

Klasifikasi Biji Chia dan Biji Selasih Menggunakan Algoritma k-Nearest Neighbors Berdasarkan Fitur Warna HSV

Veilind Maynius¹, Dylan Revelin², Michael³, Frans Mikael Sinaga^{*4}

^{1,2,3,4}Universitas Mikroskil, Jl. Thamrin No. 112, 124, 140, Telp. (061) 4573767, Fax. (061) 4567789

^{1,2,3,4}Fakultas Informatika, Teknik Informatika, Universitas Mikroskil, Medan

e-mail: ¹221111085@students.mikroskil.ac.id, ²221110096@students.mikroskil.ac.id,

³221110659@students.mikroskil.ac.id, ⁴frans.sinaga@mikroskil.ac.id

Dikirim: 03-04-2025 | Diterima: 29-04-2025 | Diterbitkan: 30-04-2025

Abstrak

Biji chia dan biji selasih sering dianggap sebagai biji yang sama, padahal sebenarnya keduanya memiliki beberapa perbedaan. Salah satu perbedaan yang membedakannya adalah warna. Meskipun tampak serupa, biji selasih memiliki warna yang lebih gelap dibandingkan dengan biji chia. Oleh karena itu, kami memutuskan untuk menggunakan *Hue, Saturation, dan Value* (HSV) sebagai fitur model warna untuk mendapatkan informasi warna dari kedua biji ini. Selain itu, metode *Gray Level Co-occurrence Matrix* (GLCM) dan deteksi tepi juga digunakan untuk membantu meningkatkan akurasi dalam mengklasifikasikan kedua jenis biji ini. Kami menerapkan algoritma *k-Nearest Neighbor* (k-NN) untuk melatih model dan mengukur akurasi dari setiap nilai "k" yang diuji menggunakan *Stratified 5-Fold Cross-Validation*. Total data gambar yang digunakan untuk melatih model dalam penelitian ini adalah 100 gambar untuk masing-masing jenis biji. Ukuran gambar yang diuji diubah menjadi 200×200 piksel. Akurasi tertinggi yang diperoleh dalam penelitian ini mencapai 94,4% pada data uji dengan parameter "k" sebesar 3 dan jumlah data citra yang digunakan adalah 90. Hasil penelitian ini menunjukkan bahwa jumlah data gambar yang digunakan untuk melatih dan menguji model akan mempengaruhi tingkat akurasi yang diperoleh. Selain itu, pemilihan nilai parameter "k" juga berpengaruh terhadap seberapa akurat model dalam mendeteksi suatu gambar.

Kata kunci: Biji Chia, Biji Selasih, Klasifikasi, HSV, kNN

Abstract

*Chia seeds and basil seeds often being recognized as the same seed, indeed they have difference in some points. One of the differences of them are their color. Although looks like same, basil seeds are darker than chia seeds. Hence, we decided to use Hue, Saturation, and Value (HSV) as our color model feature to get the color information of these both seeds. Besides that, GLCM and edge detection also being involved to help classified these both seeds.. We implemented the k-Nearest Neighbor (k-NN) algorithm to train a model and measured the accuracy of each tested "k" value using Stratified 5-Fold Cross-Validation. The total data image that we used to train the model were 100 images for each seed. The image sized tested are resized into 200*200 pixels. The highest accuracy got in this research reached 94.4% for the data tested with the parameter "k" is 3 and the amount of the image data is 90. The result of this research showed that the amount of image data that used for training and testing will affect the accuracy that you will get, and the parameter of "k" value that been chosen will also affect in how accurate the model will detect an image.*

Keywords: Chia Seeds, Basil Seeds, Classification, HSV, kNN

1. PENDAHULUAN

Di antara contoh sumber pangan potensial yang menunjukkan peningkatan tren pasar yang cukup signifikan adalah biji *chia* (*chia seed*) dan biji selasih (*basil seeds*). Biji selasih memiliki morfologi kecil, berwarna hitam, bulat panjang. Biji *chia* dan selasih secara visual memiliki karakteristik fisik yang hampir sama, akan tetapi jika dilihat dengan seksama maka dapat dilihat perbedaan satu sama lainnya. Biji Selasih memiliki warna hitam dan berbentuk bulat, sedangkan biji *chia* berwarna coklat-putih dengan bentuk oval [1].

Kedua biji ini memiliki manfaatnya tersendiri. Menurut [2], *chia seed* mengandung lebih banyak protein daripada biji-bijian yang lainnya. *Chia seed* berbentuk biji kecil seukuran biji wijen, bulat, pipih, dan lonjong. Ada tiga warna yang dimiliki oleh *chia seed*, yaitu hitam yang dikenal *black chia*, putih, dan keabu-abuan [3]. Di sisi lain, biji selasih memiliki total serat pangan (*total dietary fiber/TDF*) sebesar 62.85g/100g [4]. Warna pada biji selasih umumnya adalah hitam dan tidak terlalu bervariasi. Secara tradisional, selasih telah digunakan sebagai obat untuk menyembuhkan beberapa penyakit seperti, sakit kepala, batuk, diare, sembelit, penyakit kulit, penyakit cacing dan gagal ginjal [5].

Dikarenakan kedua biji ini memiliki karakteristik yang hampir mirip, membedakan antara mana yang merupakan *chia seed* dan biji selasih menjadi sebuah tantangan. Masyarakat sering salah dalam membedakan kedua jenis biji ini. Di pasaran, harga *chia seed* bisa jauh lebih mahal dari biji selasih, yang mana untuk harga *chia seed* sendiri bisa mencapai Rp100.000-150.000/100g, sedangkan biji selasih hanya berkisar di harga Rp15.000/100g. Kita dapat menerapkan teknologi di bidang informatika untuk membedakan kedua jenis biji ini dengan teknologi pengolahan citra digital 2-dimensi. Teknologi ini tidak hanya bermanfaat untuk kebutuhan akademik, namun juga memiliki aplikasi nyata dalam industri makanan, terutama dalam proses *quality control*, pendeteksian bahan baku di pabrik pengolahan, dan pengecekan keaslian produk oleh distributor serta konsumen.

Ada beberapa penelitian yang menggunakan HSV dan disertai algoritma kNN, salah satunya oleh [6] untuk membedakan citra kematangan buah tomat, di mana di sini warna menjadi acuan dan *pixel* gambar mempengaruhi akurasi yang didapatkan. Dari penelitian ini didapatkan akurasi sebesar 92.5%. Di sisi lain, penelitian yang dilakukan oleh [7] dengan metode CNN untuk mengklasifikasikan kematangan buah pisang memiliki akurasi 88% dengan *pixel* gambar berukuran 200x200. Selain itu penelitian yang dilakukan oleh [8] menggunakan perpaduan GLCM dan HSV untuk mengidentifikasi motif dari kain tenun NTT, yang mana bisa digunakan untuk mengidentifikasi *chia seed* dan biji selasih yang berukuran kecil. Pada sebuah penelitian yang dilakukan oleh [9], uji yang dilakukan untuk klasifikasi pada kematangan buah manggis memperoleh nilai akurasi sebesar 86.6% dengan menggunakan metode HSV. Penelitian mengenai prediksi penyakit jantung menggunakan algoritma k-NN yang dipadukan dengan metode PSO oleh [10] mencapai tingkat akurasi sebesar 97.67%.

