

BAB II

LANDASAN TEORI

2.1 Keteraturan Tulisan Tangan

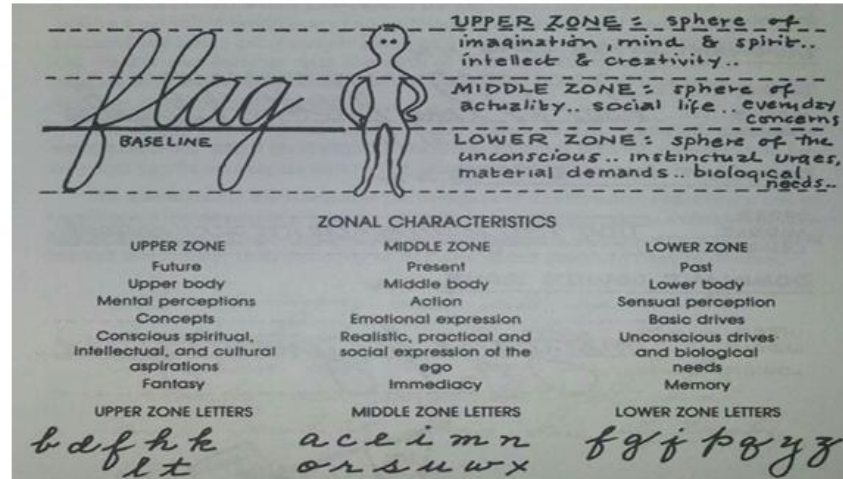
Tulisan tangan merupakan aktivitas kompleks yang melibatkan koordinasi antara sistem motorik, persepsi visual, dan proses kognitif. Aktivitas ini tidak hanya berfungsi sebagai sarana komunikasi, tetapi juga mencerminkan kondisi fisiologis dan psikologis seseorang. Keteraturan tulisan tangan berhubungan erat dengan kemampuan koordinasi dan kontrol otot kecil yang berperan dalam proses penulisan.

Dari sudut pandang neuropsikologis, tulisan tangan juga mencerminkan aktivitas otak yang kompleks, di mana setiap goresan melibatkan hubungan antara area motorik, sensorik, dan linguistik. Analisis tulisan tangan dapat digunakan sebagai alat bantu dalam penilaian neuropsikologis karena variasi pola tulisan dapat menunjukkan perubahan fungsi kognitif seseorang (8). Oleh karena itu, keteraturan tulisan tangan dapat menjadi indikator penting dalam mendeteksi gangguan neurologis maupun psikologis.

Selain aspek motorik dan neurologis, keteraturan tulisan tangan juga berkaitan dengan faktor kepribadian dan kondisi emosional. Respons neural terhadap tulisan tangan dipengaruhi oleh ciri-ciri kepribadian individu, yang menunjukkan bahwa setiap orang memiliki gaya menulis unik yang dapat mencerminkan karakter psikologisnya (9). Dalam konteks ini, tingkat keteraturan tulisan tangan dapat dijadikan salah satu parameter untuk mengevaluasi stabilitas emosional serta fokus seseorang pada saat menulis.

Dalam grafologi, keteraturan tulisan tangan dianalisis melalui beberapa aspek utama, seperti konsistensi bentuk huruf, kestabilan jarak antarhuruf dan antar-kata, serta sesuai dengan zona tulisan yang terdiri dari zona atas (*upper zone*), zona tengah (*middle zone*), dan zona bawah (*lower zone*). Pembagian zona ini digunakan untuk melihat kestabilan struktur huruf, kontrol motorik, dan kecenderungan emosional penulis. Prinsip-prinsip grafologi tersebut menjadi acuan visual dalam membedakan tulisan teratur

dan tidak teratur sebelum dianalisis lebih lanjut menggunakan metode pengolahan citra (10).



Gambar 2.1 Zona Tulisan

Kajian terhadap aspek keteraturan tulisan tangan memberikan manfaat signifikan dalam bidang informatika, psikologi, maupun kesehatan, terutama dalam pengembangan sistem otomatis berbasis pengolahan citra untuk menilai atau mendeteksi pola tulisan tangan secara objektif.

2.2 Pengolahan Citra Digital

Pengolahan citra digital merupakan bidang kajian dalam ilmu komputer yang berkaitan dengan bagaimana suatu citra dapat dimanipulasi, dianalisis, dan ditingkatkan kualitasnya dengan bantuan perangkat digital. Tujuan utamanya adalah memperoleh informasi visual yang lebih jelas serta dapat digunakan untuk keperluan analisis lebih lanjut. Dalam praktiknya, pengolahan citra melibatkan beberapa tahapan seperti peningkatan kualitas gambar (enhancement), penghilangan derau (noise reduction), segmentasi, dan ekstraksi fitur yang menjadi dasar bagi proses klasifikasi citra (11).

Kemajuan teknologi komputasi memungkinkan berbagai metode pengolahan citra dikembangkan dengan hasil yang semakin presisi. Proses pra-pemrosesan (preprocessing) menjadi tahap penting yang berfungsi mempersiapkan citra sebelum dianalisis oleh algoritma pembelajaran mesin.

Tahapan ini biasanya mencakup konversi warna menjadi skala keabuan, normalisasi pencahayaan, penajaman tepi, serta pemisahan objek dari latar belakang. Hasil dari tahap ini sangat menentukan keberhasilan sistem dalam mengenali pola atau karakteristik tertentu dari citra (12).

Dalam konteks pengenalan tulisan tangan, pengolahan citra berperan penting untuk memastikan bentuk tulisan yang bervariasi dapat dikonversi menjadi data numerik yang konsisten dan representatif. Proses peningkatan citra dan segmentasi membantu meminimalkan gangguan visual seperti bayangan, perbedaan tekanan pena, atau variasi ukuran huruf. Tahapan-tahapan tersebut menjadi landasan bagi metode ekstraksi fitur seperti Local Binary Pattern (LBP), yang digunakan dalam penelitian ini untuk mengidentifikasi pola tekstur tulisan tangan secara lebih akurat (12).

Secara keseluruhan, pengolahan citra digital tidak hanya berfungsi sebagai tahap awal dalam analisis citra, tetapi juga menjadi fondasi penting dalam penerapan metode berbasis machine learning. Melalui kombinasi teknik pra-pemrosesan dan ekstraksi fitur yang tepat, sistem klasifikasi dapat bekerja secara lebih efisien dan memberikan hasil yang objektif serta konsisten.

2.3 Pre – Processing

Tahap *pre-processing* merupakan proses awal dalam pengolahan citra digital yang bertujuan untuk meningkatkan kualitas citra agar siap digunakan pada tahap analisis berikutnya, seperti ekstraksi fitur dan klasifikasi. Langkah ini penting untuk menghilangkan *noise*, menyesuaikan tingkat pencahayaan, serta mempertegas pola tekstur pada citra. Tahapan umum yang dilakukan meliputi konversi citra berwarna (*RGB*) menjadi *grayscale*, penyaringan *noise* dengan median *filter*, invers citra, serta peningkatan kontras menggunakan *Contrast Limited Adaptive Histogram Equalization (CLAHE)* (13).

