

BAB 2

DASAR TEORI

2.1 KAJIAN PUSTAKA

Pada penelitian [6] jumlah kelas penyakit yang diteliti adalah dua yang semua disebabkan oleh jamur yang menginfeksi tanaman padi yaitu *brown spot* dan *Blastt*. Metode yang digunakan yaitu klasifikasi dengan *K-Nearest Neighbour* (KNN) dan dalam proses identifikasi penyakit pada daun akan menggunakan fitur *geometric* seperti *Area*, *Major Axis*, *Minor Axis* dan *perimeter* diantara infeksi penyakit pada daun. Sedangkan untuk penelitian oleh penulis akan menggunakan bantuan metode fraktal. Pada penelitian ini citra *Red Green Blue* (RGB) akan dikonversi ke *Hue Saturation Value* (HSV) dan menggunakan 330 *dataset* yang menggunakan ratio 60:40 dimana terbagi 60% untuk data latih dan 40% sisanya untuk data uji. Dari penelitian ini mendapatkan jumlah nilai akurasi sebesar 76,59%.

Pada penelitian [10] model pengolahan citra digunakan untuk mendeteksi penyakit pada daun padi yaitu *Blast*. Penelitian ini menggunakan metode dimensi fraktal serta teori *Chaos Game* dimana untuk mengidentifikasi melalui pola pada daun dalam dimensi fraktal dan menggunakan perhitungan ratio dengan *Box-Counting* dan teori *chaos game* yang akan digunakan untuk menyusun ulang pola penyakit daun sehingga akan dapat melakukan identifikasi sesuai fitur *self-similarity* yang ada pada fraktal. Setelah analisis fraktal untuk penyakit daun berjalan dengan *threshold* 0,5 dan *box count dimension* 1,43 berhasil ditemukan enam sudut yang diidentifikasi sesuai pola penyakit daun menunjukkan pola bentuk *hexagonal*. Dari tujuh titik sampel dikembangkan untuk membuat fraktal dengan *chaos game* yang selanjutnya akan dibandingkan dengan fraktal pada pola daun padi yang sehat. Namun pada penelitian ini tidak diberi penjelasan mengenai *dataset* yang digunakan.

Pada penelitian [14] melakukan identifikasi penyakit pada daun padi menggunakan metode *Adaptive Neuro Fuzzy Inference System* (ANFIS) dengan proses yang diawali dari *thresholding* citra menggunakan metode Otsu dan

dilanjutkan dengan pengelompokkan data dengan menggunakan *K-Means* untuk mendapatkan nilai koefisien adaptif. Pelatihan dan uji coba untuk dilakukan untuk mengenali sebanyak dua jenis penyakit yaitu bercak Coklat Sempit dan *Blast*. Memiliki jumlah data uji sebanyak 40 diperoleh nilai rata-rata akurasi terendah 80.7% dan dengan hasil akurasi terbaik sebesar 84.5% dengan maksimum error sebesar 1×10^{-12} .

Pada penelitian [15] melakukan identifikasi terhadap tiga jenis penyakit pada daun padi yaitu *Leaf Smut*, *Brown Spot*, dan Hawar daun Bakteri (*Bacterial Leaf Blight*) dengan menggunakan sebanyak 120 *dataset* yang diambil dari UCI *repository*. Penelitian ini menggunakan metode klasifikasi dengan *K-Nearest Neighbour* dengan ekstraksi ciri fitur yang menggunakan *Gray Level Co-occurrence Matrix* (GLCM). KNN pada penelitian ini menggunakan nilai maksimum *k value* 1 hingga 20 dan diperoleh hasil terbaik pada $k = 11$ dengan 65,83% dan kappa 0,485.

Pada penelitian [16] merupakan bentuk prototipe untuk sistem deteksi dan klasifikasi penyakit pada daun padi yang dikembangkan dengan beberapa teknis yang digunakan pada pengolahan termasuk GLCM. Sistem ini dicoba untuk mendeteksi tiga citra jenis penyakit yaitu *Bacterial leaf blight*, *Brown spot*, *Leaf smut* yang diambil langsung dari sawah oleh peneliti menggunakan kamera digital dengan jumlah 120 data citra yang terbagi menjadi masing-masing 40 gambar dari setiap jenis penyakitnya. Dalam penelitian ini menggunakan empat teknik *background removal* dan tiga teknik segmentasi serta melakukan ekstraksi warna, bentuk, dan tekstur. Dengan menggunakan *Support Vector Machine* (SVM) sebagai klasifikasi *multiclass* mendapatkan hasil akurasi pada data latih sebesar 93,33% dan data uji dengan akurasi 73,33% pada *dataset*. Peneliti juga menambahkan 5 dan 10 *fold cross-validations* dengan hasil yang didapatkan adalah akurasi sebesar 83,80% dan 88,87%.

Pada penelitian [17] mendeteksi penyakit pada daun padi dengan menggunakan metode *Convolutional Neural Network* (CNN) dengan jumlah *dataset* sebanyak 900 sampel citra daun padi dimana terbagi menjadi tiga kelas penyakit yang akan dideteksi yaitu *Blast*, *Brown Spot* dan Hispa dengan penggunaan 780 data latih dan 180 data pengujian. Setiap citra diolah menjadi format yang sama berupa 224 x 224 piksel Untuk pemodelan menggunakan empat

convolutional layers dan empat *max Pooling layer* dengan hasil rata-rata akurasi data latih sebesar 92% serta memiliki akurasi 77% untuk data validasi.

Pada penelitian [18] mendeteksi penyakit pada daun padi menggunakan metode klasifikasi jaringan saraf tiruan yaitu *Backpropagation Neural Network* (BPNN) dengan menggunakan sebanyak 150 *dataset* untuk data latih dibagi menjadi lima kelas penyakit yang dideteksi yaitu *Blast*, *Bacterial leaf blight*, *Brown spot*, *Tungro*, dan *Sheath blight*. Data yang didapatkan berasal dari Litbang pertanian kota Surakarta, untuk data yang diuji menggunakan 30 *dataset*. Ekstraksi fitur pada penelitian ini menggunakan metode *Gray-Level Co-Occurance Matrix* (GLCM) dengan lima fitur. Hasil pengujian sistem klasifikasi penyakit daun padi ini mendapatkan akurasi 80%.

