

BAB III METODE PENELITIAN

3.1 Objek Penelitian

Objek penelitian yang digunakan berupa dataset *image* yang diperoleh dengan cara foto citra menggunakan kamera Cannon 3000d yang terdiri dari kumpulan gambar buah jeruk segar, kurang segar, dan busuk. Dataset yang disediakan berisi 435 gambar yang terdiri dari 348 data training dan 87 data testing. Objek penelitian kali ini adalah buah jeruk medan dengan tingkat kesegaran segar dan busuk. Dataset *image* ini akan digunakan sebagai data utama untuk mengklasifikasikan apakah suatu *image* menunjukkan buah jeruk segar, kurang segar, atau busuk.

3.2 Lokasi dan Waktu Penelitian

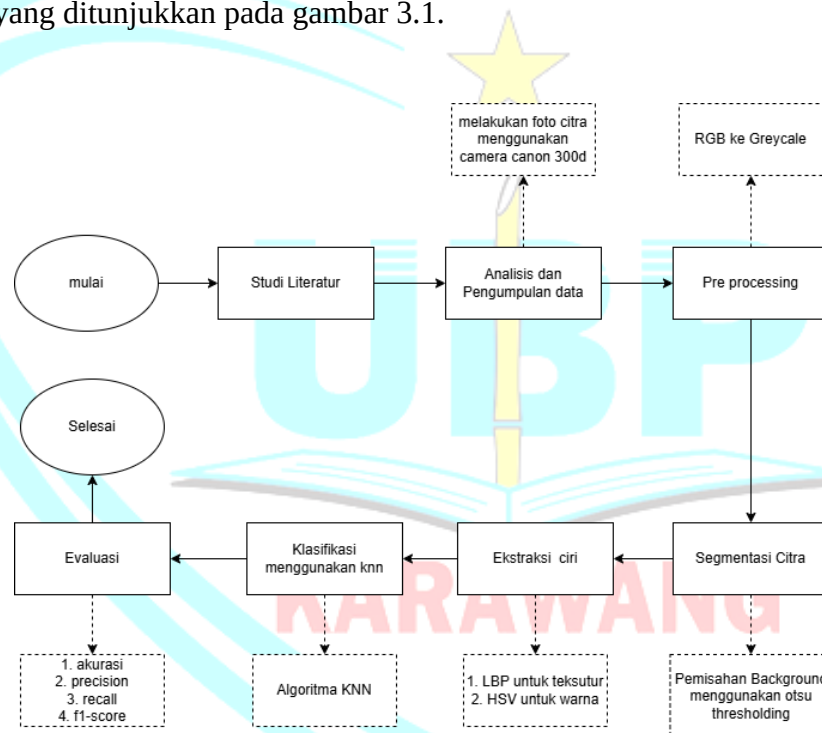
Penelitian ini dilaksanakan di Laboratorium Riset Universitas Buana Perjuangan Karawang. Adapun waktu pelaksanaannya dimulai pada bulan November hingga penelitian selesai.

Tabel 3. 1 Waktu Penelitian

No	Kegiatan	Bulan 1				Bulan 2				Bulan 3			
		1	2	3	4	1	2	3	4	1	2	3	4
1	Studi Literatur	■	■	■									
2	Analisis dan Pengumpulan data		■	■	■	■							
3	Pre processing			■	■	■	■						
4	Segmentasi citra				■	■	■	■					
5	Ekstraksi ciri					■	■	■	■				
6	Klasifikasi menggunakan KNN						■	■	■	■			
7	Hasil pengujian							■	■	■	■		
8	Evaluasi								■	■	■	■	
9	Membuat laporan									■	■	■	■

3.3 Prosedur Penelitian

Untuk mencapai hasil penelitian yang optimal, diperlukan pendekatan yang terencana dengan matang serta terstruktur dengan baik. Proses penelitian ini mencakup serangkaian langkah-langkah yang dilakukan secara sistematis untuk memastikan bahwa setiap tahap dapat terlaksana dengan efektif dan mendukung satu sama lain. Sebagai gambaran yang lebih jelas mengenai alur penelitian ini, prosedur tersebut disajikan dalam bentuk diagram alur (*flowchart*) yang menggambarkan proses dari setiap tahapan secara menyeluruh dan terperinci. Seperti yang ditunjukkan pada gambar 3.1.



Gambar 3. 1 *Flowchart* Prosedur Penelitian

3.4 Studi Literatur

Pada tahapan ini melakukan *review* pengumpulan data dengan mencari referensi data dari buku, jurnal paper yang berkaitan dengan pokok bahasan penelitian klasifikasi menggunakan algoritma *K-Nearest Neighbor* untuk klasifikasi kesegaran buah jeruk berdasarkan warna dan tekstur. Tahap selanjutnya yaitu penulis akan melakukan pengumpulan dan analisis data.