Konversi citra ke *grayscale* dilakukan untuk mengurangi kompleksitas data dengan mempertahankan informasi intensitas cahaya tanpa melibatkan

komponen warna. Proses konversi tersebut dapat dituliskan dengan Persamaan (2.1):

$$Gray = (0.299) * R + (0.587) * G + (0.114 * B) \quad (2.1)$$

di mana R , G , dan B merupakan nilai intensitas dari masing-masing komponen warna merah, hijau, dan biru pada citra, sedangkan $Gray$ adalah hasil konversi ke tingkat keabuan. Tahapan ini menghasilkan citra yang lebih sederhana dan siap diproses pada tahap pengolahan berikutnya.

Dalam konteks pengenalan tulisan tangan, *pre-processing* berperan penting untuk mempertahankan detail bentuk huruf dan pola tekstur tulisan agar mudah dianalisis. Tahapan seperti konversi ke grayscale, pengurangan noise menggunakan median filter, invers citra, serta peningkatan kontras menggunakan metode CLAHE membantu memperjelas kontur dan tekstur tulisan tangan sebelum proses ekstraksi fitur dilakukan (14). Dengan demikian, *pre-processing* menjadi langkah penting dalam menghasilkan citra berkualitas tinggi yang siap diproses menggunakan metode *Local Binary Pattern (LBP)* dan *Support Vector Machine (SVM)*.

2.4 Local Binary Pattern (LBP)

2.4.1 Pengertian Local Binary Pattern (LBP)

Local Binary Pattern (LBP) merupakan salah satu metode ekstraksi ciri berbasis tekstur yang digunakan untuk mengenali pola pada citra digital. Teknik ini bekerja dengan cara membandingkan nilai intensitas piksel pusat terhadap piksel-piksel tetangganya di area tertentu. Setiap hasil perbandingan dikonversi menjadi nilai biner (0 atau 1) berdasarkan apakah piksel tetangga memiliki intensitas lebih besar atau lebih kecil dari piksel pusat. Nilai-nilai biner tersebut kemudian disusun menjadi kode desimal yang merepresentasikan pola tekstur lokal dari suatu citra. Dengan demikian, LBP mampu menggambarkan karakteristik visual suatu objek secara efisien tanpa dipengaruhi oleh perubahan pencahayaan. Metode ini banyak digunakan pada berbagai penelitian pengenalan pola, termasuk dalam

klasifikasi citra dan analisis tulisan tangan (13).

LBP menjadi populer karena kesederhanaannya dan kemampuannya menghasilkan fitur yang stabil serta mudah diintegrasikan dengan algoritma klasifikasi. Dalam proses penerapannya, hasil ekstraksi dari metode ini biasanya direpresentasikan dalam bentuk histogram yang menunjukkan distribusi pola tekstur pada citra. Histogram inilah yang kemudian digunakan sebagai masukan untuk proses klasifikasi dengan metode tertentu.

Pada penelitian pengenalan tulisan tangan, LBP terbukti efektif dalam mendeskripsikan perbedaan pola guratan yang muncul pada setiap karakter karena metode ini peka terhadap variasi tekstur yang halus (14).

2.4.2 Konsep Dasar Local Binary Pattern (LBP)

Konsep dasar metode Local Binary Pattern (LBP) terletak pada proses pembentukan pola biner berdasarkan perbandingan nilai intensitas antar piksel dalam area lokal citra. Setiap piksel pusat dianalisis bersama tetangganya untuk menentukan hubungan tekstur mikro yang ada di sekitarnya. Perbandingan ini menghasilkan pola biner yang mencerminkan variasi tekstur pada area tersebut. Pola biner yang terbentuk kemudian dikonversi menjadi nilai desimal yang digunakan untuk membentuk histogram distribusi fitur tekstur pada citra (15).

Selain itu, dalam penerapan metode LBP diperlukan penentuan parameter jumlah tetangga (P) dan radius (R) yang digunakan pada proses pembentukan pola biner. Konfigurasi $P = 8$ dengan $R = 1$ merupakan pengaturan yang paling umum digunakan karena memberikan keseimbangan optimal antara kemampuan menangkap detail tekstur lokal dan efisiensi komputasi (16). Konfigurasi $P = 8$ dianggap sebagai *default neighborhood configuration* pada banyak

penelitian LBP, sementara radius 1 efektif untuk memodelkan pola mikro pada tekstur berukuran kecil. Oleh karena itu, penelitian ini menggunakan parameter $P = 8$ dan $R = 1$.

Secara matematis, nilai LBP dapat dihitung menggunakan persamaan berikut:

$$LBP_{P,R}(x, y) = \sum_{i=0}^{p-1} S(g_p - g_c) 2^i \quad (2.2)$$

dengan fungsi threshold:

$$S(x) = \begin{cases} 0, & \text{jika } x < 0 \\ 1, & \text{jika } x \geq 0 \end{cases} \quad (2.3)$$

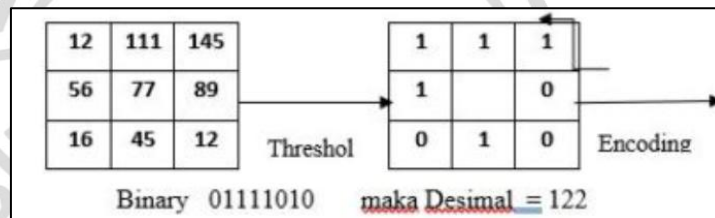
di mana g_c adalah nilai intensitas piksel pusat dan g_p merupakan nilai intensitas piksel tetangga ke- p . Nilai P menunjukkan jumlah tetangga yang digunakan, sedangkan R adalah jarak radius dari piksel pusat. Nilai hasil konversi biner menjadi desimal inilah yang nantinya digunakan sebagai fitur utama dalam proses klasifikasi. Metode ini dinilai sederhana namun efektif, serta memiliki ketahanan terhadap perubahan pencahayaan karena bekerja pada perbandingan relatif intensitas, bukan nilai absolut (17), (18).

2.4.3 Langkah Kerja Local Binary Pattern (LBP)

Langkah kerja metode Local Binary Pattern (LBP) terdiri dari beberapa proses yang dilakukan secara berurutan untuk menghasilkan representasi tekstur citra dalam bentuk histogram fitur. Setiap tahap menggambarkan proses pembentukan pola tekstur lokal mulai dari perbandingan intensitas piksel, pembentukan pola biner, hingga penghitungan distribusi nilai LBP. Ilustrasi lengkap mengenai proses LBP disajikan melalui lima gambar berikut.

