

BAB 2

TINJAUAN PUSTAKA

2.1 KAJIAN PUSTAKA

Penelitian mengenai deteksi penyakit daun padi dengan algoritma *Convolutional Neural Network* (CNN) dilakukan oleh beberapa peneliti Rizal Amegia Saputra *et al* tahun 2021 melakukan klasifikasi penyakit daun padi yang berdasarkan citra dengan menggunakan *MobileNetV1* berdasarkan ekstraksi fitur. Dataset berasal dari *UCI Repository* sebanyak 120 yang terdiri dari 3 penyakit daun padi yaitu *bacterial leaf blight*, *brown spot*, *leaf smut*. Berdasarkan hasil pelatihan dan pengujian menggunakan citra penyakit daun padi yang berukuran 224x224 piksel didapatkan hasil nilai akurasi pelatihan mencapai 1.0 dan nilai akurasi validasi mencapai 0.8333. Nilai akurasi pada *confusion matrix* yaitu sebesar 92% [8]. Perbedaan penelitian dengan Rizal Amegia Saputra adalah penelitian akan menggunakan pendekatan algoritma KNN.

Penelitian Jani Kusanti *et al* 2018 melakukan klasifikasi penyakit daun padi berdasarkan ekstraksi fitur *Gray Level Co-occurrence Matrix* (GLCM) interval 4 sudut. Pada penelitian ini dibuatlah suatu program aplikasi yang dapat mengidentifikasi hama penyakit padi. Hal ini menjadi penting untuk dibuat sistem klasifikasi penyakit pada daun tanaman padi. Sistem ini menggunakan metode *backpropagation* untuk mengklasifikasi jenis penyakit berdasarkan ekstraksi fitur GLCM dengan 4 sudut. Hasil yang diperoleh dari 30 data mencapai 80% data valid dengan waktu 16 detik [9]. Perbedaan penelitian dengan Jani Kusanti adalah menggunakan pendekatan algoritma KNN berdasarkan fitur warna tidak berdasarkan ekstraksi fitur GLCM dengan interval 4 sudut dan pada penelitian tidak sampai tahap membuat aplikasi.

Penelitian Endang Anggiratih *et al* 2021 melakukan klasifikasi penyakit tanaman padi menggunakan *deep learning efficientnet B3* dengan *transfer learning*. Pada penelitian ini menggunakan metode *deep learning* khususnya CNN dengan arsitektur *efficientnet B3* untuk klasifikasi penyakit bercak cokelat dan

penyakit bakteri. Hasil pengujian tercapai akurasi 79,53% dengan *loss* 0,012 [11]. Perbedaan penelitian dengan Endang Anggiratih adalah menggunakan pendekatan algoritma KNN berdasarkan fitur warna dan *invariant moment* bukan CNN dengan arsitektur *efficientnet* B3 dan pada penelitian ini menggunakan tiga jenis penyakit tanaman padi.

Penelitian Zulkifli tahun 2021 melakukan sistem pendeteksi penyakit tanaman padi berbasis kecerdasan buatan dan sistem pakar yang sama optimalnya dengan diagnosa para pakar. Metode yang digunakan adalah metode *Expert System Development Life Cycle* (ESDLC) yang terdiri dari penilaian, pemetaan pengetahuan pakar, desain, dan pengujian. Adapun hasil yang didapatkan dalam penelitian ini adalah adanya sebuah aplikasi pendeteksi penyakit tanaman padi berbasis kecerdasan buatan dengan menggunakan metode sistem pakar yang dapat digunakan oleh petani untuk mendiagnosa penyakit tanaman padi secara mandiri tanpa harus bertemu langsung dengan pakarnya [12]. Perbedaan penelitian dengan Zulkifli adalah menggunakan pendekatan algoritma KNN berdasarkan fitur warna dan *invariant moment* bukan berfokus dalam pembuatan aplikasi berbasis *Artificial Intelligence* (AI).

Penelitian Mohtar Khoiruddin *et al* tahun 2022 mengenai klasifikasi daun padi. Metode yang digunakan adalah metode CNN. Jenis daun padi yang diklasifikasi pada penelitian ini diantaranya: *bacterial leaf blight*, *rice blast*, dan *rice tungro* virus. Jumlah dataset yang digunakan 6000 yang terbagi menjadi data *training* 80%, data validasi 10%, dan data *testing* 10%. Hasil yang didapatkan untuk *epoch* 25, 50, 75 dan 100 memiliki akurasi yang beragam. Hasil akurasi *training* terbaik ada di *epoch* 100 dengan akurasi 98% [13]. Perbedaan penelitian dengan Mohtar Khoiruddin *et al* adalah penelitian ini menggunakan algoritma KNN dengan fitur warna HSV dan *invariant moment*.

