

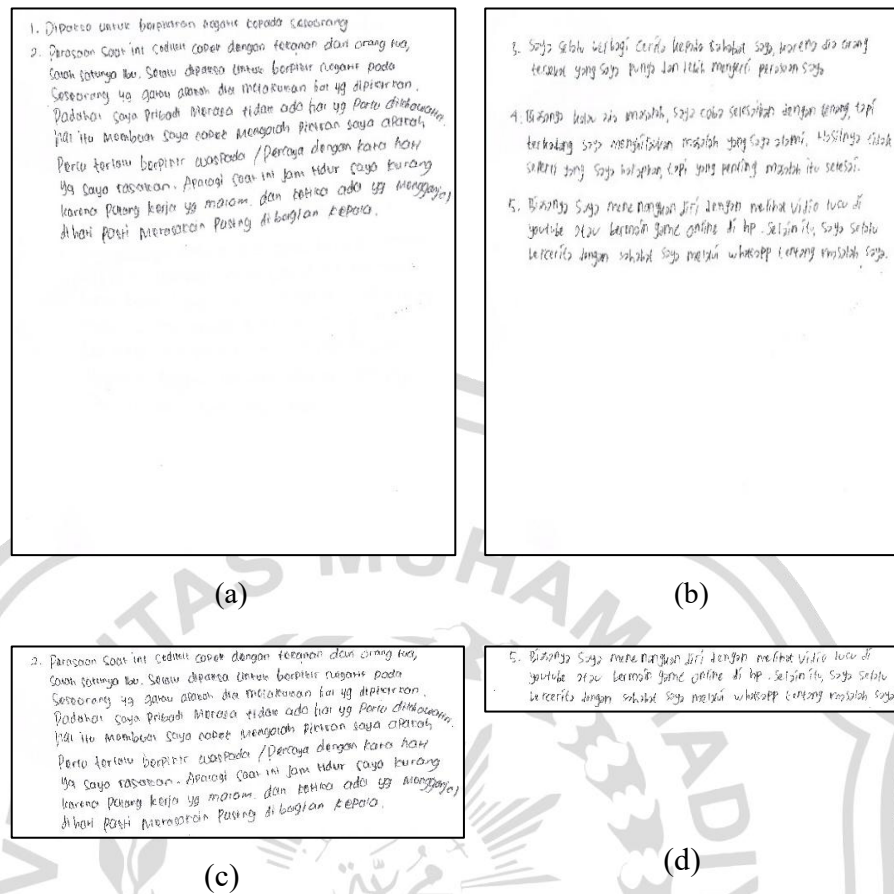
## **BAB III**

### **ANALISA DAN PERANCANGAN SISTEM**

#### **3.1 Analisis Sistem**

Penelitian ini membahas proses pendeteksian keteraturan tulisan tangan secara offline menggunakan pendekatan pengolahan citra digital. Penilaian keteraturan tulisan tangan yang selama ini dilakukan secara manual sering kali menghasilkan hasil yang berbeda antarindividu karena bergantung pada persepsi masing-masing. Kondisi tersebut menimbulkan perlunya sistem otomatis yang mampu menilai tingkat keteraturan tulisan tangan secara objektif dan konsisten.

Sebelum dilakukan pengolahan lebih lanjut, citra tulisan tangan dilakukan proses pemotongan (cropping) untuk memfokuskan area analisis pada bagian tulisan dan menghilangkan area kosong. Pada penelitian ini, tulisan tangan teratur didefinisikan sebagai tulisan dengan bentuk huruf yang konsisten, jarak antarhuruf dan antar-kata yang proporsional. selain itu tulisan teratur menempatkan tulisan sesuai dengan zona (**Gambar 2.1**). Sebaliknya, tulisan tangan tidak teratur memiliki variasi bentuk huruf yang tinggi, jarak tulisan yang tidak seragam, zona tulisan lebih banyak di zona tengah karena adanya ketidaksesuaian kaidah penulisan. Hal ini akan membentuk tekstur yang tidak teratur pula. Perbandingan citra sebelum dan sesudah proses cropping pada kategori tulisan teratur dan tidak teratur ditampilkan pada **Gambar 3.1**



**Gambar 3.1** Perbandingan citra tulisan tangan sebelum dan sesudah proses cropping. (a) Citra tulisan tangan teratur utuh, (b) Citra tulisan tangan tidak teratur utuh (c) Citra tulisan tangan teratur setelah di-crop (d) Citra tulisan tangan tidak teratur setelah di-crop

