkat Lunak	Versi	Fungsi
1	Windows 11 Home Single Language 64-bit	64-bit	Sistem operasi
2	Python	3.13.7	Bahasa pemrograman utama
3	Visual Studio Code	Terbaru	Pengembangan program
4	Streamlit	1.55.0	Implementasi antarmuka sistem berbasis web
5	OpenCV (cv2)	4.12.0.88	Pengolahan citra digital
6	NumPy	2.2.6	Operasi numerik dan pengolahan array

7	Pandas	2.3.2	Pengolahan dataset
8	Scikit-image	0.26.0	Ekstraksi fitur tekstur GLCM
9	Scikit-learn	1.7.2	Implementasi algoritma SVM, normalisasi data, dan evaluasi model
10	PySwarms	1.3.0	Optimasi parameter SVM menggunakan PSO
11	Matplotlib	3.10.6	Visualisasi grafik hasil penelitian
12	Joblib	1.5.2	Penyimpanan model dan StandardScaler

Berdasarkan Tabel 3.2, perangkat lunak yang digunakan dalam penelitian terdiri atas sistem operasi, bahasa pemrograman, lingkungan pengembangan, pustaka pengolahan citra dan data, pustaka machine learning, pustaka optimasi, serta perangkat lunak pendukung visualisasi dan penyimpanan model. **Python 3.13.7** digunakan sebagai bahasa pemrograman utama dalam mengimplementasikan seluruh tahapan penelitian, sedangkan **Visual Studio Code** digunakan sebagai lingkungan pengembangan program.

OpenCV (cv2) digunakan untuk mendukung proses pengolahan citra digital, sedangkan **NumPy** digunakan dalam operasi numerik dan pengolahan data berbentuk array. **Pandas** digunakan untuk mengelola dataset dan data hasil ekstraksi fitur. Pada tahap ekstraksi fitur tekstur, **Scikit-image** digunakan untuk memperoleh fitur GLCM. Sementara itu, fitur warna HSV diimplementasikan sebagai bagian dari proses pengolahan citra.

Pada tahap pembangunan model klasifikasi, **Scikit-learn** digunakan untuk implementasi **Support Vector Machine (SVM)**, normalisasi data menggunakan **StandardScaler**, serta proses evaluasi model. **PySwarms** digunakan untuk menerapkan metode **Particle Swarm Optimization (PSO)** dalam proses pencarian parameter SVM yang optimal. Hasil pengolahan dan evaluasi model kemudian dapat divisualisasikan menggunakan **Matplotlib**.

Setelah model selesai dilatih, **Joblib** digunakan untuk menyimpan model SVM dan *StandardScaler* sehingga keduanya dapat digunakan kembali pada proses deteksi tanpa harus melakukan pelatihan ulang. Selanjutnya, **Streamlit** digunakan untuk membangun antarmuka sistem berbasis web yang memungkinkan pengguna mengunggah citra dan memperoleh hasil klasifikasi *Real* atau *Fake* beserta informasi *confidence* yang dihasilkan oleh sistem.

Dengan demikian, perangkat lunak yang digunakan memiliki fungsi yang saling mendukung pada setiap tahapan penelitian, mulai dari pengolahan dataset, preprocessing citra, ekstraksi fitur HSV dan GLCM, pembentukan dan optimasi model SVM, evaluasi model, penyimpanan model, hingga implementasi aplikasi deteksi berbasis web.

3.5 Teknik Pengumpulan Data

Teknik pengumpulan data dalam penelitian ini dilakukan melalui dua tahapan utama yaitu studi literatur dan pengumpulan dataset citra digital. Tahapan ini dilakukan untuk memperoleh data dan informasi yang dibutuhkan dalam proses penelitian sistem deteksi gambar wajah palsu menggunakan metode machine learning.

Studi literatur dilakukan dengan mengumpulkan berbagai referensi ilmiah yang berkaitan dengan penelitian, seperti jurnal nasional maupun internasional, buku, prosiding seminar, dan artikel penelitian yang membahas digital image forensics, fake image detection, Gray Level Co-occurrence Matrix (GLCM), Hue Saturation Value (HSV), Support Vector Machine (SVM), serta Particle Swarm Optimization (PSO). Studi literatur bertujuan untuk memperoleh landasan teori, memahami metode yang digunakan pada penelitian sebelumnya, dan menentukan metode yang sesuai untuk diterapkan pada penelitian ini.

Selain studi literatur, pengumpulan data juga dilakukan dengan memperoleh dataset gambar wajah asli dan gambar wajah palsu dari sumber dataset publik. Dataset yang digunakan terdiri dari total 4.656 gambar yang meliputi citra wajah asli dan citra hasil manipulasi atau sintesis berbasis Artificial Intelligence (AI).

Data tersebut digunakan sebagai data utama dalam proses pelatihan dan pengujian model machine learning.

Setelah dataset diperoleh, dilakukan proses seleksi dan persiapan data untuk memastikan kualitas citra yang digunakan sesuai dengan kebutuhan penelitian. Selanjutnya dataset dibagi menjadi dua bagian yaitu data training sebesar 80% dan data testing sebesar 20%. Data training digunakan untuk proses pelatihan model klasifikasi, sedangkan data testing digunakan untuk menguji performa sistem deteksi gambar wajah palsu.

3.6 Analisis Data

Analisis data dalam penelitian ini dilakukan menggunakan metode pengolahan citra digital dan machine learning untuk mendeteksi gambar wajah palsu. Proses analisis data dimulai dari tahap preprocessing citra, ekstraksi fitur, klasifikasi menggunakan Support Vector Machine (SVM), optimasi parameter menggunakan Particle Swarm Optimization (PSO), hingga evaluasi performa model.

Tahap pertama dalam analisis data adalah preprocessing citra. Pada tahap ini dilakukan proses resize gambar untuk menyeragamkan ukuran seluruh citra agar mempermudah proses pengolahan data. Selanjutnya dilakukan konversi warna dari RGB ke grayscale untuk kebutuhan ekstraksi fitur tekstur menggunakan metode Gray Level Co-occurrence Matrix (GLCM). Selain itu dilakukan juga konversi ruang warna RGB ke HSV untuk memperoleh karakteristik warna citra.

Tahap berikutnya adalah ekstraksi fitur citra. Ekstraksi fitur tekstur dilakukan menggunakan metode Gray Level Co-occurrence Matrix (GLCM) untuk memperoleh informasi tekstur gambar berupa contrast, homogeneity, energy, dan entropy. Sementara itu, ekstraksi fitur warna dilakukan menggunakan metode Hue Saturation Value (HSV) dengan mengambil nilai statistik berupa mean dan standar deviasi dari komponen hue, saturation, dan value. Seluruh fitur hasil ekstraksi kemudian digabungkan menjadi feature vector yang digunakan sebagai data input pada model klasifikasi.

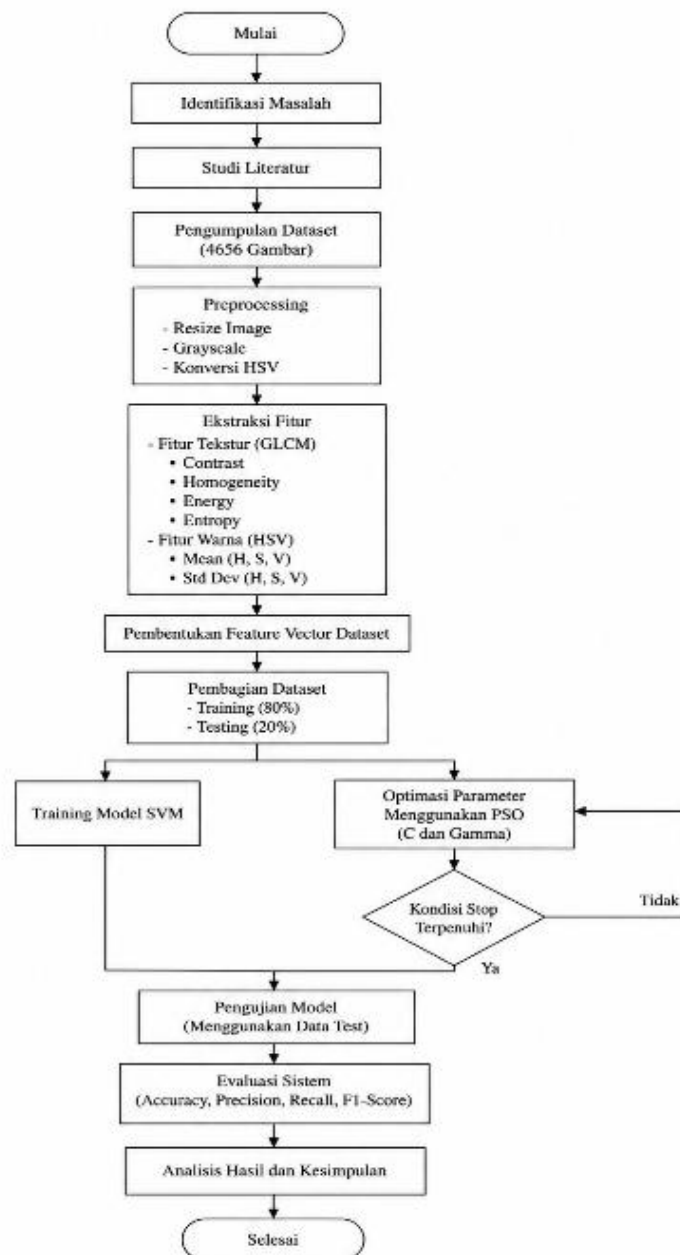
Feature vector hasil ekstraksi kemudian digunakan pada proses klasifikasi menggunakan algoritma Support Vector Machine (SVM) dengan kernel Radial Basis Function (RBF). Algoritma SVM digunakan untuk mengklasifikasikan gambar wajah ke dalam kategori gambar asli atau gambar palsu berdasarkan pola fitur yang diperoleh dari proses ekstraksi.

