

## BAB 2

### KAJIAN PUSTAKA

#### 2.1 Landasan Teori

##### 2.1.1 Pengolahan Citra Digital

Secara harfiah, citra (*image*) adalah gambar pada bidang dwimatra (dua dimensi), yang dapat dilihat oleh mata manusia atau perangkat penginderaan. Dalam konteks matematis, citra dapat dianggap sebagai fungsi kontinu dari intensitas cahaya pada bidang dwimatra. Proses terbentuknya citra dimulai dengan adanya sumber cahaya yang menerangi objek-objek, dan objek tersebut memantulkan sebagian dari cahaya yang diterimanya. Cahaya yang dipantulkan ini kemudian ditangkap oleh alat-alat optik seperti mata manusia, kamera, *scanner*, dan sebagainya. Hasil dari proses ini adalah rekaman atau bayangan objek yang terbentuk dan disebut sebagai citra (Ahmad, 2005).

Dalam citra digital, citra tersebut direpresentasikan dalam bentuk diskret atau digital menggunakan bilangan bulat atau bit. Citra digital terdiri dari kumpulan piksel (elemen gambar terkecil) yang memiliki nilai numerik yang mewakili tingkat kecerahan atau warna pada lokasi yang tepat dalam citra. Pengambilan gambar menggunakan perangkat seperti kamera digital atau sensor penginderaan mengubah cahaya yang dipantulkan oleh objek menjadi sinyal listrik. Sinyal listrik tersebut kemudian diubah menjadi nilai numerik yang merepresentasikan intensitas cahaya pada setiap piksel dalam citra, membentuk citra digital.

Pengolahan citra digital melibatkan penggunaan metode matematika, statistik, dan algoritme untuk memanipulasi, menganalisis, dan meningkatkan citra digital. Hal ini meliputi berbagai operasi seperti filtrasi, segmentasi, ekstraksi fitur, restorasi, dan rekonstruksi citra. Pengolahan citra digital memiliki berbagai aplikasi yang luas, termasuk komunikasi visual, komputer grafis, pengenalan pola, pengolahan medis, pemrosesan citra satelit, pengolahan citra forensik, dan banyak lagi.

Dengan demikian, citra digital adalah representasi diskret atau digital dari citra pada bidang dwimatra, dihasilkan melalui proses pengambilan

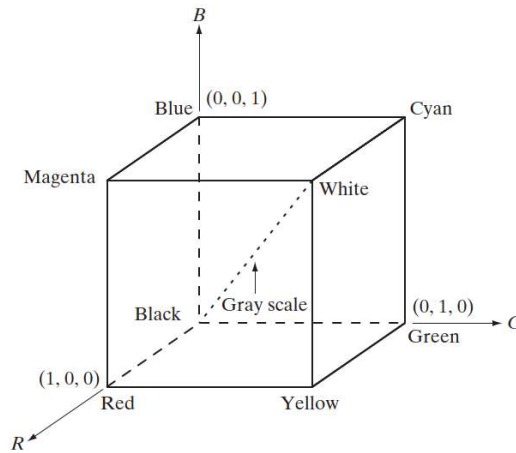
gambar dan direpresentasikan menggunakan piksel-piksel dengan nilai numerik. Pengolahan citra digital melibatkan manipulasi dan analisis menggunakan teknik matematis dan algoritme untuk meningkatkan dan menganalisis citra tersebut.

## **2.1.2 Ruang Warna**

Warna adalah cara sistem visual manusia (*HVS - Human Visual System*) mengukur sebagian dari spektrum elektromagnetik, kira-kira antara 300 hingga 830 nm. Sedangkan ruang warna adalah notasi yang digunakan untuk menentukan warna berdasarkan persepsi manusia terhadap spektrum elektromagnetik yang terlihat (Tkalčič & Tasič, 2003).

