

BAB II TINJAUAN PUSTAKA

2.1 Citra

Menurut [4] citra digital adalah representasi objek 2 dimensi dalam bentuk kumpulan titik-titik atau piksel-piksel yang berwarna. Manusia dalam kehidupan sehari-hari memanfaatkan citra digital sebagai representasi objek yang berada di sekitar kita dengan berbagai macam tujuan. Nilai yang dimiliki citra digital tersebut mendorong munculnya banyak penelitian tentang identifikasi identitas dari nilai yang digunakan untuk merepresentasi citra digital tersebut.

2.1.1 Pengolahan Citra

Pengolahan citra digital adalah teknik pemrosesan citra foto maupun gambar bergerak untuk meningkatkan kualitas citra agar dapat dengan mudah dimengerti baik oleh manusia atau mesin komputer[4].

Menurut [7] pengolahan citra dapat didefinisikan sebagai kegiatan untuk memeriksa citra dengan tujuan mengidentifikasi objek dan menentukan signifikansinya. Pengolahan citra secara digital pada saat ini merupakan komponen yang sangat penting dalam berbagai aplikasi industri, komersil, dan juga merupakan komponen utama dalam bidang *computer vision*.

2.1.2 Jenis Citra

Menurut [8] ada empat tipe citra digital yaitu:

- 1) *Binary Image*: merupakan tipe citra yang paling sederhana karena hanya memiliki 2 nilai yaitu hitam dan putih atau bernilai 0 dan 1. *Binary Image* juga disebut sebagai citra satu bit atau satu piksel karena hanya membutuhkan satu digit *binary* untuk merepresentasikan setiap piksel.
- 2) *Grayscale Image*: *grayscale image* merupakan citra yang *monochrome* atau hanya memiliki 1 warna karena hanya memiliki informasi tentang pencahayaan dan tidak memiliki informasi tentang warna. Biasanya citra ini mengandung 8-bit atau piksel yang memungkinkan citra untuk merepresentasikan 0-255 tingkat kecerahan(*gray*) citra.
- 3) *Indexed Image*: sebuah *indexed image* terdiri dari *array* dan matriks *colormap*. Nilai piksel di dalam *array* merupakan indeks langsung dari *colormap*. *Colormap* adalah matriks *m-by-3* yang berisi nilai *floating point* (0 sampai 1). Setiap baris merepresentasikan komponen *RGB* dari satu warna.
- 4) *RGB Image*: citra *RGB* tidak menggunakan *color map*, citra direpresentasikan dengan 3 komponen warna yaitu *RGB*. Citra *RGB* menggunakan standar *monochrome* 8-bit dan mempunyai 24 piksel dimana setiap 8 pikselnya merepresentasikan masing-masing warna dari *RGB*.

2.2 Penyakit Daun Jagung

Menurut [9] penyakit pada tanaman jagung dapat menyerang berbagai macam bagian pada tanaman jagung seperti bibit, daun, batang, dan biji jagung. Pada penelitian ini penyakit daun jagung *common rust*, *gray leaf spot*, dan *northern leaf blight* yaitu:

- 1) *Common rust*: penyakit pada daun jagung ini disebabkan oleh jamur *Puccinia sorghi*, pertama karat atau *rust* mulai muncul dalam bentuk lingkaran dengan warna hijau muda, tapi setelah sekian lama lingkaran tersebut semakin banyak seolah-olah berkumpul dan berubah warna menjadi coklat keemasan atau coklat kemerahan dengan titik seperti karat pada tengah lingkaran tersebut. Penyakit ini dapat tumbuh pada bagian atas atau bawah daun dan batang, jika dibiarkan maka daun akan berwarna kuning dan mati. Untuk mengatasi penyakit ini disarankan menjaga suhu antara 60 sampai 77 derajat fahrenheit, menanam jagung pada musim yang tepat, dan bisa juga menanam bibit jagung *hybrid* yaitu hasil perkawinan antara jagung yang tahan dengan penyakit ini.
- 2) *Gray leaf spot*: penyakit pada daun jagung ini disebabkan oleh jamur *Cercospora zea-maydis*, pertama muncul bintik oval dengan warna coklat kemerahan dengan titik coklat di tengahnya, kemudian bintik oval tadi akan berkelompok dan menjadi semakin banyak dan memanjang sekitar 0,5 inci sampai