Selanjutnya dilakukan optimasi parameter menggunakan algoritma Particle Swarm Optimization (PSO). Optimasi dilakukan terhadap parameter C dan gamma pada model SVM untuk memperoleh kombinasi parameter terbaik yang mampu meningkatkan performa klasifikasi sistem. Proses optimasi dilakukan secara iteratif berdasarkan nilai fitness yang diperoleh dari hasil akurasi model.

Tahap akhir analisis data adalah evaluasi performa sistem menggunakan confusion matrix. Evaluasi dilakukan menggunakan beberapa parameter pengukuran yaitu accuracy, precision, recall, dan f1-score. Hasil evaluasi digunakan untuk mengetahui tingkat performa sistem dalam mendeteksi gambar wajah palsu serta membandingkan performa model SVM standar dengan model SVM yang telah dioptimasi menggunakan PSO.



3.7 Teknik Pengumpulan Data



Gambar 3.1 Alur Prosedur Penelitian

Tahapan penelitian dilakukan secara bertahap dan sistematis agar penelitian berjalan sesuai dengan tujuan yang telah ditentukan. Tahapan penelitian terdiri dari:

3.7.1 Identifikasi Masalah

Tahap awal penelitian dilakukan dengan mengidentifikasi permasalahan terkait perkembangan teknologi Artificial Intelligence (AI) yang mampu menghasilkan gambar wajah palsu dengan tingkat kemiripan tinggi terhadap gambar asli. Kemajuan teknologi tersebut meningkatkan

risiko penyalahgunaan media digital sehingga diperlukan sistem yang mampu mendeteksi keaslian gambar wajah secara otomatis dan akurat.

3.7.2 Studi Literatur

Pada tahap ini peneliti melakukan pengumpulan referensi ilmiah seperti jurnal, buku, prosiding, dan artikel penelitian yang berkaitan dengan digital image forensics, fake image detection, Gray Level Co-occurrence Matrix (GLCM), Hue Saturation Value (HSV), Support Vector Machine (SVM), dan Particle Swarm Optimization (PSO). Studi literatur dilakukan untuk memperoleh dasar teori, memahami metode penelitian sebelumnya, dan menentukan metode yang sesuai untuk penelitian.

3.7.3 Pengumpulan Data

Tahap pengumpulan dataset dilakukan dengan memperoleh gambar wajah asli dan gambar wajah palsu dari sumber dataset publik. Total dataset yang digunakan sebanyak 4.656 gambar yang terdiri dari citra wajah asli dan citra hasil manipulasi AI. Dataset tersebut kemudian dipersiapkan untuk proses pengolahan data dan klasifikasi machine learning.

3.7.4 Pra Processing Data

Pada tahap preprocessing dilakukan proses penyeragaman ukuran gambar (*resize*) agar seluruh citra memiliki dimensi yang sama, yaitu 64×64 piksel, sehingga setiap citra memiliki representasi yang seragam pada saat proses ekstraksi fitur. Penyeragaman ukuran ini juga bertujuan untuk mengurangi perbedaan resolusi antar citra, mempercepat waktu komputasi, serta meningkatkan konsistensi hasil ekstraksi fitur. Selanjutnya dilakukan konversi ruang warna RGB ke grayscale untuk memperoleh informasi intensitas piksel yang diperlukan dalam ekstraksi fitur tekstur menggunakan metode Gray Level Co-occurrence Matrix (GLCM). Penggunaan citra grayscale dilakukan karena GLCM bekerja berdasarkan hubungan spasial antar tingkat keabuan piksel. Selain itu, dilakukan pula konversi ruang warna RGB ke HSV (Hue, Saturation, Value) untuk memperoleh karakteristik

warna citra. Ruang warna HSV dipilih karena mampu merepresentasikan informasi warna secara lebih baik dengan memisahkan komponen warna (*hue*), tingkat kejenuhan (*saturation*), dan tingkat kecerahan (*value*), sehingga lebih efektif dalam menggambarkan distribusi warna pada citra wajah. Hasil dari tahap preprocessing berupa citra grayscale dan citra HSV selanjutnya digunakan sebagai masukan pada proses ekstraksi fitur tekstur dan fitur warna.

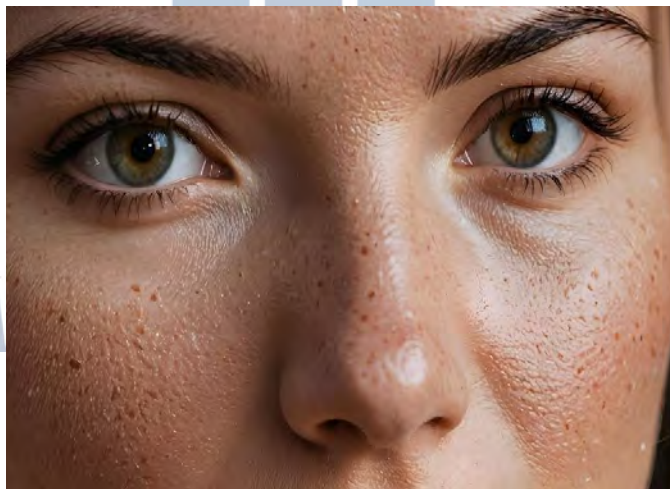
3.7.5 Ekstraksi Fitur

Tahap ekstraksi fitur dilakukan untuk memperoleh karakteristik tekstur dan warna dari citra digital.

3.7.5.1 Ekstraksi Fitur Tekstur menggunakan GLCM

Ekstraksi fitur tekstur dilakukan menggunakan metode Gray Level Co-occurrence Matrix (GLCM). Metode ini digunakan untuk memperoleh informasi tekstur gambar berupa:

- Contrast



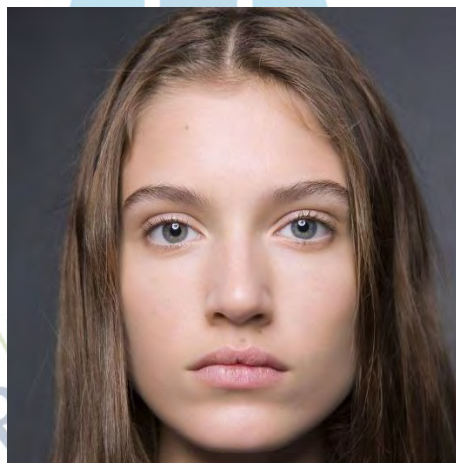
Gambar 3.2 Contoh Karakteristik Contrast

Berdasarkan Gambar 3.2, *Contrast* merupakan salah satu karakteristik tekstur yang digunakan untuk mengukur tingkat perbedaan intensitas antara piksel yang saling berdekatan pada citra. Nilai *Contrast* menggambarkan tingkat variasi intensitas pada suatu tekstur, di mana nilai yang lebih tinggi menunjukkan adanya perbedaan intensitas yang lebih besar antara piksel bertetangga

sehingga tekstur yang dihasilkan terlihat lebih kasar atau memiliki variasi terang dan gelap yang lebih jelas.

Pada gambar tersebut terlihat karakteristik tekstur kulit wajah, seperti pori-pori, lipatan halus, serta bayangan pada beberapa bagian wajah. Area di sekitar mata, hidung, dan mulut menunjukkan adanya perbedaan intensitas piksel yang cukup terlihat. Perbedaan intensitas tersebut menjadi salah satu karakteristik yang direpresentasikan oleh nilai *Contrast*. Dengan demikian, semakin besar perbedaan intensitas antara piksel yang berdekatan, semakin tinggi nilai *Contrast* yang dihasilkan. Fitur ini kemudian digunakan sebagai salah satu representasi tekstur citra dalam proses klasifikasi citra wajah Real dan Fake menggunakan metode SVM.

- Homogeneity

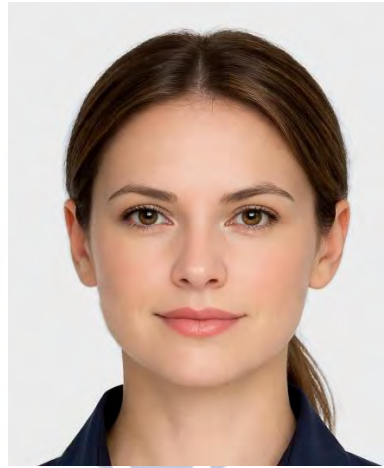


Gambar 3.3 Contoh Wajah dengan Karakteristik Homogeneity

Homogeneity yaitu Mengukur tingkat keseragaman intensitas piksel. Nilai tinggi menunjukkan tekstur yang lebih halus dan seragam.

Pada bagian pipi dan dahi terlihat perubahan intensitas yang berlangsung secara bertahap tanpa perubahan yang tajam. Hal ini menunjukkan tekstur yang relatif seragam sehingga menghasilkan nilai Homogeneity yang tinggi. Semakin mirip intensitas piksel-piksel yang berdekatan, semakin tinggi nilai homogeneity.

- Energy



Gambar 3.4 Contoh wajah dengan karakteristik Energy

Energy adalah mengukur tingkat keteraturan atau keseragaman pola tekstur pada citra. Nilai tinggi menunjukkan pola tekstur yang lebih teratur dan konsisten.

Seperti Gambar diatas dapat dilihat pencahayaan yang merata dan tekstur kulit yang konsisten akan menghasilkan distribusi pasangan piksel yang lebih teratur. Keteraturan ini menyebabkan nilai Energy menjadi lebih tinggi. Energy menggambarkan tingkat keteraturan pola tekstur pada citra.

- Entropy



Gambar 3.5 Contoh Karakteristik Entropy

Entropy yaitu mengukur tingkat kompleksitas atau ketidakteraturan tekstur. Nilai tinggi menunjukkan tekstur yang lebih kompleks dan bervariasi.

