

IDENTIFIKASI KEMATANGAN BUAH NANAS MENGGUNAKAN CITRA DIGITAL DENGAN METODE *K-NEAREST NEIGHBOR* (K-NN) DAN *SUPPORT VECTOR MACHINE* (SVM)

Jevan George Mantofani Sanu

Program Studi Teknik Informatika, STIKOM Uyelindo

Jl. Perintis Kemerdekaan, Kupang – Nusa Tenggara Timur, Indonesia

Email: jevangeorge21@gmail.com

ABSTRAK

Produksi nanas di Indonesia pada 2021 mencapai 2,89 juta ton dan pada tahun yang sama salah satu provinsi di Indonesia yaitu NTT menghasilkan sebanyak 11.241 ton nanas. Pemilihan buah nanas secara manual membutuhkan waktu yang lama serta menghasilkan pemanenan buah nanas memiliki tingkat kematangan yang tidak merata, hal ini dikarenakan tingkat konsistensi manusia sangat beragam dalam menilai kematangan. Proses ini dapat dibantu dengan sistem cerdas yang dapat menentukan kualitas berdasarkan karakteristik yang sudah ada, sistem cerdas yang digunakan untuk mengidentifikasi tingkat kematangan buah nanas adalah Support Vector Machine (SVM) dan k-Nearest Neighbor (k-NN). Pada penelitian ini menggunakan 200 citra buah nanas dengan tingkat kematangan yaitu belum matang, setengah matang dan matang. Buah nanas yang dipilih secara acak dari petani. Langkah pertama adalah preprocessing citra dengan ekstraksi warna *red*, *green*, dan *blue*, selanjutnya penentuan parameter nilai histogram yaitu mean, variance, skewness, kurtosis, dan entropy selanjutnya menggunakan metode k-NN untuk mencari nilai k terdekat. Selanjutnya Metode yang digunakan adalah Support Vector Machine (SVM), metode ini digunakan untuk membentuk model identifikasi saat proses pemanenan dan pendistribusian serta dapat mengurangi selektifitas para petani dalam menentukan buah nanas yang akan dipanen. Data set dibagi menjadi 80% dan 20% dengan 5 skenario cross validation. Hasil penelitian ini dapat disimpulkan bahwa identifikasi kematangan buah nanas dengan k-Nearest Neighbor (k-NN) dan Support Vector Machine (SVM) berhasil mengidentifikasi buah nanas matang dan belum matang dengan confusion matrix dengan 5 skenario cross validation yang memperoleh nilai akurasi terbaik pada skenario ke-1 sebesar 93% dengan nilai rata-rata akurasi sistem sebesar 81,2%.

Kata kunci : Citra Digital, *Cross Validation*, *k-Nearest Neighbor* (k-NN), *Support Vector Machine* (SVM)

ABSTRACT

Pineapple production in Indonesia in 2021 reached 2.89 million tons and in the same year one of the provinces in Indonesia, namely NTT, produced 11,241 tons of pineapples. Manual selection of pineapples takes a long time and results in pineapple harvesting with uneven levels of ripeness, this is because the level of human consistency varies greatly in assessing ripeness. This process can be assisted by an intelligent system that can determine quality based on existing characteristics, the intelligent system used to identify the level of ripeness of pineapples is Support Vector Machine (SVM) and k-Nearest Neighbor (k-NN). This study used 200 pineapple images with levels of ripeness, namely unripe, half-ripe and ripe. Pineapples are randomly selected from farmers. The first step is image preprocessing with red, green, and blue color extraction, then determining the histogram value parameters, namely mean, variance, skewness, kurtosis, and entropy, then using the k-NN method to find the nearest k value. Furthermore, the method used is Support Vector Machine (SVM), this method is used to form an identification model during the harvesting and distribution process and can reduce the selectivity of farmers in determining which pineapples to harvest. The data set is divided into 80% and 20% with 5 cross validation scenarios. The results of this study can be concluded that the identification of pineapple ripeness with k-Nearest Neighbor (k-NN) and Support Vector Machine (SVM) successfully identified ripe and unripe pineapples with a confusion matrix with 5 cross validation scenarios that obtained the best accuracy value in scenario 1 of 93% with an average system accuracy value of 81.2%.

Keywords : *Cross Validation* , *Digital Image*, *k-Nearest Neighbor* (k-NN), *Support Vector Machine* (SVM)

1. PENDAHULUAN

Nanas (*ananas comosus*) merupakan tanaman buah yang banyak tumbuh di daerah subtropis hingga tropis. Nanas umumnya mencapai kematangan setelah 120 - 270 hari. Berdasarkan data dari [1] produksi nanas di Indonesia mencapai 2,89 juta ton, jumlah tersebut tumbuh 17,95% dibandingkan pada tahun sebelumnya sebesar 2,45 juta ton dan pada tahun yang sama Nusa Tenggara Timur memproduksi sebanyak 11.541 ton nanas. Buah nanas diklasifikasikan menjadi beberapa tahap kedewasaan yang dimulai dari warna hijau tua atau nanas mentah hingga berwarna *orange* atau nanas matang penuh. Pada tahap pertama, buah nanas terbilang masih muda atau belum matang dan belum layak untuk dipetik. Kemudian pada tahap kedua, buah nanas sudah mulai memasuki tingkat permulaan kematangan, berwarna hijau tua dengan sedikit kekuning-kuningan di bagian mata dan bagian pangkal sudah dapat dipanen dan layak untuk diekspor. Pada tahap ketiga, buah nanas sudah matang, keseluruhan mata berwarna hijau dengan 1-2 mata bagian pangkal yang berwarna kuning sudah siap untuk diekspor. Pada tahap keempat, buah nanas sudah mulai masak dan sekitar 25% mata pada bagian pangkal buah nanas sudah berubah warna menjadi orange kekuning-kuningan. Selanjutnya, pada tahap kelima, mata pada buah nanas sudah 50% berubah warna menjadi orange kekuning-kuningan. Pada tahap keenam, lebih dari 75% mata buah nanas sudah berwarna *orange* kekuning-kuningan dan pada tahap ketujuh atau tahap terakhir, buah nanas sudah masak ranum dan mata dari buah nanas berubah warna menjadi *orange* kuning [2].