Berdasarkan beberapa penelitian terkait yang sudah dilakukan, metode yang diusulkan untuk diterapkan pada penelitian kali ini adalah kNN dikarenakan memiliki tingkat akurasi yang cukup tinggi dan sampel yang diperlukan juga tergolong tidak terlalu banyak untuk melatih model ini. Kedua biji ini sangat mirip dan bagi orang awam yang tidak terlalu mengenal biji-bijian ini mungkin akan salah mengira, perbedaan paling signifikannya adalah pada warna biji, seperti yang sudah disebutkan sebelumnya bahwa *chia seed* memiliki variasi warna abu, hitam, dan putih, sedangkan biji selasih hanya memiliki satu warna yaitu hitam, sehingga dalam hal segmentasi, penggunaan HSV, GLCM, dan *edge detection* dipilih untuk membedakan kedua jenis biji. Menurut [11], metode HSV adalah metode yang paling bagus untuk digunakan dalam mengenali fitur warna. Alasan penggunaan GLCM untuk ekstraksi fitur tekstur adalah karena menurut [12], GLCM terbukti sangat manjur sebagai pendeteksi karakteristik tekstur dari citra. Penelitian yang menyinggung soal penggunaan *edge detection* dalam mengenali citra yang dilakukan oleh [13] menyebutkan bahwa secara umum, pendeteksi tepi adalah proses awal dalam mendeteksi objek yang terdapat dalam sebuah citra.

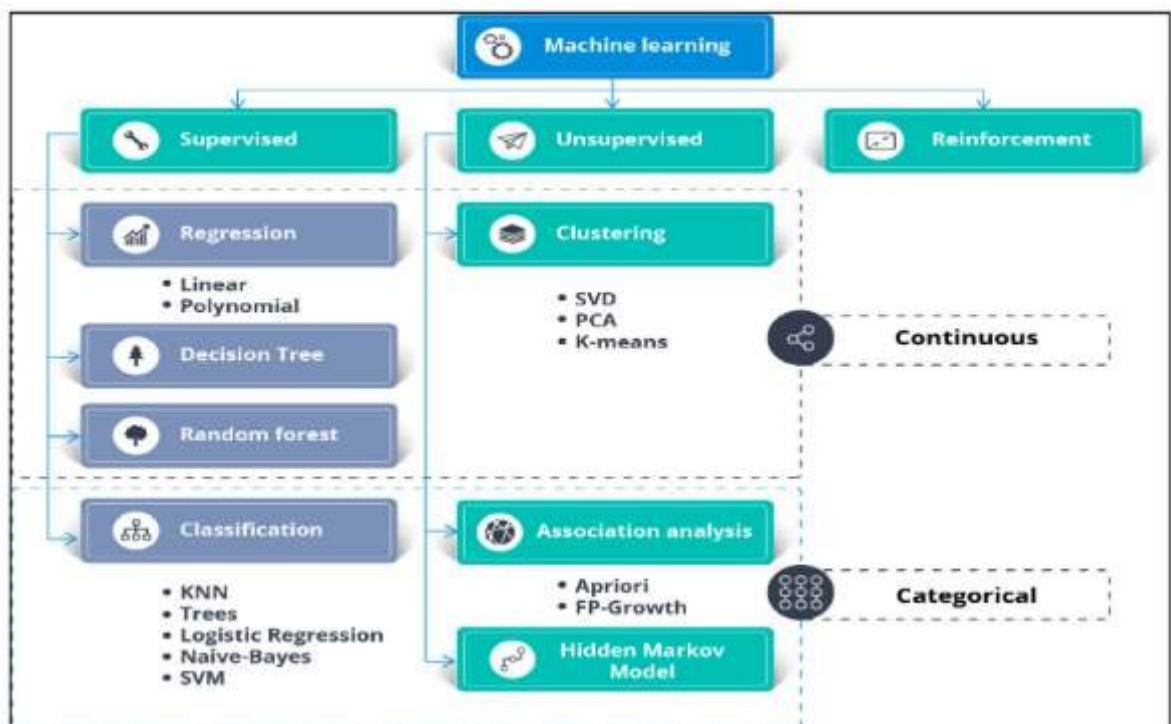
2. TINJAUAN PUSTAKA

2.1 *k*-Nearest Neighbor

Algoritma *k*-Nearest Neighbor (*k*-NN) adalah algoritma pembelajaran mesin yang sederhana namun efektif untuk masalah klasifikasi dan regresi. Algoritma ini bekerja dengan cara mencari nilai *k* dari tetangga yang terdekat dengan data baru dalam ruang fitur dan mengklasifikasikannya berdasarkan mayoritas kelas dari tetangga tersebut. Jarak yang umum digunakan untuk menentukan tetangga terdekat adalah jarak Euclidean, yang dapat dihitung dengan formulasi berikut :

$$d(p, q) = \sqrt{\sum_{i=1}^n (p_i - q_i)^2} \quad (1)$$

Algoritma *k*NN termasuk dalam kategori teknik pembelajaran mesin dengan tipe *supervised learning* dan dianggap sebagai salah satu algoritma yang paling mudah digunakan dalam pembelajaran mesin. Walaupun algoritma ini cocok untuk klasifikasi dan regresi, algoritma ini banyak digunakan terutama untuk mengklasifikasikan objek [14].



Gambar 1. Klasifikasi Algoritma *Machine Learning*

Penggunaan algoritma *k*-NN efektif dalam berbagai aplikasi, termasuk pada pengenalan pola dan klasifikasi citra. Jika dibandingkan dengan algoritma lain, *k*-NN menunjukkan kinerja yang baik dalam pengklasifikasian data yang memiliki dimensi tinggi, sehingga bisa dijadikan pilihan yang tepat untuk masalah klasifikasi citra. Adapun cara kerja atau tahapan dari algoritma klasifikasi *k*NN yaitu:

1. Inisiasi nilai *K*
2. Hitung jarak antara sampel *input* dan sampel *training*.
3. Uritkan setiap titik berdasarkan jarak terdekat.
4. Ambil nilai *K* tetangga terdekat.
5. Terapkan mayoritas sederhana.
6. Prediksi label kelas dengan lebih banyak tetangga untuk sampel *input*.

Namun, pemilihan nilai k yang tepat sangat penting untuk kinerja algoritma. Nilai k yang terlalu kecil dapat menyebabkan *overfitting*, sedangkan nilai k yang terlalu besar dapat mempengaruhi hasil keputusan, sehingga mengakibatkan kesalahan klasifikasi. Oleh karena itu, teknik seperti validasi silang sering digunakan untuk menentukan nilai k yang optimal untuk dataset tertentu [15].

2.2 RGB (Red, Green, Blue) Color Model

Model warna RGB (*Red, Green, Blue*) adalah model dasar dalam penggambaran warna digital, di mana warna dihasilkan melalui penggabungan cahaya merah, hijau, dan biru. Setiap saluran warna biasanya memiliki rentang dari 0 hingga 255, sehingga terdapat lebih dari 16 juta kemungkinan warna. Model RGB sangat efektif untuk teknologi tampilan, karena sesuai dengan cara layar memancarkan cahaya.

Penelitian menunjukkan bahwa meskipun model RGB sederhana, model warna ini memiliki keterbatasan dalam menangani variasi pencahayaan dan persepsi warna. Model RGB memiliki peran signifikan dalam representasi warna yang akurat dalam citra digital, yang sangat penting untuk tugas klasifikasi citra dan ekstraksi fitur [16].

