

BAB III METODOLOGI PENELITIAN

1.1 Objek Penelitian

Objek dalam penelitian ini adalah untuk mendeteksi penggunaan pemutih pada beras menggunakan algoritma *Support Vector Machine* (SVM), serta mengevaluasi tingkat akurasi model dalam proses deteksi pemutih pada beras. Penelitian dilakukan dengan dataset berbentuk citra beras yang diambil secara pribadi untuk mendeteksi penggunaan pemutih pada beras. Dataset terdiri dari dua kelas yaitu beras normal dan beras yang mengandung pemutih. Dataset dikumpulkan melalui proses pengambilan sampel secara pribadi. Dalam dataset ditemukan ukuran citra yang berbeda-beda sehingga memerlukan tahapan pra-proses data untuk membantu penelitian. Dataset yang digunakan memiliki dua sebaran dataset yaitu sebaran data beras normal dan beras yang mengandung pemutih.

1.2 Lokasi dan Waktu Penelitian

Penelitian dilakukan mulai dari bulan Januari 2025 sampai bulan April 2025. Tempat penelitian dilakukan di Laboratorium Riset Universitas Buana Perjuangan Karawang. Adapun rincian tahapan penelitian dapat dilihat pada Tabel 3.1 dibawah.

Tabel 3. 1 Waktu Penelitian

No	Kegiatan	NOV				DES				JAN				FEB				MAR				APR			
		1	2	3	4	1	2	3	4	1	2	3	4	1	2	3	4	1	2	3	4	1	2	3	4
1	Studi Literatur																								
2	Pengumpulan Dataset																								
3	<i>Preparation</i> Dataset																								
4	Perancangan Model																								
5	Pengujian																								

1.3 Alat Penelitian

Dibutuhkan sistem perangkat keras berupa:

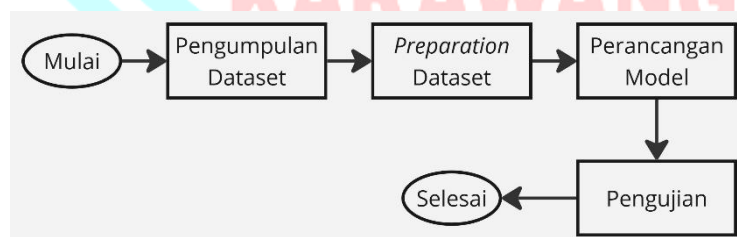
1. Satu unit laptop Asus TUF Gaming A15 dengan spesifikasi prosesor AMD Ryzen 5 7535HS with Radeon Graphics dengan base clock 3.3 GHz Up to NVIDIA GeForce RTX™, RAM 16 GB, dan SSD 512 GB dengan sistem operasi Windows 11 Home Versi 32H2.
2. Iphone 11 Pro dengan spesifikasi resolusi kamera 12 MP, sensor *triple camera*, *aperture f/1.8(wide)*, dan *autofocus dengan phase detection*.

Penelitian ini juga menggunakan beberapa perangkat lunak meliputi:

1. *browser* yang dipakai yaitu menggunakan *Google Chrome*,
2. Untuk pelatihan model berbasis *cloud* yaitu dengan menggunakan *Google Colab*.

1.4 Prosedur Penelitian

Penelitian dilakukan dengan tahapan yang terdiri dari studi literatur, pengumpulan dataset, dilanjutkan dengan pelatihan model svm untuk deteksi pemutih pada beras, dan terakhir melakukan pengujian tingkat akurasi model dengan menggunakan *confusion matrix*. Diagram alur penelitian ditunjukkan seperti pada Gambar 3.1.



Gambar 3. 1 Flowchart Prosedur Penelitian

1.4.1 Pengumpulan Dataset

Pengumpulan adalah langkah berikutnya setelah melakukan studi literatur. Data yang dikumpulkan adalah citra beras yang mengandung pemutih dan citra beras tanpa pemutih. Citra beras diperoleh dengan pengambilan gambar secara manual, menggunakan kamera Iphone 11 Pro yang memiliki spesifikasi resolusi 12 MP dan menghasilkan format gambar .HEIC dengan resolusi 4032x3024 piksel. Sebanyak 400 citra beras yaitu 200 untuk citra beras normal dan 200 untuk

citra beras yang mengandung pemutih akan diambil dan dibagi menjadi data latih sebesar 80% sedangkan data uji sebesar 20%.