Pada umumnya untuk menentukan tingkat kematangan buah nanas dipilih secara manual, hal ini menjadi tidak efisien secara waktu maupun tenaga apabila buah nanas yang dipilih atau diseleksi dalam jumlah yang besar, maka dari itu salah satu cara untuk mempermudah dalam menentukan tingkat kematangan buah nanas adalah dengan menggunakan pendekatan sistem cerdas untuk dapat melakukan identifikasi buah nanas tersebut. Proses untuk menentukan kualitas atau mengidentifikasi suatu objek dapat dibantu dengan sistem yang dapat menentukan kualitas berdasarkan karakteristik yang dimilikinya, salah satu cara untuk menentukan hal tersebut dengan penggunaan metode sistem cerdas. Terdapat beberapa metode dalam sistem cerdas, dalam hal ini metode yang digunakan untuk menentukan identifikasi tingkat kematangan buah nanas adalah *Support Vector Machine (SVM)* dan *k-Nearest Neighbor (k-NN)*.

Penelitian yang dilakukan oleh [3] mengklasifikasikan tingkat kemanisan mangga berdasarkan fitur warna menggunakan SVM dan k-NN mendapatkan hasil dengan tingkat akurasi 87,5% (*SVM*) dan 83,35% (*k-NN*). Pada penelitian tersebut disimpulkan bahwa metode SVM memiliki tingkat akurasi lebih tinggi namun memiliki kelemahan yaitu sangat sulit diaplikasikan dalam jumlah dimensi yang sangat besar sedangkan metode *k-NN* memiliki tingkat akurasi lebih rendah namun lebih mudah untuk diaplikasikan. Kemudian penelitian yang dilakukan oleh [4] tentang penentuan kualitas buah pepaya california menggunakan metode *k-NN* dengan fitur warna R (*red*) dan G (*green*) serta fitur ukuran *mirror axis* dan *major axis* mendapatkan hasil dengan tingkat akurasi sebesar 86,67%, dengan data tersebut menunjukkan penerapan metode *k-NN* menunjukkan performa penentuan kualitas yang baik. Pada penelitian yang dilakukan oleh [5] dengan judul *Selection of Superior Rice Seed Features Using Deep Learning Method* mendapatkan kesimpulan bahwa klasifikasi dengan *k-Nearest Neighbor* memperoleh uji validasi sensitivitas hingga 100%, spesifisitas 87,5% dan akurasi 92,85% pada lipatan ke-7.

Pembentukan model klasifikasi *SVM* dengan kernel polynomial dapat menghasilkan buah nanas yang dipanen memiliki tingkat kematangan yang seragam karena kerja metode konsisten dan algoritma yang digunakan mempunyai tingkat akurasi yang tinggi serta lebih otomatis sehingga dapat menguntungkan berbagai pihak yaitu petani, penjual, dan konsumen [6]. Para petani seringkali kesulitan dalam mengidentifikasi kematangan buah nanas dalam jumlah yang besar dengan kasat mata atau dengan cara yang konvensional hal ini membutuhkan waktu, tenaga serta biaya yang besar.

2. METODE PENELITIAN

Penelitian dilakukan di Daerah Kabupaten Kupang, dimana penelitian ini menggunakan data citra buah nanas yang diambil menggunakan kamera digital. Selain itu juga disarikan dari jurnal – jurnal penelitian sebelumnya mengenai citra digital, metode k-NN dan SVM.

Citra Digital

Citra digital adalah sebuah representasi numerik dari suatu citra dua dimensi (2D) yang termasuk ke dalam jenis vektor atau *bitmap* yang tergantung pada resolusi citra. Citra *bitmap* mempunyai jumlah piksel yang terbatas dan banyaknya piksel di baris dan kolom sebuah citra *bitmap* tidak dapat berubah. Piksel sendiri merupakan sebuah elemen individu terkecil pada sebuah citra serta menyimpan nilai keterangan warna di titik tersebut. apabila citra *bitmap* diperbesar, maka citra terlihat pecah. Pada umumnya, piksel tersimpan di komputer sebagai citra *bitmap* atau *raster map*, berbentuk susunan 2D *integer* kecil serta ditransmisi maupun disimpan dalam bentuk terkompres. Citra digital

dapat dilihat pada perangkat lunak seperti “*Windows photo viewer*”. *Web browser* dapat melihatkan format citra yang standar, seperti *GIF*, *JPEG*, dan *PNG* serta *Web browser* akan menampilkan format *SVG* juga [7].

k-NN (*k-Nearest Neighbor*)

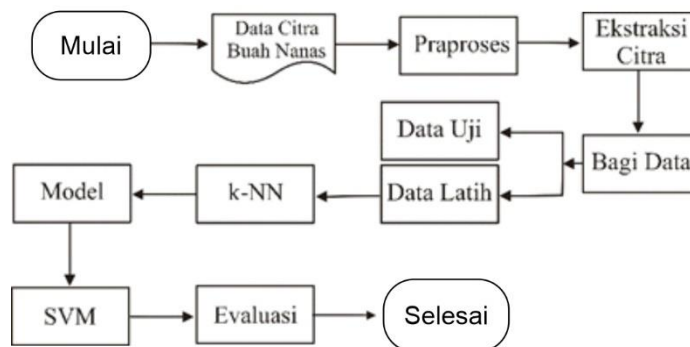
Metode Ekstraksi Ciri Statistik (*Variance*) dan *k-NN* dapat digunakan untuk mengetahui kematangan buah nanas. Berdasarkan nilai grayscale, variance dan hasil pencocokan buah nanasnya berdasarkan hasil nilai euclidean distance [8]. Klasifikasi *k-NN* merupakan metode non parametrik sederhana untuk klasifikasi. Gagasan dalam metode *k-Nearest Neighbor* bertujuan untuk mengidentifikasi sampel *k* dalam set pelatihan yang variabel independen *x* mirip dengan *u* serta menggunakan sampel *k* ini untuk mengklasifikasikan sampel baru ini ke dalam kelas *v*. Terlepas dari kesederhanaan algoritma, kinerjanya sangat baik, dan merupakan metode tolak ukur yang penting. Klasifikasi *k-NN* membutuhkan metrik dan integer positif. Aturan *k-NN* memegang posisi sampel pelatihan dan kelas mereka. Saat memutuskan tentang data masuk baru. Tujuan algoritma ini adalah untuk mengklasifikasikan objek baru berdasarkan nilai atribut dan data latih. Beberapa langkah-langkah klasifikasi algoritma *k-NN* yang pertama adalah tentukan parameter nilai *k* = banyaknya jumlah tetangga terdekat. Kedua, hitung jarak antara data baru dengan semua data training dan yang ketiga, urutkan jarak dan tetapkan tetangga terdekat berdasarkan jarak minimum ke-*k* 4 [9]. *k-Nearest Neighbor (k-NN)* merupakan teknik pengelompokan data baru berdasarkan *K* jarak tetangga terdekat antara informasi latih dan informasi uji. Nilai *K* yang digunakan dalam penelitian ini adalah *Euclidean Distance*.