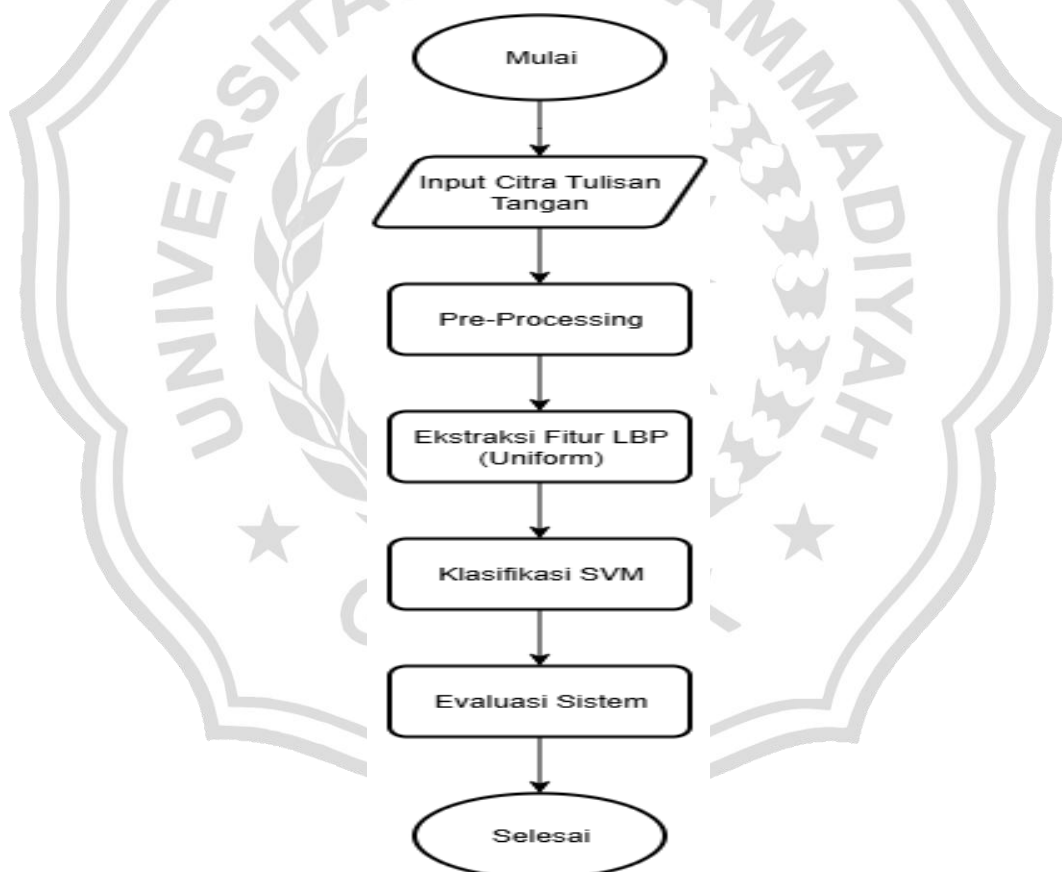
Proses cropping ini berfungsi untuk memastikan sistem hanya menganalisis area tulisan yang relevan, sehingga hasil ekstraksi fitur dapat dilakukan secara lebih fokus dan akurat.

### 3.2 Perancangan Sistem

Perancangan sistem pada penelitian ini bertujuan untuk menggambarkan alur kerja sistem secara menyeluruh, mulai dari proses input citra tulisan tangan hingga tahap evaluasi hasil klasifikasi. Sistem dirancang agar mampu mendeteksi tingkat keteraturan tulisan tangan melalui beberapa tahapan

yang saling berkaitan, yaitu prapemrosesan citra untuk meningkatkan kualitas visual, ekstraksi fitur menggunakan metode Local Binary Pattern (LBP) Uniform, serta klasifikasi dengan algoritma Support Vector Machine (SVM). Alur kerja keseluruhan sistem divisualisasikan dalam bentuk flowchart yang ditampilkan pada **Gambar 3.2**.

Data yang digunakan dalam sistem ini merupakan citra tulisan tangan yang dikelompokkan menjadi dua kategori, yaitu tulisan teratur dan tidak teratur. Citra diperoleh melalui proses pemindaian atau pemotretan, kemudian digunakan sebagai data masukan pada tahap prapemrosesan. Seluruh rangkaian proses, mulai dari pengolahan citra hingga klasifikasi, diimplementasikan menggunakan perangkat lunak MATLAB R2021a.