Pada karakteristik entropy tampak detail alami kulit seperti pori-pori, rambut halus, garis ekspresi, dan variasi pencahayaan. Banyaknya variasi tekstur tersebut menyebabkan distribusi intensitas menjadi lebih kompleks sehingga menghasilkan nilai Entropy yang tinggi. Entropy menunjukkan tingkat keragaman atau kompleksitas tekstur pada citra.

Fitur tekstur tersebut digunakan untuk membedakan pola permukaan antara gambar wajah asli dan gambar wajah palsu.

3.7.5.2 Ekstraksi Fitur Warna Menggunakan HSV

Ekstraksi fitur warna dilakukan menggunakan metode Hue Saturation Value (HSV). Dari masing-masing komponen hue, saturation, dan value dihitung nilai statistik berupa mean dan standar deviasi. Fitur warna digunakan untuk memperoleh informasi distribusi warna pada citra wajah.

Mean digunakan untuk mengetahui nilai rata-rata warna pada setiap kanal HSV. Semakin besar nilai mean, semakin tinggi rata-rata

nilai komponen warna tersebut pada citra. Nilai mean dihitung menggunakan persamaan berikut.

$$\mu_c = \frac{1}{N} \sum_{i=1}^N c_i$$

Berdasarkan rumus di atas dapat diketahui bahwa:

μ_c = nilai rata-rata (mean) pada komponen warna H , S , atau V .

c_i = nilai piksel ke- i pada komponen yang dihitung.

N = jumlah seluruh piksel pada citra.

Standar deviasi digunakan untuk mengukur tingkat penyebaran nilai piksel terhadap rata-ratanya. Nilai standar deviasi yang tinggi menunjukkan bahwa distribusi warna pada citra lebih bervariasi, sedangkan nilai yang rendah menunjukkan bahwa warna pada citra lebih seragam. Untuk menghitung Standar deviasi dapat menggunakan persamaan berikut.

$$\sigma_c = \sqrt{\frac{1}{N} \sum_{i=1}^N (c_i - \mu_c)^2}$$

σ_c = standar deviasi pada komponen warna (H , S , atau V).

c_i = nilai piksel ke- i pada komponen warna yang dihitung.

μ_c = nilai rata-rata (mean) komponen warna.

N = jumlah seluruh piksel pada citra.

3.7.6 Pembentukan Feature Vector Dataset

Seluruh fitur yang diperoleh dari proses ekstraksi fitur tekstur menggunakan **Gray Level Co-occurrence Matrix (GLCM)** dan fitur warna menggunakan **Hue, Saturation, Value (HSV)** kemudian digabungkan menjadi satu **feature vector**. Feature vector merupakan representasi numerik yang menggambarkan karakteristik setiap citra berdasarkan informasi tekstur dan warna yang telah diekstraksi. Penggabungan kedua jenis fitur tersebut dilakukan untuk menghasilkan representasi citra yang lebih lengkap sehingga dapat digunakan sebagai masukan dalam proses klasifikasi. Seluruh feature vector yang terbentuk kemudian disusun menjadi dataset fitur, di mana setiap

baris merepresentasikan satu citra dan setiap kolom merepresentasikan nilai fitur hasil ekstraksi. Dataset fitur tersebut selanjutnya digunakan sebagai data input pada proses klasifikasi menggunakan algoritma **Support Vector Machine (SVM)**.

3.7.7 Pembagian Dataset

Dataset fitur yang telah terbentuk kemudian dibagi menjadi data *training* dan data *testing* dengan perbandingan 80:20 dari total 4.656 data, yaitu sebanyak 3.724 data digunakan sebagai data pelatihan dan 932 data digunakan sebagai data pengujian. Perbandingan 80:20 dipilih karena mampu memberikan keseimbangan antara proses pembelajaran model dan proses evaluasi. Sebagian besar data digunakan untuk melatih algoritma Support Vector Machine (SVM) agar model dapat mempelajari pola karakteristik citra wajah asli dan citra wajah palsu secara optimal, sedangkan sebagian data lainnya digunakan sebagai data uji untuk mengevaluasi kemampuan model dalam melakukan klasifikasi terhadap data yang belum pernah dipelajari sebelumnya. Pembagian ini juga mendukung proses optimasi parameter menggunakan Particle Swarm Optimization (PSO) sehingga parameter terbaik dapat diperoleh dari data pelatihan yang memadai, sementara hasil pengujian tetap bersifat objektif karena dilakukan pada data yang terpisah dari proses pelatihan. Oleh karena itu, penggunaan perbandingan 80:20 diharapkan mampu menghasilkan model dengan kemampuan generalisasi yang baik serta memberikan nilai evaluasi yang lebih representatif.

3.7.8 Training Model Support Vector Machine (SVM)

Pada tahap ini dilakukan proses pelatihan model menggunakan algoritma **Support Vector Machine (SVM)** dengan kernel **Radial Basis Function (RBF)**. Tujuan pelatihan model adalah membangun fungsi klasifikasi yang mampu membedakan citra wajah **real** dan **fake** berdasarkan *feature vector* hasil ekstraksi fitur warna menggunakan **Hue Saturation Value (HSV)** dan fitur tekstur menggunakan **Gray Level Co-occurrence Matrix (GLCM)**.

Sebelum proses pelatihan dilakukan, seluruh data fitur terlebih dahulu dinormalisasi menggunakan **StandardScaler** agar setiap fitur memiliki skala yang relatif sama. Normalisasi diperlukan karena algoritma SVM bekerja berdasarkan perhitungan jarak antar data pada ruang fitur (*feature space*). Apabila rentang nilai antar fitur berbeda secara signifikan, maka fitur dengan nilai yang lebih besar akan mendominasi proses pembentukan model sehingga dapat menurunkan performa klasifikasi.

Setelah proses normalisasi selesai, dataset dibagi menjadi **80% data pelatihan** dan **20% data pengujian** menggunakan metode **train-test split**. Data pelatihan digunakan untuk membangun model SVM, sedangkan data pengujian digunakan untuk mengevaluasi kemampuan model terhadap data yang belum pernah dipelajari sebelumnya.

Pada penelitian ini digunakan kernel **Radial Basis Function (RBF)** karena kernel tersebut memiliki kemampuan yang baik dalam menangani data yang tidak dapat dipisahkan secara linear. Kernel RBF bekerja dengan memetakan data dari ruang fitur awal ke ruang berdimensi lebih tinggi sehingga data yang semula sulit dipisahkan dapat dibentuk menjadi dua kelompok yang memiliki batas keputusan (*decision boundary*) yang lebih optimal.

Parameter utama yang digunakan pada kernel RBF adalah **C** dan **Gamma**. Parameter **C** berfungsi mengontrol keseimbangan antara tingkat kesalahan klasifikasi dan kompleksitas model. Nilai **C** yang besar akan berusaha mengurangi kesalahan klasifikasi pada data pelatihan, sedangkan nilai yang terlalu kecil menghasilkan batas keputusan yang lebih sederhana namun berpotensi menurunkan akurasi. Sementara itu, parameter **Gamma** mengatur luas pengaruh setiap data pelatihan terhadap pembentukan *decision boundary*. Nilai **Gamma** yang terlalu besar dapat menyebabkan model mengalami *overfitting*, sedangkan nilai yang terlalu kecil dapat menyebabkan *underfitting*.

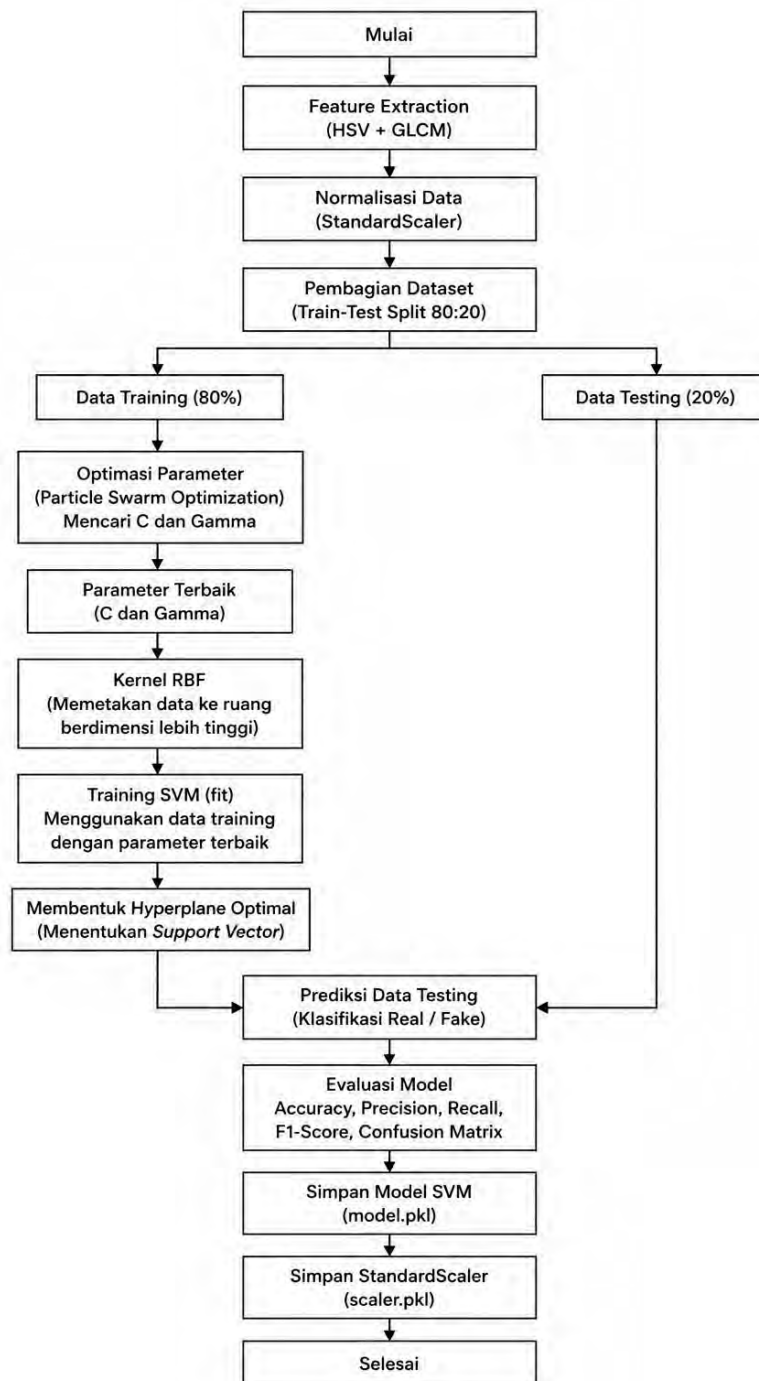
Untuk memperoleh kombinasi parameter terbaik, dilakukan optimasi menggunakan metode **Particle Swarm Optimization (PSO)**. Setiap partikel dalam PSO merepresentasikan pasangan nilai **C** dan **Gamma**. Selanjutnya

setiap kombinasi parameter dievaluasi menggunakan nilai akurasi hasil klasifikasi SVM sebagai fungsi objektif. Proses iterasi dilakukan hingga diperoleh kombinasi parameter yang menghasilkan nilai fungsi objektif terbaik.