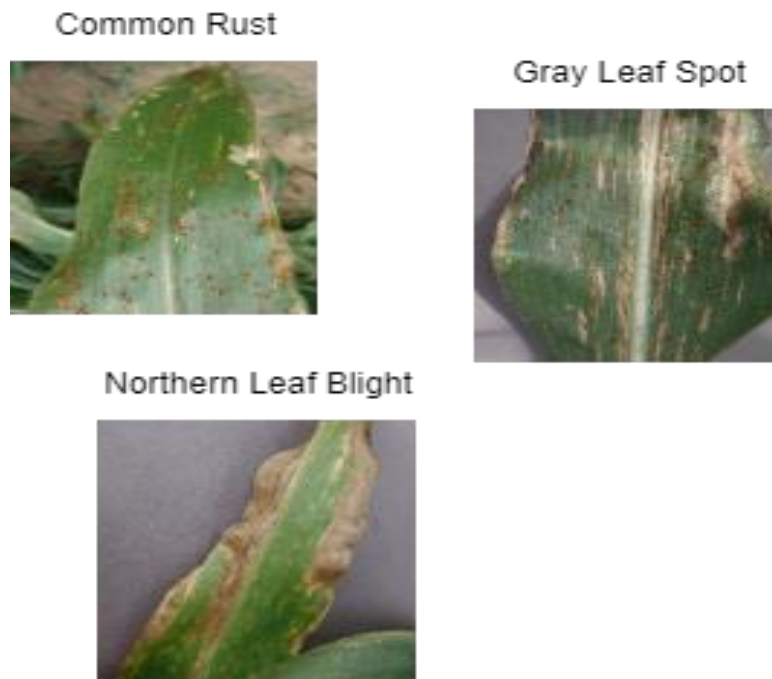
2 inci untuk panjangnya dan berubah warna keabu-abuan. Jamur penyebab penyakit ini masih tersimpan dalam permukaan tanah walaupun tanaman telah dicabut dan menyebabkan penyakit ini menjadi lebih ganas untuk tanaman jagung selanjutnya. Penyakit ini dapat dicegah dengan menanam bibit jagung *hybrid* yang tahan terhadap penyakit ini.

- 3) *Northern leaf blight*: penyakit ini disebabkan oleh jamur *Exserobolium turcicum*, pertama *penyakit* ini tumbuh dengan jumlah yang tidak banyak tetapi dalam bentuk oval dan dalam area yang luas pada permukaan daun, yang kemudian menjadi lebih besar dan lebih panjang sehingga membuat layu daun jagung. Untuk penyakit ini juga dapat dicegah dengan menanam bibit jagung *hybrid* yang tahan terhadap penyakit ini.

Gambaran ketiga penyakit tersebut dapat dilihat pada Untuk ketiga penyakit daun jagung diatas khususnya penyakit pada daun jagung, menurut [9] manajemen tanaman jagung untuk menghindari penyakit ini adalah dengan cara:

- 1) Memilih bibit jagung yang merupakan bibit unggul hasil perkawinan tanaman jagung yang tahan terhadap masing-masing penyakit daun jagung.
- 2) Melakukan rotasi terhadap tanaman jagung yaitu mencabut atau memanen jagung paling tidak dalam jangka waktu 1 tahun.

- 3) Mengatur residual dari jagung, seperti mengganti permukaan tanah bekas tanaman jagung yang terkena penyakit.
- 4) Memberi fungisida khusus untuk daun jagung.



