

BAB II

TINJAUAN PUSTAKA

1.1 Penelitian Terkait

Penelitian terdahulu ini menjadi salah satu acuan penulis dalam melakukan penelitian sehingga penulis dapat memperkaya teori yang digunakan dalam mengkaji penelitian yang dilakukan. Dari penelitian terdahulu, penulis tidak menemukan penelitian dengan judul yang sama seperti judul penelitian penulis. Namun penulis mengangkat beberapa penelitian sebagai referensi dalam memperkaya bahan kajian pada penelitian penulis. Berikut merupakan penelitian terdahulu berupa beberapa jurnal terkait dengan penelitian yang dilakukan penulis adalah sebagai berikut.

Penelitian dilakukan oleh Pratama, dkk 2022 [1]. Pada penelitian ini untuk mengklasifikasi tingkat kematangan buah tomat dengan menggunakan metode K-NN peneliti mengambil citra buah tomat mawar secara langsung dengan memakai kamera Samsung A20s dengan resolusi 13 megapiksel. Ada total 150 dataset yang dikumpulkan dengan rincian 50 citra matang, 50 citra mengkal dan 50 citra mentah. Selanjutnya dataset citra dipisah masing-masing diambil 30 citra matang, 30 citra mengkal, 30 citra mentah yang akan dijadikan sebagai data latih untuk pembuatan model klasifikasi dan 20 citra matang, 20 citra mengkal, 20 citra mentah yang akan dijadikan sebagai data testing untuk menguji model klasifikasi yang telah dibuat.

Data akan diuji menggunakan *library OpenCV* dengan model klasifikasi K-NN. Adapun sampel data latih yakni seperti gambar dibawah ini.

Penelitian ini dilakukan oleh Humaira, dkk 2021 [2] dengan judul “Klasifikasi tingkat kualitas dan kematangan buah tomat berdasarkan fitur warna menggunakan jaringan saraf tiruan”. Pada penelitian buah tomat ini, tingkatan kematangan buah tomat yang diteliti ialah 4, mentah, mengkal, masak dan busuk. Dimana setiap tingkatannya itu terdapat 30 citra, sehingga jumlah dataset yang digunakan sebanyak 120 buah citra. Ruang warna yang digunakan pada tahap processing citra ialah RGB kemudian ditransformasikan ke LAB, dimana pada LAB ini kami hanya menggunakan 2 ruang warna yaitu A dan B. Penggunaan 2 ruang warna yaitu *a dan *b bertujuan agar citra yang akan disegmentasi itu hasilnya lebih maksimal. Ruang warna a* dan b* untuk komponen warna hijau-merah dan biru-kuning. Dengan menggunakan ruang warna *a dan *b akan menghasilkan background berwarna purple cenderung merah muda atau pink (magenta) dan warna yang dihasilkan pada objek adalah warna biru. Sehingga, apabila digunakan citra dengan ruang warna seperti ini maka akan di peroleh segmentasi citra yang maksimal.

Penelitian ini dilakukan oleh Bagaskara, dkk 2023 [3] dengan judul “Identifikasi Tingkat Kematangan Buah Tomat dengan Citra Warna Berdasarkan Warna Kulit Buah”. Pengujian dilakukan dengan menguji fungsi utama dari aplikasi yaitu fungsi ekstraksi HSI dan klasifikasi KNN, dengan melakukan perhitungan manual maupun dengan perhitungan dalam sistem aplikasi. Jika sistem dapat menghitung hasil tes yang dilakukan sama atau mendekati angka perhitungan

manual maka bisa dikatakan pengujian berhasil. Data uji yang digunakan dalam pengujian menggunakan nilai mean H, mean S, mean I, dan pengujian nilai K pada KNN sesuai dengan yang ditentukan. Penelitian pada pengujiannya nilai K yang digunakan adalah 3, 5, dan 7. Pengujian dilakukan 10 kali disetiap masing masing kategori citra gambar. Adapun kategori (kelas) datanya ada 3, yaitu matang, setengah matang, dan mentah.

Penelitian ini dilakukan oleh Riska, dkk 2016 [4] dengan judul “klasifikasi level kematangan buah tomat berdasarkan fitur warna menggunakan MULTI-SVM”. Data yang digunakan pada penelitian ini sebanyak 108 citra tomat yang dibagi menjadi 66 data training dan 42 data testing. Pengambilan Citra dilakukan dengan cara mengambil foto buah tomat pada salah satu sisinya untuk mewakili level kematangan tomat. Citra tomat yang digunakan diambil dengan kamera smartphone dengan posisi tegak lurus terhadap objek. Hal tersebut digunakan untuk menghindari adanya efek bayangan pada permukaan citra. Dataset yang digunakan pada penelitian ini merupakan citra buah tomat yang dapat dikelompokkan menjadi 6 level.