Setelah parameter optimal diperoleh, model SVM dibangun menggunakan parameter tersebut melalui fungsi **fit()** pada pustaka **Scikit-learn**. Pada proses ini algoritma SVM mempelajari pola hubungan antara sebelas fitur hasil ekstraksi dengan label kelas (**real** dan **fake**) sehingga terbentuk sebuah model klasifikasi yang mampu memisahkan kedua kelas menggunakan *hyperplane* optimal.

Model yang telah selesai dilatih kemudian disimpan menggunakan pustaka **Joblib** agar dapat digunakan kembali pada proses klasifikasi tanpa perlu melakukan pelatihan ulang. Selain model SVM, objek **StandardScaler** juga disimpan untuk memastikan bahwa data citra baru mengalami proses normalisasi yang sama dengan data pelatihan sebelum dilakukan prediksi.





Gambar 3.6 Diagram Alur Pelatihan Model Support Vector Machine

3.7.9 Optimasi Parameter menggunakan Particle Swarm

Optimization (PSO)

Tahap optimasi dilakukan menggunakan algoritma **Particle Swarm Optimization (PSO)**. Algoritma PSO digunakan untuk mencari kombinasi parameter terbaik pada model **Support Vector Machine (SVM)**, khususnya parameter **C** dan **Gamma (γ)** pada kernel **Radial Basis Function (RBF)**.

Proses optimasi dilakukan secara iteratif hingga diperoleh kombinasi parameter yang mampu menghasilkan performa klasifikasi terbaik berdasarkan nilai fungsi objektif yang digunakan.

Particle Swarm Optimization merupakan algoritma optimasi berbasis populasi (*population-based optimization*) yang terinspirasi dari perilaku kawanan burung atau ikan ketika mencari sumber makanan. Pada algoritma ini terdapat sejumlah **partikel (particles)** yang bergerak di dalam ruang pencarian (*search space*). Setiap partikel merepresentasikan sebuah kandidat solusi, yaitu pasangan nilai **C** dan **Gamma** yang akan digunakan pada model SVM. Selama proses iterasi, setiap partikel akan memperbarui posisi dan kecepatannya berdasarkan pengalaman terbaik yang pernah diperoleh oleh partikel tersebut (*personal best* atau **pBest**) serta pengalaman terbaik seluruh populasi (*global best* atau **gBest**).

Pada penelitian ini ruang pencarian parameter dibatasi pada rentang tertentu agar proses optimasi lebih terarah. Nilai **C** dicari pada rentang **0,1 hingga 10**, sedangkan nilai **Gamma** berada pada rentang **0,0001 hingga 1**. Setiap kombinasi parameter yang dihasilkan oleh partikel selanjutnya digunakan untuk membangun model SVM sementara menggunakan data pelatihan. Model tersebut kemudian diuji menggunakan data pengujian, dan nilai akurasi yang diperoleh digunakan sebagai **fungsi objektif (objective function)**.

Karena pustaka **PySwarms** bekerja dengan konsep **minimisasi**, sedangkan tujuan penelitian adalah memperoleh nilai akurasi yang setinggi mungkin, maka fungsi objektif diubah menjadi nilai negatif dari akurasi (*negative accuracy*). Dengan demikian, proses minimisasi yang dilakukan oleh PSO secara tidak langsung akan menghasilkan kombinasi parameter yang memberikan nilai akurasi tertinggi.

Selama proses optimasi, setiap partikel akan terus memperbarui posisi berdasarkan persamaan kecepatan dan posisi hingga jumlah iterasi yang telah ditentukan tercapai. Pada setiap iterasi dilakukan evaluasi terhadap seluruh partikel untuk menentukan nilai **pBest** dan **gBest**. Proses tersebut

berlangsung secara berulang sampai diperoleh kombinasi parameter yang memberikan nilai fungsi objektif terbaik.

Parameter PSO yang digunakan pada penelitian ini terdiri atas **20 partikel** (**n_particles** = 20) dengan **30 iterasi** (**iters** = 30). Selain itu digunakan parameter **c1** = 0,5, **c2** = 0,3, dan **w** = 0,9. Parameter **c1 (cognitive coefficient)** mengatur kecenderungan partikel mengikuti pengalaman terbaiknya sendiri, **c2 (social coefficient)** mengatur kecenderungan partikel mengikuti solusi terbaik seluruh populasi, sedangkan **w (inertia weight)** berfungsi mengendalikan keseimbangan antara proses eksplorasi dan eksploitasi selama pencarian solusi.

Setelah proses optimasi selesai dilakukan, diperoleh kombinasi parameter **C** dan **Gamma** terbaik. Parameter tersebut kemudian digunakan pada tahap pelatihan model **Support Vector Machine (SVM)** sehingga model yang dibangun memiliki kemampuan klasifikasi yang lebih optimal dibandingkan apabila parameter dipilih secara manual atau menggunakan nilai bawaan (*default*).

Secara keseluruhan, penerapan metode **Particle Swarm Optimization (PSO)** pada penelitian ini bertujuan untuk mengoptimalkan proses pencarian parameter SVM secara otomatis, sehingga diperoleh kombinasi parameter yang mampu meningkatkan kemampuan model dalam membedakan citra wajah **real** dan **fake**.

3.7.10 Pengujian Model

Model **Support Vector Machine (SVM)** yang telah selesai dilatih menggunakan parameter optimal hasil optimasi **Particle Swarm Optimization (PSO)** selanjutnya memasuki tahap pengujian (*testing*). Tahap ini bertujuan untuk mengevaluasi kemampuan model dalam mengklasifikasikan citra wajah yang belum pernah digunakan selama proses pelatihan sehingga dapat diketahui tingkat generalisasi model terhadap data baru.

Pada penelitian ini, proses pengujian dilakukan menggunakan **20% data testing** yang diperoleh melalui proses **train-test split**, sedangkan **80%**

data lainnya digunakan sebagai data pelatihan. Seluruh data uji terlebih dahulu melalui proses ekstraksi fitur menggunakan kombinasi metode **Hue Saturation Value (HSV)** dan **Gray Level Co-occurrence Matrix (GLCM)**. Feature vector yang dihasilkan kemudian dinormalisasi menggunakan **StandardScaler** yang sama dengan proses pelatihan agar skala data tetap konsisten.

Selanjutnya, data uji yang telah dinormalisasi dimasukkan ke dalam model **Support Vector Machine (SVM)** untuk dilakukan proses klasifikasi. Model akan menentukan apakah citra tersebut termasuk ke dalam kelas **real** atau **fake** berdasarkan pola yang telah dipelajari selama proses pelatihan. Selain menghasilkan label prediksi, model juga menghasilkan **nilai probabilitas (confidence)** menggunakan fungsi `predict_proba()`, yang menunjukkan tingkat keyakinan model terhadap hasil klasifikasi yang diberikan.

Hasil prediksi model kemudian dibandingkan dengan label sebenarnya (*ground truth*) pada data pengujian. Perbandingan tersebut digunakan untuk menghitung berbagai metrik evaluasi yang menggambarkan performa model dalam melakukan klasifikasi.

Pada penelitian ini, performa model dievaluasi menggunakan beberapa metrik, yaitu:

- **Accuracy**, untuk mengukur persentase keseluruhan data yang berhasil diklasifikasikan dengan benar.
- **Precision**, untuk mengukur ketepatan model dalam memberikan prediksi pada setiap kelas.
- **Recall**, untuk mengukur kemampuan model dalam menemukan seluruh data pada masing-masing kelas.
- **F1-Score**, untuk mengukur keseimbangan antara nilai *precision* dan *recall*.
- **Confusion Matrix**, untuk mengetahui jumlah prediksi benar dan salah pada masing-masing kelas sehingga kesalahan klasifikasi dapat dianalisis secara lebih rinci.

Hasil evaluasi tersebut selanjutnya digunakan untuk menilai tingkat keberhasilan model dalam membedakan citra wajah **real** dan **fake**, serta menjadi dasar dalam menganalisis kelebihan dan keterbatasan metode yang diterapkan pada penelitian ini.

3.7.11 Evaluasi Sistem

Tahap evaluasi sistem dilakukan untuk mengukur tingkat keberhasilan model **Support Vector Machine (SVM)** yang telah dioptimasi menggunakan metode **Particle Swarm Optimization (PSO)** dalam mengklasifikasikan citra wajah **real** dan **fake**. Evaluasi dilakukan menggunakan data pengujian yang tidak terlibat selama proses pelatihan sehingga hasil yang diperoleh dapat menggambarkan kemampuan model dalam melakukan klasifikasi terhadap data baru (*generalization*).