Berikut ini pada Tabel 2.1 menampilkan Kajian Pustaka secara lebih ringkas dari penelitian serupa mengenai klasifikasi dan deteksi penyakit pada daun padi serta perbedaan yang ada terhadap penelitian yang diambil.

Tabel 2.1 Ringkasan Kajian Pustaka

JUDUL	PENULIS	PENYAKIT	METODE	PERBEDAAN
Recognition of Diseases in Paddy Leaves Using kNN Classifier [6]	M. Suresha, Shreekanth, dan Thirumalesh B, V.	<i>Blast</i> dan <i>Brown Spot</i> (Bercak Coklat) Jumlah: 330 Data	KNN, Fitur <i>Geometric</i> meliputi Area. Perimeter, <i>Minor</i> dan <i>Major Axis</i>	Fitur ekstraksi yang digunakan.
Chaos Game and Fractal Dimension in <i>Image processing</i> to Detect Rice Leaf <i>Blast</i> Disease [10]	Mia Cruz Torres-dela, Murugananthan Velayutham, Vinukumar Luckose, dan Venu, Surendrababu.	<i>Blast</i> Jumlah: 300 Data	<i>Chaos game</i> dan Fraktal	Hanya deteksi dengan bantuan <i>chaos game theory</i> penyakit yang diidentifikasi hanya <i>Blast</i>

JUDUL	PENULIS	PENYAKIT	METODE	PERBEDAAN
Implementasi Citra Digital Untuk Identifikasi Penyakit Pada Daun Padi Menggunakan ANFIS [14]	Candra Dewi, Elok Fatma Anjarwati, dan Imam Cholissodin	Bercak Coklat dan <i>Blast</i> Jumlah: 140 Data	ANFIS dan segmentasi Otsu	Metode yang digunakan berbeda karena menggunakan ANFIS dan juga penyakit yang diidentifikasi yang berbeda
Rice Leaf Disease Image Classifications Using KNN Based on GLCM Feature Extraction [15]	R. A. Saputra, Suharyanto, S. Wasiyanti, D. F. Saefudin, A. Supriyatna, dan A. Wibowo.	<i>Bacterial Leaf Blight</i> (hawar daun bakteri), Bercak Coklat dan <i>Leaf Smut</i> Jumlah: 120 Data	<i>K-Nearest Neighbour</i> dan <i>Gray-Level Co-Occurance Matrix</i> (GLCM)	Menggunakan KNN namun tidak dengan fraktal dan perbedaan pada penyakit yang diidentifikasi.
Detection and Classification of Rice Plant Diseases [16]	Harshadkumar B. Prajapati, Jitesh P. Shah, dan Vipul K. Dabhi	<i>Bacterial Leaf Blight</i> (hawar daun bakteri), <i>Brown spot</i> dan <i>Leaf Smut</i> Jumlah: 120 Data	<i>Gray-Level Co-Occurance Matrix</i> dan <i>Support Vector Machine</i>	Perbedaan pada metode dan <i>dataset</i> penyakit yang diidentifikasi serta data yang RGB akan diconvert ke HSV

JUDUL	PENULIS	PENYAKIT	METODE	PERBEDAAN
Deteksi Penyakit Pada Tanaman Padi Menggunakan Pengolahan Citra dengan Metode Convolutional Neural Network [17]	Syaikhul Anam A., Musthafa Aziz, dan Oddy V. Putra	<i>Blast</i> , Bercak Coklat dan Hispa Jumlah: 900 Data	<i>Convolution Neural Network</i> (CNN)	Berbeda dari segi metode dan juga penyakit daun padi yang diidentifikasi kecuali <i>Blast</i>
Klasifikasi Penyakit Daun Padi Berdasarkan Hasil Ekstraksi Fitur GLCM Interval 4 Sudut [18]	Jani Kusanti dan Noor Abdul Haris	<i>Blast</i> , <i>Bacterial leaf blight</i> , <i>Brown spot</i> , Tungro, dan <i>Sheath blight</i> Jumlah: 180 Data	<i>Backpropagation Neural Network</i> (BPNN) dan GLCM	Menggunakan metode saraf tiruan dan kelas penyakit yang lebih banyak
Klasifikasi Penyakit Pada Daun Padi Menggunakan Metode Fraktal Dan <i>K-Nearest Neighbour</i>	Rahadian Agsis Darmawan	<i>Bacterial Leaf Blight</i> , <i>Blast</i> dan tungro Jumlah : 900 Data	Fraktal dan KNN	Menggunakan ekstraksi ciri fraktal yang diklasifikasikan dengan KNN

2.2 DASAR TEORI

2.2.1 Padi

Padi merupakan termasuk makanan pokok paling banyak digemari sebagai panganan pokok konsumsi di seluruh dunia. Tanaman ini banyak ditanam terutama

pada daerah Amerika Selatan dan tentunya di Asia. Padi memiliki nama ilmiah *Oryza sativa L.* adalah tanaman yang dibudidaya meski ada juga yang merupakan padi liar. Padi sendiri diduga dimulai dari India atau Indocina, pada tahun 1500 SM mulai terjadi pembudidayaan tanaman padi beberapa daerah di Indonesia. Di negara agraris seperti India, Cina, Indonesia dan Bangladesh, padi merupakan tanaman komoditas yang sering dijumpai di sebagian besar daerah. Padi jadi penghasil sebagian besar makanan pokok konsumsi masyarakat. Secara budidaya padi dibagi menjadi dua jenis yaitu padi gogo dan padi rawa. Tidak seperti padi pada umumnya, padi gogo tidak ditanam di sawah melainkan ditanam di ladang atau kebun. Padi gogo memiliki kelebihan yaitu tidak diperlukannya sistem irigasi khusus. Padi gogo sering ditemui di daerah dengan sumber air yang kurang melimpah, seperti di daerah Lombok. Sedangkan padi rawa adalah padi yang sering ditanam di persawahan sehingga padi jenis ini membutuhkan genangan air terus menerus maka diperlukan irigasi secara konsisten [19].