**Gambar 3.2** Flowchart Perancangan Sistem

Pada proses klasifikasi keteraturan tulisan tangan, tahapan dimulai dengan penginputan citra tulisan tangan yang diperoleh dari hasil

pemindaian atau pemotretan. Setiap citra berformat RGB dimasukkan ke dalam sistem sebagai data awal yang akan diproses lebih lanjut. Citra tersebut kemudian melewati tahap preprocessing, di mana dilakukan konversi citra dari RGB ke grayscale menggunakan fungsi `rgb2gray`, serta dilakukan peningkatan kualitas citra melalui pengaturan kontras dan penghilangan noise. Tujuan dari tahap ini adalah untuk memperjelas pola tulisan dan mengurangi gangguan visual yang dapat memengaruhi hasil analisis.

Tahapan selanjutnya adalah ekstraksi fitur menggunakan metode Local Binary Pattern (LBP) Uniform. Proses ini dilakukan dengan membandingkan nilai intensitas piksel pusat terhadap piksel-piksel di sekelilingnya untuk menghasilkan pola biner yang mewakili tekstur lokal tulisan tangan. Hasil dari proses ini kemudian dikonversi menjadi nilai numerik dan disusun dalam bentuk histogram fitur yang merepresentasikan karakteristik tekstur dari setiap citra. Informasi tekstur inilah yang nantinya digunakan sebagai masukan pada tahap klasifikasi.

Pada tahap akhir, dilakukan klasifikasi menggunakan algoritma Support Vector Machine (SVM). Data hasil ekstraksi fitur ULBP digunakan untuk melatih model agar dapat membedakan antara tulisan tangan yang teratur dan tidak teratur. Algoritma SVM akan mencari hyperplane terbaik yang memisahkan kedua kelas dengan margin maksimum. Setelah proses klasifikasi selesai, sistem menghasilkan keluaran berupa kategori tulisan tangan dan selanjutnya dilakukan evaluasi untuk mengetahui tingkat akurasi, presisi, dan performa sistem dalam mendeteksi keteraturan tulisan tangan secara otomatis.

### 3.2.1 Dataset

Citra tulisan tangan yang digunakan dalam penelitian ini adalah data hasil pemindaian tulisan tangan dari beberapa responden yang terbagi menjadi dua kategori, yaitu tulisan tangan teratur dan tulisan tangan tidak teratur, seperti terlihat pada **Gambar 3.3**.

R:150 G:150 B:160	R:159 G:159 B:169	R:188 G:188 B:198	R:160 G:160 B:170
R:161 G:161 B:173	R:177 G:177 B:187	R:179 G:179 B:189	R:201 G:201 B:211
R:184 G:184 B:196	R:190 G:190 B:202	R:191 G:191 B:203	R:176 G:176 B:188
R:201 G:201 B:213	R:195 G:195 B:207	R:212 G:212 B:224	R:189 G:189 B:201
R:213 G:212 B:226	R:215 G:214 B:228	R:195 G:194 B:208	R:203 G:202 B:216
R:223 G:223 B:235	R:222 G:222 B:234	R:215 G:215 B:227	R:207 G:206 B:220

s) Citra tersebut dengan tingkat dari setiap support, semangot dan perisai & keran.

(a)

(b)

**Gambar 3.3** (a). Citra tulisan tangan, (b). Nilai RGB tulisan tangan

Dalam citra RGB, setiap piksel memiliki tiga kanal warna, yaitu Red, Green, dan Blue. Misalnya, pada citra dengan ukuran  $4962 \times 537$  piksel, jumlah elemen warna yang diproses adalah  $4962 \times 537 \times 3 = 7.993.782$ . Hal ini menunjukkan bahwa citra berwarna membutuhkan komputasi yang cukup besar, sehingga diperlukan konversi ke citra grayscale untuk menyederhanakan proses pengolahan.