Pada tahap ini, hasil prediksi model dibandingkan dengan label sebenarnya (*ground truth*) pada data pengujian. Selanjutnya dilakukan perhitungan **Confusion Matrix** sebagai dasar untuk memperoleh berbagai metrik evaluasi. Confusion Matrix memberikan informasi mengenai jumlah prediksi yang benar maupun salah pada masing-masing kelas, sehingga kesalahan klasifikasi dapat dianalisis secara lebih rinci.

Parameter evaluasi yang digunakan pada penelitian ini meliputi:

- **Accuracy**, yaitu persentase keseluruhan data yang berhasil diklasifikasikan dengan benar terhadap seluruh data pengujian.
- **Precision**, yaitu tingkat ketepatan model dalam memberikan prediksi pada setiap kelas.
- **Recall**, yaitu kemampuan model dalam mengenali seluruh data yang sebenarnya termasuk ke dalam suatu kelas.
- **F1-Score**, yaitu nilai yang menunjukkan keseimbangan antara **Precision** dan **Recall**, sehingga dapat memberikan gambaran performa model secara lebih komprehensif.

Selain metrik evaluasi tersebut, penelitian ini juga menggunakan **Confusion Matrix** untuk mengetahui jumlah **True Positive (TP)**, **True Negative (TN)**, **False Positive (FP)**, dan **False Negative (FN)**. Informasi

tersebut digunakan untuk menganalisis jenis kesalahan klasifikasi yang dilakukan oleh model serta mengetahui kemampuan model dalam membedakan citra wajah **real** dan **fake**.

Hasil evaluasi yang diperoleh selanjutnya digunakan sebagai dasar untuk menilai performa model **Support Vector Machine (SVM)** yang telah dioptimasi menggunakan **Particle Swarm Optimization (PSO)**. Nilai **Accuracy, Precision, Recall, F1-Score**, dan **Confusion Matrix** kemudian dibahas secara lebih rinci pada **Pembahasan** sebagai dasar dalam menganalisis keberhasilan maupun keterbatasan sistem deteksi citra wajah palsu yang dikembangkan.

3.7.12 Alur Sistem Aplikasi

Setelah model **Support Vector Machine (SVM)** selesai dilatih dan disimpan, model tersebut diimplementasikan ke dalam aplikasi berbasis **Streamlit** sehingga dapat digunakan untuk mendeteksi citra wajah **real** maupun **fake** secara otomatis. Pada tahap ini, pengguna berinteraksi langsung dengan sistem melalui antarmuka (*user interface*) tanpa perlu menjalankan proses pelatihan model kembali.

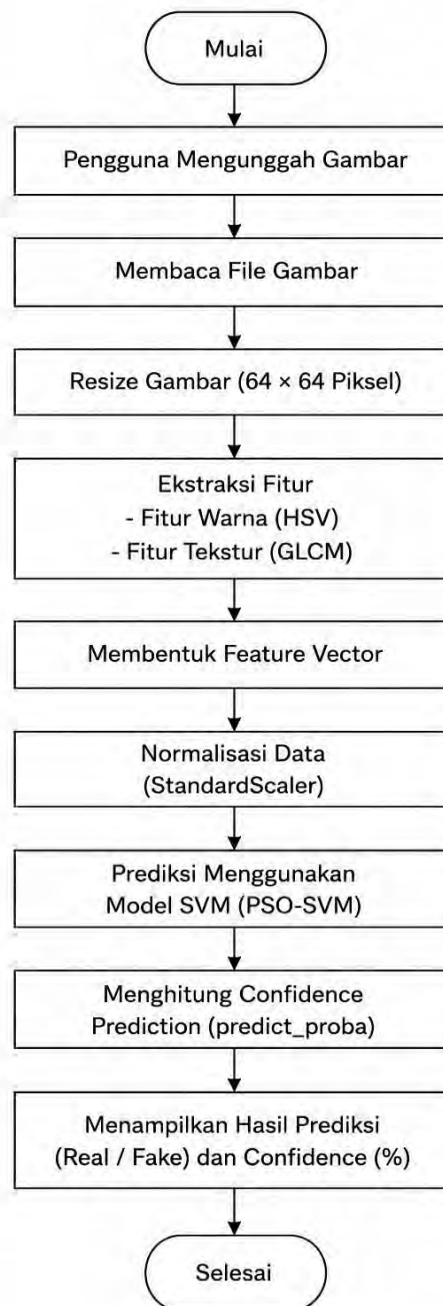
Alur kerja sistem dimulai ketika pengguna mengunggah sebuah citra wajah melalui halaman utama aplikasi. Selanjutnya sistem membaca citra yang diunggah dan melakukan proses *preprocessing* berupa *resize* citra menjadi ukuran **64 × 64 piksel** agar sesuai dengan data pelatihan. Setelah itu dilakukan proses ekstraksi fitur warna menggunakan metode **Hue Saturation Value (HSV)** dan ekstraksi fitur tekstur menggunakan metode **Gray Level Co-occurrence Matrix (GLCM)** sehingga diperoleh *feature vector* yang merepresentasikan karakteristik citra.

Feature vector yang dihasilkan kemudian dinormalisasi menggunakan objek **StandardScaler** yang telah disimpan pada saat proses pelatihan model. Normalisasi dilakukan agar distribusi nilai fitur citra baru memiliki skala yang sama dengan data pelatihan sehingga proses prediksi dapat dilakukan secara konsisten.

Tahap berikutnya adalah proses klasifikasi menggunakan model **Support Vector Machine (SVM)** yang telah dioptimasi menggunakan **Particle Swarm Optimization (PSO)**. Model akan memprediksi kelas citra menjadi **Real** atau **Fake** berdasarkan pola yang telah dipelajari selama proses pelatihan. Selain menghasilkan label prediksi, sistem juga menghitung nilai **confidence** menggunakan fungsi `predict_proba()` sebagai ukuran tingkat keyakinan model terhadap hasil prediksi.

Hasil klasifikasi kemudian ditampilkan pada antarmuka aplikasi berupa label prediksi (**Real** atau **Fake**) beserta nilai **confidence** dalam bentuk persentase. Dengan demikian, pengguna dapat mengetahui hasil identifikasi citra secara langsung tanpa harus melakukan proses pelatihan model kembali.





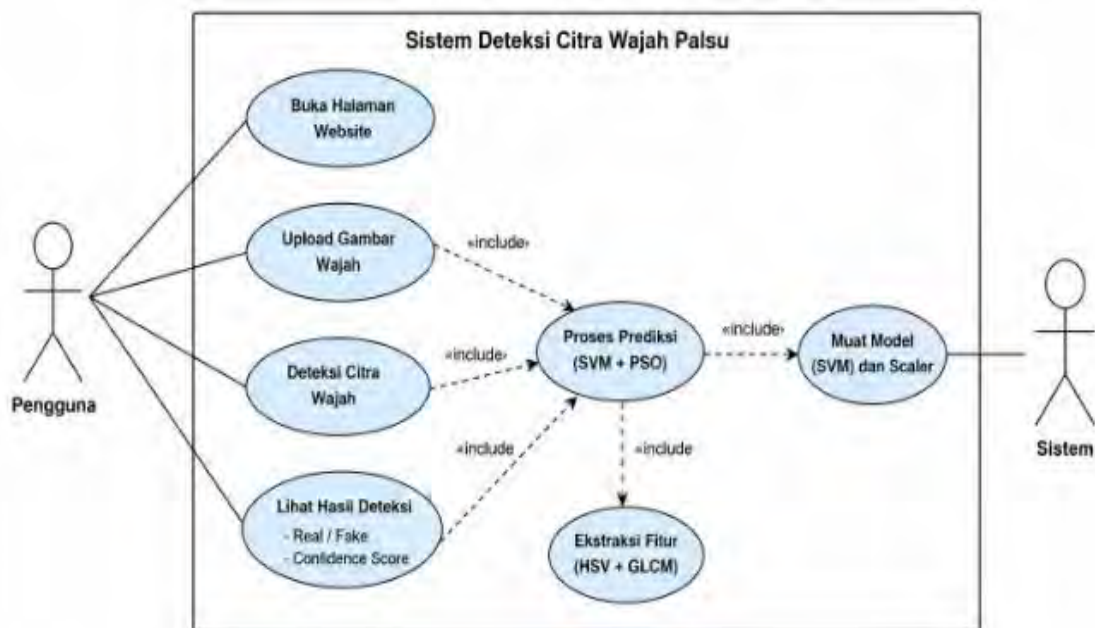
Gambar 3.7 Alur Kerja Sistem Deteksi Citra Wajah Real dan Fake

Berdasarkan gambar 3.7 menunjukkan alur kerja sistem deteksi citra wajah real dan fake. Proses dimulai ketika pengguna mengunggah citra wajah melalui antarmuka aplikasi. Sistem kemudian membaca citra, melakukan *resize* menjadi ukuran 64×64 piksel, dan mengekstraksi fitur warna menggunakan metode HSV serta fitur tekstur menggunakan metode GLCM. Seluruh fitur digabungkan menjadi *feature vector* dan dinormalisasi

menggunakan StandardScaler sebelum diproses oleh model Support Vector Machine (SVM) yang telah dioptimasi menggunakan Particle Swarm Optimization (PSO). Selanjutnya sistem menghasilkan prediksi kelas (*real* atau *fake*) beserta nilai *confidence* yang ditampilkan kepada pengguna.

3.7.13 Use Case Diagram Sistem

Use Case Diagram digunakan untuk menggambarkan interaksi antara pengguna dengan sistem deteksi citra wajah palsu yang dikembangkan. Diagram ini menunjukkan fungsi utama yang dapat diakses oleh pengguna serta batasan sistem dalam menjalankan proses deteksi citra wajah. Sistem dirancang dalam bentuk aplikasi berbasis web sehingga pengguna dapat melakukan proses deteksi melalui antarmuka yang telah disediakan tanpa harus menjalankan proses pengolahan citra dan klasifikasi secara manual.