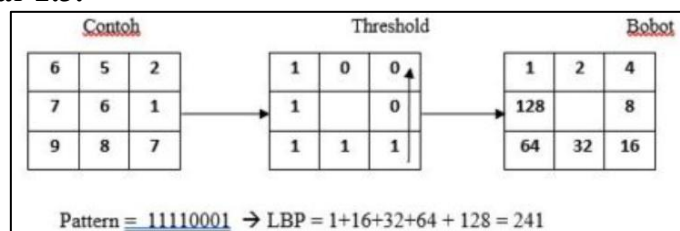
Tahap pertama dalam proses LBP adalah melakukan thresholding pada nilai intensitas piksel. Pada tahap ini, sebuah piksel pusat (center

pixel) dipilih sebagai referensi, kemudian nilai intensitasnya dibandingkan dengan nilai delapan piksel tetangga di sekelilingnya. Jika nilai tetangga lebih besar atau sama dengan nilai pusat, maka piksel tersebut diberi nilai 1; jika lebih kecil maka diberi nilai 0. Proses ini menghasilkan delapan digit biner yang mewakili pola tekstur lokal di sekitar piksel pusat. Pola biner inilah yang menjadi dasar pembentukan nilai LBP yang kemudian digunakan pada tahap penghitungan bobot dan nilai akhir, sebagaimana ditunjukkan pada **Gambar 2.2.**



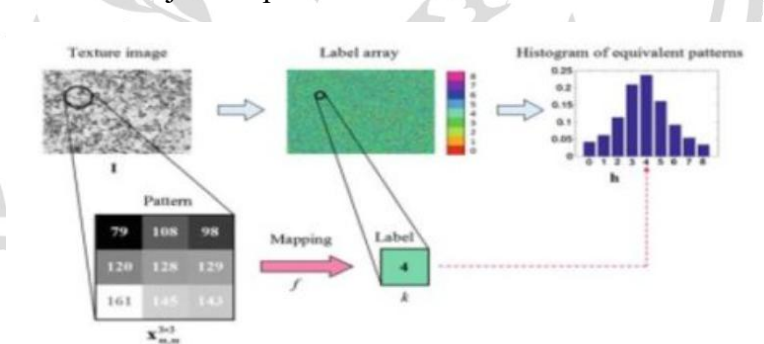
Gambar 2.2 Tahapan proses threshold dan encoding pada LBP (19)

Tahap selanjutnya adalah proses perhitungan nilai LBP berdasarkan pola biner yang telah diperoleh pada setiap piksel. LBP merepresentasikan pola tekstur lokal pada daerah tersebut. Setelah delapan digit biner terbentuk, masing-masing posisi biner memiliki nilai bobot tertentu yang diurutkan secara searah jarum jam. Nilai biner tersebut kemudian dikalikan dengan bobot posisinya, lalu dijumlahkan. Nilai inilah yang akan menggantikan nilai yang ada di tengah ketetanggan. Perhitungan ini menunjukkan bagaimana variasi pola biner dapat menggambarkan karakteristik tekstur seperti tepi, titik, maupun area datar pada citra, sebagaimana ditunjukkan pada **Gambar 2.3.**



Gambar 2.3 Contoh perhitungan dengan menggunakan LBP (19)

Kerangka kerja LBP secara global menggambarkan bagaimana fitur tekstur diekstraksi dari sebuah citra melalui serangkaian proses yang saling terhubung. Citra awal yang telah melalui proses cropping dan konversi ke grayscale diperlakukan sebagai *texture image* yang menjadi masukan utama. Setiap piksel pada citra tersebut kemudian diproses menggunakan metode LBP untuk membentuk pola biner, yaitu hasil perbandingan nilai intensitas piksel pusat terhadap delapan piksel tetangganya. Pola biner yang dihasilkan selanjutnya melalui tahap *mapping*, yang bertujuan mengubah pola biner tersebut ke dalam nilai desimal atau kategori tertentu. Pada varian seperti Uniform LBP (ULBP), proses pemetaan ini juga mencakup pengelompokan pola ke dalam kategori *uniform* dan *non-uniform* sehingga representasinya lebih ringkas dan stabil. Hasil proses mapping membentuk sebuah *label array*, yaitu citra baru yang setiap pikselnya menunjukkan nilai LBP hasil transformasi dari citra asli. Tahap terakhir adalah membangun *histogram of patterns*, yaitu menghitung frekuensi kemunculan setiap nilai LBP dalam *label array*. Histogram ini kemudian menjadi vektor fitur yang merepresentasikan karakteristik tekstur citra dan digunakan pada proses klasifikasi, sebagaimana ditunjukkan pada **Gambar 2.4**.

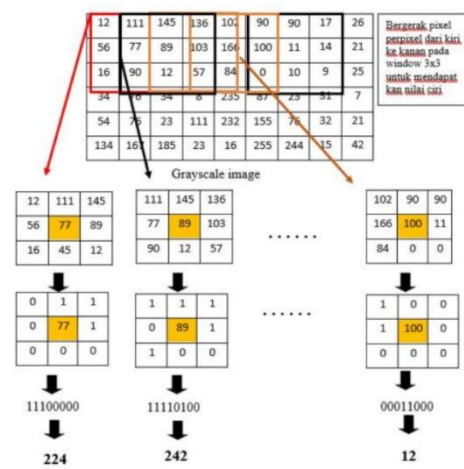


Gambar 2.4 Kerangka kerja dari LBP sebagai metode ekstraksi ciri (19)

Tahap berikutnya adalah penerapan Local Binary Pattern menggunakan teknik *sliding window* berukuran 3×3 piksel. Window

ini digeser ke seluruh area citra secara sistematis, satu piksel demi satu piksel, baik secara horizontal maupun vertikal. Pada setiap posisi window, satu piksel bertindak sebagai piksel pusat, sementara delapan piksel yang mengelilinginya berfungsi sebagai tetangga. Nilai intensitas piksel pusat kemudian dibandingkan dengan masing-masing tetangga untuk membentuk pola biner 8-bit.

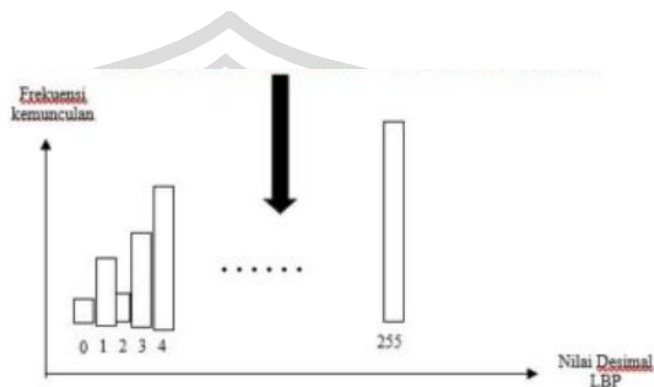
Setelah pola biner terbentuk, pola tersebut dikonversi menjadi nilai desimal sehingga menghasilkan nilai LBP untuk piksel tersebut. Proses ini diulang pada seluruh posisi window sehingga setiap piksel di dalam citra yang dapat diproses memiliki nilai LBP masing-masing. Contoh proses pergeseran window dan perhitungan nilai LBP ditunjukkan pada **Gambar 2.5**, yang memperlihatkan bagaimana window bergerak dari kiri ke kanan dan dari atas ke bawah untuk mengekstraksi fitur tekstur secara menyeluruh.