Sebuah penelitian dilakukan oleh Gilvy dkk (2021) [3], yang berjudul “Sistem Informasi Persediaan Stok Bahan Baku Toko Kopi Dengan Menggunakan Metode *Waterfall*”. Hasil dari penelitian ini yaitu sistem persediaan bahan baku dapat menunjang kegiatan bisnis kopi secara berkelanjutan dengan mengontrol jumlah persediaan bahan baku sesuai kebutuhan yang diperlukan.

Penelitian yang dilakukan oleh Nur Hidayati (2019) [4], yang berjudul “Penggunaan Metode *Waterfall* Dalam Rancang Bangun Sistem Informasi

Penjualan”. Hasil dari penelitian ini adalah Metode yang digunakan dalam rancang bangun sistem ini menggunakan *Waterfall*. Dengan penggunaan metode ini, maka akan lebih mempermudah dalam pembuatan rancang bangun sistem penjualan.

Penelitian lain yang dilakukan Ibnu Lesmono (2018) [5], yang berjudul “Rancang Bangun Sistem Informasi Penjualan Sepatu Berbasis *Website* dengan menggunakan metode *Waterfall*”. Hasil dari penelitian ini pengembangan *website* untuk mengatasi persaingan bisnis sekarang ini khususnya penjualan sepatu, dalam melakukan kegiatan transaksi mulai dari proses order, menyimpan data produk, mengubah data produk serta menghapus data produk sehingga menghasilkan informasi yang cepat, tepat, akurat dan menarik bagi pelanggan.

Sebuah penelitian yang dilakukan oleh Yahya dkk (2019) [6], yang berjudul “Sistem Informasi Penjualan Tiket Wisata Berbasis *Web* menggunakan metode *Waterfall*”. Penelitian ini menghasilkan sebuah sistem informasi Penjualan Tiket Wisata Berbasis *Web* Menggunakan Metode *Waterfall*.

Penelitian terakhir oleh tara dkk (2022) [7], yang berjudul “Sistem Informasi Penjualan Pakaian Berbasis *Web* Pada Toko Kolor Murah Ungaran Menggunakan Metode *Waterfall*”. Hasil dari penelitian ini yaitu Berdasarkan hasil uji coba sistem *black box* menghasilkan fitur yang layak dan uji coba *usability* pada data kuesioner menghasilkan 84,0% yang berarti hasil pengujian tersebut sangat layak dan memenuhi aspek *usability*.

1.2 Landasan Teori

1.2.1 Buah Tomat

Tanaman tomat tergolong tanaman semusim (*annual*). Artinya, tanaman berumur pendek yang hanya satu kali berproduksi dan setelah itu mati. Tanaman ini berbentuk perdu atau semak, mempunyai batang lunak dan bulat. Secara taksonomi tanaman tomat digolongkan sebagai berikut:

<i>Kingdom</i>	: <i>Plantae</i>
<i>Divisio</i>	: <i>Magnoliophyta</i>
<i>Kelas</i>	: <i>Magnoliopsida</i>
<i>Subkelas</i>	: <i>Asteridae</i>
<i>Ordo</i>	: <i>Solanales</i>
<i>Famili</i>	: <i>Solanaceae</i>
<i>Genus</i>	: <i>Solanum</i>
<i>Species</i>	: <i>Solanum lycopersicum</i>

Tanaman tomat merupakan tanaman herba semusim dari keluarga *Solanaceae*.

Batang tanaman tomat bervariasi ada yang tegak atau menjalar, padat dan merambat, berwarna hijau, berbentuk silinder dan ditumbuhi rambut-rambut halus terutama dibagian yang berwarna hijau. Daunnya berbentuk oval dan bergerigi dan termasuk daun majemuk. Daun tanaman tomat biasanya berukuran panjang sekitar 20 – 30 cm serta lebarnya 16 – 20 cm. Daun tanaman tomat memiliki jarak yang dekat dengan ujung dahan sementara tangkai daunnya berbentuk bulat berukuran 7 – 10 cm. Bunga tomat berwarna kuning cerah, termasuk *hermaprodit* dan dapat

menyerbuk sendiri. Tanaman tomat memiliki akar tunggang dengan akar samping yang menjalar ke samping. Berikut tanaman tomat ditunjukkan pada gambar 2.1