Tabel 2. 1 Perbandingan Metode dan Klasifikasi

Pengarang	Tujuan	Metode	Perbedaan
Rizal Amegia Saputra dkk	Deteksi penyakit pada daun padi	Dataset <i>Bacterial leaf blight</i> , <i>Brown spot</i> , <i>Leaf smut</i> dari	Penelitian rujukan pertama menggunakan metode CNN.

Pengarang	Tujuan	Metode	Perbedaan
(2021)	menggunakan CNN.	<i>UCI Repository</i> dengan metode menggunakan citra CNN, <i>Confusion matrix</i> , <i>Feature Extraction</i> .	Sedangkan, penelitian ini mendekati yang digunakan dengan metode KNN.
Jani Kusanti dkk (2018)	Klasifikasi penyakit daun padi menggunakan ekstraksi fitur GLCM interval 4 sudut.	Ekstraksi fitur <i>Gray Level Co-occurrence Matrix</i> (GLCM), <i>Energy</i> , <i>Entropi</i> , <i>Contrast</i> , <i>Homogeneity</i> , <i>Correlation</i> , <i>Backpropagation</i> , interval 4 sudut dengan sudut 0°, 45°, 90°, dan 135°.	Penelitian rujukan kedua menggunakan ekstraksi fitur GLCM dengan interval 4 sudut dan sampai pada tahap pembuatan aplikasi. Sedangkan, penelitian ini menggunakan metode algoritma KNN berdasarkan fitur warna.
Endang Anggiratih dkk (2021)	Klasifikasi penyakit tanaman padi menggunakan <i>deep learning</i> <i>Efficientnet B3</i> dengan <i>transfer learning</i> .	<i>Deep learning</i> , CNN, <i>Efficientnet B3</i> , <i>Mobilenet V3</i> , Dataset <i>brown spot</i> dan <i>bacterial leaf</i> , <i>Transfer learning</i> , <i>Tensorflow</i> .	Penelitian rujukan ketiga menggunakan metode CNN dengan arsitektur <i>efficientnet B3</i> dan menggunakan dua klasifikasi jenis penyakit. Sedangkan, penelitian ini menggunakan metode KNN berdasarkan fitur warna dan <i>invariant moment</i> serta menggunakan tiga

Pengarang	Tujuan	Metode	Perbedaan
			kelas jenis penyakit daun padi.
Zulkifli (2021)	Mendeteksi penyakit tanaman padi menggunakan <i>Artificial Intelligence</i> .	Metode <i>Expert System Development Life Cycle</i> (ESDLC), <i>Fuzzy</i> .	Penelitian rujukan keempat menggunakan metode ESDLC dan berfokus dalam pembuatan aplikasi berbasis AI. Sedangkan, penelitian ini menggunakan metode algoritma KNN berdasarkan fitur warna dan <i>invariant moment</i> .
Mohtar Khoiruddin dkk (2022)	Klasifikasi tanaman daun padi menggunakan CNN.	Dataset <i>Bacterial leaf blight, Rice blast, Rice tungro</i> virus, metode menggunakan <i>Convolution Neural Network</i> dengan <i>epoch</i> 25, 50, 75 dan 100.	Penelitian rujukan kelima menggunakan metode CNN dengan menggunakan <i>epoch</i> 25, 50, 75 dan 100 dengan 2 kelas jenis penyakit daun padi. Sedangkan, penelitian ini menggunakan metode KNN berdasarkan fitur warna dan <i>invariant moment</i> serta menggunakan tiga kelas jenis penyakit daun padi.

2.1 DASAR TEORI

2.2.1 TANAMAN PADI

Padi ialah salah satu komoditas tumbuhan utama warga Indonesia. Padi dengan latin *Oryza Sativa* adalah komoditi tanaman pangan [14]. Padi menghasilkan beras yang ialah makanan utama untuk dikonsumsi warga. Peningkatan kualitas beras bisa menaikkan produksi beras itu. Pada pelaksanaan budidaya padi harus sesuai agar bisa mendatangkan produksi padi dengan cara maksimal. Akan tetapi, saat ini banyak ditemui hambatan oleh petani pada perihal budidayanya [15].