Gambar 2.5 Tahapan proses dari local binary pattern untuk menghitung nilai biner (19)

Tahap terakhir dalam proses ekstraksi fitur LBP adalah membentuk histogram dari seluruh nilai LBP yang telah diperoleh melalui proses sliding window. Setiap nilai LBP yang muncul pada label array direkap berdasarkan frekuensi kemunculannya sehingga menghasilkan distribusi pola tekstur, sebagaimana ditunjukkan pada

Gambar 2.6. Setiap batang pada histogram merepresentasikan seberapa sering suatu pola LBP tertentu muncul pada citra, sehingga histogram tersebut menjadi vektor fitur yang kompak dan mampu menggambarkan karakteristik tekstur citra secara menyeluruh. Vektor fitur inilah yang kemudian digunakan sebagai masukan pada tahap klasifikasi, karena memiliki sifat yang stabil, ringkas, dan mudah dianalisis.



Gambar 2.6 histogram berdasarkan frekuensi nilai dari nilai fitur dengan operator LBP (19)

Untuk memperoleh representasi tekstur yang lebih efisien dan stabil, penelitian ini menggunakan varian Uniform Local Binary Pattern (ULBP), yaitu pengembangan dari LBP yang mengelompokkan pola biner dengan jumlah transisi maksimal dua sehingga menghasilkan fitur yang lebih ringkas namun tetap informatif.

2.4.4 Uniform Local Binary Pattern (ULBP)

Uniform Local Binary Pattern (ULBP) merupakan pengembangan dari metode *Local Binary Pattern (LBP)* konvensional yang dirancang untuk meningkatkan efisiensi dalam proses ekstraksi ciri tekstur. Berbeda dengan LBP standar yang memperhitungkan seluruh kombinasi pola biner, ULBP hanya mempertimbangkan pola dengan maksimal dua transisi biner untuk meningkatkan efisiensi dan

kestabilan fitur. Pada metode ini, pola biner dikategorikan sebagai *uniform* apabila hanya memiliki paling banyak dua transisi antara nilai 0 dan 1 pada urutan delapan piksel tetangga di sekitar piksel pusat. Contohnya, pola seperti 00000000 atau 11110000 termasuk ke dalam pola *uniform* karena perubahan bit-nya hanya terjadi satu kali dalam satu siklus.

Pendekatan ini secara signifikan mengurangi jumlah pola yang mungkin dari 256 menjadi 59 pola utama. Dengan demikian, proses ekstraksi fitur menjadi lebih efisien tanpa kehilangan karakteristik penting dari tekstur citra (7), (20). Pola *non-uniform* yang memiliki lebih dari dua transisi biasanya dikelompokkan ke dalam satu kategori umum agar histogram fitur yang dihasilkan lebih sederhana dan stabil terhadap variasi pencahayaan serta rotasi citra.

ULBP digunakan untuk menghasilkan representasi tekstur yang lebih robust, terutama dalam aplikasi pengenalan pola dan analisis citra medis. (20) menunjukkan bahwa penerapan ULBP mampu menurunkan ukuran vektor fitur hingga sekitar 3,5% dari ukuran awalnya, tanpa mengurangi akurasi klasifikasi secara signifikan. Hal ini karena fitur yang dihasilkan tetap mampu menggambarkan informasi lokal dari citra secara efisien.

Sebelum menentukan apakah suatu pola termasuk *uniform* atau tidak, jumlah transisi antara $0 \rightarrow 1$ atau $1 \rightarrow 0$ pada urutan piksel tetangga dihitung menggunakan fungsi berikut:

$$uLBP_{P,R} = \sum_{i=0}^p |S(g_i - g_c) - s(g_{i-1} - g_c)| \quad (2.4)$$

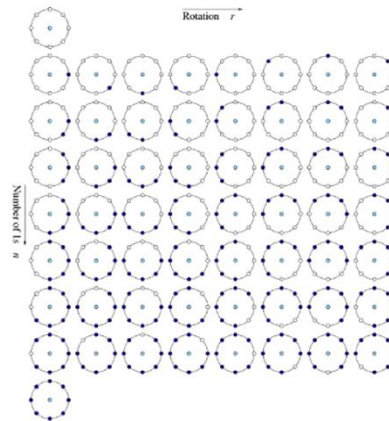
Pola dikatakan *uniform* apabila memenuhi syarat transisi:
 $U(LBP_{P,R}) \leq 2$.

Nilai ULBP dihitung menggunakan persamaan LBP standar dengan penambahan syarat *uniform*. Secara matematis dapat dinyatakan sebagai:

$$uLBP_{P,R} = \begin{cases} \sum_{i=0}^{p-1} S(g_p - g_c)2^i, & \text{Jika } U(LBP_{P,R}) \\ P + 1, & \\ \leq 2 \text{ Lainnya} & \end{cases} \quad (2.5)$$

di mana g_c adalah nilai intensitas piksel pusat, g_p adalah nilai intensitas tetangga, dan fungsi $s(x)$ menghasilkan nilai 1 jika $x \geq 0$ serta 0 jika sebaliknya.

Gambar berikut menunjukkan contoh pola uniform pada metode ULBP, mulai dari perbandingan nilai piksel tetangga terhadap piksel pusat hingga pembentukan pola biner.

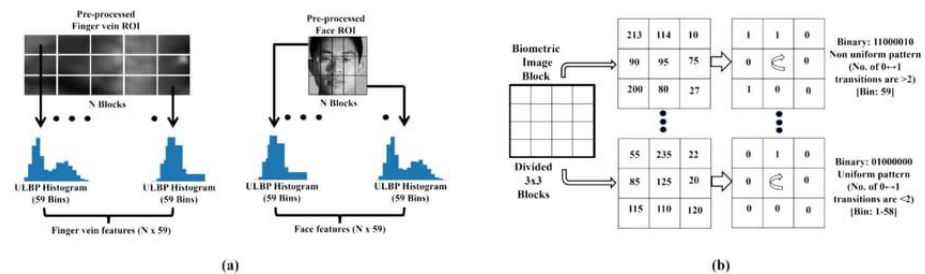


Gambar 2.7 Pola Uniform Local Binary Pattern (uLBP) (7).