Gambar 2.1 Buah Tomat

Tomat adalah buah dari bagian *tanamansolanum lycopersicum* yang biasa dikenal sebagai tanaman tomat dan termasuk ke dalam keluarga *Solanaceae*, merupakan tanaman asal Amerika dan telah menyebar ke wilayah Eropa. Nama lain buah tomat di berbagai negara diantaranya buah tomat (*Spanyol*), Nahuatl (bahasa *Aztec*), tomat pertama kali dikenal pada tahun 1595[19].

1.2.2 Citra Digital

1.2.2.1 Konsep Dasar

Citra *digital* adalah sebuah fungsi 2D, $f(x,y)$, yang merupakan fungsi intensitas cahaya, dimana nilai x dan y merupakan koordinat spasial dan nilai fungsi di setiap titik (x,y) merupakan tingkat keabuan citra pada titik tersebut. Citra *digital* dinyatakan dengan sebuah matriks dimana baris dan kolomnya menyatakan suatu titik pada citra tersebut dan elemen matriksnya (yang disebut sebagai elemen gambar atau piksel) menyatakan tingkat keabuan pada titik tersebut. Matriks dari citra digital berukuran $N \times M$ (tinggi \times lebar), dimana $N =$ jumlah baris $0 < y \leq N -$

1 $M =$ jumlah kolom $0 \leq x \leq M - 1$ $L =$ derajat keabuan $0 \leq f(x,y) \leq L - 1$. Dimana indeks baris (x) dan indeks kolom (y) menyatakan suatu koordinat titik pada citra, sedangkan $f(x,y)$ merupakan intensitas (derajat keabuan) pada titik (x,y).

Berdasarkan jenisnya, citra digital dapat dibagi menjadi 3 (Sutoyo, 2009), yaitu

1. Citra Biner (Monokrom)

Memiliki 2 buah warna, yaitu hitam dan putih. Warna hitam bernilai 1 dan warna putih bernilai 0. Untuk menyimpan kedua warna ini dibutuhkan 1 bit di memori.

2. Citra *Grayscale*

Citra *Grayscale* mempunyai kemungkinan warna hitam untuk nilai minimal dan warna putih untuk nilai maksimal. Banyaknya warna tergantung pada jumlah bit yang disediakan di memori untuk menampung kebutuhan warna tersebut. Semakin besar jumlah bit warna yang disediakan di memori, maka semakin halus gradasi warna yang terbentuk.

3. Citra Warna (*True Color*)

Setiap piksel pada citra warna mewakili warna yang merupakan kombinasi tiga warna dasar, yaitu merah, hijau, dan biru ($RGB = Red, Green, Blue$). Setiap warna dasar menggunakan penyimpanan 8 bit = 1 *byte* (nilai maksimum 255 warna), jadi satu piksel pada citra warna diwakili oleh 3 *byte*.

1.2.2.2 Ruang Warna

Definisi ruang warna (*color space*) adalah sistem koordinat atau model matematis yang digunakan untuk mempresentasikan warna dalam bentuk angka. Ruang warna mendeskripsikan bagaimana warna dan diinterpretasikan oleh perangkat seperti monitor, kamera, printer, dan perangkat lunak grafis. Ruang warna menentukan bagaimana warna di tampilkan atau direproduksi dalam media digital maupun cetak.

1.2.2.3 Konsep Ruang Warna

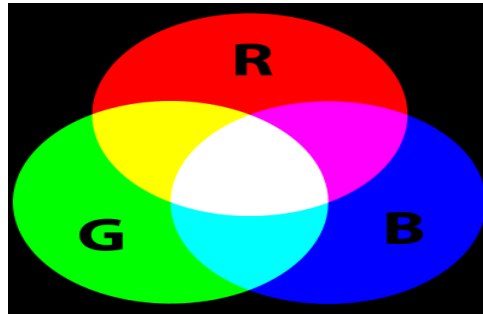
Konsep ruang warna adalah sebuah sistem atau model yang digunakan untuk menggambarkan dan mewakili warna secara numerik. Ini memungkinkan untuk memetakan warna ke dalam sistem koordinat, misalnya dalam bentuk angka, dan mengelola warna dengan lebih akurat dalam berbagai aplikasi, seperti desain grafis, fotografi, dan perangkat lunak.