2.2.2 Penyakit Daun Padi

Penyakit padi dapat sangat berdampak pada penurunan hasil panen yang umumnya disebabkan oleh bermacam jenis bakteri, virus, atau jamur. Di Indonesia, banyak jenis penyakit yang biasa ditemukan pada tanaman padi ialah *Bacterial Leaf Blight* (*Xanthomonas campestris pv. oryzae*), penyakit tungro (virus tungro), *Blast*, bercak hitam dan masih banyak lainnya. Contoh hasil panen yang tidak maksimal yang diakibatkan oleh gangguan *Bacterial Leaf Blight* mengurangi jumlah panen hingga berkisar antara 15-24%. Perubahan iklim global yang berdampak terhadap anomali iklim mendorong perkembangan hama maupun gulma dan penyakit yang mengancam keselamatan produksi padi [20].

2.2.2.1 Bacterial Leaf Blight

Pada Penyakit Retak / Hawar Daun (Indonesia) / Lodoh (Jawa) atau BLB (*Bacterial Leaf Blight*) atau HDB disebabkan oleh bakteri *Xanthomonas campestris pv. Oryzae*. Akibat serangan OPT ini, pertumbuhan tanaman menjadi lambat, mutu benih buruk, dan proporsi beras patah tinggi. Gejala ini bervariasi Kresek /Layu

Daun Terjadi pada tanaman muda (biasanya muncul 1-6 minggu setelah tanam) dan gejala awal ditemukan sebagai basah, garis berbintik-bintik pada tepi daun atau bagian daun yang terkena. Bintik-bintik itu kemudian meluas menjadi abu-abu kehijauan, seluruh daun menyusut dan akhirnya layu seperti terbakar. Gejala pemisahan adalah gejala kesulitan yang paling merusak. Hawar Terjadi pada Terjadi pada tanaman dewasa, gejala pertama berupa bercak basah pada salah satu atau kedua sisi daun beberapa cm dari ujung daun. Lebar abu-abu kehijauan, basah, daun keriting, kering dan abu-abu muda. pertama kali muncul sebagai garis-garis berisi air pada tepi daun [21]. Pada Gambar 2.1 menampilkan contoh citra gambar daun padi yang terjangkit penyakit *Bacterial leaf blight* yang diambil dari *dataset* penelitian ini.



Gambar 2.1 Daun Padi Terjangkit *Bacterial Leaf Blight* [13].

2.2.2.2 Blast

Penyakit *Blast* awalnya ditemukan pada padi yang ditanam pada lahan tadah hujan atau yang lebih dikenal dengan nama padi gogo penyakit ini akibat adanya jamur *Pyricularia grisea* pada saat tidak berkembang biak dan *Magnaporthe grisea* pada saat fase menebarkan spora namun mulai tersebar juga di beberapa lahan sawah irigasi. Menyerang daun dan menimbulkan gejala dengan bentuk bercak coklat berbentuk wajik. Selama masa pertumbuhan reproduksi tanaman padi, gejala *Blast* yang dikenal sebagai *Blast* leher berkembang di batang/leher *spurge*. Infeksi dapat menyerang bagian biji-bijian dan buah dapat membawa patogen sebagai

patogen yang terbawa benih. Penyakit *Blast* juga dapat berkembang pada tanaman selain padi, seperti sorgum, gandum, dan rerumputan. Gejala yang muncul seperti leher busuk atau patah mengganggu proses pertumbuhan leher padi dan banyak menghasilkan gabah kosong yang sangat berpengaruh pada penurunan hasil kualitas biji. [22]. Pada Gambar 2.2 dibawah memperlihatkan daun padi yang mengalami bercak coklat berbentuk wajik yang merupakan ciri dari penyakit daun padi *Blast*.



Gambar 2.2 Daun Padi Terjangkit *Blast* [13].




2.2.2.3 Tungro

Tungro merupakan salah satu penyakit tanaman padi yang paling sering ditemui di negara-negara Asia tropis, Tungro disebabkan oleh infeksi kompleks dari dua virus, yaitu *Rice Tungro Bacilliform Virus* (RTBV) dan *Rice Tungro Spherical Virus* (RTSV). Gejala penyakit tungro pada daun muda yang terinfeksi tungro virus, terlihat bercak dan klorosis diantara. Kurap jenis ini disebabkan oleh RTBV, tetapi RTBV hanya dapat ditularkan jika ada RTSV. RTBV menyebabkan gejala mulai dari daun menguning hingga kemerahan dan kerdil atau menyebabkan tinggi tanaman tidak rata sehingga dapat ditemukan tanaman memiliki gabah hamp atau kosong, sedangkan RTSV berperan dalam penularan kedua jenis virus tersebut melalui wereng hijau yang memiliki nama latin (*Nephotettix virescens*) [23]. Pada Gambar 2.3 menampilkan gambar tanaman padi yang terserang penyakit oleh virus Tungro dengan ciri daun menguning dan pertumbuhan daun yang melengkung atau terjadi patahan.



Gambar 2.3 Daun Padi Terjangkit Tungro [13].

Tabel 2.2 Perbandingan Fisik Jenis Penyakit Daun Padi

Jenis Penyakit	Bentuk Penyakit	Ciri Fisik
<i>Bacterial Leaf Blight</i>		Dari warna hijau dengan garis yang berwarna coklat hingga abu-abu
<i>Blast</i>		Bercak coklat dengan bentuk acak seperti melingkar dan putih keabu-abuan
Tungro		Kuning layu hingga jingga serta bentuk tulang daun padi yang tidak beraturan

Pada Tabel 2.2 diatas ditampilkan penyakit daun yang akan diklasifikasikan pada penelitian ini dengan metode yang dipilih berupa penyakit Tungro, *Blast*, dan *Bacterial leaf blight* dengan gambar atau citra diambil dari *dataset* [13] yang digunakan dalam penelitian ini beserta karakteristik daun yang terjangkit penyakitnya.