Gambar 2.1: Jenis Penyakit Pada Daun Jagung

2.3 Logika *Fuzzy*

Logika *fuzzy* adalah logika yang memiliki nilai antara benar atau salah yang sering disebut juga dengan nilai kekaburan atau *fuzziness*. Logika *fuzzy* dapat bernilai benar atau salah secara bersamaan tetapi logika *fuzzy* memiliki derajat keanggotaan dari 0 hingga 1 yang menentukan keberadaan dan kesalahan nilai tersebut. Logika *fuzzy* menterjemahkan suatu besaran yang direpresentasikan dengan menggunakan bahasa (*linguistic*), misalnya kecepatan mobil direpresentasikan dengan pelan, agak cepat, cepat, dan sangat cepat. Tidak

seperti logika klasik dimana suatu nilai hanya memiliki 2 kemungkinan yaitu bukan merupakan anggota himpunan jika derajat keanggotaan 0 dan merupakan anggota himpunan jika derajat keanggotaan bernilai 1 [10].

Logika *fuzzy* merupakan cara yang tepat untuk memetakan suatu ruang *input* ke dalam suatu ruang *output* yang mempunyai nilai kontinu. *Fuzzy* dinyatakan dengan derajat dari suatu keanggotaan dan derajat dari kebenaran, oleh sebab itu dalam logika *fuzzy* sesuatu dapat dikatakan sebagian benar dan sebagian salah pada waktu yang bersamaan. Pada teori himpunan *fuzzy* peran derajat keanggotaan sebagai penentu keberadaan elemen dalam suatu himpunan sangatlah penting karena merupakan ciri utama dalam penalaran logika *fuzzy*. Derajat keanggotaan juga berperan penting pada operasi himpunan *fuzzy* untuk proses inferensi dan penalaran sebagai *fire strength* atau α -predikat [10]. Menurut [10] ada beberapa hal yang perlu diketahui dalam memahami sistem *fuzzy* yaitu:

- 1) Variabel *fuzzy* merupakan variabel yang dibahas dalam sistem *fuzzy*.
- 2) Himpunan *fuzzy* merupakan suatu kelompok yang mewakili suatu keadaan tertentu atau suatu kondisi dalam suatu variabel *fuzzy*.
- 3) Semesta pembicaraan adalah keseluruhan nilai yang diijinkan untuk dioperasikan dalam suatu variabel *fuzzy*.
- 4) Domain himpunan *fuzzy* adalah keseluruhan nilai yang diperbolehkan dalam semesta pembicaraan dan diijinkan untuk dioperasikan dalam suatu himpunan *fuzzy*.

2.4 Segmentasi Citra

Segmentasi citra merupakan proses yang paling pertama dilakukan untuk dapat di analisis dan diproses oleh komputer dengan mengklasifikasikan *pixel* dari sebuah gambar. Segmentasi citra memiliki tujuan untuk membagi gambar menjadi beberapa bagian untuk mendapatkan wilayah pada gambar yang memiliki suatu kesamaan berdasarkan variabel yang ditentukan. Secara umum segmentasi citra dapat dibagi menjadi 4 kategori yaitu *thresholding*, *edge detection*, *region extraction*, dan *clustering*[11].

Menurut [11] secara umum terdapat 4 buah kategori segmentasi citra yaitu:

- a. *Thresholding*: mensegmentasi citra dengan berdasarkan pada *gray level* atau seberapa intensitas nilai dari *pixel*. Tantangannya dalam metode ini adalah menentukan *gray level* yang sesuai untuk membagi setiap *pixel* menjadi 2 kategori yaitu gelap dan terang.
- b. *Edge detection*: membagi citra dengan mendeteksi tepi dari citra lalu mengelompokkannya ke beberapa bagian untuk merepresentasikan pembatas antar objek citra
- c. *Region extraction*: membagi seluruh citra menjadi beberapa bagian kecil berdasarkan kriteria yang telah ditentukan, biasanya menggunakan kesamaan dari intensitas, warna, dan tekstur sebagai kriteria.

- d. *Clustering*: mensegmentasi citra dengan mengklasifikasikan pola atau objek ke dalam beberapa *cluster* dengan karakteristik yang mirip.