Beberapa ruang warna tersebut antara lain adalah sebagai berikut:

1. RGB (*Red Green Blue*)

Model warna RGB adalah model warna yang berdasarkan pada konsep penambahan kuat cahaya primer yaitu *Red*, *Green* dan *Blue*. Dalam suatu ruang yang sama sekali tidak ada cahaya, maka ruangan tersebut adalah gelap total [11]. Pada perhitungan program-program komputer model warna direpresentasi dengan nilai komponennya, seperti dalam RGB (r, g, b) masing-masing nilai antara 0 hingga 255 (intensitas RGB) sesuai dengan tugas masing-masing yaitu pertama *Red*, kedua *Green* dan ketiga *Blue* dengan demikian masing-masing komponen ada 256 tingkat (*level*). Apabila dikombinasikan maka ada 256x256x256 atau

16.777.216 kombinasi warna RGB yang dapat dibentuk. Ruang warna RGB dapat dilihat pada Gambar 2.2 berikut.



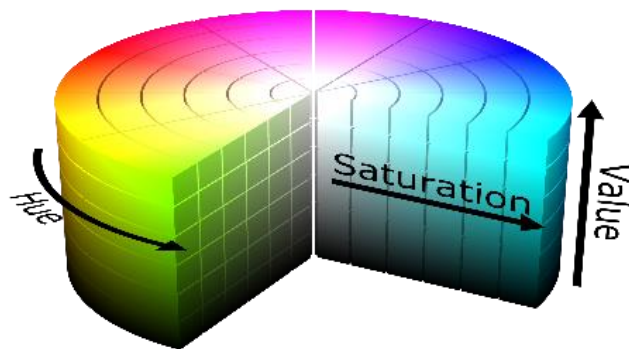
Gambar 2.2 Ruang Warna RGB

RGB adalah ruang warna aditif yang bermakna bahwa semua warna dimulai dari hitam dan dibentuk dengan menambah R, G dan B. Setiap warna yang tampak merupakan kombinasi dari tiga komponen R, G dan B. Ruang warna RGB adalah ruang warna berdasarkan konsep penambahan kuat cahaya primer yaitu Red, Green dan Blue. Ruang yang gelap total berarti tidak ada sinyal gelombang cahaya yang diserap oleh mata atau RGB (0,0,0). Bila ditambahkan cahaya merah maka ruangan akan berubah menjadi merah dan semua benda menjadi berwarna merah, misalnya RGB (255,0,0). Ruang warna standar yang didasarkan pada hasil akuisisi frekuensi warna oleh sensor elektronik yang bentuk keluarannya berupa sinyal analog adalah Ruang warna RGB (Red, Green, Blue), yang amplitudonya di digitalisasikan dan dikodekan dalam 8 bit untuk setiap warnanya.

2. HSV (*Hue Saturation Value*)

Model warna HSV mendefinisikan warna dalam terminology *Hue*, *Saturation* dan *Value*. *Hue* menyatakan warna yang sebenarnya, seperti merah, violet, dan kuning. *Hue* digunakan untuk membedakan warna-warna dan menentukan kemerahan (*redness*), kehijauan (*greenness*), dari cahaya. *Hue*

berasosiasi dengan Panjang gelombang cahaya. Saturation menyatakan tingkat kemurnian atau warna [12]. *Value* merupakan atribut yang menyatakan banyaknya cahaya yang diterima oleh mata tanpa memperdulikan warna. Ruang warna HSV dapat dilihat pada Gambar 2.3 berikut.



Gambar 2.3 Ruang Warna HSV

Warna HSV adalah hasil persepsi dari cahaya dalam spektrum wilayah yang terlihat oleh retina mata, dan memiliki panjang gelombang antara 400nm sampai dengan 700nm. Keuntungan HSV adalah terdapat warna-warna yang sama dengan yang ditangkap oleh indra manusia. Sedangkan warna yang dibentuk model lain seperti RGB merupakan hasil campuran dari warna-warna primer [21].