Pola pada gambar tersebut menunjukkan representasi visual dari uniform LBP. Pola dikategorikan sebagai uniform apabila hanya terdapat dua kali perubahan nilai biner dari 0 ke 1 atau sebaliknya pada satu putaran penuh delapan piksel tetangga. Dengan membatasi pola hanya pada transisi yang minimal, ULBP mampu mengurangi kompleksitas kombinasi menjadi 59 pola utama sehingga lebih efisien dan stabil terhadap perubahan rotasi serta pencahayaan.

Selain contoh pola uniform, gambar berikut memperlihatkan proses lengkap ULBP dimulai dari pembagian blok citra,

pembentukan pola biner, klasifikasi pola uniform/non-uniform, hingga pemetaan nilai ke histogram 59 bin.



Gambar 2.8 Ilustrasi proses ULBP dari pembagian blok hingga histogram (21).

Gambar tersebut memperlihatkan bagaimana nilai intensitas piksel dalam setiap blok diubah menjadi kode biner, kemudian dianalisis transisinya untuk menentukan apakah pola tersebut masuk kategori uniform atau non-uniform. Pola non-uniform ditempatkan pada bin terakhir (bin ke-59), sedangkan pola uniform dipetakan pada salah satu dari 58 bin pertama. Proses ini menghasilkan histogram ULBP yang digunakan sebagai vektor fitur untuk tahap klasifikasi.

2.5 Support Vector Machine (SVM)

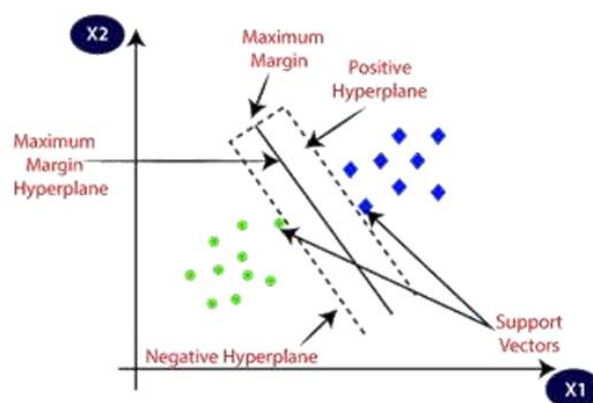
2.5.1 Pengertian Support Vector Machine (SVM)

Support Vector Machine (SVM) merupakan salah satu algoritma dalam bidang machine learning yang digunakan untuk menyelesaikan permasalahan klasifikasi dan prediksi. Prinsip utama dari SVM adalah mencari hyperplane terbaik yang berfungsi sebagai batas pemisah antara dua kelas data dalam ruang fitur. Hyperplane ini ditentukan sedemikian rupa agar memiliki jarak maksimum terhadap titik data dari masing-masing kelas, di mana titik-titik terdekat disebut sebagai support vectors. Titik-titik ini memiliki peran penting karena menjadi penentu posisi hyperplane yang optimal dalam proses pemisahan data (22).

SVM bekerja dengan memetakan data ke dalam ruang berdimensi tinggi menggunakan fungsi kernel, sehingga data yang awalnya tidak dapat dipisahkan secara linear di ruang asli menjadi dapat dipisahkan dengan garis atau bidang pemisah di ruang fitur tersebut. Melalui pendekatan ini, SVM dapat menangani berbagai jenis data baik linear maupun non-linear dengan tingkat akurasi yang tinggi. Keunggulan lain dari SVM adalah kemampuannya dalam menghindari overfitting karena fokus pada margin maksimum antara kelas data (23).

2.5.2 Konsep Dasar Support Vector Machine (SVM)

Konsep dasar dari metode Support Vector Machine (SVM) berpusat pada pencarian bidang pemisah (*hyperplane*) yang mampu memaksimalkan jarak antar kelas atau *margin* dalam ruang fitur. Tujuan utama algoritma ini adalah menemukan *hyperplane* dengan posisi optimal sehingga dapat meminimalkan tingkat kesalahan dalam proses klasifikasi. Titik-titik data yang berada paling dekat dengan bidang pemisah disebut *support vectors*, karena kedudukannya berpengaruh langsung terhadap pembentukan batas keputusan optimal (23).



Gambar 2.9 Ilustrasi konsep dasar Support Vector Machine (SVM) (24)

Gambar tersebut memperlihatkan bagaimana SVM mencari hyperplane optimal yang memisahkan dua kelas data. Garis putus-putus menunjukkan batas margin, sedangkan titik-titik yang berada paling dekat dengan margin disebut sebagai support vectors. Jarak antar margin inilah yang dimaksimalkan oleh SVM untuk menghasilkan pemisahan kelas yang paling optimal. Dengan cara ini, SVM dapat menemukan keputusan klasifikasi yang stabil dan memiliki generalisasi yang baik.

Untuk data yang tidak dapat dipisahkan secara linier, SVM memanfaatkan fungsi kernel guna memproyeksikan data ke dalam ruang berdimensi lebih tinggi sehingga memungkinkan pemisahan yang lebih optimal. Beberapa jenis fungsi kernel yang umum digunakan meliputi linear kernel, polynomial kernel, dan radial basis function (RBF). Transformasi ini membantu sistem dalam menemukan batas keputusan yang ideal terhadap data yang kompleks.

Secara matematis, keputusan SVM dapat dinyatakan melalui persamaan berikut:

$$f(x) = \text{sign}(w \cdot x + b) \quad (2.6)$$

dengan w sebagai vektor bobot, x sebagai vektor fitur, dan b sebagai bias. SVM menentukan bidang pemisah optimal dengan meminimalkan fungsi objektif:

$$\text{minimize } \left(\frac{1}{2} \| W \|^2 \right) \quad (2.7)$$

dengan batasan:

$$y_i(w \cdot x_i + b) \geq 1 \quad (2.8)$$

di mana y_i merepresentasikan label kelas untuk data ke- i . Melalui proses optimasi ini, algoritma SVM memperoleh *hyperplane* dengan margin maksimum yang dapat memisahkan data secara efisien. Keunggulan tersebut membuat SVM menjadi salah satu algoritma yang efektif dalam pengenalan pola dan klasifikasi citra berbasis

tekstur, termasuk pada penelitian yang mengombinasikan metode ekstraksi fitur seperti Local Binary Pattern (LBP) (22).

2.5.3 Langkah Kerja Support Vector Machine (SVM)

Proses kerja algoritma Support Vector Machine (SVM) dalam klasifikasi citra terdiri dari beberapa tahapan utama yang saling berkaitan. Tahapan pertama adalah persiapan data, yaitu mengumpulkan dataset citra yang akan digunakan dalam proses klasifikasi. Data tersebut dibagi menjadi dua bagian, yaitu data training dan data testing, untuk memastikan bahwa model mampu belajar dari pola data latih dan kemudian diuji pada data baru yang belum dikenali. Setelah itu dilakukan ekstraksi fitur untuk mengambil informasi penting dari citra, seperti warna, tekstur, atau bentuk, yang akan digunakan sebagai masukan (input feature vector) bagi SVM (25).