Pengumpulan data penelitian dilakukan dengan menggunakan sampel beras Rojolele, yang merupakan jenis beras muncul yang dibudidayakan di wilayah Jawa Tengah dan Jawa Timur. Sampel beras yang digunakan merupakan beras yang dikonsumsi sehari-hari dengan usia simpan 1 bulan setelah panen, sampel beras dibagi menjadi dua kategori yaitu beras normal dan beras pemutih. Untuk beras pemutih, sampel beras Rojolele diproses dengan larutan klorin mengikuti protokol terukur, yakni perendaman dalam campuran 600 ml air dan 100 ml larutan klorin selama 10 menit, dilanjutkan dengan masa inkubasi semalaman pada suhu ruang. Setelah proses tersebut selesai, dilakukan pengambilan citra terhadap kedua kategori sampel untuk digunakan dalam analisis selanjutnya.

Contoh citra beras yang diambil secara manual dengan berbagai kondisi ditunjukkan pada Gambar 3.2.

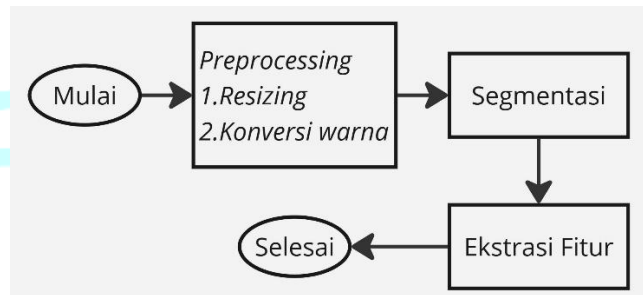


Gambar 3. 2 Contoh Citra Beras (a) alami (b) mengandung pemutih

Setelah proses pengumpulan dataset citra beras selesai, langkah selanjutnya adalah mengonversi format berkas citra dari HEIC ke JPG. Proses konversi ini dilakukan menggunakan *platform* konversi heic.online, yang memungkinkan perubahan format citra agar lebih mudah untuk digunakan dalam proses pengolahan data. Setelah seluruh citra berhasil dikonversi ke format JPG, kemudian melakukan pengorganisasian dataset dengan membuat folder terpisah berdasarkan kelas masing-masing citra sesuai dengan kategorinya. Dengan pengelompokan ini, dataset menjadi lebih terstruktur dan memudahkan proses analisis atau pelatihan model pengenalan citra pada tahap berikutnya.

1.4.2 Preparation Dataset

Setelah mendapatkan dataset gambar beras dan label yang ditulis padanya, langkah selanjutnya adalah melakukan proses penyiapan pada gambar beras. *Preparation* data dilakukan untuk meningkatkan kualitas data citra sehingga dapat memudahkan proses klasifikasi model SVM. Tahapan-tahapan *preparation* data ditunjukkan seperti pada Gambar 3.3.



Gambar 3. 3 Flowchart *Preparation* Dataset

Adapun penjelasan untuk flowchart pada Gambar 3.3 yaitu:

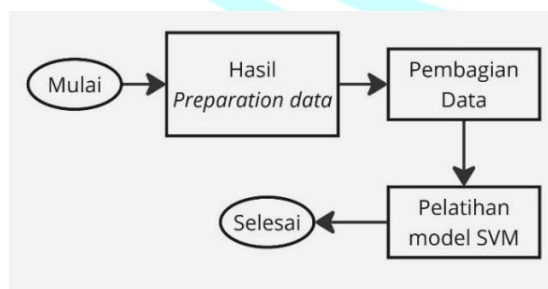
1. *Resizing* menjadi ukuran 256x256 piksel dilakukan untuk menyesuaikan ukuran citra ke dimensi standar. Memastikan bahwa semua citra memiliki ukuran yang konsisten, sehingga dapat diproses secara seragam oleh model atau algoritma. Proses *resizing* dilakukan menggunakan metode interpolasi, seperti *nearest neighbor* untuk citra kategori, *bilinear interpolation* untuk hasil yang lebih halus, atau *bicubic interpolation* untuk mempertahankan detail yang lebih baik. Setelah proses *resizing*, hasilnya harus diperiksa untuk memastikan tidak ada kehilangan informasi penting dalam citra. Dengan menggunakan ukuran 256x256, untuk memastikan bahwa citra yang digunakan dalam analisis tetap memiliki kualitas yang baik dan sesuai dengan kebutuhan penelitian.
2. Konversi warna, selanjutnya dilakukan konversi warna untuk mengubah citra berwarna (RGB) menjadi *grayscale*, yaitu citra hitam-putih dengan satu kanal intensitas. Transformasi ini bertujuan menyederhanakan data dengan mengurangi jumlah kanal warna dari tiga (merah, hijau, biru) menjadi satu, serta fokus pada intensitas piksel tanpa memperhatikan informasi warna. *Grayscale* dapat dilakukan dengan metode rata-rata intensitas piksel menggunakan rumus $(R+G+B)/3$ atau menggunakan bobot

tertentu berdasarkan sensitivitas mata terhadap warna dengan rumus $0.2989R + 0.5870G + 0.1140B$. Proses ini biasanya diotomatisasi menggunakan pustaka pemrosesan citra seperti *OpenCV*, *PIL*, atau *skimage*.