3. HSI (*Hue Saturation Intensity*)

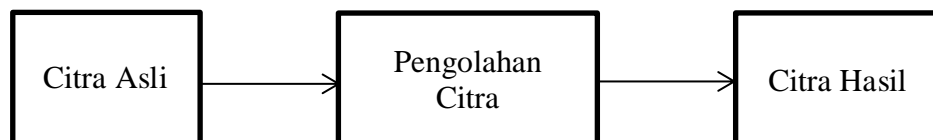
HIS merupakan model warna yang berfungsi untuk mendefinisikan warna dalam terminology *Hue*, *Saturation* dan *Intensity*. *Hue* menyatakan warna sebenarnya, seperti merah, violet, dan kuning juga digunakan untuk membedakan warna-warna dan menentukan kemerahan (*rednrss*), kehijauan (*greenrss*) serta warna lainnya dari cahaya. *Hue* berasosiasi dengan Panjang gelombang cahaya. *Saturation* menyatakan tingkat kemurnian suatu warna, yaitu dengan mengindikasikan seberapa banyak warna putih diberikan pada warna. Sedangkan

intensity adalah atribut yang menyatakan banyaknya cahaya yang diterima oleh mata tanpa memperdulikan ruang warna [13].

1.2.2.4 Pengolahan Citra

Pengolahan citra (*image processing*) merupakan proses mengolah piksel-piksel didalam citra digital untuk memperbaiki kualitas citra. Namun dengan berkembangnya dunia komputasi yang ditandai dengan semakin meningkatnya kapasitas dan kecepatan proses komputer serta munculnya ilmu-ilmu komputasi yang memungkinkan manusia dapat mengambil informasi dari suatu citra [13]. Proses pengolahan citra secara diagram proses dimulai dari pengambilan citra, perbaikan kualitas citra, sampai dengan pernyataan representatif citra yang dicitrakan.

Operasi pengolahan citra dilakukan untuk mentransformasi suatu citra menjadi citra lain.



Gambar 2.4 Alur Pengolahan Citra

Terdapat satu hal yang penting dalam konsep pengolahan citra yaitu mengekstrak informasi dari citra tersebut, sebagai contoh seperti pengenalan berdasarkan bentuknya. Salah satu metode pengolahan citra yaitu *edge detection*, yaitu pemrosesan citra terhadap bidang *feature detection* dan *feature extraction*. *Edge* merupakan batas-batas dalam sebuah objek yang digunakan untuk mendeteksi dan mengidentifikasi tepi dalam citra digital [14].

1.2.2.5 Pengolahan Citra Digital

Proses pengolahan data teks berbeda dengan pengolahan citra. Dalam pengolahan citra ini dilakukan dengan beberapa teknik untuk memanipulasi citra agar dapat diketahui perbedaan atau ciri khas yang membedakan antara citra yang satu dengan citra yang lain. Pengolahan citra adalah proses memanipulasi citra dengan mesin komputer dengan maksud agar kualitas citra tersebut menjadi lebih baik [13]. Terdapat satu hal yang penting dalam konsep pengolahan citra yaitu mengekstrak informasi dari citra tersebut, sebagai contoh seperti pengenalan berdasarkan bentuknya. Salah satu metode pengolahan citra yaitu *edge detection*, yaitu pemrosesan citra terhadap bidang *feature detection* dan *feature extraction*. *Edge* merupakan batas-batas dalam sebuah objek yang digunakan untuk mendeteksi dan mengidentifikasi tepi dalam citra *digital* [14].

1. Dasar-dasar Pengolahan Citra

Pengolahan citra merupakan sebuah metode hasil pengembangan pemahaman dasar visual manusia. Dalam prosesnya, pengolahan citra meliputi beberapa proses dasar seperti pengambilan citra, image sampling dan kuantisasi.

- a. Pengambilan Citra, dilakukan menggunakan *array* CCD. *Array* CCD (*Charge Couple Device*) banyak diaplikasikan pada kamera *digital*, yang berfungsi sebagai sensor optik pada kamera *digital*. Sensor ini

memiliki hasil yang proporsional terhadap integral energi cahaya yang diproyeksikan ke permukaan sensor.

- b. *Image Sampling* dan Kuantisasi. *Sampling* merupakan proses pendigitalisasian nilai koordinat, sedangkan kuantisasi merupakan pendigitalisasian nilai *amplitude*. *Sampling* menyatakan besarnya kotak-kotak yang disusun dalam baris dan kolom atau dengan kata lain besar kecilnya ukuran *pixel* pada citra, sedangkan kuantisasi menyatakan besarnya nilai tingkat kecerahan yang dinyatakan dalam tingkat keabuan sesuai dengan jumlah *bit biner* yang dipakai.