2.2.3 *Machine Learning*

Machine learning adalah aplikasi atau bagian dari kecerdasan buatan yang memungkinkan sistem belajar secara otomatis tanpa diprogram. Fokus pengembangan program adalah untuk dapat mengakses data dan belajar dari data tersebut. Selain itu, *machine learning* dapat didefinisikan sebagai algoritma yang bertujuan untuk menemukan pola dalam data. Sejarah *machine learning* dimulai pada tahun 1943 dimana Warren Mcculloch memperkenalkan konsep dan cara kerja jaringan saraf. Perkembangan pada abad 21 dimulai dengan tahun 2012 Google melalui program Google Brain mengembangkan deep neural network untuk mendeteksi objek-objek dalam Youtube. Perkembangan saat ini *machine learning* banyak merambah beberapa sektor diantaranya adalah analisis data penjualan personalisasi, deteksi atau klasifikasi, sistem rekomendasi dan *dynamic pricing*. [24].

2.2.4 *Jenis Metode Machine Learning*

Machine learning memiliki beberapa jenis pembelajaran yaitu *Supervised Learning* yang merupakan algoritma belajar dengan membandingkan *output* sebenarnya dengan *output* yang benar untuk menemukan *error* atau kesalahan. Pembelajaran ini dibagi menjadi beberapa metode yang terdiri dari metode berdasarkan klasifikasi dan regresi. Pembelajaran jenis selanjutnya adalah *Semi-Supervised Learning* yang menggabungkan contoh data *input* berlabel dan tidak berlabel untuk menghasilkan fungsi yang sesuai. *Unsupervised Learning* digunakan untuk mengeksplorasi data dan menemukan strukturnya, karena data yang diolah tidak memiliki label dan sistem tidak mengetahui jawaban atau *output* yang tepat, pembelajaran ini menggunakan cara *clustering*. Selanjutnya *Reinforcement Learning* yang mempelajari kebijakan tentang pengambilan keputusan berdasarkan pengamatan. Setiap tindakan memiliki beberapa dampak serta memberikan umpan balik yang memandu algoritma pembelajaran [25]. Dalam penelitian ini menggunakan cara spesifik berupa klasifikasi dibawah ini beberapa penjelasan mengenai jenis metode pada *machine learning* klasifikasi.

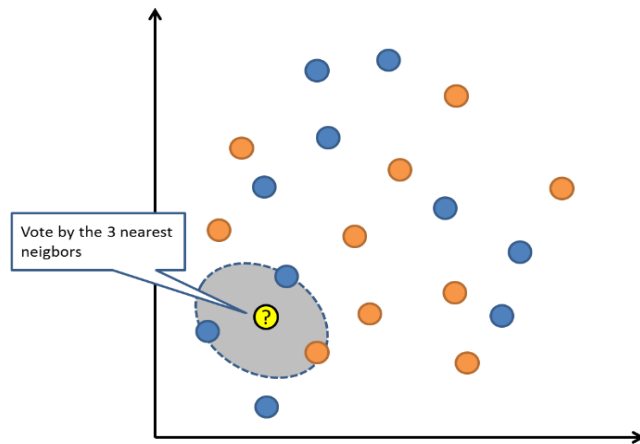
2.2.4.1 K-Nearest Neighbour

KNN merupakan algoritma klasifikasi yang bekerja dengan cara memasukkan sejumlah nilai K data terdekat (tetangganya) sebagai acuan untuk menentukan kelas dari data baru. Algoritma ini mengklasifikasikan data berdasarkan *similarity* atau kemiripan atau kedekatannya terhadap data lainnya [26]. KNN memiliki cara kerja dimana *input* data dari pelatihan berupa fitur karakteristik kemudian akan ditentukan jumlah nilai K yang selanjutnya dilakukan penghitungan jarak sehingga diperoleh tetangga terdekat yang kemudian untuk melakukan prediksi ditentukan dengan cara *vote* mayoritas dimana titik terbanyak akan menjadi hasil prediksi. Tujuan dari algoritma KNN adalah untuk mengklasifikasikan objek baru berdasarkan atribut dan sampel pelatihan. Hasil sampel uji baru diklasifikasikan ke dalam sebagian besar kategori KNN. Proses klasifikasi algoritma ini tidak menggunakan model apapun untuk pencocokan dan hanya mengandalkan memori. Algoritma KNN menggunakan klasifikasi tetangga terdekat. Beberapa contoh jarak yang digunakan pada KNN adalah *Euclidean Distance*, *Manhattan Distance*, *Minkowski Distance*, *Chebychev Distance*. sebagai nilai prediksi dari data uji coba baru. Jarak *Euclidean* adalah jarak yang paling umum digunakan dalam data numerik [27].

Rumus *Euclidean* yang digunakan untuk mencari jarak antara dua titik yaitu titik pada dua dimensi untuk mengukur jarak koordinat *latitude* dan *longitude*.

$$D = \sqrt{(x_2 - x_1)^2 + (y_2 - y_1)^2} \quad (2.1)$$

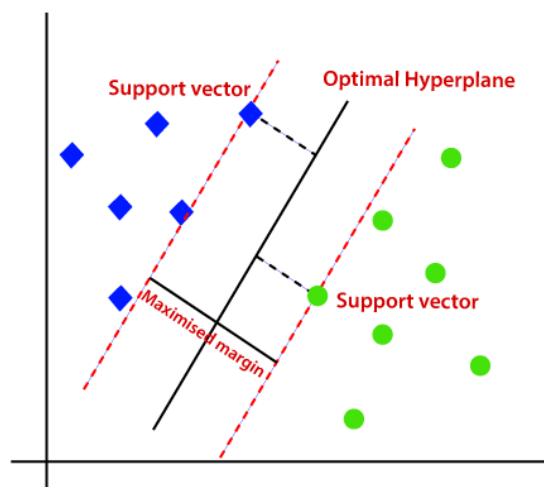
Dimana (x_1, y_1, \dots, x_n) adalah koordinat dari titik pertama dan (x_2, y_2, \dots, x_n) adalah koordinat dari titik kedua. Rumus ini dapat diperluas ke ruang berdimensi lebih tinggi dengan menambahkan lebih banyak koordinat (x_3, x_4, \dots, x_m) untuk setiap titik, dan (x_3, x_4, \dots, x_m) untuk titik kedua. Pada Gambar 2.4 menampilkan ilustrasi penggunaan algoritma KNN dalam menentukan kelas yang didapatkan dari menghitung jarak ke tetangga terdekat.