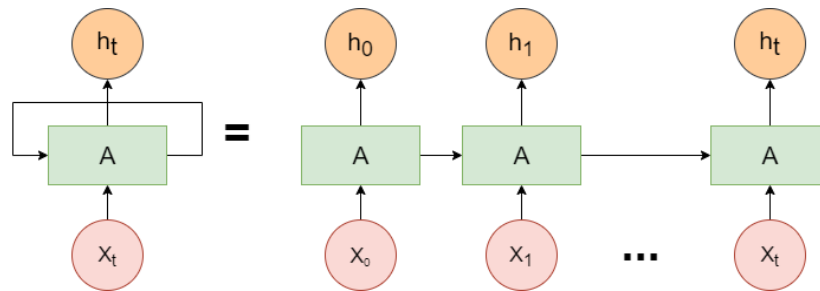
Selain itu, dengan munculnya *deep learning* maka muncul teknik segmentasi yang baru seperti *CNN* dan *RNN*, pada penelitian ini digunakan *LSTM* dan akan dibandingkan dengan gabungan antara algoritme *clustering* yang cukup terkenal yaitu *Fuzzy C-Means* dengan *LSTM*.

2.4.1 Segmentasi Citra Berbasis *Deep Learning*

RNN adalah sebuah *class* dari jaringan saraf tiruan dimana koneksi antara *nodes* pada setiap *layer* membentuk *directed graph* dengan variabel yang berurutan. *RNN* memiliki beberapa variasi seperti *GRU* (*Gated Recurrent Units*) dan *LSTM* (*Long Short-Term Memory Network*) yang meningkatkan performa dari algoritme *RNN*[12].

Struktur *RNN* terdiri dari *input layer*, satu atau lebih *hidden layer*, dan *output layer*. *RNN* memiliki struktur yang mirip dengan rantai yang terdiri dari modul yang berulang. Modul tersebut digunakan dengan tujuan sebagai memori untuk menyimpan informasi penting dari langkah proses sebelumnya. *RNN* juga menyertakan *feedback loop* yang memungkinkan jaringan saraf tiruan untuk menerima urutan *input*. Oleh karena itu, *output* dari langkah $t-1$ dimasukkan kembali ke jaringan untuk mempengaruhi hasil dari langkah t . Pada Gambar 2.2 dapat dilihat gambaran sederhana tentang cara kerja algoritme *RNN* yang terdiri dari satu *input unit*, satu

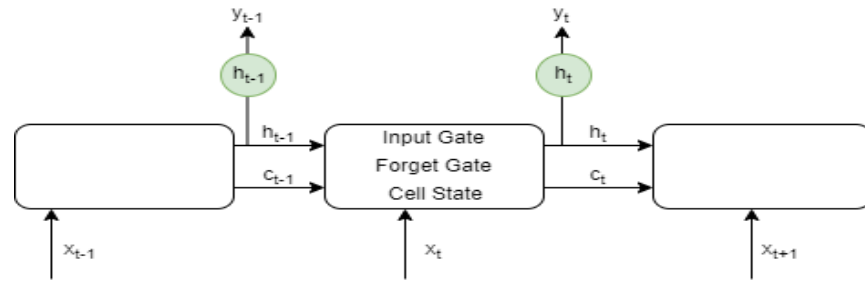
output unit, dan *hidden unit* yang berulang dan berkembang menjadi jaringan penuh, nilai X_t adalah *input* dari *time step* t dan h_t adalah *output* dari *time step* ke t [13].



Gambar 2.2: Gambaran Cara Kerja *RNN*

Penelitian sebelumnya telah berhasil melakukan implementasi *RNN* untuk *image processing* seperti [14] yang telah berhasil melakukan implementasi *LSTM* untuk identifikasi penyakit pada tanaman, penelitian [12] yang telah berhasil meng-implementasi *Attention-Based RNN* untuk deteksi penyakit pada tanaman, dan penelitian [15] yang menggunakan variasi dari *RNN* yaitu *Dense RNN* untuk segmentasi citra jantung.