2. Operasi Pengolahan Citra

Pengolahan citra pada dasarnya dilakukan dengan cara memodifikasi setiap titik dalam citra tersebut sesuai keperluan. Secara garis besar, modifikasi tersebut dikelompokkan menjadi enam yaitu:

- a. Operasi titik, pada operasi ini setiap titik diolah secara tidak gayut terhadap titik yang lain.
- b. Operasi global, karakteristik global biasanya berupa sifat statistik dari citra yang digunakan untuk memodifikasi nilai setiap titik.
- c. Operasi temporal/berbasis bingkai, sebuah citra diolah dengan cara dikombinasikan dengan citra lain.
- d. Operasi geometri, dimana bentuk, ukuran, atau orientasi citra dimodifikasi secara geometris.

- e. Operasi banyak titik bertetangga, pada operasi ini data dari titik-titik yang bersebelahan (bertetangga) dengan titik yang ditinjau ikut berperan dalam mengubah nilai.

1.2.3 Ekstraksi Fitur

Fitur adalah objek yang memiliki karakteristik atau sifat. Sedangkan ekstraksi fitur merupakan proses pengambilan nilai fitur atau ciri dari sebuah gambar. Objek dalam citra dibutuhkan parameter-parameter yang bisa mencirikan objek tersebut. Ekstraksi ciri merupakan proses pengambilan ciri atau karakteristik objek yang dapat digunakan sebagai pembeda dari objek-objek lainnya [14]. Karakteristik inilah yang dipakai sebagai parameter untuk menggambarkan sebuah objek.

Salah satu kategori fitur yang umum diekstraksi adalah fitur bentuk, yang menggambarkan sifat geometris dari objek dalam citra. Komponen utama dari fitur bentuk meliputi area, perimeter, dan bentuk relatif. Area mencerminkan luas dari suatu objek dalam citra dan memberikan informasi mengenai ukuran objek, sedangkan perimeter menunjukkan panjang garis batas objek dan berkaitan dengan kompleksitas bentuk objek. Fitur bentuk relatif seperti kebulatan, aspek rasio, dan eksentrisitas digunakan untuk menggambarkan struktur dan proporsi objek secara lebih mendetail. Ekstraksi bentuk juga dapat dilakukan melalui identifikasi kontur objek menggunakan algoritma seperti Canny atau metode flood fill, serta melalui penggunaan deskriptor bentuk seperti momen invarian Zernike yang tetap konstan terhadap transformasi geometris. Salah satu metode populer dalam deteksi bentuk adalah transformasi Hough, yang mampu mendeteksi garis atau lingkaran dengan mentransformasikan representasi spasial ke domain parameter.

Selain bentuk, fitur tekstur juga merupakan informasi penting yang mengekspresikan distribusi spasial dari intensitas piksel. Fitur tekstur berguna untuk membedakan objek berdasarkan pola permukaan atau keteraturan visualnya. Metode ekstraksi tekstur yang umum digunakan adalah matriks ko-occurrence tingkat abu-abu (GLCM), yang menghitung hubungan spasial antar pasangan piksel untuk menghasilkan fitur seperti energi, kontras, homogenitas, dan korelasi. Alternatif lain adalah matriks semivariogram, yang mengukur variabilitas spasial piksel, serta transformasi wavelet, yang memungkinkan analisis tekstur dalam berbagai skala dan arah. Selain itu, deskriptor statistik seperti rata-rata, standar deviasi, skewness, dan kurtosis juga dapat digunakan untuk mengekstraksi ciri tekstur secara sederhana.

Fitur warna merupakan aspek visual yang kuat dalam analisis citra, terutama dalam klasifikasi dan segmentasi objek. Informasi warna dapat direpresentasikan dalam berbagai sistem warna seperti RGB, HSV, Lab, atau Luv. Salah satu metode yang paling umum adalah menggunakan histogram warna, yaitu distribusi frekuensi intensitas warna dalam citra yang menggambarkan dominasi dan penyebaran warna. Fitur lain seperti rata-rata warna dan standar deviasi warna juga dapat digunakan sebagai deskriptor warna yang menggambarkan karakteristik warna secara numerik. Ekstraksi fitur warna biasanya dimulai dengan transformasi ke sistem warna yang sesuai, kemudian dihitung histogram atau deskriptor lainnya dari masing-masing saluran warna.