### **2.1.2.1 Red Green Blue (RGB)**

Ruang warna yang paling umum digunakan dalam teknologi *computer vision* dan pengolahan citra digital adalah ruang warna *RGB* karena secara langsung menggambarkan saluran merah (*Red*), hijau (*Green*), dan biru (*Blue*), yang merupakan komponen warna primer yang erat kaitannya dengan sistem visual manusia (Ngadi & Liu, 2010). Model ruang warna ini menggunakan sistem koordinat Cartesius. Pada model ini, ruang warna ditampilkan dalam bentuk kubus, seperti yang terlihat pada (Gambar 2.1). Nilai-nilai primer *RGB* diletakkan di tiga sudut; sementara warna sekunder *cyan*, *magenta*, dan *kuning* diletakkan di tiga sudut lainnya; warna hitam berada di titik pusat kubus; dan warna putih berada di sudut terjauh dari titik pusat. Skala abu-abu dalam model ini berjalan sepanjang garis diagonal yang menghubungkan titik hitam dan titik putih. Selain itu, diasumsikan bahwa semua nilai warna telah dinormalisasi, sehingga kubus yang ditampilkan pada (Gambar 2.1) merupakan kubus satuan, dan semua nilai *R*, *G*, dan *B* diasumsikan berada dalam kisaran  $[0, 1]$  (Gonzalez & Woods, 2008).



**Gambar 2.1** Model Warna *RGB*

Sumber : (Gonzalez & Woods, 2008)

### 2.1.2.2 Hue Saturation Value (*HSV*)

Model warna *HSV* (*Hue*, *Saturation*, dan *Value*) adalah cara yang lebih intuitif untuk mendefinisikan warna. Parameter *Hue* menggambarkan warna sebenarnya, seperti merah, violet, dan kuning, dan digunakan untuk membedakan berbagai warna serta menentukan tingkat kemerahan, kehijauan, atau warna lain dari cahaya. Rentang nilai *Hue* berkisar antara 0 hingga 1, dimulai dari warna merah, melalui kuning, hijau, cyan, biru, magenta, dan kembali lagi menjadi merah pada nilai 1. Selain itu, model *HSV* memiliki parameter *Saturation* yang menggambarkan tingkat saturasi warna. Nilai *Saturation* berkisar antara 0 hingga 1, di mana nilai 0 menghasilkan warna yang tidak tersaturasi (keabuan) atau mendekati warna putih, sementara nilai 1 menghasilkan warna dengan saturasi penuh (tidak tercampur dengan warna putih). Model warna *HSV* sering digunakan dalam desain grafis, pengolahan gambar, dan pengaturan warna karena kemudahannya dalam mengontrol karakteristik warna (Situmorang, 2021). Perhitungan konversi *RGB* ke *HSV* dapat dirumuskan sebagai berikut (Smith, 1978) :

1. Ubah nilai warna *RGB* dalam rentan  $[0,1]$

$$R' = \frac{R}{255}, G' = \frac{G}{255}, B' = \frac{B}{255} \quad (2.1)$$

2. Menghitung nilai *Value* dan *X*

$$V = \max(r, g, b), X = \min(r, g, b) \quad (2.2)$$

3. Menghitung nilai *Saturation*

$$S = \begin{cases} 0, & \text{jika Value} = 0 \\ \left(\frac{V - X}{V}\right), & \text{jika lainnya} \end{cases} \quad (2.3)$$

4. Menghitung nilai *r, g, b*

$$r = \frac{V - R'}{V - X}; g = \frac{V - G'}{V - X}; b = \frac{V - B'}{V - X} \quad (2.4)$$

5. Menghitung nilai *Hue*

$$H = \begin{cases} R' = V \text{ then } G = X, H = 5 + b, G \neq X, 1 - g \\ G' = V \text{ then } B = X, H = 1 + r, B \neq X, 3 - b \\ B' = V \text{ then } R = X, H = 3 + g, R \neq X, 5 - r \end{cases} \quad (2.5)$$

6. Normalisasi nilai *Hue*

$$H = \frac{H}{6} \quad (2.6)$$

Keterangan :

*S* : nilai *Saturation*

*V* : nilai *Value*

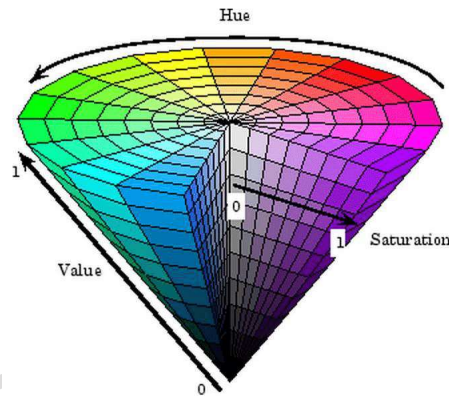
*H* : nilai *Hue*

*X* : nilai minimum dari normalisasi *RGB*.