Menurut [15] gambaran secara garis besar cara kerja variasi metode *RNN* yaitu *LSTM* dapat dilihat pada Gambar 2.3. Setiap sel *LSTM* mendapatkan informasi dari sel sebelumnya lalu mengirimkan informasi yang didapat ke sel selanjutnya.



Gambar 2.3: Gambaran Cara Kerja LSTM

Menurut [16] LSTM memiliki *memory cell* dan *gate inputs* (*input gate*, *forget gate*, *cell gate*, dan *output gate*). Pada *forget gates* setiap data yang masuk, diolah lalu dipilih yang mana yang dibuang dan yang mana yang disimpan, pada *gate* ini fungsi aktivasi yang digunakan adalah *sigmoid* (Jika bernilai 1 data disimpan, jika bernilai 0 data dibuang), berikut ini adalah rumus yang digunakan pada *forget gate*:

$$f_t = \sigma(W_f \cdot [h_{t-1}, x_t] + b_f)$$

Selanjutnya *input gate* dilakukan 2 hal yaitu pertama menentukan nilai mana yang diperbaharui dengan fungsi aktivasi *sigmoid* dan dilanjutkan dengan fungsi *tanh* yang membuat *vector* nilai baru yang disimpan pada *memory cell*. Berikut adalah rumus yang digunakan pada *input gate*:

$$i_t = \sigma(W_i \cdot [h_{t-1}, x_t] + b_i)$$

$$\hat{c}_t = \tanh(W_c \cdot [h_{t-1}, x_t] + b_c)$$

Setelah itu pada *cell gate* dilakukan penggantian nilai *memory cell* sebelumnya dengan nilai yang baru yang didapatkan dengan

menggabungkan nilai yang didapat dari *forget gate* dan *input gate* dengan rumus sebagai berikut:

$$c_t = f_t * c_{t-1} + i_t * \hat{c}_t$$

Terakhir pada *output gate* dilakukan seleksi nilai pada *memory cell* yang dikeluarkan dengan fungsi aktivasi *sigmoid* lalu dimasukkan nilai pada *memory cell* dengan fungsi aktivasi *tanh* dan nilai kedua proses tersebut dikalikan sehingga menghasilkan nilai keluaran. Berikut ini adalah rumus yang digunakan pada *output gate*.

$$o_t = \sigma(W_o \cdot [h_{t-1}, x_t] + b_o)$$

$$h_t = o_t \tanh(c_t)$$

2.4.2 Segmentasi Citra Berbasis *Clustering*

Dari penelitian [17] telah dilakukan penelitian tentang teknik segmentasi citra menggunakan teknik *clustering* pada penelitian ini digunakan metode *Fuzzy C-Means*.

Fuzzy C-Means adalah algoritme *clustering* yang menggunakan model pengelompokan *fuzzy* sehingga data dapat menjadi anggota dari semua *cluster* terbentuk dengan tingkat keanggotaan yang berbeda antara 0 hingga 1. Menurut [17] [10] algoritme *Fuzzy C-Means* adalah sebagai berikut:

1. Input data untuk parameter *clustering* yaitu X , data berupa matriks berukuran $n \times m$ dimana n adalah jumlah sampel data

dan m adalah atribut setiap data. X_{ij} adalah data sampel ke- i ($i=1,2,\dots,n$), atribut ke- j ($j = 1,2,\dots,m$)

2. Tentukan:

- Jumlah *cluster* = c
- Pangkat = w
- Iterasi maksimum = $MaxIter$
- *Error* terkecil yang diharapkan = ε
- Fungsi objektif awal = $P_0 = 0$
- Iterasi awal = $t = 1$

3. *Generate* bilangan acak μ_{ik} dengan nilai $i = 1,2,\dots,n$ dan nilai $k = 1,2,\dots, c$; sebagai elemen-elemen matriks partisi awal U . Lalu hitung nilai setiap kolom (atribut) dengan rumus:

$$Q_j = \sum_{k=1}^c \mu_{ik} \text{ dengan } j = 1,2, \dots, m$$

4. Hitung pusat dari *cluster-k* yaitu V_{kj} dengan $k = 1,2,\dots,c$ dan $j = 1,2,\dots,m$:

$$V_{kj} = \frac{\sum_{i=1}^n ((\mu_{ik})^w * X_{ij})}{\sum_{i=1}^n (\mu_{ik})^w}$$