Gambar 2.4 Bentuk Penentuan Nilai K.

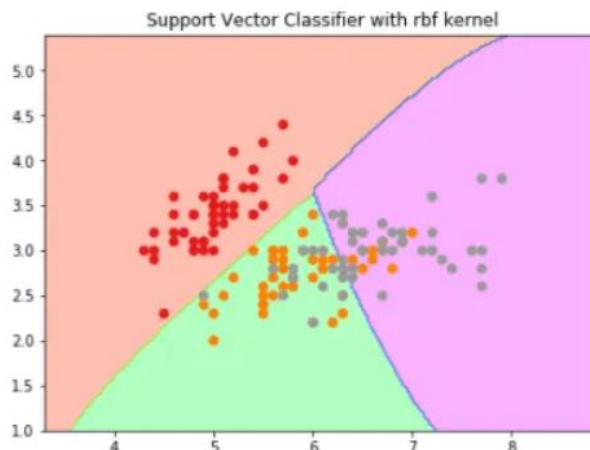
2.2.4.2 Support Vector Machine

Support Vector Machine (SVM) adalah algoritma *supervised learning* yang berfungsi pada kumpulan data yang lebih kecil digunakan untuk tugas klasifikasi dan regresi. SVM bekerja dengan mencari *hyperplane* (garis atau bidang) terbaik yang memisahkan dua kelas data sehingga jarak antara garis tersebut dengan titik-titik data dari kedua kelas adalah maksimal. Dasar dari klasifikasi menggunakan SVM adalah mengetahui empat konsep seperti memisahkan *hyperplane*, mendapatkan *margins* maksimal *hyperplane*, menentukan *soft margins*, dan menggunakan fungsi kernel. Memiliki dua cara penghitungan yang pertama adalah *Linear* dan *Non-Linear* (Kernel).



Gambar 2.5 Bentuk Penentuan Data pada SVM [28].

Pada Gambar 2.5 diatas merupakan ilustrasi klasifikasi data yang dilakukan dengan algoritma SVM. Selanjutnya fungsi Kernel adalah metode matematis yang memungkinkan SVM untuk melakukan klasifikasi secara dua dimensi dari semula yang hanya memiliki satu dimensi data. Secara umum, fungsi kernel memproyeksikan data dari ruang berdimensi rendah ke ruang dimensi yang lebih tinggi. Penggunaan kernel yang tepat akan mendapatkan hasil yang lebih tinggi pada ruang dimensi. Beberapa jenis kernel diantaranya adalah *Polynomial Kernel*, *Sigmoid Kernel*, *Bessel Function Kernel*, dan *Radial Base Function (RBF) Kernel* [29]. Gambar 2.6 menampilkan salah satu penggunaan kernel yaitu kernel RBF pada algoritma SVM.



Gambar 2.6 Penggunaan Kernel pada SVM [28].

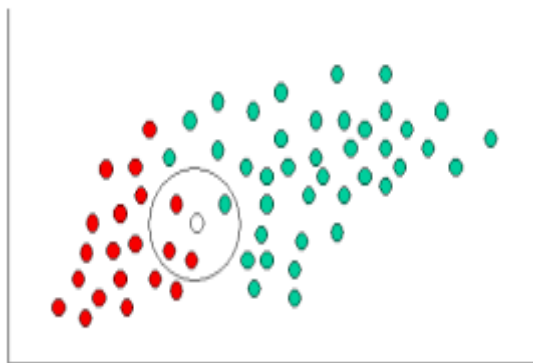
2.2.4.3 Naive Bayes

Naive Bayes classifier adalah algoritma *supervised learning* yang populer digunakan untuk tugas klasifikasi seperti klasifikasi teks. Merupakan algoritma pembelajaran yang generatif yaitu berarti memodelkan distribusi *input* untuk kelas atau kategori tertentu. Pendekatan ini didasarkan pada asumsi bahwa fitur-fitur dari data *input* bersifat independen secara kondisional berdasarkan kelasnya, memungkinkan algoritma untuk membuat prediksi dengan cepat dan akurat. Dalam statistik, *Naive Bayes* dianggap sebagai pengklasifikasi probabilistik sederhana yang menerapkan teorema *Bayes*.

$$P(A|B) = \frac{P(B|A) * P(A)}{P(B)} \quad (2.2)$$

Dalam Persamaan 2.2 diatas dengan menggunakan teorema Bayes. A adalah hipotesis, dan B adalah bukti. $P(B|A)$ adalah probabilitas B jika A benar. $P(A)$ dan $P(B)$ adalah probabilitas bebas dari A dan B. Teorema ini didasarkan pada probabilitas hipotesis. Naive Bayes classifier mengasumsikan bahwa semua fitur dalam data input tidak tergantung satu sama lain, yang seringkali tidak benar dalam skenario dunia nyata. Naive Bayes classifier digunakan secara luas karena efisiensi dan kinerjanya yang baik di banyak aplikasi dunia nyata. Teknik ini melibatkan penggunaan fungsi kernel untuk memperkirakan fungsi banyaknya probabilitas dari data input yang memungkinkan pengklasifikasi untuk meningkatkan kinerjanya dalam skenario kompleks dimana distribusi data tidak terdefinisi dengan baik [30].

Pada Gambar 2.7 menampilkan ilustrasi data baru yang didapatkan algoritma *naive bayes* yang akan diklasifikasikan.