Padi memiliki umur yang berbeda-beda. Padi yang dapat dipanen berumur kurang dari 90 hari, tanaman padi bisa dipanen ketika umur lebih dari 6 bulan. Pada umumnya padi dipanen saat umur 3-4 bulan setelah tanam. Tumbuhan padi memiliki kapasitas beradaptasi di hampir seluruh kawasan, mulai dari dataran rendah sampai tinggi (2000 meter dpl), wilayah tropis sampai subtropis, wilayah basah (rawa- rawa) hingga kering (padang pasir), serta lain- lain.

Beras bisa memenuhi sebagian besar kebutuhan vitamin. Tingkat konsumsi beras bangsa Indonesia menggapai 139,15 kilogram per jiwa tahun. Hasil analisis membuktikan bahwa beras mempunyai kandungan zat makanan yang terdiri dari karbohidrat, protein, lemak, air, besi, magnesium, fosfor, kalium, seng, vit B1, B2, B3, B6, B9, serta serat [16]. Tindakan pengendalian sebagai prioritas budidaya padi karena tingginya kehilangan hasil tanaman padi diakibatkan karena serangan hama dan penyakit [14].

2.2.2 PENYEBAB PENYAKIT DAUN PADI

Hama dan penyakit tumbuhan ialah organisme yang mengganggu tumbuhan. Serangan ini menyebabkan kerusakan dan kehilangan panen dalam skala besar yang merugikan petani [14]. Padi dapat terkena penyakit dikarenakan jamur, bakteri, dan virus. Selain itu, pemicu penyakit tanaman padi adanya 3 faktor utama yaitu patogen penyebab penyakit yang ganas, tanaman inang yang

rentan, serta area yang mensupport. Faktor-faktor tersebut terjadi pada waktu dan tempat yang sama. Adapun beberapa penyakit padi yang menurunkan produksi:

1. Hawar Daun Bakteri

Hawar Daun Bakteri atau *Bacterial Leaf Blight* adalah penyakit padi yang disebabkan oleh bakteri gram negatif *Xanthomonas oryzae* pv. *Oryzae*. Bakteri ini menyerang pada vegetatif dan generatif [14].



Gambar 2. 1 Penyakit hawar daun bakteri [17]

Gambar 2.1 ialah penyakit hawar daun bakteri pada daun padi yang sering ditemukan pada masa vegetatif akhir dan generatif awal. Penurunan produksi yang diakibatkan oleh penyakit ini mencapai 10-20% [17].

2. Bercak Cokelat

Bercak Cokelat atau *Brown Spot* adalah penyakit padi yang disebabkan oleh jamur *Helminthosporium oryzae*. Penyakit ini dapat mematikan padi muda dan menurunkan kualitas padi [17].



Gambar 2. 2 Penyakit bercak cokelat

(a) gejala pada tanaman

(b) spora patogen [17]

Gambar 2.2 ialah penyakit bercak cokelat pada daun padi yang sering ditemukan pada tanah yang kurang subur. Pada kondisi tanah ini menimbulkan kerugian produksi padi mencapai 50-90% [17].

3. Tungro

Tungro adalah penyakit padi yang disebabkan oleh virus yang disebarkan melalui wereng hijau. Tanaman padi muda lebih muda dan lebih sering terkena infeksi tungro daripada padi yang sudah tua [18].

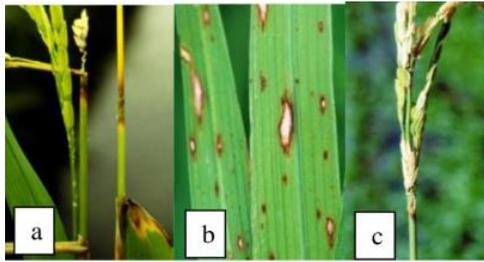


Gambar 2. 3 Penyakit tungro [17]

Gambar 2.3 ialah penyakit tungro pengganggu pertumbuhan tanaman padi yang menimbulkan kerugian hingga 90% [19]. Gejala yang paling mudah dalam pengenalan penyakit tungro adalah tanaman yang berubah menjadi kuning dengan jumlah anakan yang semakin dikit dan lambatnya pertumbuhan padi.