3. Segmentasi dilakukan untuk memisahkan objek utama dari latar belakangnya dengan menggunakan metode *Thresholding*. Proses ini membantu sistem untuk mengidentifikasi area-area penting dalam gambar sehingga memudahkan sistem untuk analisis pengenalan pola, deteksi objek, atau ekstraksi fitur.
4. Ekstraksi fitur adalah proses mengidentifikasi dan mengekstrak karakteristik penting dari objek yang telah melewati tahap segmentasi, dengan menganalisis nilai-nilai intensitas warna dari setiap gambar. Pada penelitian ini digunakan beberapa metode ekstraksi fitur, yaitu *Gray Level Co-occurrence Matrix* (GLCM) untuk menganalisis tekstur citra, *Histogram of Oriented Gradients* (HOG) untuk mendeteksi bentuk dan kontur, dan Histogram untuk menggambarkan distribusi intensitas warna. Hasil dari ekstraksi digunakan sebagai parameter untuk mengklasifikasi apakah beras mengandung pemutih atau tidak.

1.4.3 Perancangan Model SVM

Proses pemodelan SVM diawali dengan melakukan penginputan data citra dari dataset sampai didapatkannya hasil deteksi beras mengandung pemutih atau tidak, adapun alur pemodelan SVM ditunjukkan pada Gambar 3.4:



Gambar 3. 4 Flowchart Perancangan Model SVM

Adapun penjelasan dari flowchart diatas yaitu:

1. Setelah dilakukannya *preparation data*, langkah selanjutnya dataset akan dibagi menjadi dua subset yaitu data *training* dan data *testing*. Data *training*

digunakan untuk melatih model agar model dapat belajar mengenali pola dari data yang diberikan. Sedangkan data *testing* digunakan untuk mengevaluasi model dalam menggeneralisir terhadap data baru. Pembagian data akan dilakukan dengan proporsi umum yaitu 80% untuk data *training* dan 20% untuk data *testing*.

2. Pelatihan model SVM dilakukan dengan menggunakan dataset *training* yang telah disiapkan. Pelatihan model SVM dilakukan untuk menemukan *hyperplane* optimal yang memisahkan dua kelas (beras berpemutih dan beras alami) dengan *margin* maksimal. Model dilatih untuk mengenali hubungan antara fitur-fitur yang telah diekstraksi dari citra dengan label kelas. Proses pelatihan dimulai memilih linear kernel, yang merupakan pilihan yang tepat ketika data dapat dipisahkan secara *linear*. *Linear* kernel bekerja dengan cara mengubah data kedalam bentuk yang lebih sederhana, dimana pemisahan antara kelas dilakukan menggunakan garis lurus atau bidang data.

1.4.4 Pengujian Model SVM

Setelah model SVM dilatih, langkah berikutnya adalah menguji kinerja model dengan menggunakan *confusion matrix*. Dengan menunjukkan jumlah prediksi yang benar dan salah untuk setiap kelas, *confusion matrix* menunjukkan cara model mengklasifikasikan data. Matriks ini terdiri dari empat elemen yaitu *True Positives* (TP), *True Negatives* (TN), *False Positives* (FP), dan *False Negatives* (FN). *Confusion matrix* membantu dalam memahami kesalahan klasifikasi yang dilakukan oleh model. Selain itu, metrik juga digunakan untuk menghitung metrik evaluasi seperti *accuracy*, *precision*, dan *recall*. Presisi mengukur seberapa akurat prediksi model untuk kelas positif (yang mengandung pemutih), sedangkan *recall* mengukur seberapa efektif model dalam mengidentifikasi semua kasus positif. Evaluasi ini memberikan wawasan tentang kinerja model dalam mengklasifikasikan citra dengan tepat dan dapat membantu dalam penyempurnaan model lebih lanjut jika diperlukan. Tabel 3.2 menunjukkan tabel *confusion matrix*.

Tabel 3. 2 *Confusion Matrix*

<i>Predicted Class</i> <i>Actual Class</i>	<i>Positive</i>	<i>Negative</i>
<i>Positive</i>	<i>True Positive (TP)</i>	<i>False Negative (FN)</i>
<i>Negative</i>	<i>False Positive (FP)</i>	<i>True Negative (TN)</i>

Untuk menghitung *accuracy*, *precision*, dan *recall* didapatkan dari fungsi Persamaan 9, 10, dan 11:

$$Accuracy = \frac{TP + TN}{TP + TN + FP + FN} \times 100\% \quad (9)$$

$$Precision = \frac{TP}{FP + TP} \times 100\% \quad (10)$$

$$Recall = \frac{TP}{FN + TP} \times 100\% \quad (11)$$