Meskipun model RGB banyak digunakan, model HSV dapat memberikan hasil yang lebih konsisten dalam kondisi pencahayaan yang berbeda. Hal ini menunjukkan bahwa pemilihan model warna yang tepat sangat penting untuk aplikasi tertentu dalam pengolahan citra [17].

2.3 HSV (Hue, Saturation, Value) Color Model

Model warna HSV (*Hue, Saturation, Value*) adalah representasi warna yang lebih intuitif dibandingkan dengan model RGB. *Hue* menunjukkan jenis warna, *saturation* menunjukkan intensitas warna, dan *value* menunjukkan kecerahan. Konversi dari RGB ke HSV dapat dilakukan dengan formulasi berikut:

$$H = \begin{cases} 0 & \text{Jika } C_{max} = R \\ 60 \times \left(\frac{G - B}{C_{max} - C_{min}} + 6 \right) & \text{Jika } C_{max} = G \\ 60 \times \left(\frac{B - R}{C_{max} - C_{min}} + 4 \right) & \text{Jika } C_{max} = B \end{cases} \quad (2)$$

Di mana :

H = Nilai *Hue*

R, G, B = Nilai *Red, Green, Blue*

C_{max} = Nilai maksimum dari RGB

C_{min} = Nilai minimum dari RGB

Penelitian terbaru menunjukkan bahwa model HSV sangat efektif dalam pengolahan citra, terutama dalam deteksi objek dan pengenalan. Sebuah studi oleh [16] menunjukkan bahwa penggunaan model HSV dalam analisis citra medis dapat meningkatkan akurasi deteksi objek dibandingkan dengan penggunaan model RGB.

Model HSV juga memiliki keunggulan dalam kondisi pencahayaan yang bervariasi, karena memisahkan informasi warna dari intensitas. Hal ini memungkinkan segmentasi berbasis warna yang lebih akurat dalam citra, yang sangat penting untuk aplikasi dalam visi komputer [18].

2.4 Gray Level Co-occurrence Matrix (GLCM)

Gray Level Co-occurrence Matrix (GLCM) adalah metode yang digunakan untuk mengekstrak fitur tekstur dari citra. GLCM menghitung frekuensi pasangan piksel dengan nilai intensitas tertentu yang muncul dalam jarak dan arah tertentu. Metode ini sangat efektif dalam analisis citra karena dapat memberikan informasi yang mendalam tentang pola tekstur yang ada dalam citra.

Dalam konteks pengolahan citra, GLCM dapat dikombinasikan dengan teknik lain untuk meningkatkan hasil ekstraksi fitur. Penggunaan GLCM dalam pengolahan citra untuk analisis tekstur yang lebih mendalam menunjukkan bahwa GLCM dapat digunakan untuk mengevaluasi fitur tekstur yang membantu dalam klasifikasi citra dengan akurasi yang lebih baik [19]. Dengan demikian, GLCM

tidak hanya berfungsi sebagai alat analisis tekstur, tetapi juga sebagai komponen penting dalam pengolahan citra yang lebih kompleks.

Penggunaan GLCM dalam klasifikasi biji chia dan biji selasih dapat memberikan wawasan yang lebih baik tentang perbedaan tekstur antara kedua jenis biji tersebut. Dengan memanfaatkan fitur yang diekstrak dari GLCM, model klasifikasi seperti k-NN dapat dilatih untuk membedakan antara biji chia dan biji selasih dengan lebih akurat.

2.5 Edge detection

Edge detection adalah teknik penting dalam pengolahan citra yang digunakan untuk mendeteksi batas objek dalam citra. Metode *edge detection* menggabungkan beberapa langkah, termasuk penghalusan citra, perhitungan gradien, dan penekanan non-maksimum untuk menghasilkan tepi yang lebih akurat dan terdefinisi dengan baik yang di mana sangat efektif dalam mengurangi *noise* dan meningkatkan kualitas deteksi tepi [20]. Dalam konteks klasifikasi biji chia dan biji selasih, *edge detection* dapat membantu dalam mengidentifikasi tepi dan bentuk biji, yang merupakan fitur penting untuk membedakan antara kedua jenis biji ini.

Dalam penelitian terbaru oleh [21], metode *edge detection* yang ditingkatkan digunakan untuk meningkatkan akurasi segmentasi objek dalam citra. Penelitian ini menunjukkan bahwa dengan menggabungkan teknik *edge detection* dengan algoritma pembelajaran mesin, akurasi klasifikasi dapat ditingkatkan secara signifikan. Hasil penelitian ini relevan untuk klasifikasi biji chia dan biji selasih, di mana deteksi tepi yang akurat dapat membantu dalam membedakan antara kedua jenis biji berdasarkan bentuk dan ukuran.

2.6 Image Processing (Pemrosesan Citra)

Pemrosesan citra melibatkan manipulasi citra digital untuk meningkatkan kualitasnya atau mengekstrak informasi yang berguna. Teknik seperti *filtering*, segmentasi, dan ekstraksi fitur adalah dasar dalam bidang ini.

Kemajuan dalam pembelajaran mendalam telah merevolusi pemrosesan citra, memungkinkan pengembangan model yang dapat secara otomatis mempelajari fitur dari citra tanpa pemrograman eksplisit. Ini telah menyebabkan peningkatan signifikan dalam tugas seperti deteksi objek dan klasifikasi citra, menjadikan pemrosesan citra sebagai bidang yang dinamis dan terus berkembang [22].

2.7 Confusion matrix

Confusion matrix adalah sebuah metode yang dapat dengan mudah dan efektif menunjukkan performa dari klasifikasi yang dilakukan [23]. Model yang diklasifikasikan dengan benar hanya akan ditunjukkan oleh diagonal dalam matriks, sedangkan sisanya yang berupa model yang diklasifikasikan salah akan diklasifikasikan di luar dari diagonal [24].

Tabel 1. *Confusion Matrix*

Confusion matrix		True Class (Actual)	
		P	N
Hypothesized class (Predicted)	Y	True Positives	False Positives
	N	False Negatives	True Negatives

Cara perhitungan hasil dari *confusion matrix* termasuk perhitungan *accuracy*, *precision*, dan *recall* sebagai berikut