*R, G, B* : nilai kanal warna *red* (merah), *green* (hijau), *blue* (biru)

*R', G', B'* : nilai kanal warna *red* (merah), *green* (hijau), *blue* (biru) dalam rentang [0,1]

*r, g, b* : nilai kanal warna *red* (merah), *green* (hijau), *blue* (biru) ternormalisasi

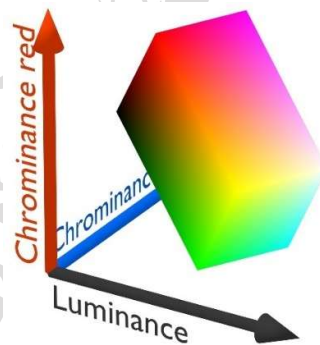


**Gambar 2.2** Model Warna *HSV*

Sumber : (Tkalčič & Tasič, 2003)

### 2.1.2.3 Dasar Warna *YCbCr*

Ruang *YCbCr* akan memisahkan citra ke dalam komponen luminance dan komponen warna. Informasi luminance dipresentasikan oleh komponen *Y*, sedangkan informasi warna dipresentasikan pada komponen *Cb* dan *Cr* (Awaludin, 2016).



**Gambar 2.3** Model Warna *YCbCr*

Sumber : (Pamungkas, 2016)

Komponen *Cb* merupakan selisih antara komponen *Blue* dengan suatu nilai referensi, dan komponen *Cr* adalah selisih antara komponen *Red* dengan suatu nilai referensi. Ruang warna *YCbCr* dapat dengan mudah diperoleh dari ruang *RGB* dengan matrik sederhana (Awaludin, 2016),

Konversi antara ruang warna *RGB* dan ruang warna *YCbCr* dijelaskan melalui persamaan-persamaan berikut (Ganesan et al., 2015):

$$\begin{pmatrix} Y \\ Cb \\ Cr \end{pmatrix} = \begin{pmatrix} 0 \\ 128 \\ 128 \end{pmatrix} + \begin{pmatrix} 0,2999 & 0,587 & 0,114 \\ -0,168 & -0,331 & 0,5 \\ 0,5 & -0,418 & -0,081 \end{pmatrix} \begin{pmatrix} R \\ G \\ B \end{pmatrix} \quad (2.7)$$

### 2.1.3 Momen Warna

Momen warna adalah ukuran yang dapat digunakan untuk membedakan gambar berdasarkan fitur-fitur warnanya. Dasar momen warna terletak pada asumsi bahwa distribusi warna dalam suatu gambar dapat diinterpretasikan sebagai distribusi probabilitas. Distribusi probabilitas ditandai oleh sejumlah momen unik (misalnya, distribusi Normal dibedakan oleh rata-rata dan variansnya). Oleh karena itu, jika warna dalam suatu gambar mengikuti distribusi probabilitas tertentu, maka momen-momen dari distribusi tersebut dapat digunakan sebagai fitur untuk mengidentifikasi gambar tersebut berdasarkan warna (Noah Keen, 2005).

Informasi mengenai distribusi warna diatur dalam tiga urutan momen, yaitu rerata (*mean*), standar deviasi (*standard deviation*), dan *skewness*. Dengan demikian, setiap kanal warna akan menghasilkan tiga nilai yang mewakili karakteristik distribusi warna. Pada Skripsi ini, peneliti menggunakan tiga ruang warna, yaitu *RGB*, *HSV*, dan *YCbCr*, sebagai perbandingan untuk klasifikasi objek. Setiap ruang warna memiliki tiga kanal warna, dan masing-masing akan menghasilkan sembilan nilai sebagai identitas sebuah citra.