### 3.2.2 Pemrosesan Data Awal (Pre-processing)

Citra yang diperoleh dari hasil cropping kemudian dikonversi ke dalam format grayscale untuk menghasilkan citra keabuan. Proses ini bertujuan menyederhanakan tahapan pengolahan citra selanjutnya, karena citra keabuan hanya memiliki satu kanal intensitas piksel dengan rentang nilai antara 0 hingga 255. Nilai intensitas keabuan diperoleh dari kombinasi tiga komponen warna utama, yaitu merah (Red), hijau (Green), dan biru (Blue) pada setiap piksel. Sebagai contoh, jika sebuah piksel memiliki nilai intensitas  $R = 150$ ,  $G = 150$ , dan  $B = 160$ , Maka nilai keabuan dihitung menggunakan **persamaan**

(2.1) sebagai berikut:

$$\begin{aligned}
 Gray &= (0.299) * 150 + (0.587) * 150 \\
 &\quad + (0.114 * 160) \\
 &= 44,85 + 88,05 + 18,24 \\
 &= 151,14
 \end{aligned}$$

Berdasarkan hasil perhitungan tersebut, diperoleh nilai intensitas keabuan sebesar 151, yang menunjukkan bahwa piksel tersebut memiliki tingkat kecerahan menengah cenderung terang pada citra hasil konversi.

Setelah proses konversi ke citra keabuan, citra grayscale terlebih dahulu diproses melalui tahapan pre-processing lanjutan, yaitu pengurangan noise menggunakan median filter, invers citra, serta peningkatan kontras menggunakan metode Contrast Limited Adaptive Histogram Equalization (CLAHE).

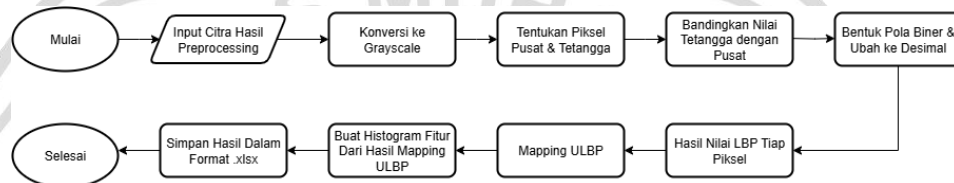
Tahapan-tahapan tersebut bertujuan untuk menonjolkan area tulisan tangan dan meningkatkan kejelasan tekstur citra dengan tetap mempertahankan informasi intensitas piksel yang dibutuhkan pada tahap ekstraksi fitur menggunakan metode Local Binary Pattern (LBP). Oleh karena itu, proses binerisasi tidak diterapkan pada tahap ini.

### 3.2.3 Proses Ekstraksi Fitur

Setelah proses *preprocessing* dilakukan, citra hasil olahan selanjutnya melalui tahap ekstraksi fitur untuk memperoleh karakteristik utama yang akan digunakan dalam proses klasifikasi. Pada penelitian ini, metode Uniform Local Binary Pattern (ULBP) digunakan untuk mengekstraksi pola tekstur mikro dari citra tulisan tangan. Proses ini diimplementasikan menggunakan fungsi *built-in* MATLAB `extractLBPFeatures()`, yang menghitung pola biner berdasarkan perbandingan intensitas antara piksel pusat dan piksel tetangga di area lokal citra.

Dalam implementasinya, digunakan parameter radius = 1 dan numNeighbors = 8, yang berarti setiap piksel pusat dibandingkan dengan delapan piksel di sekelilingnya pada jarak satu piksel. Parameter Upright diset ke *true* agar hasil ekstraksi tidak dipengaruhi oleh rotasi citra. Nilai hasil perbandingan intensitas kemudian dikonversi menjadi pola biner delapan bit, diubah ke bentuk desimal, dan dipetakan ke pola *uniform* untuk menghasilkan histogram fitur yang lebih stabil terhadap variasi tekstur.

Tahapan proses ekstraksi fitur menggunakan metode ULBP secara keseluruhan ditunjukkan pada **Gambar 3.4**.



**Gambar 3.4** Flowchart Tahapan Ekstraksi Fitur ULBP

Flowchart ini menunjukkan tahapan proses ekstraksi fitur menggunakan metode Uniform Local Binary Pattern (ULBP). Proses diawali dengan citra hasil preprocessing yang dikonversi ke grayscale, kemudian nilai piksel pusat dibandingkan dengan piksel tetangga untuk membentuk pola biner. Pola tersebut dikonversi menjadi nilai desimal dan dipetakan ke dalam pola uniform sebelum menghasilkan histogram fitur. Hasil akhir proses ini disimpan dalam format .xlsx sebagai dataset untuk tahap klasifikasi berikutnya.