$$Accuracy = \frac{TP + TN}{TP + FP + TN + FN} \quad (3)$$

$$Precision = \frac{TP}{TP + FP} \quad (4)$$

$$Recall = \frac{TP + TN}{TP + FN} \quad (5)$$

2.8 K-fold Cross Validation

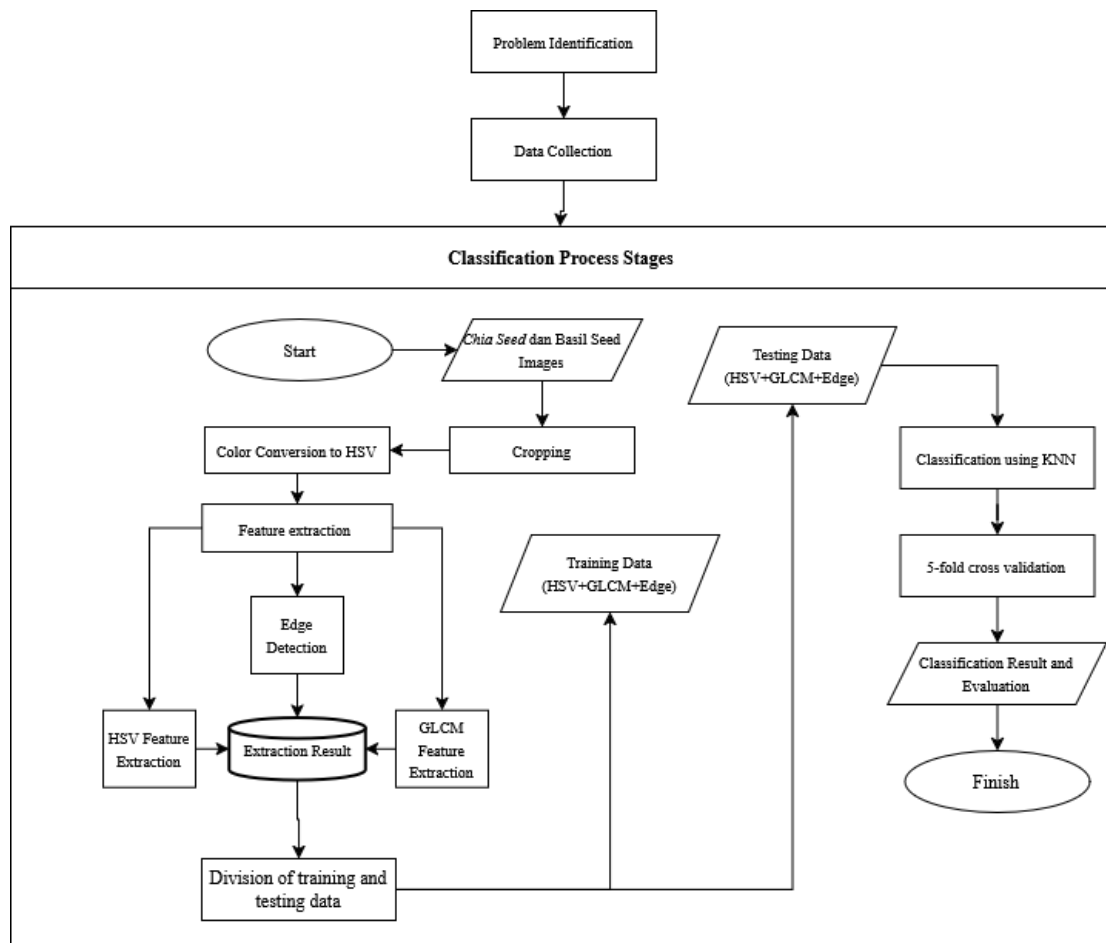
K-fold cross validation merupakan teknik evaluasi model yang banyak digunakan dalam pembelajaran mesin, khususnya pada klasifikasi citra. Metode ini membagi *dataset* menjadi k bagian (*folds*) yang sama besar, kemudian secara bergantian menggunakan satu *fold* sebagai data uji dan sisanya sebagai data latih. Proses ini diulang sebanyak k kali sehingga setiap bagian *dataset* berperan sebagai data uji sekali. Pendekatan ini membantu mengurangi bias evaluasi model serta memberikan estimasi performa yang lebih stabil dibandingkan dengan pembagian data tunggal. Dalam konteks klasifikasi citra, penggunaan *k-fold cross validation* memungkinkan pemanfaatan optimal dari *dataset* terbatas dengan memastikan bahwa setiap gambar diuji dan dilatih secara menyeluruh.

Menurut [25], penerapan teknik *k-fold cross validation* sangat penting untuk mendapatkan hasil akurasi yang andal pada aplikasi diagnostik berbasis *machine learning*. Penelitian tersebut menekankan bahwa metode ini membantu menghindari *overfitting* sekaligus memberikan gambaran performa model pada data baru secara realistis.

Meskipun *k-fold cross validation* adalah metode standar untuk seleksi model dalam *machine learning*, tantangan tetap ada terutama terkait distribusi data antar *folds* dan ukuran sampel kecil pada *dataset* citra kompleks [26].

3. METODE PENELITIAN

Pada penelitian ini dilakukan beberapa tahapan untuk mengklasifikasikan *chia seed* dan juga biji selasih menggunakan algoritma kNN, HSV, GLCM, dan juga *edge detection*. Setelah itu, hasil klasifikasi dilakukan *cross validation* sebanyak 5 fold yang mana dapat dilihat pada Gambar 2.



Gambar 2. Metodologi Penelitian

3.1 Pengumpulan data

Seperti yang diketahui ada 2 jenis biji yang digunakan dalam penelitian ini yaitu *chia seed* dan juga biji selasih. Sampel diambil dari pembelian langsung di pasar tradisional yaitu Pasar Tavip Binjai yang selanjutnya dilakukan proses pengambilan foto citra dengan latar belakang kertas putih dan diakuisisi menggunakan kamera Ipad Air M2 dengan format file adalah JPG. Pengambilan data citra dilakukan dengan memerhatikan beberapa posisi dan dengan pengaturan cahaya yang berbeda-beda. Data yang dihasilkan berjumlah 200 foto, di mana *chia seed* terdiri dari 100 foto, dan biji selasih juga 100 foto. Setelah citra didapatkan, citra dipotong kembali dengan beberapa ukuran untuk menghasilkan bentuk yang lebih mendetail.

Gambar 3. *Chia Seed* (Kiri), Biji Selasih (Kanan)

Hasil pengambilan foto dari citra memiliki beberapa ukuran dengan *width* antara 200 *pixel* sampai 1600 *pixel*, dan *height* antara 300 *pixel* sampai 1200 *pixel*.

3.2 Pra-Pemrosesan

Pada tahap selanjutnya, citra yang sudah didapatkan dilakukan proses pemotongan (*cropping*) seperti pada Gambar 3 dan 4. Pemotongan dilakukan dengan mengubah ukuran citra menjadi 200×200 piksel untuk memastikan keseragaman pada tahap pemrosesan selanjutnya.

Gambar 4. Pemotongan Citra *Chia Seed*

Gambar 5. Pemotongan Citra Biji Selasih

Untuk mendapatkan hasil yang lebih akurat, warna yang ada pada setiap citra dikonversi terlebih dahulu yaitu dari RGB menjadi HSV sesuai dengan metode yang digunakan pada penelitian kali ini.

3.3 Ekstraksi Ciri

Tahapan selanjutnya untuk menghasilkan data hasil klasifikasi yaitu dengan melakukan proses ekstraksi ciri dari citra yang telah dikumpulkan. Warna pada citra dikonversi ke HSV dengan menggunakan *library* OpenCV yang bisa melakukan konversi secara otomatis [27].

Beberapa tahapan proses yang dilakukan yaitu:

1. Normalisasi nilai RGB dari citra ke rentang $[0,1]$;

$$R' = \frac{R}{255} ; G' = \frac{G}{255} ; B' = \frac{B}{255}$$

Di mana :

R = nilai *Red* (merah)

G = nilai *Green* (hijau)

B = nilai *Blue* (biru)

Untuk memperoleh nilai V pada HSV maka kita mengambil nilai maksimal dari nilai RGB yang sudah kita normalisasi pada langkah 1. Lalu setelah itu, kita ambil sebuah variabel untuk menghitung nilai minimum (X). Nilai H dan S berpengaruh pada nilai maksimum dan minimum dari komponen RGB [26]. Jika nilai maksimum (V) dan nilai minimum (X) sangat dekat, berarti warna hampir tidak memiliki saturasi.