Dalam (Noah Keen, 2005)  $P_{ij}$  didefinisikan sebagai nilai intensitas warna pada piksel ke- $j$  dalam kanal warna ke- $i$  pada citra. Ketiga momen warna didefinisikan sebagai berikut :

**Moment 1 – Mean :**

$$E_i = \sum_N^{j=1} \frac{1}{N} P_{ij} \quad (2.8)$$

*Mean* adalah nilai rata-rata dari komponen warna dalam citra.

**Moment 2 – Standard Deviation**

$$\sigma_i = \sqrt{\left( \frac{1}{N} \sum_{j=1}^N (P_{ij} - E_i)^2 \right)} \quad (2.9)$$

*Standard Deviation* adalah akar kuadrat dari varians dari distribusi atau jangkauan tersebarnya data dari *mean*.

### **Moment 3 – Skewness**

$$s = \sqrt[3]{\left( \frac{1}{N} \sum_{j=1}^N (P_{ij} - E_i)^3 \right)} \quad (2.10)$$

*Skewness* adalah ukuran yang digunakan untuk menilai sejauh mana suatu distribusi data cenderung menjadi tidak simetris.

#### **2.1.4 Klasifikasi**

Klasifikasi adalah proses identifikasi dan pengelompokan objek atau data ke dalam kategori atau kelas yang berbeda berdasarkan sejumlah fitur atau atribut tertentu. Klasifikasi merupakan salah satu tugas penting dalam *Machine Learning* yang bertujuan untuk mengelompokkan atau mengkategorikan data ke dalam kelas atau label yang sudah ditentukan. Klasifikasi digunakan dalam berbagai bidang, termasuk pengenalan pola, pengolahan bahasa alami, pengenalan wajah, sistem rekomendasi, dan banyak lagi.

Dalam konteks klasifikasi, data pelatihan terdiri dari sejumlah sampel data yang masing-masing memiliki atribut atau fitur tertentu, serta label kelas yang menunjukkan keanggotaan ke dalam suatu kelas atau kategori. Misalnya, dalam klasifikasi email spam, atribut dapat berupa teks email dan labelnya adalah "spam" atau "bukan spam." Dan dalam penelitian ini atributnya berupa halaman dokumen dan labelnya adalah "Warna tinggi", "Warna rendah", dan "Hitam putih". Tujuan dari *machine learning* klasifikasi adalah untuk mengembangkan model yang dapat memprediksi label kelas dari data baru berdasarkan atribut yang diberikan dalam penelitian ini.

#### **2.1.4.1 K- Nearest Neighbours (KNN)**

Metode *K-Nearest Neighbours (KNN)* merupakan salah satu algoritme *supervised learning*. Metode *K-Nearest Neighbours (KNN)* digunakan dalam proses klasifikasi citra dengan berpacu pada hasil ekstraksi fitur yang sebelumnya sudah dilakukan *Training* (pelatihan) (Khotimah et al., 2019). Langkah-langkahnya melibatkan perhitungan jarak antara data uji dengan semua data pelatihan, lalu menentukan data pelatihan yang paling mirip, yang disebut sebagai tetangga terdekat. Setelah itu, metode ini melakukan klasifikasi. Titik-titik "k" terdekat yang telah ditentukan sebelumnya dalam data pelatihan digunakan untuk menghitung probabilitas kelas dan menentukan kelas mana yang memiliki probabilitas tertinggi. Metode ini populer karena implementasinya yang mudah dipahami dan kemampuannya dalam mengklasifikasikan data dengan baik (García et al., 2019).

Langkah-langkah yang dilakukan untuk pada metode *K-Nearest Neighbours* adalah sebagai berikut (Rizarta, 2019) :

- a. Menentukan data sampel yang akan digunakan sebagai data latih.
- b. Inialisasi titik K sebagai titik pusat (*centroid*) awal.
- c. Hitung jarak setiap objek data set dengan data training dengan rumus perhitungan jarak.

#### **2.1.4.2 Euclidean Distance**

*Euclidean distance* (jarak Euclidean) adalah metrik yang digunakan untuk mengukur jarak antara dua titik dalam ruang Euclidean (ruang geometri yang mengikuti konsep yang diperkenalkan oleh Euclid, seorang matematikawan Yunani pada tahun 300 B.C.E.) (Rizarta, 2019). Ini adalah metode yang umum digunakan dalam berbagai aplikasi, termasuk analisis data, pengenalan pola, dan pemrosesan citra.