**Tabel 3.1** Contoh Hasil Ekstraksi Fitur ULBP

No	F1	F2	F3	...	F57	F58	F59
1	0.008	0.002	0.0004	...	0.0004	0.999	0.019
2	0.016	0.004	0.0009	...	0.0007	0.999	0.035
3	0.013	0.003	0.0007	...	0.0005	0.999	0.028
4	0.012	0.003	0.0006	...	0.0005	0.999	0.026
5	0.015	0.004	0.0008	...	0.0007	0.999	0.034

Sebanyak 59 fitur dihasilkan dari proses ekstraksi menggunakan metode Uniform Local Binary Pattern (ULBP). Fitur-fitur tersebut merepresentasikan pola tekstur lokal yang terbentuk berdasarkan distribusi nilai biner pada setiap area citra tulisan tangan. Nilai-nilai fitur ini menjadi representasi karakteristik tekstur yang digunakan sebagai masukan pada tahap klasifikasi.

#### 3.2.4 Klasifikasi SVM

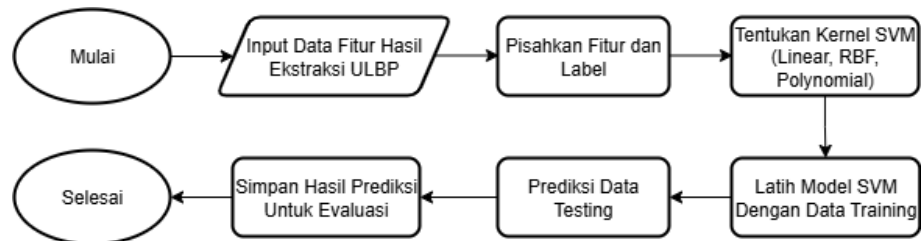
Setelah proses ekstraksi fitur ULBP dilakukan, tahap berikutnya adalah klasifikasi menggunakan algoritma Support Vector Machine (SVM). Tahap ini bertujuan untuk membedakan antara tulisan tangan yang tergolong teratur dan tidak teratur berdasarkan fitur tekstur yang telah diperoleh dari hasil ekstraksi sebelumnya. Algoritma SVM dipilih karena mampu melakukan pemisahan data dengan batas (hyperplane) yang optimal sehingga hasil klasifikasi menjadi lebih akurat dan efisien.

Pada tahap ini, data fitur hasil ekstraksi dibagi menjadi dua bagian, yaitu data pelatihan (training) dan data pengujian (testing). Data pelatihan digunakan untuk membangun model klasifikasi, sedangkan data pengujian berfungsi untuk menguji kemampuan model dalam mengenali pola baru. Setiap citra tulisan tangan memiliki label tertentu sebagai penanda kategori kelas, yaitu tulisan tangan teratur dan tidak teratur.

Dalam implementasinya, fungsi `fitcecoc` pada MATLAB R2021a digunakan untuk melatih model SVM. Beberapa jenis kernel diterapkan untuk mendapatkan hasil klasifikasi terbaik, meliputi linear, radial basis function (RBF), dan polynomial dengan orde tiga. Penggunaan berbagai kernel ini dimaksudkan untuk membandingkan performa model dalam menghasilkan tingkat akurasi yang paling optimal pada data pengujian. Pada tahap implementasi dan pengujian, parameter pada masing-masing kernel SVM dapat disesuaikan untuk

memperoleh performa klasifikasi yang optimal. Pemilihan kernel terbaik ditentukan berdasarkan hasil akurasi tertinggi yang diperoleh dari data uji.

Tahapan proses klasifikasi SVM secara umum dapat dilihat pada **Gambar 3.5**.



**Gambar 3.5** Flowchart Tahapan Klasifikasi SVM

Flowchart pada Gambar 3.5 menggambarkan urutan proses klasifikasi menggunakan algoritma SVM, dimulai dari input data fitur hasil ekstraksi ULBP, dilanjutkan dengan pemisahan antara fitur dan label kelas. Setelah itu, sistem menentukan jenis kernel SVM yang digunakan, seperti linear, RBF, atau polynomial. Selanjutnya dilakukan pelatihan model menggunakan data training, kemudian model tersebut digunakan untuk memprediksi data testing. Hasil prediksi yang diperoleh akan digunakan pada tahap evaluasi sistem untuk menghitung performa klasifikasi berdasarkan nilai akurasi, precision, recall, dan F1-score.

### 3.2.5 Evaluasi sistem

Setelah hasil prediksi diperoleh dari proses klasifikasi menggunakan algoritma SVM, tahap selanjutnya adalah melakukan evaluasi sistem untuk menilai kinerja model dalam mengenali tingkat keteraturan tulisan tangan. Evaluasi ini dilakukan dengan menggunakan Confusion Matrix, yang berfungsi untuk membandingkan antara hasil prediksi sistem dan data sebenarnya. Matriks ini menunjukkan jumlah data yang terklasifikasi dengan benar

maupun yang salah untuk masing-masing kategori, yaitu tulisan tangan teratur dan tidak teratur.