2. Hitung nilai V (*Value*) = $\max(R', G', B')$; (7)

3. Hitung nilai X = $\min(R', G', B')$; (8)

4. Hitung nilai S (*Saturation*) = $\frac{V-X}{V}$, jika $V \neq 0$ (9)

Jika $V = 0$ maka: $S = 0$ (warna hitam, tanpa saturasi)

Dalam memperoleh nilai H, nilai yang diperoleh di modulo 360 dikarenakan rentang dari nilai HSV adalah 0-360. Mod 360 memastikan nilai tetap dalam batas yang dapat dipahami dalam sistem HSV.

5. Hitung nilai H (*Hue*)

$$\text{Jika } V = R' \text{ maka: } H = 60 \times \left(4 + \frac{G'-B'}{V-X}\right) \bmod 360 \quad (10)$$

$$\text{Jika } V = G' \text{ maka: } H = 60 \times \left(2 + \frac{B'-R'}{V-X}\right) \bmod 360 \quad (11)$$

$$\text{Jika } V = B' \text{ maka: } H = 60 \times \left(4 + \frac{R'-G'}{V-X}\right) \bmod 360 \quad (12)$$

$$\text{Jika } V = X \text{ maka: } H = 0 \text{ (warna netral atau grayscale)} \quad (13)$$

Dikarenakan library yang digunakan adalah OpenCV (Open Source Computer Vision Library), maka kita perlu membagi nilai H yang didapatkan karena library ini memiliki rentang 0-180 dan citra disimpan dalam format 8-bit.

6. Hitung nilai H akhir yaitu $H_{\text{opencv}} = \frac{n}{2}$ (14)

Selanjutnya, GLCM digunakan untuk menganalisis tekstur citra berdasarkan frekuensi kemunculan pasangan piksel dengan intensitas tertentu pada jarak dan arah tertentu untuk memberikan informasi tentang pola distribusi intensitas piksel yang dapat digunakan untuk menghitung fitur tekstur seperti kontras dan energi [28].

7. Hitung nilai Kontras mengukur variasi intensitas lokal dalam citra

$$\text{Contrast} = \sum_{i,j} |i - j|^2 P(i, j) \quad (15)$$

di mana $P(i,j)$ adalah elemen matriks GLCM pada posisi (i,j) .

8. Hitung Energi untuk mengukur homogenitas atau keteraturan tekstur citra

$$\text{Energy} = \sum_{i,j} P(i, j)^2 \quad (16)$$

Lalu menggunakan *Canny Edge detection* untuk menemukan batas objek dalam citra untuk mengurangi noise, mempertajam tepi serta thresholding hysteresis untuk menentukan tepi akhir.

9. Menghitung Rasio jumlah piksel tepi terhadap total area citra

$$\text{Edge Ratio} = \frac{\text{Jumlah Piksel Tepi}}{\text{Total Piksel}} \quad (17)$$

Setelah deteksi tepi dilakukan oleh algoritma Canny, kontur-kontur objek akan diekstrak dan difilter. Dari kontur terbesar dipilih fitur bentuk seperti luas area, keliling, serta *circularity* yang menggambarkan seberapa bulat bentuk tersebut [29].

10. Menghitung *circularity* untuk mengukur seberapa mendekati bentuk kontur ke lingkaran sempurna

$$Circularity = \frac{4\pi \times Area}{Perimeter^2} \quad (18)$$

Nilai *circularity* berkisar antara 0 hingga 1, yang di mana semakin mendekati 1, semakin sempurna bentuk lingkarannya.

3.4 Pembagian Data

Setelah melalui tahap ekstraksi, langkah selanjutnya adalah membagi data yang ada menjadi data latih dan data uji. Pembagian data dilakukan dengan metode *stratified shuffle split* [30], di mana 80% dari total data dijadikan sebagai data latih (*training set*), dan 20% dari total data dijadikan data uji (*testing set*). Untuk membuat pembagian *dataset* tetap konsisten maka digunakan algoritma *random state* [31] sebesar 46. Penggunaan *random state* adalah untuk memastikan bahwa *dataset* terbagi dengan cara yang sama setiap model dijalankan.

3.5 Pengujian Dan Pengukuran Akurasi

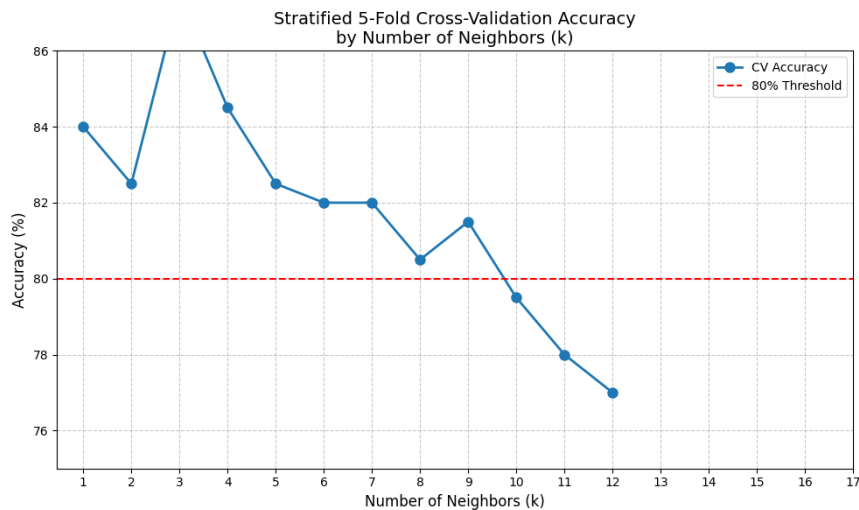
Dikarenakan pengklasifikasian menggunakan algoritma kNN, maka data dari tetangga terdekat yang akan dijadikan sebagai acuan untuk mengecek kesesuaian gambar. Untuk pemilihan nilai parameter *k* terbaik dalam algoritma *k-Nearest Neighbors* (k-NN), model dijalankan dengan batas performa sebesar 80%, jika performa berada di bawah 80% sebanyak 3 kali maka perulangan akan berhenti dan didapatkan nilai *k* yang menghasilkan akurasi tertinggi. Evaluasi dilakukan menggunakan metode *stratified 5-Fold cross validation*, yaitu membagi data menjadi 5 subset yang proporsional terhadap kelas, lalu secara bergantian digunakan sebagai data uji dan data latih. Penggunaan stratifikasi memastikan bahwa distribusi kelas tetap seimbang di setiap *fold*. Akurasi yang dihasilkan akan ditampilkan menggunakan *confusion matrix* [32].

4. HASIL DAN PEMBAHASAN

Setelah dilakukan ekstraksi pada citra, diperoleh nilai HSV, kontras, dan juga rasio tepi untuk masing-masing citra. Pengujian dilakukan dengan menggunakan *K-Fold Cross-Validation* untuk mencari nilai *k* terbaik. Hasil pengujian yang didapatkan dengan menggunakan citra berukuran 200 x 200 *pixel* sebagai berikut.