Dalam ruang Euclidean, setiap titik memiliki koordinat yang diberikan oleh beberapa bilangan riil, dan *Euclidean distance* mengukur panjang jarak lurus (garis lurus) antara dua titik tersebut. Ini dihitung menggunakan formula Pythagoras, yang merupakan dasar dari jarak Euclidean. Secara matematis,



jarak Euclidean antara dua titik, misalnya A (x1, y1) dan B (x2, y2), dihitung sebagai:

$$d = \sqrt{(x_2 - x_1)^2 + (y_2 - y_1)^2} \quad (2.11)$$

### 2.1.5 Evaluasi Sistem

Evaluasi sistem adalah tahap penting dalam penelitian yang bertujuan untuk mengukur kinerja sistem yang dikembangkan. Evaluasi ini penting untuk menilai sejauh mana sistem mampu mencapai tujuan yang diinginkan.

#### 2.1.5.1 K-Fold Cross Validation

*K-Fold Cross Validation* adalah metode validasi model yang digunakan untuk mengukur kinerja algoritme atau model machine learning. Dalam *K-Fold Cross Validation*, data dibagi menjadi k bagian, dan proses berikut dilakukan (Mardiana et al., 2022) :

1. Data dibagi menjadi k bagian yang sama ukurannya.
2. Model dilatih menggunakan k-1 bagian data dan diuji pada bagian yang tersisa.
3. Proses ini diulang sebanyak k kali, dengan setiap bagian data menjadi data uji satu kali.
4. Akurasi model dihitung sebagai rata-rata dari akurasi pada setiap iterasi.

#### 2.1.5.2 Confusion Matrik

Salah satu metrik utama yang digunakan dalam evaluasi sistem adalah matrik konfusi. Matriks konfusi dihitung dengan membandingkan antara kategori aktual (Actual Class) dan hasil prediksi (Predicted Class) untuk mengevaluasi sejauh mana model mampu mengklasifikasikan data dengan benar (Novianti, 2021).

		True/Actual Class	
		Positive (P)	Negative (N)
Predicted Class	True (T)	True Positive (TP)	False Positive (FP)
	False (F)	False Negative (FN)	True Negative (TN)
		$P = TP + FN$	$N = FP + TN$

**Gambar 2.4** Confusion matrix pada kelas biner

Sumber: (Tharwat, 2018)

Berikut adalah penjelasan tentang keempat elemen dalam confusion matrix diatas:

True Positive (TP): Jumlah sampel yang benar-benar diklasifikasikan sebagai positif oleh model.

True Negative (TN): Jumlah sampel yang benar-benar diklasifikasikan sebagai negatif oleh model.

False Positive (FP): Jumlah sampel yang salah diklasifikasikan sebagai positif oleh model.

False Negative (FN): Jumlah sampel yang salah diklasifikasikan sebagai negatif oleh model.

		True Class		
		A	B	C
Predicted Class	A	$TP_A$	$E_{BA}$	$E_{CA}$
	B	$E_{AB}$	$TP_B$	$E_{CB}$
	C	$E_{AC}$	$E_{BC}$	$TP_C$

**Gambar 2.5** Confusion matrix pada multi-kelas

Sumber: (Tharwat, 2018)

(Gambar 2.6) menunjukkan matriks konfusi untuk masalah klasifikasi multi-kelas dengan tiga kelas (A, B, dan C).  $TP_A$  adalah jumlah sampel yang diklasifikasikan dengan benar dari kelas A, dan  $E_{AB}$  adalah jumlah sampel dari kelas A yang salah diklasifikasikan sebagai kelas B. False negative dalam kelas A ( $FN_A$ ) dihitung sebagai jumlah  $E_{AB}$  dan  $E_{AC}$  ( $FN_A = E_{AB} + E_{AC}$ ) (Tharwat, 2018).