Gambar 3.8 Use Case Diagram Sistem

Gambar 3.8 merupakan *Use Case Diagram* yang menggambarkan interaksi antara pengguna dengan sistem pada proses deteksi citra wajah palsu. Pada *Use Case Diagram* terdapat satu aktor utama, yaitu **Pengguna**, yang berinteraksi secara langsung dengan sistem. Sementara itu, sistem deteksi citra wajah **real** dan **fake** digambarkan sebagai batas sistem (*system*

boundary) yang di dalamnya terdapat berbagai fungsi untuk melakukan proses deteksi.

Interaksi dimulai ketika pengguna membuka aplikasi dan mengunggah citra wajah yang akan diperiksa. Setelah citra berhasil diterima, sistem secara otomatis memproses citra tersebut menggunakan tahapan yang telah dirancang, yaitu pengolahan citra, ekstraksi fitur warna menggunakan **Hue, Saturation, Value (HSV)**, ekstraksi fitur tekstur menggunakan **Gray Level Co-occurrence Matrix (GLCM)**, pembentukan *feature vector*, normalisasi menggunakan **StandardScaler**, dan klasifikasi menggunakan model **Support Vector Machine (SVM)**.

Fitur HSV digunakan untuk merepresentasikan karakteristik warna pada citra, sedangkan GLCM digunakan untuk memperoleh karakteristik tekstur. Hasil ekstraksi kedua metode tersebut kemudian digabungkan menjadi *feature vector* yang digunakan sebagai masukan pada proses klasifikasi. Proses ekstraksi fitur pada citra yang diunggah menggunakan metode yang sama dengan metode yang digunakan pada saat pembangunan model sehingga jumlah dan bentuk fitur yang dihasilkan sesuai dengan kebutuhan model klasifikasi.

Selanjutnya, sistem memuat **model SVM** dan **StandardScaler** yang telah disimpan sebelumnya. **StandardScaler** digunakan untuk melakukan normalisasi terhadap *feature vector* agar skala fitur sesuai dengan data yang digunakan pada proses pelatihan. Model SVM digunakan untuk menentukan kelas citra berdasarkan karakteristik fitur yang telah diperoleh.

Model SVM yang digunakan telah melalui proses optimasi parameter menggunakan **Particle Swarm Optimization (PSO)**. PSO digunakan pada tahap pembangunan model untuk memperoleh nilai parameter **C** dan **Gamma** yang optimal pada SVM dengan kernel **Radial Basis Function (RBF)**. Oleh karena itu, proses optimasi PSO dan pelatihan SVM **tidak dilakukan kembali** ketika pengguna melakukan deteksi melalui aplikasi. Sistem hanya menggunakan model SVM dan *scaler* yang telah tersedia untuk melakukan prediksi terhadap citra baru.

Hasil klasifikasi terdiri atas dua kelas, yaitu **Real** dan **Fake**. Kelas **Real** menunjukkan bahwa citra diklasifikasikan sebagai citra wajah asli, sedangkan kelas **Fake** menunjukkan bahwa citra diklasifikasikan sebagai citra wajah palsu atau hasil manipulasi. Selain hasil klasifikasi, sistem juga menampilkan **confidence score** sebagai informasi mengenai tingkat keyakinan model terhadap hasil prediksi. Sebagai contoh, apabila sistem menghasilkan prediksi **Fake** dengan *confidence score* sebesar **75,73%**, maka citra tersebut diklasifikasikan sebagai kelas *Fake* dengan tingkat keyakinan sebesar 75,73%.

Secara keseluruhan, *Use Case Diagram* menunjukkan bahwa pengguna hanya perlu membuka aplikasi, mengunggah citra wajah, dan melihat hasil deteksi. Seluruh proses pengolahan citra dan klasifikasi dilakukan secara otomatis oleh sistem. Dengan demikian, terdapat pemisahan antara **proses pembangunan model** dan **proses penggunaan aplikasi**. Proses pelatihan SVM dan optimasi parameter menggunakan PSO dilakukan terlebih dahulu untuk menghasilkan model yang siap digunakan, sedangkan pada saat aplikasi digunakan, sistem memanfaatkan model dan *scaler* yang telah tersedia untuk melakukan prediksi terhadap citra baru **tanpa melakukan *training ulang***.

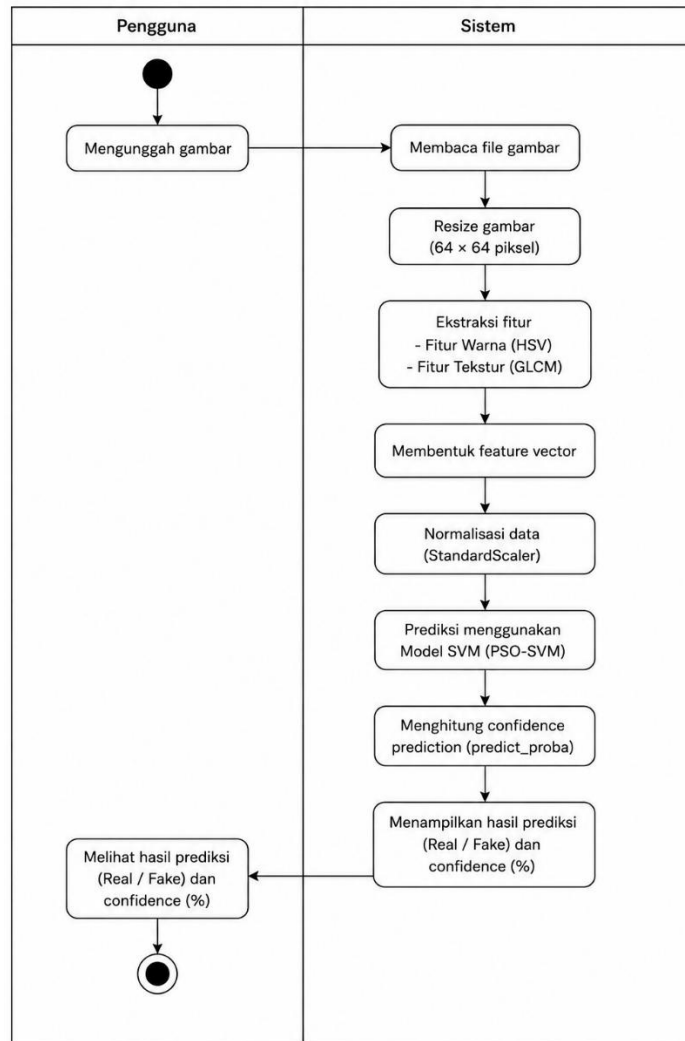
3.7.14 Activity Diagram Sistem

Activity Diagram digunakan untuk menggambarkan urutan aktivitas yang terjadi pada sistem ketika pengguna melakukan proses deteksi citra wajah. Diagram ini menunjukkan pembagian aktivitas antara pengguna dan sistem melalui dua *swimlane*, yaitu **Pengguna** dan **Sistem**.

Proses pada *Activity Diagram* berfokus pada tahapan penggunaan aplikasi setelah model SVM selesai dilatih dan disimpan. Oleh karena itu, proses pelatihan SVM dan optimasi PSO tidak dilakukan kembali ketika pengguna melakukan deteksi. Sistem menggunakan model SVM dan *StandardScaler* yang telah tersedia untuk memproses citra baru

.

.



Gambar 3.9 Activity Diagram Sistem

Berdasarkan Gambar 3.9, proses dimulai pada *initial node* ketika pengguna menjalankan proses deteksi dengan mengunggah gambar wajah ke dalam sistem. Gambar yang diunggah menjadi input utama dalam proses klasifikasi. Setelah gambar diterima, sistem membaca file citra untuk memperoleh data yang diperlukan pada tahap pengolahan berikutnya.

Citra yang telah berhasil dibaca kemudian dilakukan proses *resize* menjadi ukuran **64 × 64 piksel**. Proses *resize* bertujuan untuk menyeragamkan ukuran citra input agar sesuai dengan ukuran citra yang digunakan pada proses pembangunan model. Dengan ukuran citra yang seragam, proses ekstraksi fitur dapat dilakukan secara konsisten.

Setelah proses *resize* selesai, sistem melakukan ekstraksi fitur menggunakan dua metode, yaitu **Hue, Saturation, Value (HSV)** untuk

memperoleh karakteristik warna dan **Gray Level Co-occurrence Matrix (GLCM)** untuk memperoleh karakteristik tekstur. HSV merepresentasikan karakteristik warna berdasarkan komponen *Hue*, *Saturation*, dan *Value*, sedangkan GLCM digunakan untuk memperoleh informasi tekstur berdasarkan hubungan antarintensitas piksel pada citra.

Hasil ekstraksi fitur HSV dan GLCM kemudian digabungkan menjadi satu *feature vector*. *Feature vector* merupakan representasi numerik dari karakteristik citra yang digunakan sebagai masukan bagi model klasifikasi. Selanjutnya, *feature vector* tersebut dinormalisasi menggunakan **StandardScaler**. Normalisasi bertujuan agar skala nilai fitur sesuai dengan skala data yang digunakan ketika model dibangun sehingga proses klasifikasi dapat dilakukan secara konsisten.

Setelah normalisasi selesai, *feature vector* diberikan kepada **model Support Vector Machine (SVM)** dengan kernel **Radial Basis Function (RBF)**. Model yang digunakan telah melalui proses optimasi parameter menggunakan **Particle Swarm Optimization (PSO)** pada tahap pembangunan model. PSO digunakan untuk memperoleh konfigurasi parameter SVM yang optimal, khususnya parameter **C** dan **Gamma**. Oleh karena itu, ketika pengguna melakukan deteksi melalui aplikasi, proses PSO dan pelatihan SVM tidak dijalankan kembali. Sistem langsung menggunakan model SVM yang telah dilatih dan disimpan sebelumnya.