Tahapan kedua adalah pelatihan model (training). Pada tahap ini, algoritma SVM mencari hyperplane terbaik yang dapat memisahkan data ke dalam kelas-kelas tertentu dengan margin maksimum. Support vectors yaitu titik-titik data yang berada paling dekat dengan hyperplane berperan penting dalam menentukan posisi bidang pemisah tersebut. Setelah proses pelatihan selesai, dilakukan pengujian model (testing) dengan data uji untuk mengukur kemampuan model dalam mengenali pola baru.

Selanjutnya dilakukan pemilihan fungsi kernel untuk menyesuaikan karakteristik data. Jenis kernel yang umum digunakan antara lain kernel linear, polynomial, dan Radial Basis Function (RBF), di mana masing-masing memiliki kemampuan berbeda dalam menangani data linier maupun nonlinier. Kernel RBF secara khusus banyak digunakan karena mampu memetakan data non-linear ke dalam ruang berdimensi lebih tinggi sehingga menghasilkan pemisahan kelas yang lebih optimal. Dalam praktiknya, pemilihan

kernel yang tepat sangat berpengaruh terhadap kinerja klasifikasi SVM (26).

Selain pemilihan kernel, kinerja Support Vector Machine (SVM) juga sangat dipengaruhi oleh nilai hyperparameter yang digunakan. Oleh karena itu, diperlukan proses *hyperparameter tuning* untuk memperoleh kombinasi parameter terbaik, seperti parameter C (regularisasi), gamma (parameter pada kernel Radial Basis Function), dan degree (derajat pada kernel polynomial). Proses optimasi hyperparameter bertujuan untuk meningkatkan akurasi dan stabilitas model klasifikasi. Beberapa penelitian menunjukkan bahwa penerapan tuning parameter pada SVM mampu meningkatkan nilai akurasi, precision, recall, dan F1-score secara signifikan (27).

Tahap terakhir adalah evaluasi performa model, yang umumnya dilakukan menggunakan metrik akurasi, presisi, dan F1-score. Evaluasi juga dapat divisualisasikan melalui confusion matrix untuk melihat distribusi prediksi model terhadap setiap kelas yang diuji (26).

Secara garis besar, alur kerja SVM dalam klasifikasi citra dapat dijelaskan sebagai berikut:

1. Input Data: Dataset citra diambil dan dipersiapkan.
2. Preprocessing : Dilakukan peningkatan kualitas citra, konversi ke grayscale, dan segmentasi objek.
3. Ekstraksi Fitur : Fitur tekstur atau pola dari citra diambil menggunakan metode tertentu seperti LBP.
4. Training SVM : Model dilatih menggunakan data fitur untuk menentukan hyperplane pemisah optimal.
5. Testing & Evaluasi : Model diuji menggunakan data uji, kemudian hasilnya dievaluasi menggunakan confusion matrix dan metrik performa lainnya.

Dengan alur kerja tersebut, SVM mampu menghasilkan proses klasifikasi yang efisien, stabil, serta memiliki akurasi tinggi pada berbagai jenis citra, termasuk citra tulisan tangan yang memiliki pola

tekstur kompleks.

2.6 Evaluasi Kinerja Sistem

Evaluasi kinerja sistem dilakukan untuk menilai kemampuan model klasifikasi dalam mengenali data secara tepat. Salah satu metode yang umum digunakan dalam analisis performa adalah confusion matrix, yang berfungsi untuk membandingkan antara hasil prediksi sistem dengan data aktual. Melalui confusion matrix, dapat diketahui jumlah data yang diklasifikasikan dengan benar maupun yang salah, sehingga tingkat keberhasilan sistem dapat diukur secara kuantitatif dan objektif.

Confusion matrix terdiri atas empat komponen utama, yaitu True Positive (TP), True Negative (TN), False Positive (FP), dan False Negative (FN). Keempat komponen tersebut menjadi dasar dalam menghitung ukuran kinerja seperti akurasi, presisi, recall, dan F1-Score.

- Akurasi menggambarkan sejauh mana model dapat memprediksi dengan benar terhadap seluruh data uji.
- Presisi mengukur tingkat ketepatan prediksi positif yang dihasilkan oleh sistem.
- Recall menunjukkan kemampuan sistem dalam mengenali seluruh data positif yang sebenarnya ada.
- F1-Score merupakan rata-rata harmonis antara presisi dan recall, yang memberikan keseimbangan antara keduanya, terutama saat terdapat ketidakseimbangan jumlah data antar kelas.

Rumus perhitungan untuk masing-masing metrik tersebut dapat dituliskan sebagai berikut:

$$\text{Akurasi} = \frac{TP + TN}{TP + TN + FP + FN} \times 100 \% \quad (2.9)$$

$$\text{Presisi} = \frac{TP}{TP + FP} \quad (2.10)$$

$$\text{Recall} = \frac{TP}{TP + FN} \quad (2.11)$$

$$F1 - \text{Score} = 2 \times \frac{(\text{Presisi} \times \text{Recall})}{(\text{Presisi} + \text{Recall})} \quad (2.12)$$

Hasil evaluasi dari confusion matrix memberikan gambaran menyeluruh mengenai efektivitas sistem dalam melakukan klasifikasi. Dalam konteks penelitian ini, metrik tersebut digunakan untuk menilai kinerja metode Local Binary Pattern (LBP) dan Support Vector Machine (SVM) dalam mendeteksi keteraturan tulisan tangan. Pengujian ini bertujuan memastikan bahwa sistem yang dikembangkan mampu memberikan hasil klasifikasi yang akurat, konsisten, serta dapat diandalkan dalam penerapan nyata (14).

2.7 Literatur Review

Pada bagian ini membahas Literatur Review Penelitian Terdahulu yang Berkaitan dengan Deteksi Keteraturan Tulisan Tangan Menggunakan Metode Local Binary Pattern (LBP) dan Support Vector Machine (SVM)

1. Referensi: Sari, C. A., Sari, W. S., Shelomita, V. A., Kusuma, M. R., Puspa, S. A., & Gusta, M. B. (2023). *The Involvement of Local Binary Pattern to Improve the Accuracy of Multi Support Vector-Based Javanese Handwriting Character Recognition*. *Journal of Applied Intelligent System*, 8(2), 206–215. <https://doi.org/10.33633/jais.v8i2>

Masalah yang diangkat: Minat terhadap pembelajaran aksara Jawa menurun dan sistem pengenalan tulisan tangan aksara Jawa masih kurang akurat karena kemiripan bentuk huruf.