4. Blas Kolar

Blas kolar adalah penyakit padi yang disebabkan oleh jamur *Pyricularia oryzae* yang menginfeksi di beberapa bagian padi seperti daun, batang, node, kolar, dan malai [20]. Penyakit padi ini menyerang tanaman sebagai inang utamanya dan berkembang pesat yang mengakibatkan kematian tanaman. Blas kolar menyebar melalui udara dengan kondisi suhu malam sekitar 20-25°C.



Gambar 2. 4 Penyakit blas pada berbagai bagian padi

(a) blas leher malai

(b) blas daun

(c) blas malai [17]

Gambar 2.4 ialah penyakit blas pada berbagai bagian padi, fase pertumbuhan tanaman padi mulai pesemaian hingga panen sering terserang penyakit ini hingga menyebabkan terhambatnya aktivitas pertumbuhan padi sehingga penurunan padi terjadi.

Adapun penyakit padi serta ciri-cirinya yang dijabarkan pada Tabel 2.2 [17].

Tabel 2. 2 Jenis penyakit padi dan ciri-cirinya

Jenis Penyakit	Ciri-ciri
Hawar Daun Bakteri	Dimulai pada pinggir daun ke abu-abuan dan beberapa lama kemudian daun menjadi kering.
Bercak Cokelat	Bintik-bintik cokelat tua berbentuk bulat panjang hingga bulat dengan ukuran sebesar biji wijen.
Tungro	Daun berwarna kuning oranye dan terpelintir, dengan tanaman yang kecil serta kurangnya anakan.
Blas Kolar	Terdapat bintik-bintik berbentuk belah ketupat.

Gambar 2.5 menunjukkan daun padi yang sehat dan Gambar 2.6 menunjukkan daun padi yang terkena penyakit.



Gambar 2. 5 Daun padi yang sehat [21]



Gambar 2. 6 Daun padi yang terkena penyakit [22]

Gambar 2.5 daun padi yang sehat, padi yang berkualitas mempunyai daun padi yang sehat. Kualitas padi sangat penting dalam peningkatan produksi saat masa panen. Gambar 2.6 daun padi yang terkena penyakit ialah padi dengan kualitas buruk sehingga terjadinya penurunan produksi padi yang merugikan para petani [22].

2.2.3 CITRA DIGITAL

Citra adalah gambar yang memiliki informasi berbentuk visual. Citra dihasilkan dari mesin digital sebagai citra digital [23]. Citra adalah proses *sampling* dari gambar analog 2D kontinu ke gambar diskrit . Proses *sampling*

dipecah menjadi 2, yaitu *downsampling* dan *upsampling* [24]. N baris dan M kolom yang merupakan gambar analog akan di konversi ke gambar diskrit. Citra digital ialah citra yang akan dapat di proses pengolahan melalui komputer. Komputer hanya dapat membaca angka-angka sehingga citra yang disimpan hanyalah berupa angka dari masing-masing piksel [25]. Citra berbentuk gambar maupun kumpulan gambar yang menjadi video, dan sinyal-sinyal video seperti monitor televisi yang dapat disimpan langsung pada media penyimpanan [26]. Contoh dari gambar simulasi adalah gambar tidak dapat ditampilkan di komputer seperti foto yang dicetak diatas kertas foto, yang dilukis di atas kertas, dll. Sedangkan, gambar digital adalah gambar yang dapat diolah langsung oleh komputer. Gambar digital tersusun secara *grid*. Setiap *frame* (blok) terdapat piksel dan memiliki koordinat (x,y) yang mana sumbu x (horizontal):kolom, sumbu y (vertikal): baris. Pada setiap piksel memiliki nilai fungsi pada setiap titik (x,y) yang menyatakan besar intensitas yang diterima oleh citra atau tingkat keabuan atau warna dari piksel di titik tersebut. Besarnya tergantung pada intensitas yang dipantulkan oleh objek [25]. Citra digital bisa mengalami penurunan kualitas karena terdapat *noise* atau derau. Citra yang mengalami derau akan menyulitkan proses interpretasi informasi.