Identifikasi SVM (*Support Vector Machine*)

SVM (Support Vector Machine) adalah sebuah metode dalam *supervised learning* yang biasa digunakan untuk identifikasi *Support Vector Regression* atau *Support Vector Classification*. Cara kerja algoritma *SVM* adalah dengan menemukan *hyperplane* terbaik dengan cara mencari jarak paling jauh antar setiap kelas. *Hyperplane* dalam *SVM* sendiri itu merupakan sebuah fungsi yang dapat digunakan untuk memisahkan antara satu kelas dengan kelas lainnya. Untuk fungsi dari 2 Dimensi yang akan digunakan pada klasifikasi antar kelas disebut sebagai *line whereas*, sedangkan untuk fungsi dari 3 Dimensi disebut *plane similarly*, dan untuk fungsi yang biasanya dipakai untuk klasifikasi pada ruang dimensi kelas yang lebih tinggi [10]. Konsep dasar dari *Support Vector Machine (SVM)* dapat dijelaskan secara sederhana yaitu usaha untuk mencari *hyperplane* terbaik untuk mengklasifikasikan dua buah kelas pada *input space*. Secara intuitif *hyperplane* terbaik didapatkan dengan mencari bidang pemisah dengan jarak margin maksimal. Margin dapat didefinisikan sebagai jarak antara *hyperplane* dengan data yang paling dekat dengan *hyperplane* dari masing - masing kelas [11]. Data yang paling mendekati *hyperplane* yang akan digunakan untuk mendefinisikan margin disebut sebagai *support vector*. Pada gambar 4 memperlihatkan beberapa pola yang merupakan anggota dari dua buah class + 1 dan -1. Pola yang tergabung pada class -1 disimbolkan dengan warna merah (kotak), sedangkan pola pada class + 1 disimbolkan dengan warna kuning (lingkaran). Problem identifikasi dapat dijelaskan dengan usaha menemukan garis *7 (hyperplane)* yang memisahkan antara kedua kelompok tersebut.

Analisis Data

Prosedur analisis data untuk mengolah data menjadi sebuah informasi yang dapat dimengerti dan dapat menjawab masalah-masalah yang berkaitan dengan kegiatan penelitian identifikasi kematangan buah mangga dengan metode *k-Nearest Neighbor (k-NN)* dan *Support Vector Machine (SVM)* digambarkan dalam bentuk *flowchart* sebagai berikut:



Gambar 1. *Flowchart* Penelitian

1. Data Citra Buah Nanas

Tahapan awal sebelum memproses data citra adalah proses pengumpulan data. Citra buah nanas yang diambil pada penelitian ini sebanyak 100 data citra buah nanas. Buah nanas diletakkan pada kain berwarna hitam kemudian difoto menggunakan kamera *DSLR Canon 700D* dengan jarak 40 cm antara objek buah nanas dan kamera dan diambil

dalam ruangan tanpa cahaya lampu dengan sumber cahaya untuk objek buah nanas dari *softbox*. Pengambilan data citra buah nanas disimpan dengan format warna menggunakan RGB (*red, green, blue*) dan format file *.jpg.

2. Praproses

Tahap praproses merupakan tahap menginformasikan data kesuatu format yang prosesnya lebih mudah dan efektif untuk mendapatkan nilai yang lebih aktual serta mengurangi komputasi warna dalam skala besar. Pada tahapan ini dilakukan *cropping* citra yang mengubah dimensi citra awal agar mendapat ukuran yang lebih kecil dan data yang akan diproses lebih spesifik karena mengurangi data yang tidak terpakai. Setelah tahap *cropping* selesai citra baru akan berukuran 300 x 300 *pixel*.

3. Ekstraksi citra

Ekstraksi ciri menggunakan statistik orde I (*color moment*) yang menampilkan probabilitas kemunculan nilai pixel pada suatu matriks. Matriks citra yang sebelumnya berukuran 300 x 300 *pixel* akan diubah menjadi vector 1x3 dengan proses ekstraksi ciri menggunakan citra HSV (*hue, saturation, value*). Pada tahap ini juga akan menghitung nilai dari masing masing ciri citra. Ciri – ciri yang digunakan pada ekstraksi ini antara lain *mean*, standar deviasi, dan *skewness*. Untuk mendapakan nilai *mean*, standar deviasi, dan *skewness* maka diperlukan rumus:

Standar Deviasi Hue (H)

$$SD(H) = \sqrt{\frac{\sum(Hue\ value - Mean(H))^2}{Total\ number\ of\ values}}$$

Standar Deviasi Saturation (S)

$$SD(S) = \sqrt{\frac{\sum(Hue\ value - Mean(S))^2}{Total\ number\ of\ values}}$$

Standar Deviasi Hue (V)

$$SD(V) = \sqrt{\frac{\sum(Hue\ value - Mean(V))^2}{Total\ number\ of\ values}}$$

Skewness Hue (H) :

$$Skewnes(H) = \frac{3(Mean(H) - Median(H))}{SD(H)}$$

$$Skewnes(S) = \frac{3(Mean(S) - Median(S))}{SD(S)}$$

$$Skewnes(V) = \frac{3(Mean(V) - Median(V))}{SD(V)}$$

Hitung Rata – rata (*Mean*)

$$Mean(H) = \frac{\sum Hue\ values}{Total\ number\ of\ values}$$

$$Mean(S) = \frac{\sum Hue\ values}{Total\ number\ of\ values}$$

$$Mean(V) = \frac{\sum Hue\ values}{Total\ number\ of\ values}$$

4. Bagi Data

Data citra yang sudah melalui tahapan praproses dan ekstraksi ciri kemudian dibagi dengan *k-fold*. Pada tahapan ini data citra dibagi menjadi dua bagian, yaitu data latih dan data uji. Data uji digunakan untuk melakukan pelatihan model, sedangkan data latih digunakan untuk pengujian k-NN dan SVM. Dengan menggunakan *5-fold*, maka metode yang digunakan adalah 5-fold cross validation. Untuk pembagian dataset dibagi menjadi 80% data latih dan 20% data uji.