5. Hitung fungsi objektif pada iterasi- t dengan rumus:

$$P_t = \sum_{i=1}^n \sum_{k=1}^c \left(\left[\sum_{j=1}^m (X_{ij} - V_{kj})^2 \right] (\mu_{ik})^w \right)$$

6. Hitung perubahan pada matriks partisi dengan rumus:

$$\mu_{ik} = \frac{[\sum_{j=1}^m (X_{ij} - V_{kj})^2]^{\frac{-1}{w-1}}}{\sum_{k=1}^c [\sum_{j=1}^m (X_{ij} - V_{kj})^2]^{\frac{-1}{w-1}}}$$

7. Cek kondisi berhentinya iterasi:
 - a. Jika $(|P_t - P_{t-1}| < \epsilon)$ atau $(t > MaxIter)$ maka berhenti;
 - b. Jika tidak maka $t = t+1$, lalu ulangi langkah ke-4

Untuk perbandingannya dapat dilihat pada Tabel 2.1 yang berisi algoritme bekerja pada data seperti apa, pusat data, kriteria untuk menghasilkan hasil yang terbaik, kelebihan, dan kekurangan[18].

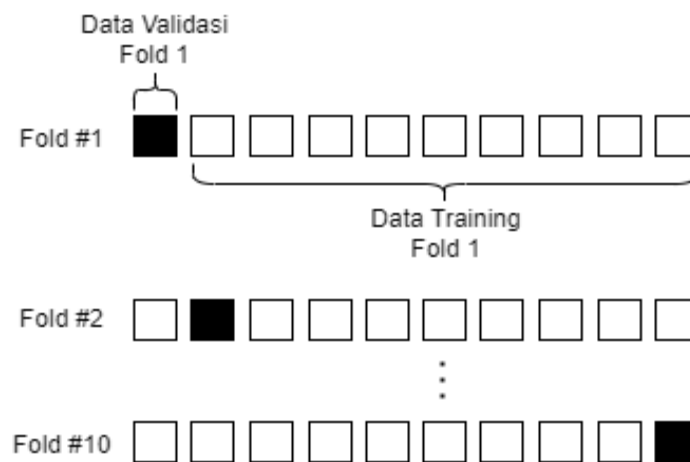
Tabel 2.1: Perbandingan Algoritme

Metode Clustering	Algoritme bekerja pada data	Pusat data	Kriteria untuk hasil yang maksimal	Kelebihan	Kekurangan
K-Means	<i>Unsupervised data</i>	Lokasi berbeda	Data terpisah dengan baik	Cepat, kokoh	Data yang memiliki <i>noise</i> dan data yang tidak linier
Fuzzy C-Means	<i>Assigning membership</i>	Pusat cluster	Data saling <i>overlapped</i>	Lebih baik dibandingkan dengan k-means	<i>Priory specification, wt. underlying factor</i>

2.5 Evaluasi Metode

Untuk validasi metode digunakan teknik *k-fold cross validation*, pada teknik *k-fold cross validation* data *training* dipartisi menjadi *k* bagian dimana *k* adalah *fold* yang ditentukan sebagai parameter masukan. Setelah itu data dilakukan *split* kemudian di-*training* untuk setiap nilai *k* tersebut. Pada Gambar 2.4 dapat

dilihat gambaran cara kerja *k-fold cross validation* dengan nilai $k = 10$, yaitu pertama-tama dilakukan *fold* yang pertama dengan membagi partisi data yang pertama untuk menjadi data validasi dan data sisanya menjadi data *training*, proses ini dilakukan sampai seluruh partisi data telah menjadi bagian dari data validasi, setelah semua data di validasi dihitung rata-rata hasil akurasi validasi dari masing-masing *fold* dan rata-rata itulah yang menjadi acuan performa dari *model* yang dilakukan *k-fold cross validation*[19].