Hasil penelitian: Untuk Penerapan metode Local Binary Pattern (LBP) dan Multi Support Vector Machine (Multi SVM) meningkatkan akurasi pengenalan tulisan tangan aksara Jawa hingga 90%, dibandingkan 80% tanpa LBP.

2. Referensi: Katili, I. F., Soeleman, M. A., & Premunendar, R. A. (2023). *Character Recognition of Handwriting of Javanese Character Image using Information Gain Based on the Comparison of Classification Method*. *Jurnal RESTI*, 7(1), 193–200.

<https://doi.org/10.29207/resti.v7i1.4488>

Masalah yang diangkat: Pelestarian aksara Jawa terancam karena pengaruh globalisasi, sementara sistem pengenalan tulisan tangan aksara Jawa masih kurang akurat akibat kemiripan bentuk huruf dan banyaknya fitur yang dihasilkan.

Hasil Penelitian: Penelitian menggabungkan metode Local Binary Pattern (LBP) dan Information Gain (IG) untuk meningkatkan akurasi klasifikasi tulisan tangan aksara Jawa. Hasil terbaik diperoleh dengan algoritma SVM linear ukuran sel 64×64 , dengan akurasi 87,86%, meningkat dari 82,5% pada penelitian sebelumnya.

3. Referensi: Gumilang, A., & Agustin, S. (2024). *Deteksi Kepribadian Melalui Margin Pada Tulisan Tangan Menggunakan Random Forest*. Jurnal Inovtek Polbeng – Seri Informatika, 9(1), 311–323. <https://doi.org/10.35314/isi.v9i1.4176>

Masalah yang diangkat: Tes kepribadian melalui tulisan tangan masih dilakukan secara manual dan subjektif, sehingga hasilnya kurang akurat serta memerlukan waktu yang lama.

Hasil penelitian: Penelitian ini menggunakan metode Random Forest untuk mendeteksi kepribadian berdasarkan margin tulisan tangan. Hasil pengujian menunjukkan akurasi sebesar 95%, presisi 95,13%, dan spesifisitas 98,34%, yang membuktikan bahwa metode ini efektif dan lebih akurat dibandingkan cara manual.

4. Referensi: Fatimah, N. S., & Agustin, S. (2025). *Klasifikasi Citra Batik Menggunakan Local Binary Pattern (LBP) dan Support Vector Machine (SVM)*. Jurnal Algoritma, 22(1), 185–196. <https://doi.org/10.33364/algoritma/v.22-1.2208>

Masalah yang diangkat: Proses pengenalan dan klasifikasi motif batik masih sulit dilakukan secara manual karena kompleksitas pola dan variasi tekstur.

Hasil penelitian: Penelitian Metode LBP dan SVM digunakan untuk klasifikasi tiga motif batik (Kawung, Megamendung, Parang) dengan

akurasi 88,89%. Kombinasi keduanya efektif mengenali pola dan tekstur khas tiap motif.

5. Referensi: Baso, B., & Risald. (2025). *Preservation of Timorese Weaving Patterns Through Pattern Recognition Using a Combination of GLCM, LBP, and SVM Multiclass*. *Jurnal Teknologi Informasi dan Ilmu Komputer (JITK)*, 11(1), 36–42.

<https://doi.org/10.33480/jitk.v11i1.6171>

Masalah yang diangkat: Pelestarian pola tenun tradisional Timor terkendala oleh menurunnya minat generasi muda dan kurangnya dokumentasi digital, serta kompleksitas pola yang sulit dikenali secara manual.

Hasil penelitian: Kombinasi GLCM dan LBP untuk ekstraksi fitur serta SVM multiclass menghasilkan akurasi terbaik 88,66% dengan kernel linear (OVO). Metode ini efektif mengenali pola tekstur global dan lokal pada motif tenun Timor.

6. Referensi: Li, C., Zhang, K., Lin, Q., Huang, S., Cheng, W., Lei, Y., Zhao, X., & Zhao, J. (2025). *Major depressive disorder recognition based on electronic handwriting recorded in psychological tasks*. *BMC Medicine*, 23(282). <https://doi.org/10.1186/s12916-025-04101-2>

Masalah yang diangkat: Diagnosis gangguan depresi mayor (MDD) masih sulit karena belum ada kriteria objektif yang mudah diterapkan secara klinis. Penelitian ini mencari alternatif melalui analisis pola tulisan tangan elektronik sebagai indikator perilaku penderita depresi.

Hasil Penelitian: Dengan data tulisan tangan dari 130 pasien MDD dan 117 individu sehat, model LightGBM mencapai akurasi 82% dan AUC 0,90. Penderita MDD menunjukkan tekanan pena lebih tinggi, kecepatan lebih cepat, dan variasi sudut pena lebih besar.

7. Referensi: Berbar, M. A. (2022). *Features extraction using encoded local binary pattern for detection and grading diabetic retinopathy*. *Health Information Science and Systems*, 10(14).

<https://doi.org/10.1007/s13755-022-00181-z>

Masalah yang diangkat: Diagnosis manual *diabetic retinopathy* sering tidak akurat karena perbedaan pencahayaan dan kualitas citra retina, sementara metode otomatis sebelumnya bergantung pada segmentasi lesi yang kompleks.

Hasil Penelitian: Latihan Penelitian mengusulkan metode Uniform Local Binary Pattern Encoded Zeros (ULBPEZ) untuk ekstraksi fitur tanpa segmentasi. Uji pada dataset Messidor-2 dan EyePACS menunjukkan akurasi hingga 98,84%, membuktikan metode ini efektif untuk deteksi dan penentuan tingkat keparahan *diabetic retinopathy* secara otomatis dan efisien.

8. Referensi: AlFayez, F., & Bouhamed, H. (2023). *Machine learning and uLBP histograms for posture recognition of dependent people via Big Data Hadoop and Spark platform. International Journal of Computers Communications & Control*, 18(1), 4981. <https://doi.org/10.15837/ijccc.2023.1.4981>

Masalah yang diangkat: Penelitian ini menyoroti pentingnya sistem pemantauan otomatis untuk mendeteksi postur manusia pada populasi lansia atau penyandang disabilitas guna mencegah kecelakaan jatuh di rumah. Tantangan utama adalah menjaga privasi pengguna serta mengurangi kompleksitas perhitungan dari metode deep learning yang umum digunakan seperti CNN dan DFFNN.

Hasil penelitian: Peneliti mengusulkan pendekatan berbasis *Uniform Local Binary Pattern* (uLBP) untuk mengekstraksi fitur tekstur citra dan menggabungkannya dengan algoritma *Random Forest*. Pendekatan ini terbukti meningkatkan efisiensi dan akurasi pengenalan postur hingga 98,4% pada dataset uji serta 92,6% pada dataset validasi, sambil tetap menjaga privasi karena hanya menyimpan histogram uLBP, bukan seluruh citra.