5. Algoritma *k-NN*

Algoritma *k-Nearest Neighbor (k-NN)* merupakan teknik pengelompokan data baru berdasarkan K jarak tetangga terdekat antara informasi latih dan informasi uji. Nilai K yang digunakan dalam penelitian ini adalah *Euclidean Distance*. Rumus *Euclidean Distance* pada persamaan:

$$d(x, y) = \sqrt{\sum p (x_i - y_i)}$$

Dimana $d(x,y)$: Jarak antara data latih dan uji

X_i : data latih

Y_i : data uji

I : variable data

P : dimensi dat

Nilai Euclidean dihitung dengan

$$\text{Jarak} = \sqrt{(x_2 - x_1)^2 + (y_2 - y_1)^2 + (z_2 - z_1)^2}$$

6. Algoritma SVM

Algoritma SVM dapat menyelesaikan masalah pengklasifikasian dengan mencari pemisah *hyperplane* yang optimal antar dua kelas pada input *space* yang memisahkan kelas negatif dan kelas positif. *Hyperplane* pemisah terbaik antara kedua kelas ditemukan dengan mengukur *margin hyperplane* tersebut dan mencari titik maksimalnya. Subset data latih yang paling dekat ini yang disebut sebagai *support vector*. Langkah – Langkah dalam pengklasifikasian menggunakan SVM antara lain menginputkan hasil ekstraksi ciri kemudian menentukan *hyperplane* terbaik antara kedua kelas. Melalui proses ini akan diperoleh model algoritma dari buah nenas yang matang dan buah nenas yang belum matang.

7. Evaluasi

Evaluasi merupakan penentuan kematangan buah nenas berdasarkan warna citra, dengan memprediksi seberapa baik identifikasi label pada data kelas tupel. Hasil dari identifikasi menggunakan *k-NN* dan SVM akan diuji menggunakan *confusion matrix*, jika data set hanya terdiri dari dua kelas, kelas yang satu dianggap sebagai positif dan kelas lainnya dianggap sebagai negatif. Dengan ketentuan *true positif* (TP) yang merupakan tupel positif yang bernilai benar pada table identifikasi. *True negatif* (TN) merupakan tupel positif yang diartikan sebagai negatif. Sensifitas dan spesifitas disebut sebagai tupel positif yang bernilai benar, sedangkan Spesifitas merupakan tupel negatif yang bernilai benar.

Confusion Matrix merupakan pengukuran performa untuk masalah klasifikasi machine learning dimana keluaran dapat berupa dua kelas atau lebih. *Confusion Matrix* bisa memiliki dua kolom atau lebih yang bergantung pada jumlah kelas. Contoh *confusion matrix* dengan dua kelas dapat dilihat pada Tabel 1.

Tabel 1. *Confusion Matrix*

<i>Predicted Class</i>		<i>Positive</i>	<i>Negative</i>
<i>Positive</i>	TP	FP	
<i>Negative</i>	FN	TN	

Dari Tabel 1 ada empat istilah yang merepresentasikan hasil proses klasifikasi pada *confusion matrix* yaitu: *true positives* yang merupakan jumlah *record positif* yang diklasifikasikan sebagai positif, *false positives* merupakan jumlah *record negatif* yang diklasifikasikan sebagai positif, *true negatives* merupakan jumlah *record negatif* yang diklasifikasikan sebagai negatif, *false negatives* merupakan jumlah *record positif* yang diklasifikasikan sebagai negatif. Kemudian data uji dimasukkan kedalam *confusion matrix* maka akan menghitung jumlah nilai sensitivitas, Spesifitas, akurasi. Sensitivitas digunakan untuk membandingkan jumlah TP terhadap jumlah *record* yang positif, sedangkan Spesifitas melakukan perbandingan jumlah TN terhadap jumlah *record* negatif . Untuk menghitung nilai tersebut dapat menggunakan persamaan sebagai berikut ini.

$$\begin{aligned} \text{Akurasi} &= \frac{TP+TN}{\text{Jumlah seluruh Data}} \times 100\% \\ \text{Sensitivitas} &= \frac{TP}{TP+FN} \\ \text{Spesifitas} &= \frac{TN}{TN+FP} \end{aligned}$$

Keterangan

- TP = jumlah true positives
- TN = jumlah true negatives
- FP = jumlah false positives
- FN = jumlah false negatives

Nilai akurasi didapatkan dari jumlah data bernilai positif yang diprediksi positif dan data bernilai negatif yang diprediksi dibagi dengan jumlah seluruh data didalam dataset. Sedangkan Spesifitas merupakan peluang kasus yang diprediksi positif yang pada kenyataannya termasuk kasus kategori positif. Sensitivitas merupakan peluang kasus dengan kategori positif yang tepat diprediksi positif.

3. HASIL DAN PEMBAHASAN

Hasil dan pembahasan terkait implementasi dan pengujian sistem yang dibangun yaitu identifikasi kematangan buah nenas dengan citra digital menggunakan metode *K-Nearest Neighbor* (k-NN) dan *Support Vector Machine* (SVM)

dengan ekstraksi ciri menggunakan color moment yaitu dengan menghitung rata-rata dari mean, standart deviasi dan skewness. Setelah itu akan dilakukan analisis hasil pengujian menggunakan *confusion matrix* yaitu menghitung akurasi, spesifitasi dan sensitivitas.

Implementasi Sistem

Berikut adalah hasil implementasi sistem berdasarkan rancangan yang telah dibuat sebelumnya:

1. Data Citra

Data citra yang digunakan adalah 200 data citra buah nanas dengan dimensi awal 3456 x 5184 *pixel*. Data citra tersebut akan diinputkan kedalam matlab untuk mendapatkan data matriks yang akan diolah.

2. Praproses Data

a. *Resize*

Pada tahapan ini citra buah nanas akan *dirresize* sehingga ukuran citra menjadi 300x300 *pixel*, sehingga dapat mengurangi beban komputasi. Citra sebelum dan sesudah *resize* dapat dilihat pada Gambar 2a dan Gambar 2b.



a. 3456 x 5184 *pixel* b. 300x300 *pixel*
Gambar 2. *Resize*

b. *Grayscale*

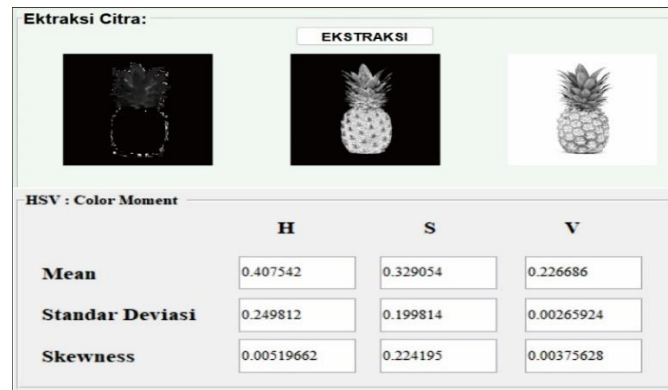
Pada tahapan ini citra buah nanas yang sudah di *resize* akan dirubah warnanya menjadi abu-abu yang fungsinya untuk menyederhanakan model citra. Citra sebelum dan sesudah di *grayscale* dapat dilihat pada Gambar 3a dan Gambar 3b.