Terdapat beberapa matrik evaluasi yang biasa digunakan untuk mengevaluasi suatu sistem atau model dan akan digunakan dalam penelitian ini yaitu akurasi, spesifitas, sensitivitas, dan f1-score yang masing-masing penjelasannya sebagai berikut:

#### **Akurasi (*Accuracy*)**

Akurasi adalah metrik evaluasi yang digunakan untuk mengukur sejauh mana sistem mampu mengklasifikasikan dokumen dengan benar ke dalam kelas yang sesuai.

$$\text{Akurasi} = \frac{TP + TN}{TP + TN + FP + FN} \quad (2.12)$$

#### **Spesifitas (*Specivity*)**

Spesifitas digunakan untuk mengukur kemampuan model mengidentifikasi dengan benar kelas negatif.

$$\text{Spesifitas} = \frac{TN}{TN + FP} \quad (2.13)$$

#### **Sensitivitas (*Sensitivity* atau *Recall*)**

Sensitivitas digunakan untuk mengukur kemampuan model mengidentifikasi dengan benar kelas positif.

$$\text{Sensitivitas} = \frac{TP}{TP + FN} \quad (2.14)$$

#### **F1-Score**

F1-Score adalah rata-rata harmonik antara presisi dan sensitivitas. Menggabungkan informasi tentang ketepatan dan recall dalam satu metrik.

$$\text{Presisi} = \frac{TP}{TP + FP} \quad (2.15)$$

$$\text{F1 - Score} = \frac{2 \times (\text{presisi} \times \text{Sensitivitas})}{\text{Presisi} + \text{Sensitivitas}} \quad (2.16)$$

## **2.2 Landasan Pustaka**

1. Astutik, dkk., (2020) dengan judul “Klasifikasi Jenis Pasir Material Bangunan Menggunakan Metode Support Vector Machine (SVM) Berdasarkan Ekstraksi Ciri Tekstur dan Warna” (Astutik et al., 2022).

Pada penelitian ini menggunakan moment RGB untuk ekstraksi ciri warna dan metode *Gray Level Co-occurrence Matrix (GLCM)* untuk ekstraksi ciri tekstur dengan arah sudut 135 derajat dengan parameter fitur correlation, dissimilarity, homogeneity, dan contrast yang kemudian menghasilkan akurasi 94% dengan menggunakan metode *SVM* sebagai *classifier*.

2. Ismail, dkk., (2023) dengan judul “Klasifikasi Kematangan Daun Tembakau Virginia Menggunakan Pengolah Citra Digital” (Ismail et al., 2023).

Pada penelitian ini, mengklasifikasikan citra daun tembakau virginia dengan kelas matang, muda, dan tua dengan ekstraksi fitur warna menggunakan *Color Moments HSV* dan metode SVM (Support Vector Machine) sebagai *classifier*. dan didapatkan akurasi yang sangat baik yaitu sebesar 98%

3. Franch, dkk., (2022) dengan judul “Identification of Banana Types with the Least-Squares Support Vector Machine (LS-SVM) Method” (Franch et al., 2022).

Pada penelitian ini Menggunakan ekstraksi fitur warna momen warna *HSV* dan *YCbCr* juga ekstraksi fitur tekstur *GLCM* dengan menggunakan metode klasifikasi *LS-SVM* untuk mengidentifikasi jenis buah pisang. Hasilnya didapat 90% akurasi didapat dari model.

4. Adenugraha, dkk., (2022) dengan judul “Klasifikasi Kematangan Buah Pisang Ambon Menggunakan Metode KNN dan PCA Berdasarkan Citra RGB dan HSV” (Adenugraha et al., 2022).

Pada penelitian ini digutakan fitur dari ekstraksi ciri HSV, RGB dan Area kemudian diklasifikasikan menggunakan metode KNN dengan penghitungan jarak *euclidean distance*. Hasil akurasi dari klasifikasi sebesar 90,9% dengan nilai K=5 yang didapat dari 10 data uji dengan klasifikasi akurat, dan 1 data uji dengan klasifikasi tidak akurat.

5. Wibowo, dkk., (2021) dengan judul “Deteksi Kematangan Buah Jambu Kristal Berdasarkan Fitur Warna Menggunakan Metode Transformasi Ruang

Warna Hsv (Hue Saturation Value) Dan K-Nearest Neighbor” (Wibowo et al., 2021).