Model SVM kemudian melakukan klasifikasi terhadap citra berdasarkan *feature vector* yang telah dinormalisasi. Hasil klasifikasi terdiri atas dua kelas, yaitu **Real** dan **Fake**. Kelas **Real** menunjukkan bahwa sistem mengklasifikasikan citra sebagai citra wajah asli, sedangkan kelas **Fake** menunjukkan bahwa sistem mengklasifikasikan citra sebagai citra wajah palsu atau hasil manipulasi.

Setelah memperoleh hasil klasifikasi, sistem menghitung nilai *confidence* menggunakan **predict_proba**. Nilai *confidence* digunakan untuk memberikan informasi mengenai tingkat keyakinan model terhadap hasil prediksi pada citra yang sedang dianalisis. Nilai tersebut tidak menunjukkan akurasi keseluruhan model, tetapi menunjukkan probabilitas

prediksi yang diberikan model terhadap kelas hasil klasifikasi pada citra tertentu.

Selanjutnya, sistem menampilkan hasil deteksi kepada pengguna berupa kelas **Real** atau **Fake** beserta nilai **Confidence (%)**. Pengguna dapat melihat hasil tersebut sebagai informasi akhir dari proses klasifikasi. Setelah hasil prediksi ditampilkan, proses berakhir pada *final node*.

Secara keseluruhan, *Activity Diagram* menunjukkan bahwa proses deteksi berlangsung secara berurutan mulai dari **pengunggahan citra**, **pembacaan file**, **resize citra**, **ekstraksi fitur HSV dan GLCM**, **pembentukan *feature vector***, **normalisasi menggunakan StandardScaler**, **klasifikasi menggunakan model SVM hasil optimasi PSO**, **perhitungan *confidence***, **hingga penampilan hasil klasifikasi Real atau Fake kepada pengguna**. Proses tersebut berlangsung secara otomatis setelah pengguna mengunggah citra dan tidak memerlukan proses *training* ulang.

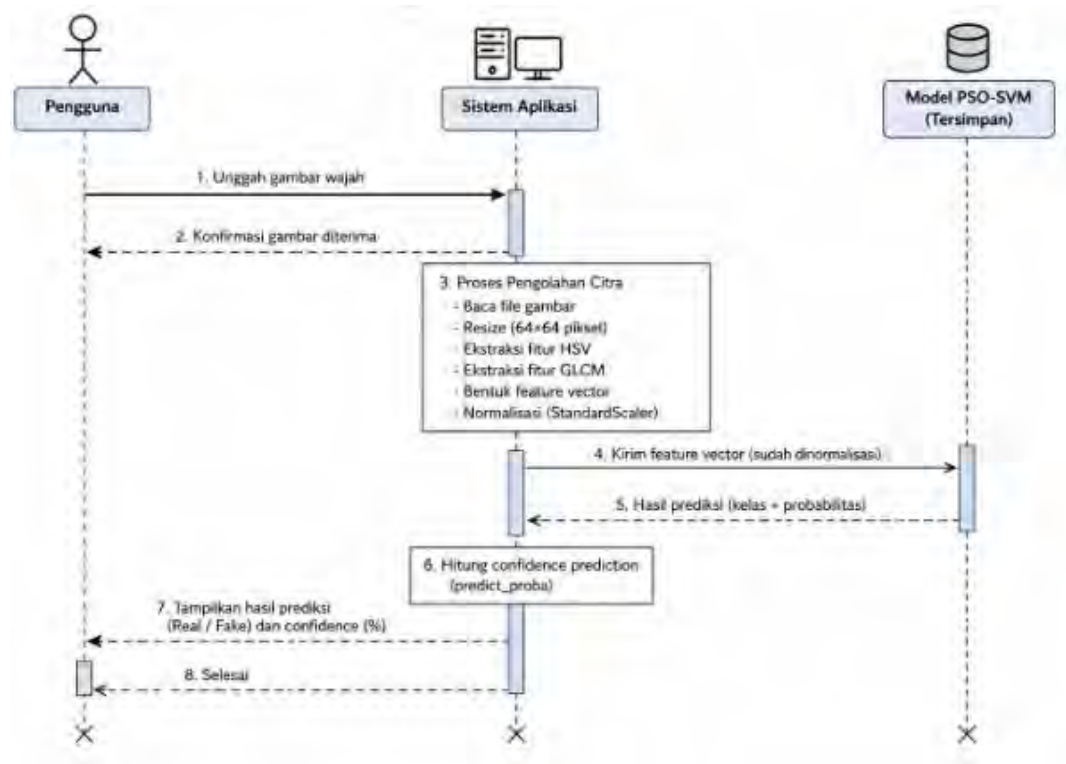
3.7.15 Sequence Diagram Sistem

Sequence Diagram merupakan diagram yang digunakan untuk menggambarkan **urutan interaksi dan pertukaran pesan antara aktor dengan komponen sistem berdasarkan urutan waktu**. Diagram ini menunjukkan bagaimana suatu permintaan dari pengguna diproses oleh sistem dan bagaimana sistem memberikan respons kembali kepada pengguna. Dengan menggunakan *Sequence Diagram*, setiap aktivitas dapat digambarkan secara berurutan sehingga hubungan antara pengguna, sistem, dan model klasifikasi dapat diketahui dengan lebih jelas.

Pada penelitian ini, *Sequence Diagram* digunakan untuk menggambarkan proses deteksi citra wajah **real** dan **fake** pada aplikasi. Diagram melibatkan tiga komponen utama, yaitu **Pengguna**, **Sistem**, dan **Model PSO-SVM**. Pengguna berperan sebagai pihak yang memberikan input berupa citra wajah dan menerima hasil deteksi. Sistem berperan dalam mengelola proses pengolahan citra, ekstraksi fitur, pembentukan *feature vector*, normalisasi, serta komunikasi dengan model klasifikasi. Sementara

itu, Model PSO-SVM berperan dalam melakukan klasifikasi terhadap fitur citra yang telah diproses.

Urutan interaksi antara ketiga komponen tersebut ditunjukkan pada Gambar 3.10.



Gambar 3.10 Sequence Diagram

Berdasarkan Gambar 3.10, proses dimulai ketika **Pengguna mengunggah gambar wajah** melalui aplikasi. Citra tersebut dikirimkan kepada **Sistem** sebagai input untuk proses deteksi. Setelah menerima citra, sistem membaca file untuk memastikan gambar dapat diproses. Selanjutnya, sistem melakukan *resize* citra menjadi ukuran **64 × 64 piksel** agar ukuran input sesuai dengan data yang digunakan pada proses pembangunan model.

Setelah proses *resize* selesai, sistem melakukan **ekstraksi fitur warna menggunakan HSV** dan **ekstraksi fitur tekstur menggunakan GLCM**. Hasil kedua proses tersebut kemudian digabungkan menjadi satu *feature vector*. Selanjutnya, sistem melakukan normalisasi *feature vector* menggunakan **StandardScaler** agar skala fitur sesuai dengan skala data yang digunakan ketika model dibangun.

Setelah proses normalisasi selesai, sistem mengirimkan *feature vector* kepada **Model PSO-SVM** untuk dilakukan klasifikasi. Model tersebut merupakan SVM dengan kernel **RBF** yang parameter **C** dan **Gamma**-nya telah ditentukan melalui proses optimasi menggunakan **PSO** pada tahap pembangunan model. Dengan demikian, proses PSO dan pelatihan SVM tidak dilakukan kembali ketika pengguna melakukan deteksi.

Model PSO-SVM kemudian melakukan prediksi dan menghasilkan kelas **Real** atau **Fake**. Hasil prediksi dikembalikan kepada sistem, kemudian sistem memperoleh nilai probabilitas menggunakan fungsi **predict_proba** yang digunakan sebagai *confidence score*. Nilai tersebut kemudian ditampilkan dalam bentuk persentase.

Selanjutnya, sistem mengirimkan hasil deteksi kepada pengguna melalui antarmuka aplikasi. Hasil yang ditampilkan berupa kelas **Real/Fake** beserta nilai **Confidence (%)**. Setelah pengguna menerima hasil tersebut, proses interaksi selesai.

Dengan demikian, *Sequence Diagram* memperlihatkan secara jelas **urutan komunikasi antar komponen sistem**, mulai dari pengguna memberikan citra, sistem melakukan pemrosesan dan ekstraksi fitur, model PSO-SVM melakukan klasifikasi, hingga sistem mengembalikan hasil deteksi beserta *confidence score* kepada pengguna.

3.8 Evaluasi Hasil Penelitian

Evaluasi hasil penelitian merupakan tahapan penting untuk menilai tingkat keberhasilan sistem deteksi gambar wajah palsu yang dikembangkan. Evaluasi dilakukan untuk mengetahui sejauh mana metode ekstraksi fitur tekstur Gray Level Co-occurrence Matrix (GLCM), fitur warna Hue Saturation Value (HSV), algoritma Support Vector Machine (SVM), serta optimasi parameter menggunakan Particle Swarm Optimization (PSO) mampu menghasilkan performa klasifikasi yang baik dalam membedakan gambar wajah asli dan gambar wajah palsu. Pada tahap ini, hasil pengujian sistem dianalisis untuk memastikan bahwa tujuan penelitian telah tercapai sesuai dengan permasalahan yang telah dirumuskan.

Evaluasi dilakukan menggunakan data testing yang telah dipisahkan dari data training. Model yang telah dilatih kemudian digunakan untuk mengklasifikasikan gambar wajah ke dalam kategori asli atau palsu. Hasil klasifikasi tersebut selanjutnya dibandingkan dengan label sebenarnya untuk mengetahui tingkat ketepatan prediksi yang dihasilkan oleh sistem. Analisis ini bertujuan untuk mengukur kemampuan model dalam mengenali pola karakteristik citra yang diperoleh melalui proses ekstraksi fitur.