a. Sebelum di *grayscale* b. Sesudah di *grayscale*
Gambar 3. Citra *Grayscale*

Ekstraksi Ciri

Pada proses ini citra yang telah melalui tahap praproses akan diekstraksi ciri (*color moment*). Ekstraksi ciri citra dilakukan dengan cara menghitung nilai rata-rata dari ekstraksi citra warna yaitu HSV dan ekstraksi citra bentuk yaitu eccentricity dan matriks. Color Moment menunjukkan probabilitas kemunculan nilai derajat *pixel* pada suatu citra. Matriks citra yang sebelumnya didapatkan dari tahap praproses berukuran 300x300 akan diubah menjadi vektor berukuran 3x3 dengan proses ekstraksi ciri menggunakan citra HSV. Pada tahapan ini juga citra akan menampilkan hasil dari Hue, Saturation, dan Value beserta nilai masing-masing ciri banyaknya data adalah 200 data citra buah mangga sehingga menghasilkan 200 vektor. Vektor inilah yang menjadi dataset masukan untuk proses klasifikasi. Berikut adalah hasil ekstraksi ciri pada Gambar 4.



Gambar 4. Ekstraksi Ciri

Tabel 2. Nilai ekstraksi warna HSV

Data ke-		Hue	Saturation	Value
1	Mean	0.414812	0.325644	0.226601
	Standar Deviasi	0.255086	0.20055	0.224193
	Skewnes	0.00553272	0.00268874	0.00375615
2	Mean	0.468371	0.333667	0.237533
	Standar Deviasi	0.231843	0.16206	0.231456
	Skewnes	0.00415396	0.00141876	0.00413317
3	Mean	0.444587	0.325685	0.237843
	Standar Deviasi	0.243488	0.178279	0.217302
	Skewnes	0.00481185	0.00188876	0.00342035
4	Mean	0.359821	0.358149	0.292215
	Standar Deviasi	0.254522	0.194822	0.226105
	Skewnes	0.00549613	0.00246486	0.00385309
5	Mean	0.437227	0.255652	0.188103
	Standar Deviasi	0.241118	0.133783	0.157308
	Skewnes	0.00467267	0.000798143	0.00129757
...
200	Mean	0.43863	0.3184	0.238156
	Standar Deviasi	0.227231	0.146456	0.187755
	Skewnes	0.00391093	0.00104714	0.00220623

Bagi Data

Pada tahapan ini, data dibagi menggunakan *k-fold cross validation*, k yang digunakan adalah 5, maka metode yang digunakan adalah *5-fold cross validation*. Dataset dibagi menjadi 80% untuk data latih dan 20% untuk data uji. Data latih dan data uji ditampilkan pada tabel 3. Data latih akan disimpan dalam file berbentuk .dat seperti pada Tabel 3.

Tabel 3. 5-fold cross validation

Skenario	Data Latih				Data Test
Skenario 1	F2	F3	F4	F5	F1 20%
	80%				
Skenario 2	F1	F3	F4	F5	F2 20%
	80%				
Skenario 3	F1	F2	F4	F5	F3 20%
	80%				
Skenario 4	F1	F2	F3	F5	F4 20%
	80%				
Skenario 5	F1	F2	F3	F4	F5 20%
	80%				

Identifikasi menggunakan k-NN

Metode *k-Nearest Neighbours* (k-NN) digunakan dalam proses klasifikasi citra dengan berpacu pada hasil ekstraksi fitur yang sebelumnya sudah dilakukan training. Metode ini memilih tetangga terdekat dari dataset training, kemudian menentukan nilai jarak yang terdekat atau nilai jarak terkecil yang akan menghasilkan keluaran klasifikasi. Pada tahapan ini fungsi k-NN adalah menghitung jarak antar data yang ingin diprediksi dengan semua data pelatihan menggunakan metrik jarak atau jarak Euclidean. Euclidean digunakan untuk mengukur seberapa dekat atau berapa besar perbedaan antara atribut dari data yang ingin diprediksi dengan data pelatihan yang ada. Data yang memiliki nilai Euclidean yang lebih kecil cenderung lebih mirip atau dekat dalam ruang atribut seperti pada Tabel 4. Nilai k 1 tingkat akurasi kemiripan lebih tinggi dibandingkan nilai k 3 atau k 7.

Tabel 4. Hasil Perhitungan Identifikasi dengan k-NN

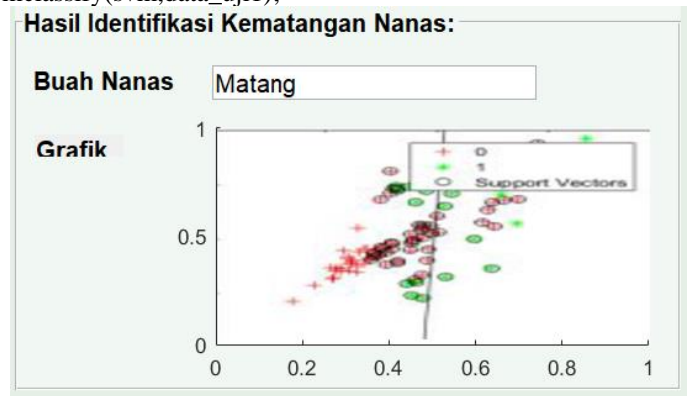
Nilai k	Data yang sesuai	Data keseluruhan	Akurasi
1	140	160	87%
3	120	160	75%
7	95	160	59%

Identifikasi Menggunakan SVM

Pada tahapan ini fitur warna menjadi informasi utama untuk mengidentifikasi kematangan buah nanas. Tahap pembuatan model untuk metode SVM menggunakan data latih dan data uji proses training dan testing. Proses training digunakan untuk menghasilkan model identifikasi dengan SVM yang akan digunakan sebagai acuan untuk mengidentifikasi buah nanas dengan data mentah yang baru. Data penelitian ini mempunyai dimensi vector matriks 200x4 citra buah nanas untuk color moment dan ditambah dengan kelas matriks. SVM bekerja dengan memisahkan dua kelas data dengan mencari garis atau *hyperplane* terbaik yang dapat memaksimalkan margin antara kelas-kelas tersebut. Margin adalah jarak antara garis atau *hyperplane* dan titik-titik terdekat dari masing-masing kelas. Hasil penelitian ini adalah mencari hasil kelas nanas terbaik pada SVM berdasarkan perhitungan jarak yang menentukan apakah berada dikelas matang atau belum matang. Berikut ini adalah *syntax* atau baris perintah pada Matlab untuk melakukan klasifikasi menggunakan SVM.

```
data_uji1 = data(1:20,1:3);
kelas_uji_asli1 = data(1:20,4);
data_latih1 = data(21:100,1:3);
kelas_latih1 = data(21:100,4);
```

```
svm = svmtrain(data_latih1,kelas_latih1);
kelas_uji_hasil1 = svmclassify(svm,data_uji1);
```