3.5 Analisis Dan Pengumpulan data

Proses pengambilan data pada penelitian ini dilakukan dengan cara mengambil gambar citra jeruk segar, kurang segar, dan busuk menggunakan kamera Cannon 3000d. Contoh data yang diambil seperti gambar dibawah ini.



Gambar 3. 2 Jeruk Medan Segar



Gambar 3. 3 Jeruk Medan Kurang Segar



Gambar 3. 4 Jeruk Medan Busuk

3.6 Pre processing

Proses pada tahapan *preprocessing* dilakukan dengan cara pengambilan sample pada citra buah jeruk. Selanjutnya akan dilakukan perubahan warna dari citra RGB menjadi grayscale.

Tahapan *preprocessing* diawali dengan mengambil gambar input sebagai data awal. Selanjutnya, gambar tersebut diubah menjadi skala abu-abu dengan memproses setiap pikselnya. Untuk setiap piksel, nilai rata-rata dari komponen warna merah (R), hijau (G), dan biru (B) dihitung, kemudian nilai piksel asli diganti dengan nilai rata-rata tersebut. Setelah semua piksel diproses, gambar input kini sepenuhnya berada dalam format skala abu-abu, yang kemudian dapat disimpan atau ditampilkan.

3.7 Segmentasi Citra

Segmentasi citra dilakukan menggunakan citra dengan *thresholding otsu*. Hal ini untuk memisahkan antara obyek pola dengan *background*. Proses dimulai dengan mempersiapkan gambar input sebagai data awal. Gambar ini kemudian dikonversi menjadi skala abu-abu dengan menghitung nilai rata-rata komponen warna (R, G, B) untuk setiap piksel dan mengganti nilai piksel asli dengan rata-rata tersebut. Setelah konversi, nilai *threshold* optimal dihitung menggunakan metode Otsu, yang menganalisis histogram intensitas piksel dan menentukan ambang batas yang memaksimalkan variansi antar kelompok piksel (objek dan latar belakang). Selanjutnya, *threshold* ini diterapkan pada setiap piksel gambar skala abu-abu: piksel yang nilainya lebih besar atau sama dengan *threshold* diubah menjadi putih, sedangkan yang lebih kecil diubah menjadi hitam. Setelah semua piksel diproses, gambar kini tersegmentasi, secara efektif memisahkan objek pola dari latar belakang. Gambar hasil segmentasi ini kemudian dapat disimpan atau ditampilkan.

3.8 Ekstraksi ciri Warna dan Tekstur

Ekstraksi ciri pada penelitian kali ini menggunakan metode HSV untuk warna dan LBP untuk tekstur untuk mengklasifikasikan kesegaran buah jeruk

3.8.1 HSV (Hue, Saturation, Value)

Fungsi ekstraksi ciri digunakan untuk mengonversi gambar dari ruang warna BGR (Blue, Green, Red) ke ruang warna HSV (Hue Saturation Value) dan kemudian mengekstraksi nilai rata-rata dari setiap channel (*H, S, V*). fitur ini akan digunakan sebagai input untuk model klasifikasi.

Proses ini dimulai dengan menyiapkan gambar input. Sebelum dilakukan konversi warna, gambar terlebih dahulu dikonversi ke grayscale, lalu dilakukan proses segmentasi menggunakan metode *thresholding Otsu* untuk menghasilkan mask biner yang memisahkan area objek (buah jeruk) dari latar belakang. Mask ini berfungsi sebagai filter sehingga hanya piksel-piksel yang merepresentasikan objek saja yang akan diproses lebih lanjut. Setelah pemisahan objek dari background, gambar asli dikonversi dari ruang warna RGB ke HSV (Hue, Saturation, Value) menggunakan fungsi konversi yang sesuai. Setelah konversi, rata-rata nilai dari masing-masing komponen HSV diekstrak secara terpisah:

pertama, komponen Hue (H) diambil dan dihitung rata-ratanya; kedua, komponen Saturation (S) diambil dan dihitung rata-ratanya; dan ketiga, komponen Value (V) diambil dan dihitung rata-ratanya. Perhitungan rata-rata ini hanya dilakukan pada area objek (berdasarkan mask hasil segmentasi). Terakhir, ketiga nilai rata-rata ini (dari Hue, Saturation, dan Value) digabungkan menjadi satu array fitur tunggal, yang siap untuk disimpan atau digunakan dalam tahap analisis selanjutnya.