Pada penelitian ini digunakan transformasi ruang warna HSV sebagai ekstraksi fitur dan KNN untuk klasifikasi, ekstraksi fitur *HSV* disini diambil dari rerata atau *average* dari masing-masing kanal *HSV*. didapatkan hasil akurasi tertinggi yaitu 95%.

6. Ezar, dkk., (2021) dengan judul “penentuan kualitas buah pepaya california menggunakan metode k-nn” (Ezar et al., 2021).

Pada penelitian ini, ekstraksi fitur yang digunakan adalah R, G, minor axis, major axis, dan defect area, kemudian diklasifikasikan dengan KNN dengan menggunakan dua perhitungan jarak yaitu Euclidean dan Manhattan untuk perbandingan dan variasi nilai-k sebesar 3, 5, 7, dan 9. Penelitian ini pun mendapatkan hasil terbaik pada nilai k-7 dengan *euclidean distance* yaitu *accuracy* sebesar 86,67%, *precision* sebesar 87,50%, dan *recall* sebesar 80,00%.

7. Wibawa, dkk., (2021) dengan judul “Penerapan Ruang Warna HSV dan Ekstraksi Fitur Tekstur Local Binary Pattern untuk Tingkat Kematangan Sangrai Biji Kopi” (Wibawa et al., 2021).

Pada penelitian ini digunakan momen warna HSV dan *Local Binary Pattern (LBP)* sebagai ekstraksi fitur dan KNN sebagai *classifier*. Akurasi yang didapatkan dari kedua metode yang digunakan yaitu untuk nilai Hue dan Saturation mendapatkan akurasi sebesar 79%, dan untuk Hue dan Saturation dengan menggunakan nilai normalisasi RGB sebesar 71,4%.

8. Li, dkk., (2021) dengan judul “SelfDoc: Self-Supervised Document Representation Learning” (Li et al., 2021).

Pada Penelitian ini diciptakan kerangka kerja (*framework*) yang disebut SelfDoc yang merupakan model pra-pelatihan (*pre-trained model*) yang dirancang khusus untuk pemahaman dokumen. SelfDoc memanfaatkan fitur-fitur seperti representasi blok semantis, pemodelan kontekstual, pembelajaran lintas modal, dan strategi pra-pelatihan sendiri pada dokumen. Penelitian ini membandingkan hasil klasifikasi dokumen menggunakan SelfDoc dengan

beberapa pra-pelatihan model lain seperti BERT dan VGG-16, dan menunjukkan SelfDoc unggul dengan akurasi tertingginya mencapai 93,81%.

9. Anggraeni D., (2021) dengan judul “Perbaikan Citra Dokumen Hasil Pindai Menggunakan Metode Simple, Adaptive-Gaussian, dan Otsu Binarization Thresholding” (Anggraeni, 2021).

Penelitian ini membandingkan kinerja dari tiga metode Thresholding yaitu Simple Thresholding, Adaptive-Gaussian Thresholding, dan Otsu Binarization untuk filterisasi citra hasil pindai dokumen. Hasil evaluasi menggunakan parameter penilaian MSE dan PSNR menunjukkan bahwa metode Simple Thresholding adalah yang terbaik dengan rata-rata nilai MSE 5.196,76 dan rata-rata nilai PSNR 13.37.

10. Rismanto, dkk., (2020) dengan judul “Optimalisasi Image Thresholding pada Optical Character Recognition Pada Sistem Digitalisasi dan Pencarian Dokumen” (Rismanto et al., 2020).

Penelitian ini mengimplementasikan *Optical Character Recognition (OCR)* dan melakukan optimalisasi pada digitalisasi dokumen. Proses ekstraksi teks yang dengan teknik *OCR* diawali Image Thresholding dengan beberapa nilai ambang yaitu 0.3, 0.4, 0.5, 0.6, 0.7, 0.8 dan 0.9. Dari hasil percobaan, nilai ambang yang menghasilkan akurasi tertinggi adalah 0.6 dengan nilai akurasi 92.568%.