Gambar 5. Tampilan *Support Vector Machine (SVM)*

Evaluasi

Evaluasi sistem dilakukan untuk mengetahui seberapa besar akurasi *Support Vector Machine (SVM)* dalam mengidentifikasi buah nanas matang dan tidak matang. Pengujian dengan menggunakan *confusion matrix* untuk mencari tingkat akurasi, spesifitasi dan sensitivitas. Sedangkan *k-fold cross validation* digunakan untuk membagi data latih dan data uji.

Tabel 5. *Fold validation ke-1*

<i>Fold 1</i>			
<i>fold</i>		<i>fold 1</i>	<i>fold 0</i>
	<i>fold 1</i>	140	5
	<i>fold 0</i>	5	10
Akurasi	$\frac{140 + 10}{160} \times 100 = 93\%$		
Sensitivitas	$\frac{140}{140 + 5} \times 100 = 96\%$		
Spesifitasi	$\frac{10}{5 + 10} \times 100 = 66\%$		

Tabel 6. *Fold validation ke-2*

<i>Fold 2</i>			
<i>fold</i>		<i>fold 1</i>	<i>fold 0</i>
	<i>fold 1</i>	120	11
	<i>fold 0</i>	9	20
Akurasi	$\frac{120 + 20}{160} \times 100 = 87\%$		
Sensitivitas	$\frac{120}{120 + 9} \times 100 = 93\%$		
Spesifitasi	$\frac{20}{11 + 20} \times 100 = 64\%$		

Tabel 7. *Fold validation ke-3*

<i>Fold 3</i>			
<i>fold</i>		<i>fold 1</i>	<i>fold 0</i>
	<i>fold 1</i>	105	35
	<i>fold 0</i>	15	5
Akurasi	$\frac{105 + 5}{160} \times 100 = 68\%$		
Sensitivitas	$\frac{105}{105 + 15} \times 100 = 87\%$		
Spesifitansi	$\frac{5}{35 + 5} \times 100 = 12\%$		

Tabel 8. *Fold validation ke-4*

<i>Fold 4</i>			
<i>fold</i>		<i>fold 1</i>	<i>fold 0</i>
	<i>fold 1</i>	123	78
	<i>fold 0</i>	15	14
Akurasi	$\frac{123 + 14}{160} \times 100 = 85\%$		
Sensitivitas	$\frac{123}{123 + 15} \times 100 = 89\%$		
Spesifitansi	$\frac{14}{78 + 14} \times 100 = 15\%$		

Tabel 9. *Fold validation ke-5*

<i>Fold 5</i>			
<i>fold</i>		<i>fold 1</i>	<i>fold 0</i>
	<i>fold 1</i>	97	21
	<i>fold 0</i>	22	21
Akurasi	$\frac{97 + 21}{160} \times 100 = 73\%$		
Sensitivitas	$\frac{97}{97 + 22} \times 100 = 81\%$		
Spesifitansi	$\frac{21}{21 + 21} \times 100 = 50\%$		

Tabel 10. Hasil Evaluasi *Confusion Matrix*

<i>Fold</i>	Akurasi	Sensitivitas	Spesifitansi
<i>Fold 1</i>	93%	96%	66%
<i>Fold 2</i>	87%	93%	64%
<i>Fold 3</i>	68%	87%	12%
<i>Fold 4</i>	85%	89%	15%
<i>Fold 5</i>	73%	81%	50%

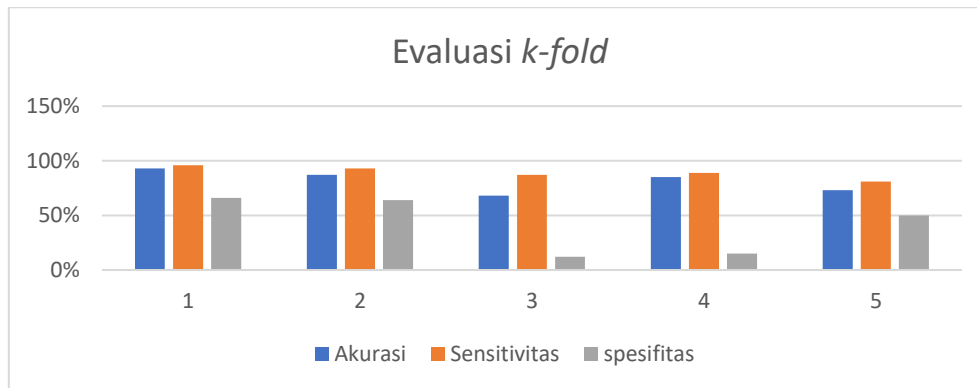
$$\text{Akurasi Sistem} = \frac{\text{Akurasi1} + \text{Akurasi2} + \text{Akurasi3} + \text{Akurasi4} + \text{Akurasi5}}{\text{Total skenario}}$$

$$\text{Akurasi Sistem} = \frac{93+87+68+85+73}{5}$$

$$\text{Akurasi Sistem} = \frac{410}{5}$$

$$\text{Akurasi Sistem} = 81,2\%$$

Berdasarkan Tabel 10, hasil evaluasi dengan *confusion matrix* yang dibagi dalam 5 *fold cross validation* dengan menghitung nilai akurasi, sensitivitas, Spesifitas mendapatkan hasil akurasi sistem sebesar 82%.