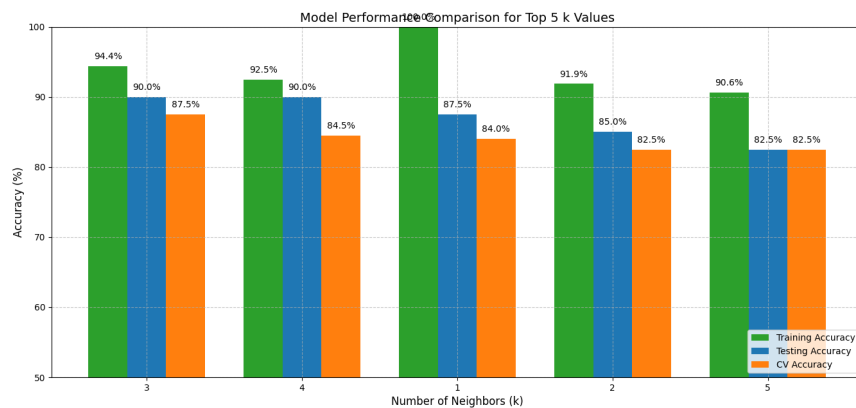
	Hue	Saturation	Value	Contrast	Energy	Edge_Ratio	Shape_Area	Shape_Perimeter	Circularity	Class
0	22.406450	72.480950	145.941825	2337.712839	0.024236	0.269750	2036.5	3806.876653	0.001766	Chia
1	27.366075	27.697425	173.326075	644.020603	0.066020	0.081525	180.5	584.825462	0.006632	Chia
2	40.420600	18.626700	157.494425	79.975226	0.069969	0.048650	2244.5	211.237589	0.632102	Chia
3	18.819725	106.871850	108.632575	2026.074146	0.007527	0.352100	1079.0	2628.581065	0.001962	Chia
4	22.000175	80.137675	120.050500	3271.597362	0.006510	0.361075	2019.5	3532.543612	0.002034	Chia
..	...	...	...	...	...	...	...	...	...	...
195	63.916300	10.346700	183.151725	455.176055	0.047650	0.037875	510.5	460.516806	0.030249	Basil
196	46.289425	4.264500	211.302775	47.298015	0.117065	0.008350	235.5	60.526911	0.807800	Basil
197	49.997325	13.268275	192.381725	317.021407	0.101479	0.040850	673.5	712.825461	0.016656	Basil
198	16.714025	39.101450	120.304200	448.748970	0.129593	0.255000	1331.0	2009.512962	0.004142	Basil
199	65.690200	12.256400	210.909125	75.633518	0.052494	0.013475	2783.5	360.149275	0.269672	Basil

Gambar 6. Hasil Ekstraksi Ciri dari Citra dengan Ukuran 200x200 *pixel*



Gambar 7. Stratified 5-Fold Cross Validation Accuracy

Dari Gambar 7, kita bisa melihat bahwa terdapat penurunan *accuracy* sebanyak 3 kali berturut-turut pada nilai $k = 10, 11$, dan 12 sehingga pengujian nilai k dihentikan, dan k yang digunakan adalah dari $1 - 9$.

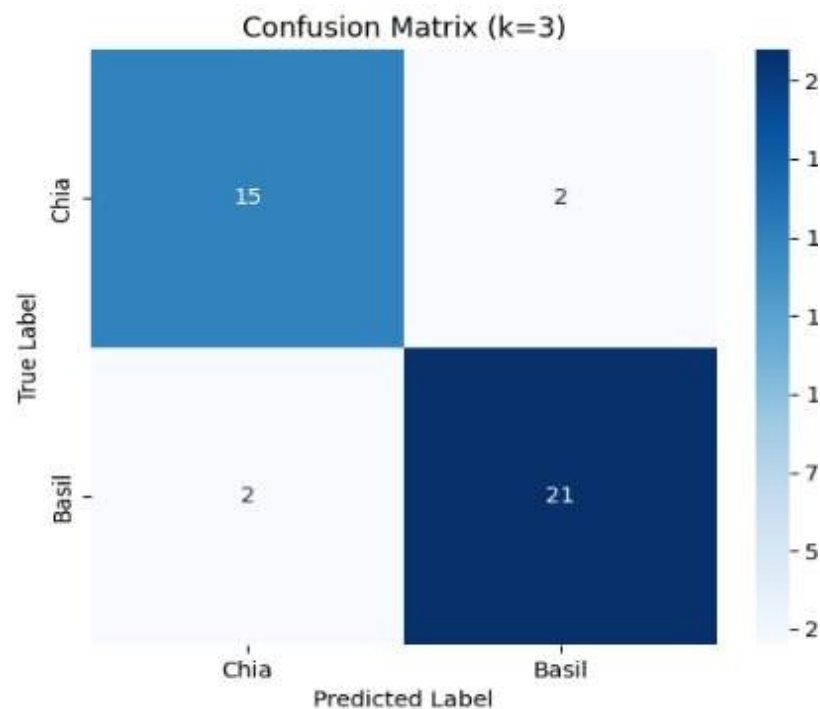


Gambar 8. Perbandingan Akurasi Data Latih, Data Uji dan Cross Validation

Tabel 2. Perbandingan Nilai Akurasi Data Latih dan Data Uji

k	Training Accuracy	Testing Accuracy	CV Accuracy
3	94.38%	90%	87.5%
4	92.5%	90%	84.5%
1	100%	87.5%	84%
2	91.88%	85%	82.5%
5	90.62%	82.5%	82.5%

Tabel 2 menunjukkan urutan nilai k dari yang tertinggi sampai terendah nilai akurasinya dan diambil sebanyak 5 nilai k dari 9 pengujian nilai k . Dari data tersebut, didapatkan bahwa rata-rata nilai akurasi untuk data latih adalah 93.87%, sedangkan untuk akurasi data uji memiliki rata-rata nilai sebesar 87%. Jika kita melihat lebih dekat lagi, nilai $k=3$ memiliki nilai akurasi maksimum untuk data uji dan juga *cross validation*, yaitu 90% untuk data uji dan 87.5% untuk *cross validation*. Setelah itu, *confusion matrix* [32] digunakan untuk menguji keakurasian dari model dengan nilai $k=3$ dengan hasil berikut.

Gambar 9. Pengukuran Akurasi dengan *Confusion Matrix*

Berdasarkan gambar pembagian ketepatan data yang diuji, kita dapat menghitung akurasinya dengan cara berikut [33].

$$Akurasi = \frac{Total\ Prediksi\ Benar}{Total\ Seluruh\ Pengujian} \times 100\%$$

$$Akurasi = \frac{TP + TN}{TP + FP + TN + FN} \times 100\%$$

$$Akurasi = \frac{15 + 21}{15 + 2 + 2 + 21} \times 100\% = 90\%$$

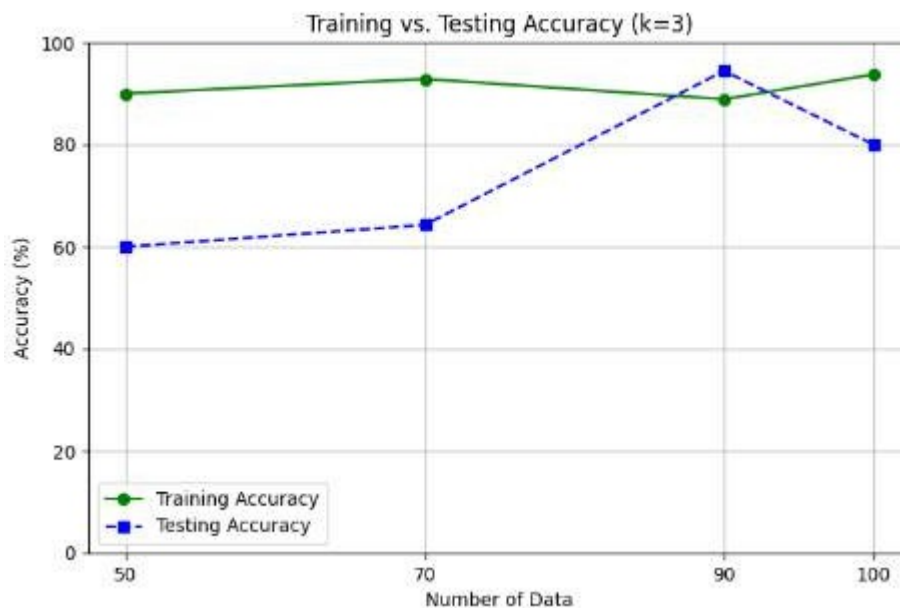
Di mana :