9. Referensi: Oktavia, V., & Wijaya, N. (2022). *Pengenalan Tulisan Tangan Huruf Latin Bersambung Menggunakan Local Binary Pattern dan K-*

Nearest Neighbor: JISKA (Jurnal Informatika Sunan Kalijaga), 7(3), 211–225 <https://doi.org/10.14421/jiska.2022.7.3>

Masalah yang diangkat: Pengenalan tulisan tangan huruf Latin bersambung sulit dilakukan karena setiap penulis memiliki variasi bentuk huruf yang berbeda.

Hasil penelitian: Metode K-Nearest Neighbor (KNN) dengan ekstraksi fitur Local Binary Pattern (LBP) digunakan untuk mengenali huruf Latin bersambung. Hasil terbaik diperoleh pada huruf kapital dengan akurasi 29,49% (k=7) dan huruf non-kapital 26,28% (k=3).

10. Referensi: Sulistianingsih, N., Astutik, F., & Rahman, A. (2024). *Optimasi Seleksi Fitur untuk Perbaikan Akurasi Support Vector Machine Classifier pada Klasifikasi Citra Tanaman Rimpang*. Jurnal Fasilkom, 14(2), 526–532 <https://doi.org/10.37859/jf.v14i2.7566>

Masalah yang diangkat: Jenis tanaman rimpang seperti kunyit, jahe, kencur, dan lengkuas sulit dibedakan karena memiliki bentuk dan tekstur yang mirip, sehingga diperlukan metode yang dapat mengidentifikasi citra tanaman dengan akurasi tinggi.

Hasil penelitian: Penelitian menerapkan metode Support Vector Machine (SVM) dengan optimasi Recursive Feature Elimination (RFE) dan ekstraksi fitur Local Binary Pattern (LBP) untuk klasifikasi citra tanaman rimpang. Hasil menunjukkan peningkatan akurasi dari 67% menjadi 81% setelah penerapan seleksi fitur SVM-RFE.

11. Referensi: Wahyu Ardiantito S., Surianto, S. J., Ramadhani, S., & Ananta, W. P. (2023). *Klasifikasi Tumor Otak Menggunakan Local Binary Pattern dan SVM Classifier*. Student Research Journal, 1(6), 182–190. <https://doi.org/10.55606/srjyappi.v1i6.823>

Masalah yang diangkat: Klasifikasi jenis tumor otak berbasis citra MRI masih membutuhkan metode otomatis yang akurat karena diagnosis manual rawan kesalahan.

Hasil penelitian: Metode LBP untuk ekstraksi fitur dan SVM untuk klasifikasi menghasilkan akurasi hingga 88%, sehingga efektif dalam membedakan jenis tumor otak.

12. Referensi: Lodong, A. T., Widodo, A. W., & Rahman, M. A. (2023). *Penentuan Mutu pada Citra Buah Jeruk Keprok Menggunakan Metode Local Binary Pattern (LBP)*. *Jurnal Pengembangan Teknologi Informasi dan Ilmu Komputer*, 7(4), 1616–1622. <https://j-ptiik.ub.ac.id/index.php/j-ptii/article/view/12550>

Masalah yang diangkat: Penilaian mutu buah jeruk keprok masih dilakukan secara manual sehingga kurang efisien, rawan ketidakkonsistenan, dan belum memenuhi kebutuhan otomasi untuk standarisasi mutu buah.

Hasil penelitian: Metode Local Binary Pattern (LBP) digunakan untuk mengekstraksi tekstur kulit jeruk pada citra, kemudian diklasifikasikan menjadi mutu baik atau buruk. Hasil terbaik mencapai akurasi 80% pada ukuran citra 100×100 piksel dengan radius R=1.

13. Referensi: Hayati, T. N., Fatimah, N. S., Fitria, L., & Agustin, S. (2024). *Klasifikasi Lahan Perkebunan Kelapa Sawit Pada Citra Foto Udara Menggunakan Metode Local Binary Pattern dan Klasifikasi SVM*. *SABER: Jurnal Teknik Informatika, Sains dan Ilmu Komunikasi*, 2(3), 138–146. <https://doi.org/10.59841/saber.v1i3.1399>

Masalah yang diangkat: Pada citra foto udara, lahan kelapa sawit dan hutan memiliki pola tekstur dan warna yang sangat mirip, sehingga sulit dibedakan secara akurat. Permasalahan ini membuat klasifikasi lahan menjadi menantang, terutama ketika terdapat bayangan atau gangguan visual lainnya.

Hasil penelitian: Metode Local Binary Pattern (LBP) digunakan untuk mengekstraksi tekstur citra, kemudian fitur tersebut diklasifikasikan menggunakan Support Vector Machine (SVM). Hasil pengujian menunjukkan akurasi 83,33%, membuktikan bahwa kombinasi LBP dan

SVM mampu mengidentifikasi lahan kelapa sawit dari citra udara dengan performa yang baik.

14. Referensi: Nasution, A. N., Lubis, A. S., & Kiswanto, D. (2024). *Identifikasi Variasi Paprika Berdasarkan Jenis Warna Paprika Berbasis Analisis Citra Digital Menggunakan Algoritma Support Vector Machine (SVM)*. Jurnal Informasi, Sains dan Teknologi, 7(2), 209–220. <https://doi.org/10.55606/isaintek.v7i2.282>

Masalah yang diangkat: Identifikasi variasi warna paprika masih dilakukan secara manual sehingga rentan kesalahan, tidak konsisten, dan kurang efisien pada proses klasifikasi dalam sektor pertanian.

Hasil penelitian: Metode ekstraksi fitur warna dan tekstur dikombinasikan dengan SVM berhasil mengklasifikasikan empat warna paprika dengan akurasi tertinggi 97,56% menggunakan kernel polynomial, menunjukkan bahwa SVM efektif dalam identifikasi objek berbasis citra digital.

15. Referensi: Patmawati. (2023). *Prediksi Penyakit Stroke Menggunakan Support Vector Machine (SVM)*. Bulletin of Network Engineer and Informatics, 1(2), 12–17. <https://doi.org/10.59688/bufnets.v1i1.5>

Masalah yang diangkat: Kasus stroke di Indonesia terus meningkat, namun proses prediksi penyakit ini masih sulit karena data medis bersifat kompleks, heterogen, dan memiliki banyak faktor risiko yang saling berkaitan. Metode prediksi sebelumnya juga masih menghasilkan akurasi yang belum optimal.

Hasil penelitian: Penelitian menerapkan kembali algoritma SVM untuk memprediksi risiko stroke menggunakan dataset publik dari Kaggle. Hasil evaluasi menunjukkan performa sangat tinggi dengan akurasi 100% dan nilai ROC-AUC 100%, sehingga membuktikan bahwa SVM mampu memberikan klasifikasi yang sangat baik untuk prediksi stroke meskipun perlu kajian lanjutan terkait kemungkinan ketidakseimbangan data.