Fitur spasial mencakup informasi tentang posisi dan distribusi spasial objek dalam citra. Ekstraksi fitur spasial dapat dilakukan melalui transformasi geometris

seperti transformasi Fourier atau transformasi kosinus diskrit (DCT) untuk menganalisis intensitas piksel dalam domain frekuensi. Selain itu, terdapat deskriptor spasial seperti histogram orientasi gradien (HOG), yang menghitung distribusi arah gradien piksel dan sangat berguna untuk mendeteksi tepi atau fitur lokal. Deteksi tepi sendiri merupakan metode penting yang digunakan dalam ekstraksi spasial, biasanya menggunakan operator seperti Sobel atau Canny untuk mengidentifikasi batas objek yang tajam.

1.2.4 K-Nearest Neighbour (K-NN)

K-Nearest Neighbour (K-NN) adalah suatu metode yang menggunakan algoritma *supervised* dimana hasil dari sampel uji yang baru diklasifikasikan berdasarkan mayoritas dari kategori pada K-NN. Tujuan dari algoritma ini adalah mengklasifikasi objek baru berdasarkan atribut dan sampel latih. Pengklasifikasian tidak menggunakan model apapun untuk mencocokkan dan hanya berdasarkan pada memori.

Diberikan titik uji, akan ditemukan sejumlah K objek (titik *training*) yang paling dekat dengan titik uji [16]. Klasifikasi menggunakan mvoting terbanyak di antara klasifikasi dari K objek. Algoritma K-NN menggunakan klasifikasi ketetanggaan sebagai nilai prediksi dari sampel uji yang baru. Dekat atau jauhnya tetangga biasanya dihitung berdasarkan jarak *Euclidean*. Langkah-langkah untuk menghitung algoritma *K-Nearest Neighbour*, yaitu:

1. Menentukan nilai K.
2. Menghitung jarak ketetanggaan terdekat, rumus yang digunakan adalah jarak *Euclidean*:

$$D_{(x,y)} = \sqrt{\sum_{i=1}^n (x_i - y_i)^2} \quad (2.1)$$

Keterangan:

x_i = data uji atau data training

y_i = data latuh atau data testing

i = record (baris) ke-I dari tabel

n = Jumlah data training

K-Nearest Neighbor (K-NN) memiliki kelebihan yaitu dapat menghasilkan data yang kuat atau jelas dan efektif jika digunakan pada data yang besar. *K-Nearest Neighbor* juga memiliki kekurangan yaitu membutuhkan nilai k , jarak dari data percobaan tidak jelas dengan tipe jarak yang digunakan, untuk memperoleh hasil yang terbaik, maka harus menggunakan semua atribut atau hanya satu atribut yang telah pasti [20].

Langkah-langkah menghitung algoritma K-NN antara lain:

1. Menentukan parameter K (jumlah tetangga terdekat)
2. Menghitung kuadrat jarak *Euclidean* (*query instance*) pada setiap objek terhadap data sampel yang diberikan.
3. Selanjutnya mengurutkan hasil no 2 secara *ascending* (berurutan dari nilai paling tinggi kerendah)
4. Mengumpulkan klasifikasi *nearest neighbor* atau kategori y .

1.2.5 Python

Python bahasa pemrograman yang memiliki struktur data tingkat tinggi yang efisien dan pendekatan yang sederhana namun efektif untuk digunakan pada pemrograman berorientasi objek. Sintaksis dari Python memiliki tipe dinamis,

dengan sifatnya yang diinterpretasikan, menjadikan bahasa yang ideal untuk skrip dan pengembangan aplikasi yang tepat di banyak area di Sebagian besar platform. Interpreter Python mudah dikembangkan dengan fungsi dan tipe data baru yang diimplementasikan dalam struktur pemrograman C atau C++ (atau bahasa lain yang bisa dipanggil dari C).

1.2.6 *Confusion Matrix*

Akurasi adalah perhitungan yang digunakan untuk mendapatkan nilai hasil dari proporsi jumlah prediksi yang benar. Metode pengujian yang digunakan untuk menguji tingkat akurasi, yaitu metode Confusion Matriks yang disebut dengan *error matrix*. *Confusion Matriks* ini pada dasarnya memberikan informasi perbandingan hasil klasifikasi yang dilakukan oleh sistem (model) dengan hasil klasifikasi sebenarnya [17]. *Confusion Matriks* digambarkan dengan tabel yang menyatakan jumlah data uji yang bernilai benar dan yang bernilai salah yang diklasifikasikan. Istilah srepresentasi hasil proses klasifikasi pada *Confusion Matrix* dapat dilihat pada Tabel 2.1 berikut.