2.2.4 EKSTRAKSI FITUR WARNA

Ekstraksi fitur ialah suatu proses untuk mendapatkan fitur-fitur dari citra. Fitur sendiri ialah ciri khas yang dapat membedakan dari objek satu dengan yang lainnya. Ekstraksi fitur dibedakan menjadi 3 yaitu berdasarkan tekstur, bentuk, serta warna [13]. Pada penelitian ini akan menggunakan ekstraksi fitur warna yang mana citra RGB akan di konversi menjadi citra HSV. Citra akan disegmentasi yang bertujuan untuk memisahkan dari objek dan *background* lalu diproses menggunakan ekstraksi warna HSV dan akan mendapatkan nilai berupa *hue*, *saturation*, dan *value* [27]. Nilai *hue* ini mempresentasikan warna dari merah ke hijau, *saturation* dari merah ke merah muda, serta untuk *value* sendiri mempresentasikan dari warna hitam ke putih [28]. Citra akan dikonversi RGB ke HSV dengan cara seperti ini [13]:

$$r = \frac{R}{(R + G + B)}, g = \frac{G}{(R + G + B)}, b = \frac{B}{(R + G + B)} \quad (2.1)$$

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

$$S = \begin{cases} 0 & , \text{jika } V = 0 \\ 1 - \frac{\min(r, g, b)}{V} & , \text{jika } V > 0 \end{cases} \quad (2.3)$$

$$H = \begin{cases} 0 & , \text{jika } S = 0 \\ \frac{60 * (g - b)}{S * V} & , \text{jika } V = r \\ 60 * \left[2 + \frac{b - r}{S * V} \right] & , \text{jika } V = g \\ 60 * \left[2 + \frac{r - g}{S * V} \right] & , \text{jika } V = b \end{cases} \quad (2.4)$$

$$H = H + 360 \text{ jika } h < 0 \quad (2.5)$$

Keterangan:

r, g, b = derajat keabuan dari sebuah piksel *red, green, dan blue*.

R, G, B = citra warna *red, green, dan blue* (0-255).

H = *hue* mempresentasikan warna dari merah ke hijau (0-255).

S = *saturation* mempresentasikan merah ke merah muda (0-255).

V = *value* mempresentasikan dari warna hitam ke putih (0-255).

2.2.5 INVARIANT MOMENT

Invariant moment ialah salah satu cara dalam membuat ekstraksi ciri khas bentuk yang ada dalam pengolahan citra. *Invariant moment* awal kali dipublikasikan Hu pada tahun 1962. Terdapat tujuh nilai yang dihasilkan dari *invariant moment* untuk ekstraksi ciri. Nilai ekstraksi ciri ini bersifat tidak terikat terhadap translasi, rotasi, dan perskalaan [7]. *Boundary* bentuk dan daerah *interior* akan memberikan informasi yang nantinya akan dihitung dan menghasilkan *invariant moment*. *Moment invariant* akan didefinisikan secara kontinu akan tetapi pada implementasinya akan dihitung dengan diskrit [29]. Hu memperkenalkan *moment invariant* untuk citra digital dengan ukuran $M \times N$ piksel, Persamaan dasar dari *Moment* suatu objek didefinisikan pada rumus [30].

$$m_{pq} = \sum_{x=1}^M \sum_{y=1}^N x^p y^q f(x, y) \quad (2.6)$$

Keterangan:

x = panjang citra.

y = lebar citra.

m_{pq} = *moment* pada kolom p dan baris q.

M = banyak kolom.

N = banyak baris.

$f(x, y)$ = nilai intensitas citra pada titik x dan y.

dengan $f(x, y)$ merupakan nilai piksel pada koordinat (x, y).

Invariant translasi dapat dihitung dengan menggunakan *central moment* yang didefinisikan dengan persamaan.

$$\mu_{pq} = \sum_{x=1}^M \sum_{y=1}^N (x - \bar{x})^p (y - \bar{y})^q f(x, y) \quad (2.7)$$

Keterangan :

(\bar{x}, \bar{y}) = *centroid* citra dengan koordinat \bar{x} dan \bar{y} .

μ_{pq} = *central moment*

Untuk menormalisasi *invariant* translasi dalam bidang citra, *centroid* citra digunakan menentukan *moment* pusat. Koordinat pusat grafitasi dari citra dihitung

dengan menggunakan persamaan 2.7 dengan $\bar{x} = \frac{m_{10}}{m_{00}}$, dan $\bar{y} = \frac{m_{01}}{m_{00}}$ [29].

$$m_{00} = \sum_{x=1}^M \sum_{y=1}^N f(x, y) \quad (2.8)$$

$$m_{10} = \sum_{x=1}^M \sum_{y=1}^N x f(x, y) \quad (2.9)$$

$$m_{01} = \sum_{x=1}^M \sum_{y=1}^N y f(x, y) \quad (2.10)$$

Kemudian *moment* pusat yang telah dinormalisasi memiliki persamaan seperti pada rumus [30]:

$$\eta_{pq} = \frac{\mu_{pq}}{(\mu_{00})^\lambda} \quad (2.11)$$

$$\lambda = \frac{(p+q)}{2} + 1 \quad (2.12)$$

Keterangan :