TP = *True Positive*

FP = *False Positive*

TN = *True Negative*

FN = *False Negative*



Gambar 10. Pengukuran Akurasi Data *Training* dan Data *Testing* dengan Beberapa Jumlah Data Citra

Dapat kita lihat dari grafik pada Gambar 10 bahwa jumlah data yang digunakan untuk melatih data *training* dan data uji juga akan mempengaruhi nilai akurasi yang akan didapat. Kita bisa melihat bahwa akurasi yang tertinggi untuk data testing didapatkan dengan jumlah data 90 citra gambar yaitu sebesar 88.89% untuk data *training* dan 94.4% untuk data *testing*. Sedangkan untuk akurasi tertinggi data *training* berada pada jumlah data sebesar 100 yaitu 93.75% untuk data *training* dan 80% untuk data testing. Dengan demikian, ada batas optimal untuk penambahan data citra yang tidak lagi meningkatkan performa model.

5. KESIMPULAN

Dari pengujian yang dilakukan, dapat disimpulkan bahwa nilai akurasi maksimum yang didapatkan adalah 94.4% untuk data yang diujikan dengan jumlah data citra untuk *chia seed* dan juga biji selasih masing-masing sebesar 90 data citra dan nilai keanggotaan k adalah 3 serta ukuran citra yang digunakan adalah 200x200 pixel dari hasil *cropping*. Hal ini membuktikan bahwa data yang terlalu banyak juga belum tentu dapat meningkatkan akurasi dari sebuah model dan dapat menurunkan performa dari model.

Adanya beberapa hal yang menjadi batasan dalam penelitian kali ini dan masih butuh untuk ditingkatkan, seperti pengambilan data citra dari *device* yang lebih memadai, menambah variasi data citra, dan pengurangan *noise* pada saat pra-pemrosesan data citra.

6. SARAN

Adapun saran dari penulis untuk pengembangan selanjutnya yaitu sebagai berikut.

1. Menambahkan variasi dari data baik biji selasih maupun *chia seed*.
2. Menggunakan metode lain seperti CNN dan membandingkan akurasinya dengan k -NN.
3. Menghitung nilai akurasi program untuk data uji gambar acak
4. Mengembangkan proyek ke dalam bentuk aplikasi dengan desain yang mudah dimengerti untuk digunakan.
5. Menghitung nilai akurasi dengan beberapa ukuran dari citra.

UCAPAN TERIMA KASIH

Penulis mengucapkan terima kasih kepada dosen pembimbing dan Universitas Mikroskil, serta pihak-pihak yang telah terlibat dalam penelitian ini yang telah memberikan kontribusi baik berupa materi maupun jasa.

DAFTAR PUSTAKA

- [1] D. R. Adawiyah, F. P. Wefiani, and K. Patricia, "Karakterisasi Serat Pangan, Kapasitas Pengikatan Air dan Kemampuan Emulsifikasi Biji Selasih dan Chia," *Jurnal Mutu Pangan : Indonesian Journal of Food Quality*, vol. 8, no. 2, pp. 63–69, Mar. 2022, doi: 10.29244/jmpi.2021.8.2.63.
- [2] R. Rasbawati and I. Irmayani, "Pemanfaatan Biji Chia (*Salvia hispanica* L.) untuk Meningkatkan Kualitas Susu Diversifikasi," *Jurnal Peternakan Indonesia (Indonesian Journal of Animal Science)*, vol. 23, no. 2, p. 159, Jun. 2021, doi: 10.25077/jpi.23.2.159-167.2021.
- [3] V. Y. Ixtaina, S. M. Nolasco, and M. C. Tomás, "Physical properties of chia (*Salvia hispanica* L.) seeds," *Ind Crops Prod*, vol. 28, no. 3, pp. 286–293, Nov. 2008, doi: 10.1016/j.indcrop.2008.03.009.
- [4] A. Hajmohammadi, M. Pirouzifard, M. Shahedi, and M. Alizadeh, "Enrichment of a fruit-based beverage in dietary fiber using basil seed: Effect of Carboxymethyl cellulose and Gum Tragacanth on stability," *LWT*, vol. 74, pp. 84–91, Dec. 2016, doi: 10.1016/j.lwt.2016.07.033.
- [5] A. D. R. Nurcahyanti *et al.*, "AKTIVITAS ANTIOKSIDAN DAN ANTIBAKTERI EKSTRAK POLAR DAN NON POLAR BIJI SELASIH (*Ocimum sanctum* Linn) [Antioxidant and Antibacterial Activities from Polar and Non Polar Basil (*Ocimum sanctum* Linn) Seed Extracts]," 2011.
- [6] S. Sanjaya, M. L. Pura, S. K. Gusti, F. Yanto, and F. Syafria, "K-Nearest Neighbor for Classification of Tomato Maturity Level Based on Hue, Saturation, and Value Colors," *Indonesian Journal of Artificial Intelligence and Data Mining*, vol. 2, no. 2, p. 101, Nov. 2019, doi: 10.24014/ijaidm.v2i2.7975.
- [7] A. I. Hanifah and A. Hermawan, "Klasifikasi Kematangan Pisang Menggunakan Metode Convolutional Neural Network," *Komputika : Jurnal Sistem Komputer*, vol. 12, no. 2, pp. 49–56, Sep. 2023, doi: 10.34010/komputika.v12i2.9999.
- [8] F. S. Lesiangi, A. Y. Mauko, and B. S. Djahi, "Feature extraction Hue, Saturation, Value (HSV) and Gray Level Cooccurrence Matrix (GLCM) for identification of woven fabric motifs in South Central Timor Regency," *J Phys Conf Ser*, vol. 2017, no. 1, p. 012010, Sep. 2021, doi: 10.1088/1742-6596/2017/1/012010.
- [9] M. K. Anam, S. Sumijan, K. Karfindo, and M. B. Firdaus, "Comparison Analysis of HSV Method, CNN Algorithm, and SVM Algorithm in Detecting the Ripeness of Mangosteen Fruit Images," *Indonesian Journal of Artificial Intelligence and Data Mining*, vol. 7, no. 2, p. 348, May 2024, doi: 10.24014/ijaidm.v7i2.29739.
- [10] E. Panca Saputra, A. Saryoko, and A. Tunjung Kusumo, "Prediction of Heart Disease Using K-Nearest Neighbor and Particle Swarm Optimization (PSO) Method," *Indonesian Journal of Artificial Intelligence and Data Mining (IJAIDM)*, vol. 8, no. 1, pp. 229–235, 2025, doi: 10.24014/ijaidm.v8i1.33373.
- [11] J. M. Chaves-González, M. A. Vega-Rodríguez, J. A. Gómez-Pulido, and J. M. Sánchez-Pérez, "Detecting skin in face recognition systems: A colour spaces study," *Digit Signal Process*, vol. 20, no. 3, pp. 806–823, May 2010, doi: 10.1016/j.dsp.2009.10.008.
- [12] F. Roberti de Siqueira, W. Robson Schwartz, and H. Pedrini, "Multi-scale gray level co-occurrence matrices for texture description," *Neurocomputing*, vol. 120, pp. 336–345, Nov. 2013, doi: 10.1016/j.neucom.2012.09.042.
- [13] R. Puspadini and M. P. U. Sitompul, "Implementation of the Laplacian of Gaussian Algorithm in Edge Detection Image Processing of Zebra Cross Damage on Highways in the Langkat Regency Area," *Journal of Artificial Intelligence and Engineering Applications (JAIEA)*, vol. 3, no. 2, pp. 601–605, Feb. 2024, doi: 10.59934/jaiea.v3i2.442.