Dari Confusion Matrix diperoleh beberapa parameter pengukuran performa, yaitu akurasi, presisi, recall, dan F1-score. Akurasi digunakan untuk mengukur tingkat ketepatan sistem dalam mengklasifikasikan data secara keseluruhan, presisi menunjukkan sejauh mana hasil prediksi benar untuk kelas tertentu, sedangkan recall menggambarkan kemampuan sistem dalam mendeteksi seluruh data yang benar dari setiap kelas. Adapun F1-score merupakan rata-rata harmonis antara presisi dan recall yang memberikan penilaian seimbang terhadap kinerja model.

Perhitungan nilai akurasi, presisi, recall, dan F1-score dilakukan menggunakan **persamaan (2.7) hingga (2.10)** yang telah dijelaskan pada Bab II. Seluruh proses perhitungan metrik evaluasi dilakukan secara otomatis oleh perangkat lunak MATLAB berdasarkan hasil prediksi model SVM, sehingga hasil yang diperoleh akurat dan konsisten. Hasil dari pengujian ini digunakan untuk mengevaluasi performa keseluruhan sistem serta menentukan tingkat efektivitas metode klasifikasi yang diterapkan.

### **3.3 Rencana pengujian**

Citra yang digunakan dalam skripsi ini yaitu :

1. Citra yang digunakan dalam skripsi ini ada 213 citra yang akan di crop secara manual dan dibagi menjadi data latih dan data uji.
2. Ekstraksi ciri tekstur dilakukan menggunakan metode Uniform Local Binary Pattern (ULBP).
3. Hasil ekstraksi fitur kemudian diklasifikasikan menggunakan metode Support Vector Machine (SVM). Skripsi ini akan melakukan uji coba kernel: linear, RBF, dan polynomial.
4. Langkah selanjutnya yaitu menghitung nilai akurasi dari hasil klasifikasi dengan membandingkan jumlah kelas yang benar

terhadap jumlah seluruh data uji dan dikalikan 100% menggunakan rumus akurasi (**persamaan 2.7**) pada tiap kernel yang digunakan dalam klasifikasi SVM.

5. Hasil pengujian ditampilkan dalam bentuk confusion matrix untuk melihat performa sistem terhadap setiap kelas. hasil percobaan ditampilkan pada **Tabel 3.2** Confusion Matrix sebagai berikut:

**Tabel 3 2** Matriks Hasil Prediksi

		Hasil Prediksi	
		Teratur	Tidak Teratur
Kelas Asli	Teratur	A	B
	Tidak Teratur	C	D

Keterangan:

A = Tulisan teratur terbaca teratur

B = Tulisan teratur terbaca tidak teratur

C = Tulisan tidak teratur terbaca teratur

D = Tulisan tidak teratur terbaca tidak teratur

### 3.4 Kebutuhan perangkat keras dan lunak

Perangkat yang digunakan dalam penelitian ini terdiri atas beberapa komponen yang berfungsi untuk mendukung proses pengolahan citra dan deteksi tulisan tangan. Adapun spesifikasi perangkat keras yang digunakan adalah sebagai berikut:

- a. Perangkat Keras (Laptop)
  - Merek : ASUS VivoBook
  - Tipe : X415JAB\_A416JA
  - Prosesor : Intel® Core™ i3-1005G1 CPU @ 1.20 GHz
  - GPU : Intel® UHD Graphics
  - RAM : 8.00 GB
  - Storage : 715 GB
  - Sistem Operasi : Windows 11 (64-bit Operating System, x64-based processor)

- Scanner : epon L3000, 600 dpi

b. Perangkat Lunak (Software)

Beberapa perangkat lunak digunakan untuk membantu proses implementasi dan pengujian sistem, diantaranya:

1. Microsoft Office Word: Versi 365, digunakan untuk penyusunan laporan skripsi.
2. Microsoft Excel: Versi 365, digunakan untuk menyimpan dan menampilkan hasil ekstraksi fitur LBP dalam format tabel.
3. MATLAB: Versi R2021a, digunakan untuk proses pengolahan citra, ekstraksi fitur, serta klasifikasi SVM.
4. Google Chrome: Digunakan untuk pencarian referensi dan literatur pendukung penelitian.