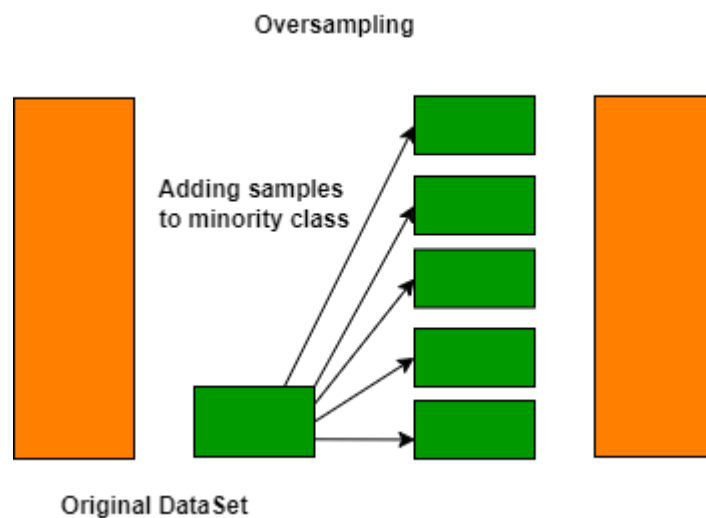
Gambar 2.4: Gambaran Cara Kerja *K-fold Cross Validation*

2.6 Imbalanced Dataset

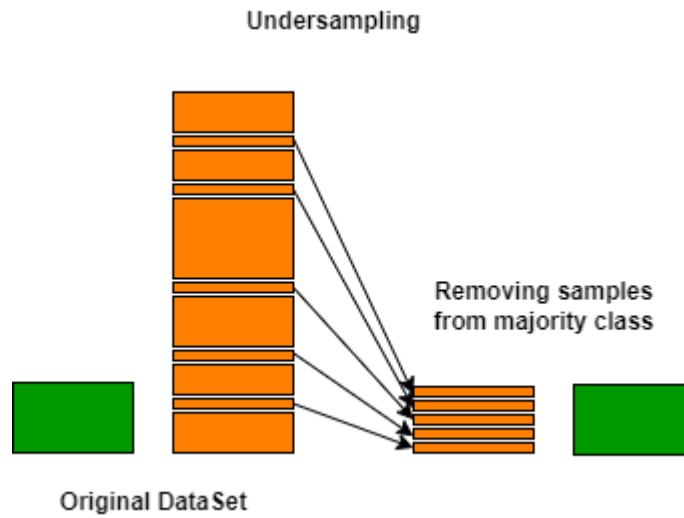
Dataset publik yang sering digunakan oleh banyak penelitian sering kali memiliki jumlah data yang tidak sama pada setiap direktori atau kelas, maka hal ini akan membuat permasalahan pada klasifikasi karena kelas yang jumlah datanya lebih banyak tentu akan lebih sering muncul saat prediksi dibandingkan dengan kelas yang datanya lebih sedikit, menurut [20] terdapat 2 cara untuk

mengatasi data yang tidak seimbang ini (*imbalanced dataset*) yaitu *oversampling* dan *undersampling*.

Oversampling adalah teknik yang menambahkan atau menduplikasi sampel pada kelas yang datanya tidak sebanyak kelas yang datanya paling banyak agar kelas yang tadinya memiliki jumlah data yang lebih sedikit daripada kelas yang datanya paling banyak dapat memiliki jumlah data yang sama. *Undersampling* adalah teknik yang menghapus data dari kelas yang memiliki data paling banyak agar memiliki jumlah data yang sama dengan kelas yang memiliki jumlah data yang paling sedikit [20]. Gambaran perbedaan *oversampling* dan *undersampling* dapat dilihat pada Gambar 2.5 dan Gambar 2.6.