- [14] M. Bansal, A. Goyal, and A. Choudhary, "A comparative analysis of K-Nearest Neighbor, Genetic, Support Vector Machine, Decision Tree, and Long Short Term Memory algorithms in machine learning," *Decision Analytics Journal*, vol. 3, p. 100071, Jun. 2022, doi: 10.1016/j.dajour.2022.100071.
- [15] E. Ozturk Kiyak, B. Ghasemkhani, and D. Birant, "High-Level K-Nearest Neighbors (HLKNN): A Supervised Machine Learning Model for Classification Analysis," *Electronics (Basel)*, vol. 12, no. 18, p. 3828, Sep. 2023, doi: 10.3390/electronics12183828.
- [16] N. A. Ibraheem, M. M. Hasan, R. Z. Khan, and P. K. Mishra, "ARPN Journal of Science and Technology:: Understanding Color Models: A Review," *ARPN Journal of Science and Technology*, vol. 2, no. 3, 2012, [Online]. Available: <http://www.ejournalofscience.org>
- [17] T. B. Plante and M. Cushman, "Choosing color palettes for scientific figures," *Res Pract Thromb Haemost*, vol. 4, no. 2, pp. 176–180, Feb. 2020, doi: 10.1002/rth2.12308.
- [18] G. B. Satria, I. N. A. Ramatryana, and S. Y. Shin, "Compressive Sensing of Medical Images Based on HSV Color Space," *Sensors*, vol. 23, no. 5, p. 2616, Feb. 2023, doi: 10.3390/s23052616.
- [19] K. Sarode, R. Savdekar, and T. Chaudhari, "Texture Feature Analysis of an Image Using Gray Level Co-Occurrence Matrix," *International Journal of Novel Research and Development*, vol. 7, no. 2, pp. 139–143, 2022.
- [20] Z. Cai, Z. Ma, Z. Zuo, Y. Xiang, and M. Wang, "An Image Edge Detection Algorithm Based on an Artificial Plant Community," *Applied Sciences*, vol. 13, no. 7, p. 4159, Mar. 2023, doi: 10.3390/app13074159.
- [21] J. Xu, S. Niu, and Z. Wang, "Object tracking method based on edge detection and morphology," *EURASIP J Adv Signal Process*, vol. 2024, no. 1, p. 45, Apr. 2024, doi: 10.1186/s13634-024-01144-0.
- [22] L. Lei, Z. Jia, and Z. Wu, "Research on Image Classification and Key Technologies Based on 3D Feature Extraction Algorithm," *J Sens*, vol. 2022, pp. 1–14, Aug. 2022, doi: 10.1155/2022/5859925.
- [23] H. Yun, "Prediction model of algal blooms using logistic regression and confusion matrix," *International Journal of Electrical and Computer Engineering (IJECE)*, vol. 11, no. 3, p. 2407, Jun. 2021, doi: 10.11591/ijece.v11i3.pp2407-2413.
- [24] C. Hung, J. Filbert Wijaya, I. Adiputra Pardosi, and F. Mikael Sinaga, "Prediksi Fluktuasi Harga Bitcoin Dengan Menggunakan Random Forest Classifier," *Jl. Thamrin No.124 Medan*, vol. 24, no. 2, pp. 95–108, Oct. 2023, doi: 10.55601/jsm.24i2.pg.
- [25] I. Tougui, A. Jilbab, and J. El Mhamdi, "Impact of the Choice of Cross-Validation Techniques on the Results of Machine Learning-Based Diagnostic Applications," *Healthc Inform Res*, vol. 27, no. 3, pp. 189–199, Jul. 2021, doi: 10.4258/hir.2021.27.3.189.
- [26] M. Rhifky Wayahdi, D. Syahputra, S. Hafiz, and N. Ginting, "EVALUATION OF THE K-NEAREST NEIGHBOR MODEL WITH K-FOLD CROSS VALIDATION ON IMAGE CLASSIFICATION," *INFOKUM*, vol. 9, no. 1, pp. 1–6, Dec. 2020, [Online]. Available: <http://infor.seaninstitute.org/index.php/infokum/index>
- [27] G. Xie and W. Lu, "Image Edge Detection Based On Opencv," *International Journal of Electronics and Electrical Engineering*, vol. 1, no. 2, pp. 104–106, 2013, doi: 10.12720/ijeec.1.2.104-106.
- [28] F. T. Kurniati, I. Sembiring, A. Setiawan, I. Setyawan, and R. R. Huizen, "GLCM-Based Feature Combination for Extraction Model Optimization in Object Detection Using Machine Learning," *Jurnal Ilmiah Teknik Elektro Komputer dan Informatika*, vol. 9, no. 4, pp. 1196–1205, Jan. 2024, doi: 10.26555/jiteki.v9i4.27842.
- [29] P. Cai, Z. Cai, Y. Fan, and W. Wu, "Image Contour Detection Based on Visual Pathway Information Transfer Mechanism," *Neural Process Lett*, vol. 56, no. 1, p. 6, Feb. 2024, doi: 10.1007/s11063-024-11486-3.
- [30] S. Ünal, O. Günay, I. Akkurt, K. Gunoglu, and H. O. Tekin, "A comparative study on breast cancer classification with stratified shuffle split and K-fold cross validation via ensembled machine learning," *J Radiat Res Appl Sci*, vol. 17, no. 4, p. 101080, Dec. 2024, doi: 10.1007/s11063-024-11486-3.

- 10.1016/j.jrras.2024.101080.
- [31] S. Dutta, A. Arunachalam, and S. Misailovic, "To Seed or Not to Seed? An Empirical Analysis of Usage of Seeds for Testing in Machine Learning Projects," in *2022 IEEE Conference on Software Testing, Verification and Validation (ICST)*, IEEE, Apr. 2022, pp. 151–161. doi: 10.1109/ICST53961.2022.00026.
- [32] Q. A'yuniyah *et al.*, "Implementasi Algoritma Naïve Bayes Classifier (NBC) untuk Klasifikasi Penyakit Ginjal Kronik," *Jurnal Sistem Komputer dan Informatika (JSON)*, vol. 4, no. 1, p. 72, Sep. 2022, doi: 10.30865/json.v4i1.4781.
- [33] J. Jasril and S. Sanjaya, "Learning Vector Quantization 3 (LVQ3) and Spatial Fuzzy C-Means (SFCM) for Beef and Pork Image Classification," *Indonesian Journal of Artificial Intelligence and Data Mining*, vol. 1, no. 2, p. 60, Oct. 2018, doi: 10.24014/ijaidm.v1i2.5024.