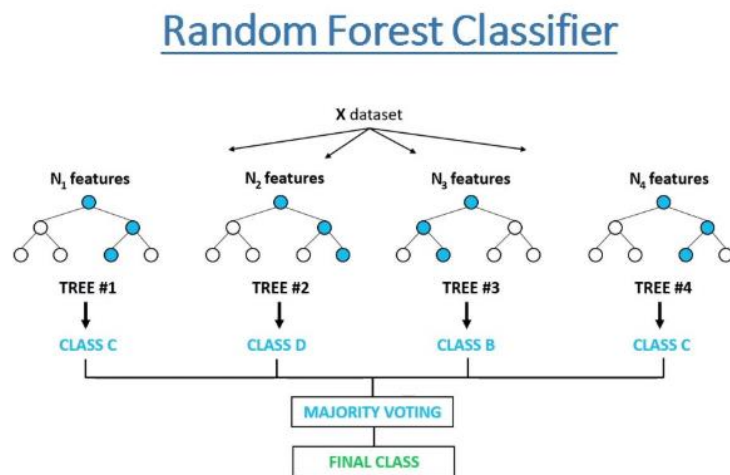


Gambar 2.7 Contoh Data pada *Naive Bayes* [29].

2.2.4.4 *Random Forrest*

Random Forrest adalah algoritma *machine learning supervised* yang dikembangkan oleh Leo Breiman dan Adele Cutler. Memanfaatkan pengambilan keputusan dari beberapa *decision tree* untuk menghasilkan prediksi atau klasifikasi. Memiliki cara dengan menggabungkan output dari setiap *decision tree*, model berbasis pohon melibatkan partisi fungsi pendefinisian *dataset* yang diberikan ke dalam dua kelompok berdasarkan kriteria tertentu sampai kondisi berhenti yang

telah ditentukan terpenuhi. Di bagian bawah pohon keputusan disebut node daun atau *leaf nodes*. Algoritma *random forest* memberikan hasil yang terkonsolidasi dan lebih akurat. Cara kerja algoritma ini yaitu pertama mengambil sampel *dataset* secara acak, berikutnya algoritma akan menciptakan *decision tree* pada setiap sampel yang terpilih sehingga akan menghasilkan *output* dari keputusan, lalu membuat *vote* pada hasil prediksi pada masalah klasifikasi akan menentukan *mode* dan pada masalah regresi akan menggunakan *mean*. Langkah terakhir untuk mendapatkan hasil ditentukan dari hasil *vote* terbanyak dan menjadikan hasil prediksi *output*. Pada Gambar 2.8 menampilkan contoh klasifikasi kelas data berdasarkan algoritma *random forest*.



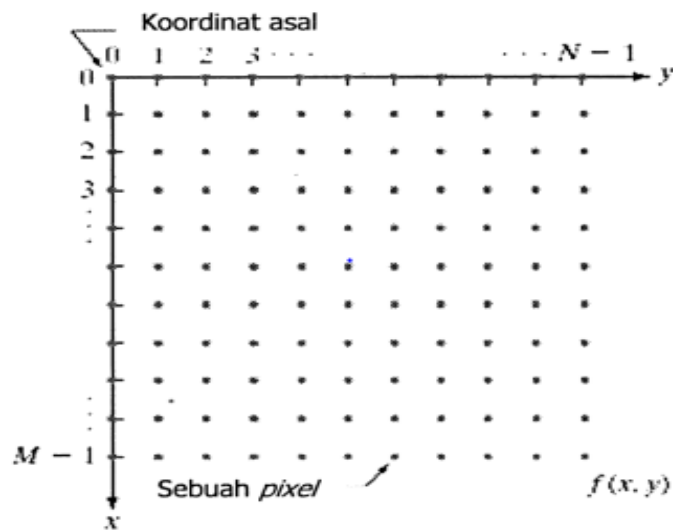
Gambar 2.8 Bentuk Algoritma *Random Forrest* [31].

2.2.5 *Image Processing*

Image processing atau pengolahan citra memungkinkan seseorang dapat mengambil suatu informasi yang ada dalam suatu gambar. Memiliki *input* adalah citra (gambar) dan keluarannya adalah informasi atau pesan tertentu seperti identitas dari sebuah objek yang direkam kamera, atau informasi jumlah objek yang terdeteksi kamera. Suatu teknik mengolah dengan cara melakukan transformasi citra input untuk mendapatkan hasil output dengan kualitas lebih baik disebut *image processing*. Pengolahan citra memiliki manfaat mengidentifikasi objek, menghilangkan cacat pada citra, dan sebagai penggabungan dengan bagian citra yang lain [32].

2.2.6 Citra Digital

Citra digital adalah pemrosesan yang berbentuk diskrit dalam fungsi intensitas cahaya pada bidang dua dimensi. Ciri ini dibentuk oleh sekumpulan piksel, atau gambar elemen. Mereka memiliki koordinat (x,y) , yang menunjukkan letak atau posisi piksel dalam gambar, dan amplitudo $f(x,y)$, yang menunjukkan luasnya. yang secara keseluruhan memiliki nilai tertentu dalam bentuk diskrit menunjukkan intensitas warna. Citra berisi nilai-nilai *real* maupun kompleks yang berbentuk *array* direpresentasikan dengan deretan bit tertentu [33]. Gambar 2.9 merupakan ilustrasi koordinat piksel yang ada pada citra digital.



Gambar 2.9 Koordinat Citra Digital [33].

2.2.7 Dataset

Dataset ini adalah citra yang akan diolah dan diklasifikasi yang akan dibagi menjadi data latih dan data uji. Data latih merupakan citra masukan sebagai parameter yang nantinya akan disimpan dalam *database* sistem. Data uji merupakan citra masukan yang akan dibandingkan dengan data citra latih yang nantinya akan disimpan juga dalam *database* sistem.