Penilaian performa sistem dilakukan menggunakan confusion matrix yang menghasilkan beberapa parameter evaluasi, yaitu accuracy, precision, recall, dan f1-score. Accuracy digunakan untuk mengukur tingkat ketepatan keseluruhan prediksi model. Precision digunakan untuk mengetahui ketepatan model dalam mengidentifikasi gambar wajah palsu, sedangkan recall digunakan untuk mengukur kemampuan model dalam menemukan seluruh gambar wajah palsu yang terdapat pada dataset. F1-score digunakan sebagai ukuran keseimbangan antara precision dan recall sehingga dapat memberikan gambaran yang lebih komprehensif mengenai performa model klasifikasi.

Selain mengevaluasi performa model, penelitian ini juga melakukan perbandingan antara model Support Vector Machine (SVM) standar dengan model SVM yang telah dioptimasi menggunakan Particle Swarm Optimization (PSO). Perbandingan tersebut dilakukan untuk mengetahui pengaruh optimasi parameter terhadap peningkatan performa klasifikasi. Hasil evaluasi digunakan untuk menganalisis kelebihan dan kekurangan masing-masing model serta menentukan model yang memiliki kemampuan deteksi terbaik.

Selanjutnya, hasil penelitian dievaluasi berdasarkan kesesuaiannya dengan teori dan penelitian terdahulu yang relevan. Analisis ini bertujuan untuk mengetahui kontribusi metode yang digunakan dalam meningkatkan kemampuan deteksi gambar wajah palsu. Dengan membandingkan hasil penelitian terhadap literatur yang ada, dapat diketahui sejauh mana metode ekstraksi fitur GLCM dan HSV serta optimasi PSO mampu memberikan hasil yang efektif dalam mendukung proses klasifikasi menggunakan algoritma SVM.

Tahap akhir evaluasi dilakukan dengan menganalisis keseluruhan hasil penelitian untuk menarik kesimpulan mengenai efektivitas sistem yang dibangun.

Hasil evaluasi diharapkan dapat menunjukkan bahwa kombinasi fitur tekstur dan warna yang dipadukan dengan algoritma SVM serta optimasi PSO mampu meningkatkan akurasi deteksi gambar wajah palsu dan memberikan kontribusi dalam pengembangan sistem deteksi manipulasi citra digital.

3.9 Timeline Penelitian

Timeline penelitian merupakan salah satu aspek penting dalam pelaksanaan penelitian karena berfungsi sebagai pedoman dalam mengatur setiap tahapan kegiatan agar dapat diselesaikan secara sistematis dan sesuai dengan target waktu yang telah ditentukan. Penyusunan timeline dilakukan untuk menggambarkan urutan pelaksanaan penelitian mulai dari tahap identifikasi masalah hingga penyusunan laporan penelitian. Dengan adanya timeline yang terstruktur, setiap tahapan penelitian dapat dilaksanakan secara lebih efektif sehingga risiko keterlambatan penyelesaian penelitian dapat diminimalkan.

Tahap awal penelitian diawali dengan identifikasi masalah dan studi literatur. Pada tahap ini dilakukan pengkajian terhadap berbagai referensi seperti jurnal ilmiah, buku, prosiding, dan artikel penelitian yang berkaitan dengan deteksi gambar wajah palsu, pengolahan citra digital, ekstraksi fitur Gray Level Co-occurrence Matrix (GLCM), Hue Saturation Value (HSV), algoritma Support Vector Machine (SVM), serta Particle Swarm Optimization (PSO). Studi literatur bertujuan untuk memperoleh landasan teori, mengetahui perkembangan penelitian terdahulu, serta menentukan metode yang sesuai dengan permasalahan yang diteliti.

Setelah memperoleh dasar teori yang memadai, penelitian dilanjutkan dengan proses pengumpulan dataset gambar wajah asli dan gambar wajah palsu yang berasal dari sumber dataset publik. Dataset yang telah diperoleh kemudian dipersiapkan melalui tahap preprocessing yang meliputi penyeragaman ukuran gambar (resize), konversi citra RGB ke grayscale untuk kebutuhan ekstraksi fitur tekstur menggunakan metode GLCM, serta konversi citra RGB ke ruang warna HSV untuk memperoleh karakteristik warna citra. Tahapan preprocessing dilakukan untuk menghasilkan data yang memiliki kualitas dan format yang seragam sehingga dapat diproses pada tahap berikutnya.

Tahap selanjutnya adalah ekstraksi fitur menggunakan metode GLCM dan HSV. Metode GLCM digunakan untuk memperoleh karakteristik tekstur citra melalui fitur contrast, homogeneity, energy, dan entropy, sedangkan metode HSV digunakan untuk memperoleh karakteristik warna melalui nilai statistik berupa rata-rata (mean) dan standar deviasi pada setiap kanal warna. Seluruh fitur hasil ekstraksi kemudian digabungkan menjadi sebuah feature vector yang akan digunakan sebagai data masukan pada proses klasifikasi.

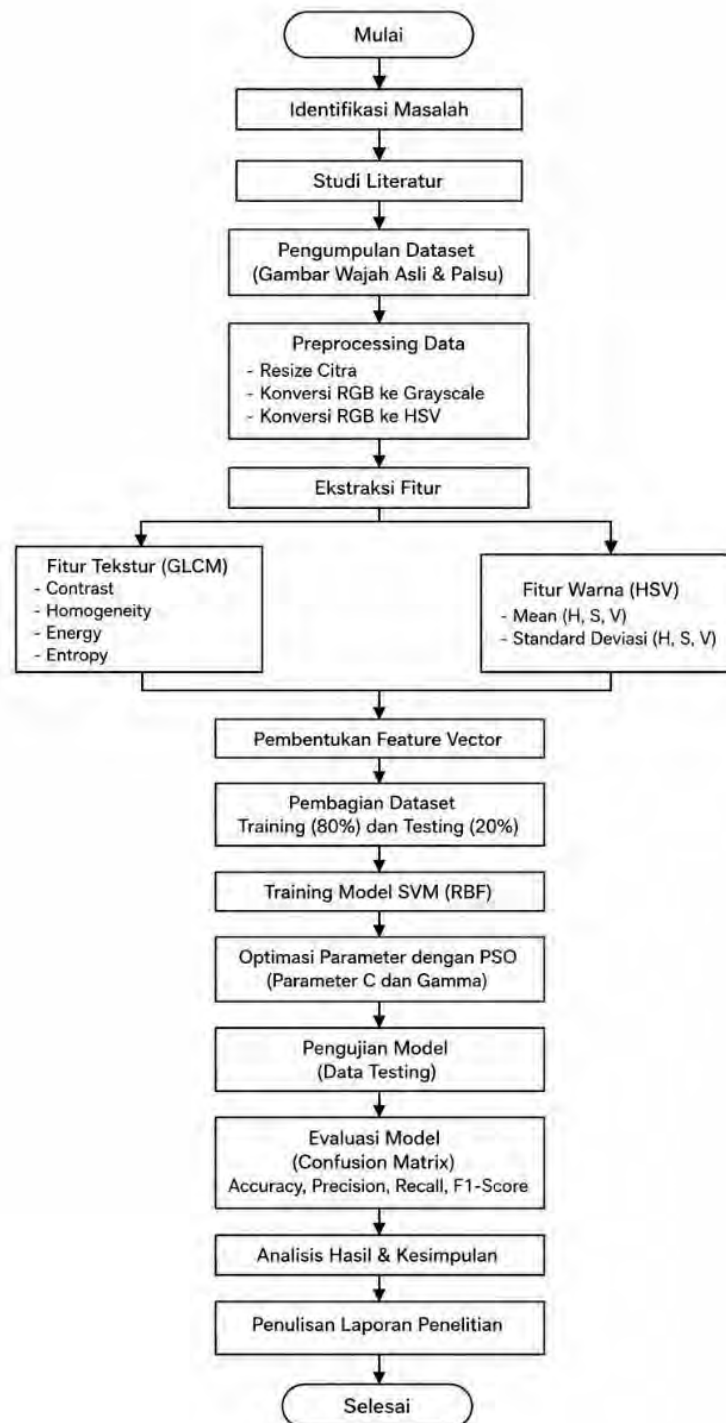
Setelah proses ekstraksi fitur selesai, dataset dibagi menjadi data pelatihan (*training*) dan data pengujian (*testing*) dengan perbandingan 80:20. Data pelatihan digunakan untuk membangun model klasifikasi menggunakan algoritma Support Vector Machine (SVM), sedangkan data pengujian digunakan untuk mengukur kemampuan model dalam mendeteksi gambar wajah asli dan gambar wajah palsu.

Untuk meningkatkan performa klasifikasi, dilakukan proses optimasi parameter SVM menggunakan algoritma Particle Swarm Optimization (PSO). Optimasi dilakukan terhadap parameter C dan gamma (γ) sehingga diperoleh kombinasi parameter terbaik yang mampu menghasilkan tingkat akurasi klasifikasi yang lebih tinggi dibandingkan dengan penggunaan parameter secara manual.

Model yang telah dilatih dan dioptimasi kemudian diuji menggunakan data pengujian. Hasil klasifikasi selanjutnya dievaluasi menggunakan confusion matrix dengan menghitung nilai accuracy, precision, recall, dan F1-score. Selain itu, dilakukan pula perbandingan antara performa model SVM tanpa optimasi dan model SVM yang telah dioptimasi menggunakan PSO untuk mengetahui pengaruh optimasi terhadap peningkatan kinerja sistem.

Tahap akhir penelitian adalah analisis hasil penelitian, penyusunan laporan tugas akhir, proses revisi berdasarkan masukan dari dosen pembimbing, serta persiapan sidang tugas akhir. Selama proses penelitian berlangsung, peneliti juga melakukan konsultasi dan evaluasi secara berkala dengan dosen pembimbing untuk memastikan setiap tahapan penelitian berjalan sesuai dengan rencana yang telah ditetapkan. Dengan adanya timeline penelitian yang tersusun secara sistematis dan realistis, diharapkan seluruh rangkaian penelitian dapat

diselesaikan tepat waktu serta menghasilkan penelitian yang berkualitas dan sesuai dengan tujuan yang telah dirumuskan.



Gambar 3.11 Alur Tugas Akhir