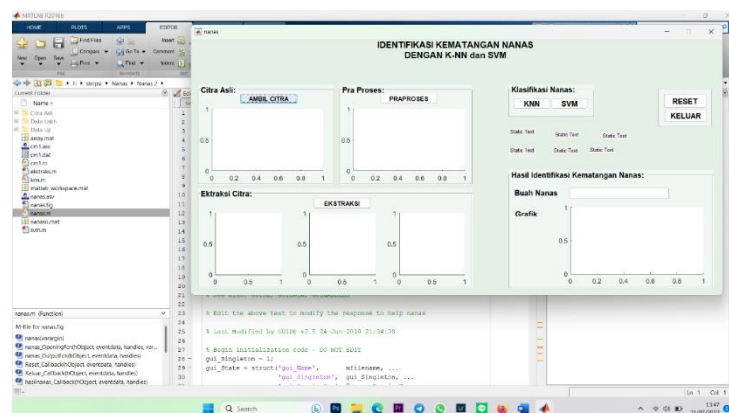


Gambar 6. Grafik Hasil Evaluasi

Dari Gambar 6 di atas terlihat bahwa hasil akurasi terendah ada pada *fold* ke 3 yaitu 68% dan tertinggi pada *fold* ke 1 yaitu 93%. Sedangkan hasil sensitivitas terendah ada pada *fold* ke 5 yaitu 81% dan tertinggi pada *fold* ke 1 yaitu 96% dan hasil spesifitas terendah ada pada *fold* ke 3 yaitu 12% dan tertinggi ada pada *fold* ke 1 yaitu 66%. Berdasarkan hasil evaluasi program dapat disimpulkan bahwa program berhasil dalam mengidentifikasi buah nanas dengan *k-fold* tertinggi dengan tingkat akurasi sebesar 93%, sensitivitas sebesar 96% dan spesifitas sebesar 66%.

Implementasi Antarmuka

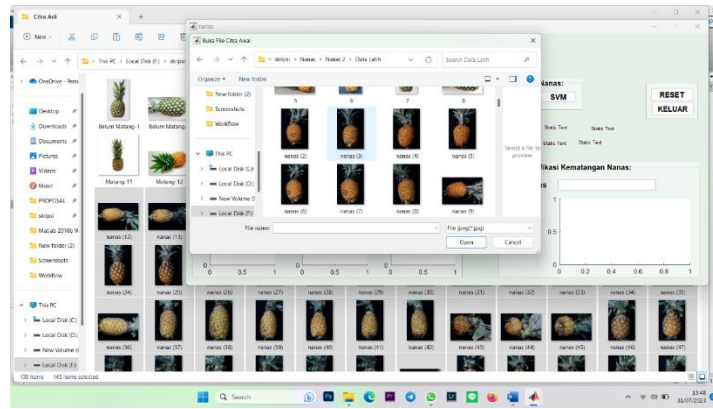
Antarmuka merupakan media penghubung antar sistem dan pengguna. Pengoperasian sistem akan dimulai pada halaman antarmuka sistem ini, sehingga memudahkan pengguna untuk menggunakan aplikasi ini, berikut pada gambar 7 adalah tampilan antarmuka sistem untuk mengidentifikasi kematangan buah nanas menggunakan citra digital dengan metode *k-Nearest Neighbor (k-NN)* dan *Support Vector Machine (SVM)*.



Gambar 7. Layout Antarmuka Sistem

1. Halaman Tampilan Lokasi Pengambilan Citra

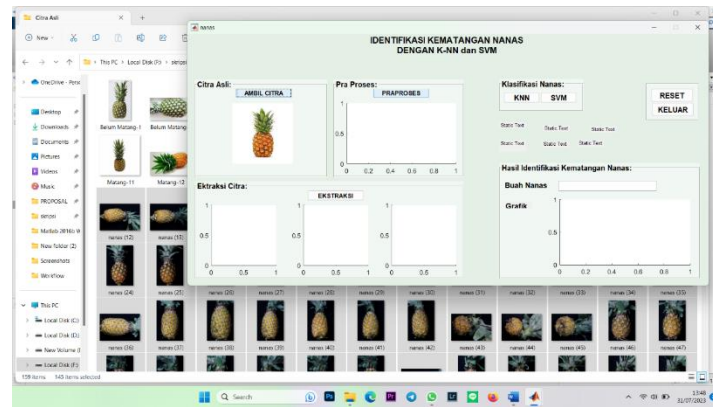
Menampilkan lokasi pengambilan citra yang akan tampil untuk dipilih menjadi citra yang akan diklasifikasi. Tampilan lokasi pengambilan citra dapat dilihat pada Gambar 8.



Gambar 8. *Layout* Lokasi Pengambilan Citra

2. Halaman Tampilan Awal

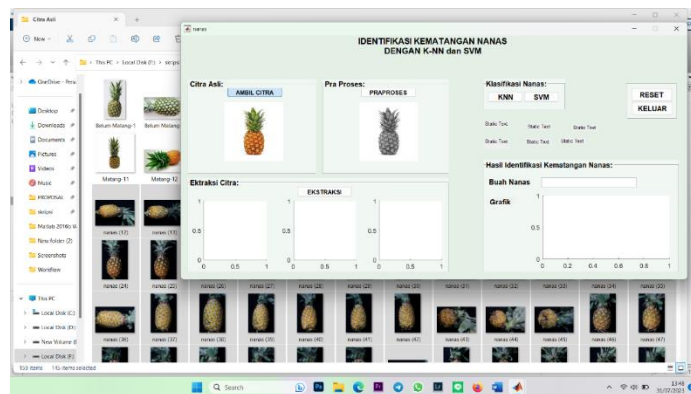
Pada tampilan awal akan menampilkan citra bunga berwarna (RGB). Dari citra awal inilah yang akan diubah ketahap selanjutnya untuk mencari hasil klasifikasi data. Tampilan Awal dapat dilihat pada Gambar 9.



Gambar 9. *Layout* Tampilan Citra Awal

3. Halaman Praproses

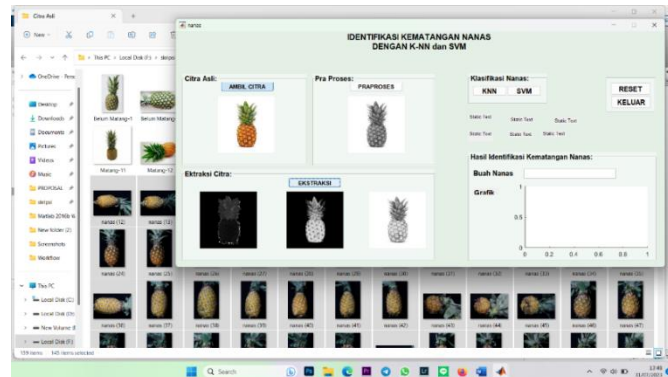
Pada tampilan praproses akan menampilkan hasil perubahan dari citra berwarna ke citra grayscale, lalu setelah sudah melakukan perubahan warna dalam praproses juga akan melakukan resize yaitu mengurangi resolusi gambar citra kain tenun menjadi 300 x 300. Tampilan Halaman Praproses dapat dilihat pada Gambar 10.