η_{pq} = normalisasi *moment* pusat

Berdasarkan *moment* ternormalisasi pada persamaan 2.11, Hu memperkenalkan tujuh *invariant* yang diberikan dalam persamaan [31]:

$$H_1 = \eta_{20} + \eta_{02} \quad (2.13)$$

$$H_2 = (\eta_{20} - \eta_{02})^2 + 4\eta_{11}^2 \quad (2.14)$$

$$H_3 = (\eta_{30} - 3\eta_{12})^2 + (3\eta_{21} - \eta_{03})^2 \quad (2.15)$$

$$H_4 = (\eta_{30} + \eta_{12})^2 + (\eta_{21} + \eta_{03})^2 \quad (2.16)$$

$$H_5 = (\eta_{30} - 3\eta_{12})(\eta_{30} + \eta_{12})[(\eta_{30} + \eta_{12})^2 - 3(\eta_{21} + \eta_{03})^2] + (3\eta_{21} - \eta_{03})(\eta_{21} + \eta_{03}) [3(\eta_{30} + \eta_{12})^2 - (\eta_{21} + \eta_{03})^2] \quad (2.17)$$

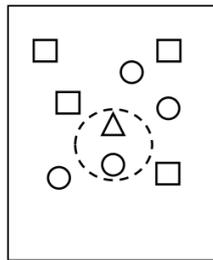
$$H_6 = (\eta_{20} - \eta_{02})[(\eta_{30} + \eta_{12})^2 - (\eta_{21} + \eta_{03})^2] + 4\eta_{11}(\eta_{30} + \eta_{12})(\eta_{21} + \eta_{03}) \quad (2.18)$$

$$H_7 = (3\eta_{21} - \eta_{03})(\eta_{30} + \eta_{12}) [(\eta_{30} + \eta_{12})^2 - 3(\eta_{21} + \eta_{03})^2] - (\eta_{30} + 3\eta_{12})(\eta_{21} + \eta_{03}) [3(\eta_{30} + \eta_{12})^2 - (\eta_{21} + \eta_{03})^2] \quad (2.19)$$

Persamaan 2.13 H1 dianalogikan *moment* inersia yang berada di sekitar pusat citra dan memiliki refleksi simetri yang tidak berubah saat dilakukan pencerminan. Persamaan 2.14 hingga 2.18 yaitu H2-H6 adalah refleksi simetris seperti H1. Persamaan 2.19 yaitu H7 adalah antisimetri yang artinya dapat berubah saat dilakukan pencerminan pada citra. Berdasarkan persamaan diatas mengenai tujuh fitur *invariant moment* yang mana memiliki fungsi untuk menjelaskan objek dengan menghitung area objek tersebut. Fitur ini menggunakan dasar *moment* pusat yang ternormalisasi seperti pada persamaan 2.12. Dengan *moment* yang dihasilkan dapat digunakan dalam pencekalan, rotasi gambar, serta translasi [32].

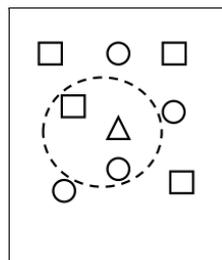
2.2.6 K-NEAREST NEIGHBOR

K-Nearest-Neighbor adalah algoritma yang mengklasifikasi objek untuk data baru. Proses pengklasifikasian objek data baru ini akan dilatih berdasarkan atribut dan data sample latih. Data pembelajaran digambarkan sebagai artibut digital berdimensi-n. Sedangkan, k adalah tetangga terdekat yang akan mencari nilai k data latih yang jaraknya paling dekat dengan data uji dengan menghitung nilai jarak yang mempresentasikan data menggunakan rumus jarak *Euclidean* [7]. Setelah mencari sejumlah k objek data (data latih) yang paling dekat jaraknya dengan data uji yang diberikan, kemudian memilih kelas dengan jumlah *voting* terbanyak [33]. KNN memiliki kelebihan, pertama yaitu relatif tidak sensitif terhadap *error* dalam dataset, dan yang kedua adalah K-NN dapat digunakan untuk memproses dataset berukuran besar [34].