3.8.2 LBP (Local Binary Pattern)

Proses ekstraksi fitur menggunakan *Local Binary Pattern* (LBP) dimulai dengan mempersiapkan gambar input buah jeruk. Penting untuk memastikan gambar sudah dalam format skala abu-abu; jika masih dalam RGB/BGR, konversi harus dilakukan terlebih dahulu. Selanjutnya, parameter LBP ditentukan, termasuk jumlah piksel tetangga (misalnya 8) dan radius (misalnya 1 piksel) untuk jarak tetangga dari piksel pusat. Setelah itu, nilai LBP dihitung untuk setiap piksel dalam gambar: nilai intensitas piksel pusat diambil sebagai acuan, kemudian dibandingkan dengan delapan piksel tetangga di sekitarnya. Piksel tetangga yang nilainya lebih besar atau sama dengan piksel pusat diberi nilai 1, sedangkan yang lebih kecil diberi nilai 0. Nilai biner yang dihasilkan dari perbandingan ini disusun, diubah menjadi nilai desimal, dan menggantikan piksel pusat dengan nilai LBP yang baru dihitung. Proses ini berlanjut dengan membentuk histogram fitur LBP yang merepresentasikan distribusi pola tekstur di seluruh gambar, dan histogram ini kemudian dinormalisasi untuk keseragaman. Terakhir, fitur LBP yang telah dinormalisasi digabungkan ke dalam satu array fitur. Array ini berisi informasi tentang pola tekstur kulit jeruk dan akan menjadi input penting untuk model klasifikasi, seperti *K-Nearest Neighbors* (KNN), guna membedakan antara kondisi jeruk segar, kurang segar, dan busuk.

3.9 Klasifikasi menggunakan KNN

Teknik ini digunakan untuk menemukan data pelatihan yang paling dekat dengan data uji dengan menghitung jaraknya. Nilai K menunjukkan jumlah tetangga terdekat yang akan dipertimbangkan dalam proses evaluasi terhadap data yang sedang dianalisis. Pseudocode klasifikasi menggunakan KNN:

1. Mulai
2. Siapkan data:

- 1) *Data training*:
Berisi informasi fitur jeruk (warna dan tekstur) dan label kelas (segar, kurang segar, dan busuk).
- 2) *Data testing*:
Berisi informasi fitur jeruk yang ingin diprediksi kelasnya.
3. Tentukan nilai K yang akan diuji:
Misalnya: $K = 1, 3, 5, 7$, dst.
4. Tentukan jumlah fold untuk *cross-validation*:
Misalnya: Gunakan *5-fold cross-validation*.
5. Cari nilai K terbaik menggunakan *cross-validation*:
 - 1) Bagi data training menjadi 5 bagian.
 - 2) Untuk setiap nilai K:
 - a. Uji KNN dengan menggunakan 4 bagian data sebagai data latih dan 1 bagian data sebagai data uji.
 - b. Untuk setiap data uji:
 1. Hitung jarak ke semua data latih menggunakan rumus *Euclidean*
 2. Urutkan jarak dari yang terkecil ke terbesar.
 3. Pilih K tetangga terdekat.
 4. Prediksi kelas dengan mengambil mayoritas kelas dari tetangga terdekat (segar, kurang segar, dan busuk).
 - 3) Hitung akurasi nilai rata-rata dari prediksi untuk nilai K ini.
 - 4) Simpan nilai K dengan akurasi tertinggi sebagai K terbaik.
6. Klasifikasikan data testing menggunakan K terbaik:
 - 1) Untuk setiap data dalam data testing:
 - a. Hitung jarak ke semua data dalam *data training* menggunakan rumus *Euclidean*.
 - b. Urutkan jarak dari yang terkecil ke terbesar.
 - c. Pilih K tetangga terdekat.
 - d. Prediksi kelas data uji dengan mengambil mayoritas kelas tetangga terdekat (segar, kurang segar, dan busuk).
 - 2) Simpan hasil prediksi.
7. Selesai

3.10 Evaluasi

Evaluasi dilakukan menggunakan *confusion matrix* untuk menghitung metrik seperti akurasi, *precision*, *recall*, dan *F1-score*, guna mengukur performa model dalam mengklasifikasikan data uji. Untuk menghitung kinerja model, terdapat empat perhitungan utama, yaitu

- a. Akurasi: Mengukur seberapa sering model melakukan prediksi yang benar, dihitung dengan membagi jumlah prediksi yang benar dengan total data.
- b. *Precision*: mengukur proporsi prediksi positif yang benar (*True Positives*) dari semua prediksi positif yang dilakukan model.
- c. *Recall*: Mengukur kemampuan model untuk menemukan semua data positif yang sebenarnya (*True Positives*) di dalam dataset.
- d. *F1-score*: Merupakan rata-rata harmonis dari nilai *precision* dan *recall*. Metrik ini berguna untuk memberikan keseimbangan antara keduanya, khususnya saat menangani dataset yang tidak seimbang.