Gambar 10. *Layout* Halaman Praproses

4. Halaman Ekstraksi Ciri Citra

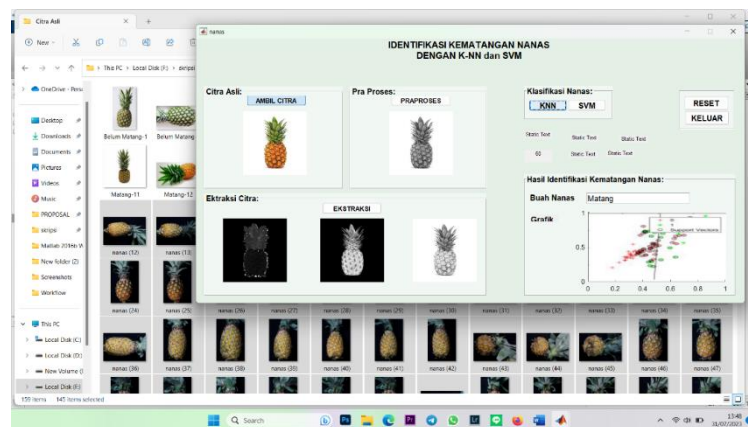
Pada tampilan ekstraksi citra akan menampilkan hasil ekstraksi warna yang dilakukan sistem untuk mendapatkan nilai rata-rata hue, saturation dan value dan ekstraksi bentuk : eccentricity dan matrik dari citra yang telah dilakukan segmentasi. Tampilan Halaman Ekstraksi Ciri Citra dapat dilihat pada Gambar 11.



Gambar 11. Layout Halaman Ekstraksi Ciri

5. Halaman Identifikasi

Pada tampilan identifikasi akan menampilkan hasil identifikasi kematangan buah nanas dengan mencocokkan antar data uji dan data latih, Tampilan Halaman Klasifikasi dapat dilihat pada Gambar 12.



Gambar 12. Layout Halaman Identifikasi

4. SIMPULAN

Berdasarkan hasil penelitian ini dapat disimpulkan bahwa identifikasi kematangan buah nanas dengan k-NN dan SVM berhasil diimplementasikan dengan baik. Hasil identifikasi buah nanas matang dan belum matang dengan *confusion matrix* dengan 5 skenario *cross validation* yang memperoleh nilai akurasi terbaik pada skenario ke-1 sebesar 93% dengan nilai rata-rata akurasi sistem sebesar 81,2%. Dengan hasil penelitian ini diharapkan dapat membentuk model identifikasi saat proses pemanenan dan pendistribusian serta dapat mengurangi selektifitas para petani dalam menentukan buah nanas yang akan dipanen. Meskipun demikian, penelitian ini memiliki batasan, seperti tidak adanya pengklasifikasian jenis-jenis buah nanas sehingga disarankan untuk melakukan penelitian lanjutan dengan memperluas cakupan penelitian atau menggunakan metode yang berbeda.

DAFTAR PUSTAKA

- [1] Badan Pusat Statistik, “Produksi Tanaman Buah-buahan, 2021-2023,” BPS. [Online]. Available : <https://www.bps.go.id/id/statistics-table/2/NjJjMg==/produksi-tanaman-buah-buahan.html> [Diakses: 10-June-2023]
- [2] A. Lustini, dan A.P. Saparudin, “Klasifikasi Tingkat Kematangan Buah Nanas Menggunakan Ruang Warna Red - Green - Blue dan Hue - Saturation - Intensity” *Jurnal Digital Teknolgi Informasi Universitas Muhammadiyah Palembang*, vol. 2, no. 1, pp. 23-31, Jul 2019. doi: 10.32502/digital.v2i1.2283
- [3] M. Ichwan, A.I. Dewi, dan S.Z. Muharom, “Klasifikasi Support Vector Machine (SVM) Untuk Menentukan TingkatKemanisan Mangga Berdasarkan Fitur Warna” *MIND Jurnal*, vol. 3. no. 2, pp.16-24, Des 2018. doi: 10.26760/mindjournal.v3i2.16-23
- [4] E. Muhammad, A. Rivan, M. Arman, dan W. Kennedy, “Penentuan Kualitas Buah Pepaya California Menggunakan Metode K-NN” *Jusikom (Jurnal Sistem Komputer Musirawas)*, vol. 6, no. 1, June 2021, doi: 10.32767/jusikom.v6i1.1175
- [5] D.A. Tonael., Y.R. Kaesmetan dan M.I.J. Lamabelawa, “Selection of Superior Rice Seed Features Using Deep Learning Method”. *IJAIDM (Indonesian Journal of Artificial Intelligence and Data Mining)*, vol. 4. no. 2, pp.78-88, Sept 2021 doi: 10.24014/ijaidm.v2i2. 13947
- [6] D. Veranda, P. Nugraha, A.T.B. Herdiawan, dan Haryati, “Analisis Data Untuk Klasifikasi Tingkat Kematangan Buah Nanas Menggunakan SVM”. *Jurnal Ilmiah Ilmu dan Teknologi Rekayasa*, vol. 4, no. 2, p. 8-17, Sept. 2022. [Online]. Available : <https://ejournal.polsub.ac.id/index.php/jiitr/article/view/104>. [Diakses: 12-Februari-2025]
- [7] K. Kusnadi, R. Dwiza dan M.S. Riana, “Pengelolaan Citra Digital Dengan Menggunakan Metode Transformasi Grayscale dan Pemerataan Histogram”. *Jurnal Teknik Informatika Kaputama*, vol. 6, No. 1, p.108-119. Jan 2022. doi : 10.59697/jtik.v6i1.369
- [8] T. Awik, S. Ardi dan W. Resty. Ekstraksi Ciri Statistik (Variance) dan KNN Untuk Identifikasi Kematangan Citra Buah Nanas, Skripsi, Universitas Nusantara PGRI Kediri, Kediri, Indonesia, 2018.
- [9] P. Putra, M. H. Akim dan S.S. Pardede, “Analisis Metode K-Nearest Neighbour (K-NN) Dalam Klasifikasi Data Iris Bunga”, *Jurnal Teknik KAPUTAMA*, no.6, vol. 1, pp. 297-305, Jan 2022. doi: 10.59697/jtik.v6i1.401
- [10] E.O.M. Laer, “Klasifikasi Kematangan Buah Mangga Menggunakan Support Vector Machine (SVM)”, Skripsi, STIKOM Uyelindo Kupang, Kota Kupang, Indonesia, 2022.
- [11] M. Fathurrohman. “Diagnosa Kerusakan Bantalan Bola Menggunakan Metode Support Vector Machine”, Skripsi, Universitas Sebelas Maret, Surakarta, Indonesia, 2019.