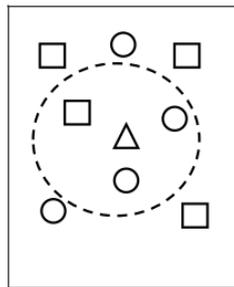
Gambar 2. 7 Ilustrasi 1-nearest neighbor[34]

Pada ilustrasi Gambar 2.7 misalkan bentuk segitiga adalah objek yang akan diprediksi kelasnya. Jika nilai k sama dengan satu maka hanya satu tetangga terdekat yang akan diperhitungkan jaraknya kemudian akan diklasifikasi berdasarkan 1 data tersebut. Pada ilustrasi bahwa bentuk lingkaran adalah 1 data terdekat sehingga bentuk segitiga akan terprediksi pada kelas bentuk lingkaran [34].



Gambar 2. 8 Ilustrasi 2-nearest neighbor[34]

Pada ilustrasi Gambar 2.8 jika nilai k sama dengan dua maka dua tetangga terdekat yang akan diperhitungkan jaraknya. Pada ilustrasi bahwa bentuk lingkaran dan persegi adalah 2 data terdekat. Bentuk lingkaran dan persegi adalah 2 kelas berbeda sehingga bentuk segitiga akan memprediksi melalui jarak terdekat dari 2 data tersebut yang mana bentuk lingkaran merupakan data terdekat, maka bentuk lingkaranlah yang akan menjadi kelas bentuk segitiga[34]



Gambar 2. 9 Ilustrasi 3-nearest neighbor [34]

Sedangkan pada ilustrasi Gambar 2.9 jika nilai k sama dengan tiga maka tiga tetangga terdekat yang akan diperhitungkan jaraknya. Pada ilustrasi bahwa bentuk lingkaran dan persegi adalah 3 data terdekat yang mana terdapat 2 data lingkaran sehingga bentuk segitiga akan terklasifikasi sebagai kelas lingkaran karena bentuk lingkaran selain memiliki jarak terdekat juga memiliki frekuensi yang lebih banyak dari bentuk persegi [34]. Algoritma KNN menggunakan klasifikasi ketetanggaan sebagai nilai prediksi dari sampel uji yang baru. Jarak yang digunakan adalah jarak *euclidean distance*. Jarak *euclidean* adalah jarak yang paling umum digunakan pada data numerik didefinisikan sebagai berikut [31]:

$$d_i = \sqrt{\sum_{i=0}^p (x_{2i} - x_{1i})^2} \quad (2.20)$$

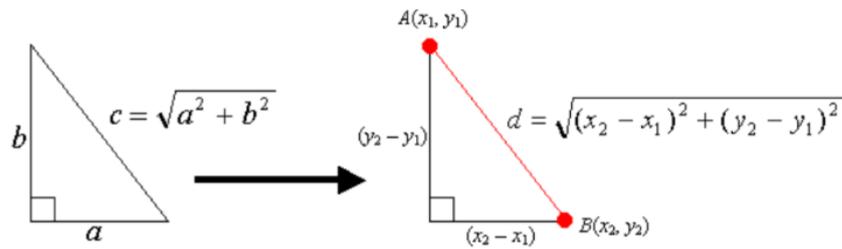
Keterangan :

x_{2i} = data *testing*

x_{1i} = data *training*

p = jumlah dimensi data

Jarak *euclidean* dapat dihitung berdasarkan fitur yang dimiliki citra. Fitur citra diantaranya adalah fitur warna, tekstur, dan bentuk. Proses perhitungannya adalah dengan melakukan *encode* dan menghasilkan 128 *deimensional face encodings*. Lalu kemudian hitung jarak terdekat dengan data yang ada di database menggunakan *euclidean distance* matrik. Hasil kedekatan yang paling dekat menjadi hasil klasifikasinya nantinya [35].



Gambar 2. 10 Ilustrasi perhitungan algoritma KNN [35]

Langkah-langkah untuk menghitung metode *K-Nearest Neighbor* antara lain :

1. Menentukan parameter k (jumlah tetangga paing dekat).
2. Menghitung kuadrat jarak euclidean objek terhadap data *training* yang diberikan.
3. Mengurutkan hasil poin secara *ascending* (berurutan dari nilai tinggi ke rendah).
4. Mengumpulkan kategori Y (klasifikasi *nearest neighbor* berdasarkan nilai k).