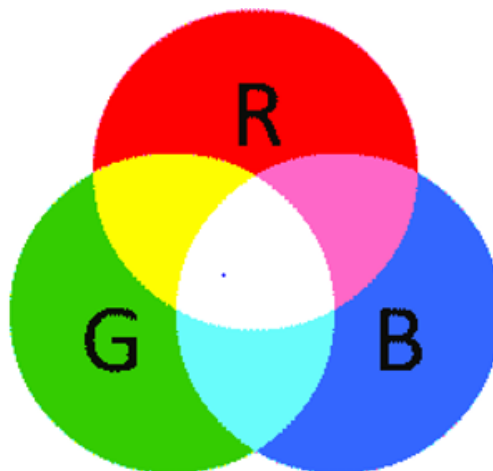
2.2.8 *Cropping*

Cropping merupakan proses untuk memotong citra pada koordinat tertentu pada area citra, pada proses ini dilakukan *cropping* untuk mengurangi *background* pada data citra agar hanya terfokus ke objek yang diinginkan agar hasil yang diperoleh lebih maksimal.

2.2.9 Sistem Warna

2.2.9.1 Sistem Warna Red Green Blue (RGB)

Citra RGB memiliki susunan utama yang terdiri dari tiga warna dominan yaitu warna merah, warna hijau, dan warna biru. Masing-masing kanal warna memiliki nilai intensitas piksel dengan kedalaman bit sebesar 8-bit yang memiliki arti variasi warna sebanyak $2^8 = 256$ derajat warna (0 s.d 255) sehingga bila ditotal maka terdapat 24 bit. Nilai intensitas setiap piksel merupakan kombinasi dari nilai R, G, dan B. Jumlah variasi citra RGB pada setiap piksel senilai $256 \times 256 \times 256$ sehingga terdapat sebanyak 16.777.216 sistem warna RGB yang dapat dikombinasikan [34]. Pada Gambar 2.10 menampilkan ilustrasi warna dominan RGB yang bila digabungkan akan menghasilkan warna lainnya.



Gambar 2.10 Sistem warna RGB [34].

2.2.9.2 Sistem Warna Grayscale

Grayscale adalah berbagai warna monokromatik dari hitam menjadi putih. Oleh karena itu, gambar *grayscale* hanya memiliki warna abu-abu. Setiap piksel dalam sebuah image dengan mode *grayscale* mempunyai nilai kecerahan (*brightness*) dengan rentang 0 (hitam) sampai dengan 255 (putih). Dengan menggunakan citra dengan derajat keabuan ini, ruang memori yang dibutuhkan dan waktu pengolahan citra lebih sedikit daripada citra berwarna [35]. Secara umum untuk mengubah citra RGB ke citra *grayscale* dapat menggunakan Persamaan 2.3 yang menghasilkan nilai derajat keabuan setiap piksel citra *grayscale* sebagai berikut:

$$\text{Gray} = I = 0.2989 \times R + 0.5870 \times G + 0.1140 \times B \quad (2.3)$$

Dengan I adalah level keabuan pada suatu koordinat yang diperoleh dengan membagi komponen warna R (merah), G (hijau), dan B (biru). Citra *grayscale* disimpan dalam format 8 bit untuk setiap sample piksel, yang memungkinkan sebanyak 256 intensitas. Citra yang ditampilkan dari citra jenis ini terdiri atas warna abu-abu yang bervariasi pada warna hitam pada bagian yang intensitas terlemah dan warna putih pada intensitas terkuat[36].

Nilai koefisien 0.2989, 0.5870, dan 0.1140 di atas berasal dari bobot yang diusulkan oleh ITU-R BT.601 dalam mengonversi warna RGB ke dalam standar luma (tingkat keabuan).

2.2.9.3 Sistem Warna Hue Saturation Value (HSV)

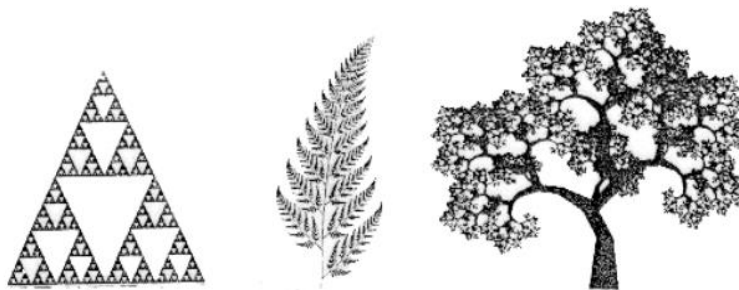
Model HSV memisahkan informasi tentang saturasi, kecerahan, dan *hue*. Ini sangat bermanfaat karena pemisahan *hue* (warna) dapat membantu membedakan objek berdasarkan warnanya dengan cara mengidentifikasi jenis warna atau tonalitas yang dimiliki piksel dalam gambar. Ini ditampilkan dalam derajat dari 0 hingga 360, dan ditampilkan dalam lingkaran warna penuh. Nilai H *zero* adalah merah murni, 120° hijau, dan 240° biru. Karena tidak terpengaruh oleh kecerahan atau intensitas, H adalah komponen HSV yang paling unik. Sedangkan *value* dan

saturasi dapat membantu mengidentifikasi tingkat saturasi dan kecerahan yang memiliki nilai berkisar antara 0 yang menghasilkan warna gelap seperti abu-abu dan hitam dan 1 untuk warna terang atau putih [37].

2.2.10 Ekstraksi Ciri

2.2.10.1 Fraktal

Fraktal merupakan bentuk objek yang memiliki kemiripan dirinya-sendiri (*self similarity*) pada skala yang berbeda, yang berarti bahwa bagian-bagian dari objek akan tampak identik satu sama lain. secara keseluruhan namun dengan ukuran berbeda. Sejarah fraktal pertama kali dikutip dari *The Fractal Geometry of Nature* adalah buku yang ditulis Benoit Mandelbrot. Geometri fraktal adalah cara yang alami untuk menunjukkan bentuk-bentuk di alam, sedangkan geometri euklidis menunjukkan bentuk yang diciptakan oleh manusia, seperti lingkaran, bola, dan bujursangkar [38]. Pada Gambar 2.11 dibawah ini menampilkan objek yang berkaitan dengan fraktal yaitu Segitiga Sierpinski, Daun Pakis Barnsley, dan Pohon Fraktal.



Gambar 2.11 Objek Fraktal [38].