Tabel 2.1 *Confusion Matrix*

<i>Correct Classification</i>	<i>Classified as</i>	
	<i>Predicted “ + “</i>	<i>Predicted “ - “</i>
<i>Actual “ + “</i>	<i>True Positives</i>	<i>False Negatives</i>
<i>Actual “ - “</i>	<i>False Positives</i>	<i>True Negatives</i>

Keterangan:

1. *True Positives* (TP): jumlah *record* data *positive* yang diklasifikasikan sebagai nilai *positive*.

2. *False Positives* (FP): jumlah *record* data *negative* yang diklasifikasikan sebagai nilai *positive*.
3. *False Negatives* (FN): jumlah *record* data *positive* yang diklasifikasikan sebagai nilai *negative*.
4. *True Negatives* (TN): jumlah *record* data *negative* yang diklasifikasikan sebagai nilai *negative*.

Evaluasi dan validasi hasil dihitung menggunakan rumus akurasi:

1. Akurasi

Perhitungan akurasi dilakukan dengan cara membagi jumlah data yang diklasifikasi secara benar dengan total *sample* data testing yang di uji.

$$Accuracy = \frac{TP+TN}{TP+FP+FN+TN} \times 100\% \quad (2.2)$$

2. *Precision*

Menghitung nilai *precision* dengan cara membagi jumlah data benar yang bernilai positif (*True Positive*) dibagi dengan jumlah data benar yang bernilai positif (*True Positive*) dan data salah yang bernilai positif (*False Negative*).

$$Precision = \frac{TP}{TP+FP} \quad (2.3)$$

3. *Recall*

Recall dihitung dengan cara membagi data benar yang bernilai positif (*True Positive*) dengan hasil perjumlahan dari data benar yang bernilai positif (*True Positive*) dan data salah yang bernilai negatif (*False Negative*).

$$Recall = \frac{TP}{TP+FN} \quad (2.4)$$

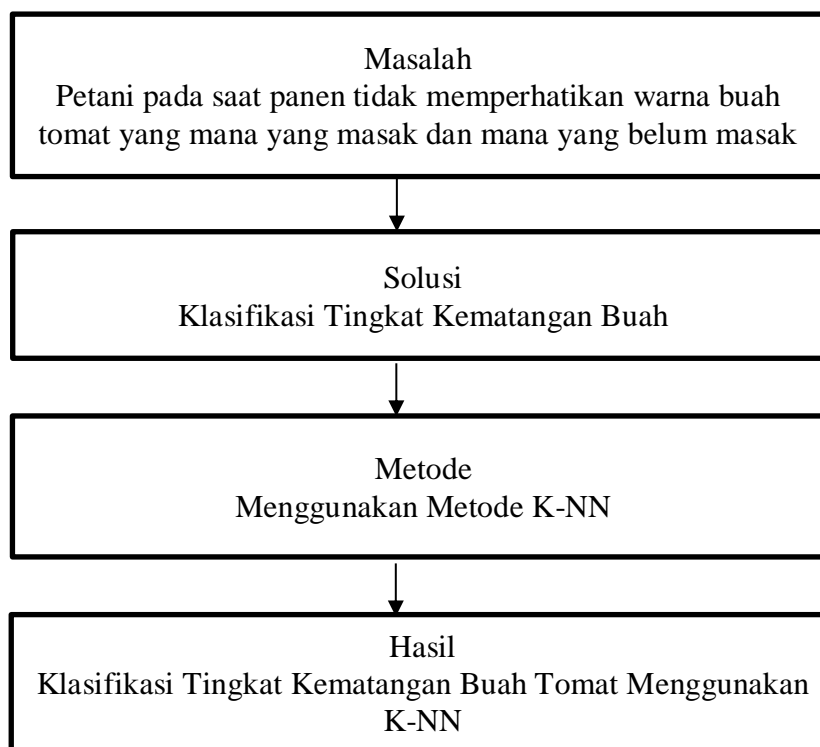
4. F-1 score

Rumus F-1 *score* adalah rata-rata dari *precision* dan *recall* dengan menggabungkan kedua nilainya.

$$F1\text{-score} = 2 \times \frac{\text{Recall} \times \text{Precision}}{\text{Recall} + \text{Precision}} \quad (2.5)$$

1.3 Kerangka Pikir

Berikut ini adalah alur kerangka pikir implementasi untuk Klasifikasi Tingkat Kematangan Buah Tomat Menggunakan K-NN. Kerangka pikir tersebut dilihat pada Gambar 2.6 berikut



Gambar 2.5 Kerangka Pikir