Perhitungan dimensi pada objek seperti segitiga *sierpinski* nilai N merupakan banyaknya jumlah objek yang identik pada perulangan tertentu pada segitiga *sierpinski* memiliki 3. Faktor pengurangan skala r pada setiap tingkat perulangan untuk segitiga *sierpinski* adalah berjumlah 2 sehingga rumus disederhanakan menjadi seperti Persamaan 2.4.

$$D(s) = \frac{\log N}{\log(r)} \quad (2.4)$$

Dimensi fraktal dimensi fraktal tidak terpengaruh oleh distorsi akibat proyeksi dan dapat memberikan informasi tentang dalam memperkirakan gradien permukaan tekstur objek. Ada beberapa metode untuk memperkirakan dimensi fraktal, seperti *dimension variation*, *regularization dimension*, *multifractal spectrum*, *lacunarity measure*, *Brownian Motion*, dan *Bouligand-Minowski* atau lebih dikenal (*Box-Counting*) [39].

Metode dimana dimensi fraktal akan ditinjau melalui jumlah batas tepi dari permukaan fraktal. Disebut juga pada geometri fraktal dengan nama *Box Counting* atau dikenal juga sebagai metode *Minkowski-Bouligand*.

Menghitung dimensi suatu himpunan s , dapat diasumsikan fraktal ini diletakkan pada suatu luasan yang dikotakkan (*grid*) secara merata. Selanjutnya dihitung banyak dari kotak yang diperlukan untuk menutup bidang ini. Dimensi *Box Counting* dihitung dengan melihat seberapa jumlah kotak tersebut berubah ketika ukuran kotaknya diperkecil. Semisal $N(r)$ merupakan jumlah kotak dengan panjang sisi r yang diperlukan sebagai penutup bidang [40]. Maka dimensi *Box Counting* didefinisikan pada Persamaan 2.5 sebagai berikut:

$$D(s) = \frac{\log N(r)}{\log \frac{1}{(r)}} \quad (2.5)$$

Dimana, r merupakan ukuran kotak, $N(r)$ merupakan jumlah kotak yang berisi gambar, dan $D(s)$ merupakan dimensi fraktal dengan kotak berukuran r yang dihitung dengan metode *Box Counting*.

2.2.10.2 Histogram

Histogram menunjukkan distribusi intensitas piksel dalam citra. Histogram dapat mengekstrak fitur seperti distribusi intensitas, kontras, dan kecerahan dari citra, seperti gambar dengan distribusi intensitas yang lebar cenderung memiliki kontras yang lebih baik. Terdapat fungsi dari histogram seperti *color histogram*

yang menghitung distribusi warna piksel untuk mendapatkan fitur, sedangkan *Histogram of Oriented Gradients* yang termasuk dalam *feature descriptor* mendefinisikan fitur setelah mengukur gradien orientasi pada citra yang kemudian memisahkan fitur pada gambar ke dalam kotak kecil dalam menjalankan metode ini citra yang digunakan akan diubah pada rasio 1:2 (lebar:tinggi) dengan ukuran 64 x 128 sehingga memudahkan dalam pembagian citra menjadi 8 x 8 atau 16 x 16 [41].

2.2.10.3 Gray-Level Co-Occurance Matrix (GLCM)

Gray-Level Co-Occurance Matrix salah satu metode untuk mendapatkan fitur tekstur secara statistik. Menggambarkan frekuensi munculnya pasangan antara beberapa pixel dengan intensitas, arah, dan jarak tertentu pada citra dalam menjalankan proses dapat menggunakan empat sudut yaitu 0° berupa horizontal dimana piksel dilihat bersebelahan pada baris yang sama, 45° berupa diagonal dari sisi kiri atas ke kanan bawah, 90° vertical berarti bersebelahan pada kolom yang sama, dan 135° diagonal dari sudut kiri bawah ke atas [15].

Tabel 2.3 Rumus pada GLCM.

Texture feature	Equation
Contrast	$\sum_{i=1}^N \sum_{j=1}^N (i - j)^2 P(i, j)$
Entropy	$- \sum_{i=1}^N \sum_{j=1}^N P(i, j) \lg P(i, j)$
Correlation	$\frac{\sum_{i=1}^N \sum_{j=1}^N (i - \bar{x})(j - \bar{y}) P(i, j)}{\sigma_x \sigma_y}$
Energy	$\sum_{i=1}^N \sum_{j=1}^N P(i, j)^2$

Dalam proses ekstraksi fitur menggunakan GLCM menggunakan beberapa statistik sebagai contoh pada Tabel 2.3 yang digunakan sesuai kebutuhan dalam pengambilan fitur. Berikut fungsi beberapa jenisnya dimulai dari kontras merupakan pengukuran variasi intensitas aras keabuan antara pasangan piksel pada citra. Pengertian entropi adalah pengukuran komplektifitas distribusi yang tidak

beraturan. Pada korelasi berarti mengukur tingkat ketergantungan atau saling berhubungan pada pasangan piksel tertentu. Energi mengukur seberapa mirip atau seragam distribusi yang dimiliki antara piksel selain itu masih ada titik ekstraksi lain seperti rata-rata dan varian [42].

2.2.10.4 Segmentasi

Dalam pengolahan citra, metode segmentasi membagi citra menjadi bagian-bagian yang memiliki karakteristik atau makna yang sama. Tujuan dari metode ini adalah untuk menemukan objek atau elemen tertentu dalam citra dapat berupa pemisahan latar depan dari latar belakang atau pengelompokan wilayah piksel berdasarkan kesamaan warna atau bentuk. Segmentasi citra berguna untuk mengelompokkan area atau segmen yang serupa dalam citra di bawah label kelasnya dapat digunakan menjadi lebih kompleks seperti klasifikasi gambar dan deteksi objek. Beberapa jenis segmentasi adalah *Thresholding*, *Clustering*, *Edge Detection* dengan contoh metode yang sering digunakan dalam membantu proses klasifikasi pada pengolahan citra adalah seperti metode otsu, *Fuzzy C-Means*, dan *K-means*.