Gambar 2.5: Gambaran Cara Kerja *Oversampling*



Gambar 2.6: Gambaran Cara Kerja *Undersampling*

2.7 Penelitian Terkait

Pada sub-bab ini, dibahas tentang penelitian-penelitian yang telah dilakukan sebelumnya untuk menjelaskan tentang *research gap* yang ada pada penelitian sebelumnya tentang *Fuzzy C-Means (FCM)* dan *RNN-LSTM*.

Permasalahan pada paper [17] adalah *FCM* sensitif pada citra yang memiliki *noise*, tetapi karena dalam mendeteksi *node* pada daun jagung *noise* tidak terlalu berpengaruh.

Solusi yang ditawarkan adalah menggunakan algoritme *FCM* dengan bantuan teknik *Morphology* untuk deteksi otomatis penyakit pada daun jagung.

Hasil yang didapat adalah metode yang ditawarkan oleh penulis memiliki performa yang baik jika diterapkan di permasalahan segmentasi citra antara penyakit pada daun jagung dan daun jagung yang sehat.

Permasalahan yang diangkat pada paper [7] adalah pendeteksian citra daun tanaman seringkali tidak murah dan kurang akurat, maka digunakan perbandingan algoritme *K-Means*, klasifikasi, dan *Fuzzy C-Means* untuk menemukan algoritme yang terbaik untuk mendeteksi penyakit pada daun tanaman.

Solusi yang ditawarkan adalah pertama-tama area yang merupakan penyakit terdeteksi oleh segmentasi *FCM*, setelah itu dilakukan ekstraksi fitur, terakhir dilakukan klasifikasi untuk mendeteksi penyakit pada citra daun.

Hasil yang didapatkan adalah dari algoritme klasifikasi, *k-means*, dan *Fuzzy C-Means* didapat tingkat akurasi dari 80% sampai 90%, karena paper ini merupakan paper survey dari 3 algoritme klasifikasi maka akan mempermudah para peneliti yang akan datang untuk memilih algoritme yang paling baik untuk diterapkan pada penelitiannya.

Saran dari penulis adalah jumlah data pada basis data bisa lebih diperbanyak lagi dan dianjurkan untuk menggunakan data dari dunia nyata.

Permasalahan yang diangkat pada penelitian [21] adalah segmentasi citra kadang-kadang memiliki hasil yang kurang memuaskan karena banyak faktor yang mempengaruhi kualitas dari segmentasi citra seperti *noise*, dan pencahayaan.

Solusi yang ditawarkan adalah mengkombinasi *Fuzzy C-Means* dengan *statistical region merging* untuk melakukan segmentasi citra.

Hasil yang didapat adalah metode *FCMSRM* yang dimodifikasi dapat melakukan segmentasi citra dimana dibagi menjadi 2 proses yaitu proses pertama dengan menggunakan metode *Fuzzy C Means* dengan *spatial function*, lalu dilanjutkan dengan *preprocessing* dengan menggunakan metode *SRM* untuk menyederhanakan kompleksitas warna pada citra yang akan diuji agar lebih mudah untuk disegmentasi. Algoritme *Fuzzy C-Means* pada penelitian yang diusulkan ini menggunakan perhitungan metrik evaluasi *Xie Beni Index* sebagai fungsi objektifnya. Hasil Evaluasi dihitung waktu eksekusi dan jumlah iterasi, didapat rata-rata jumlah iterasi 20, rata-rata waktu eksekusi 80 detik, rata-rata *XB Index* 0.5248133, dan jumlah cluster paling optimal adalah 3 untuk “horses.png”. Untuk citra uji “chain.png” diperoleh rata-rata jumlah iterasi adalah 67, rata-rata waktu eksekusi 247 detik, rata-rata *XB Index* 0.2530522, dan jumlah cluster paling optimal adalah 2. Lalu yang terakhir pada citra uji “carriage.png” diperoleh rata-rata jumlah iterasi adalah 45, rata-rata waktu eksekusi 141 detik, rata-rata *XB Index* 0.1420811, dan jumlah *cluster* paling optimal adalah 2.

Permasalahan yang diangkat pada penelitian [22] adalah karena banyaknya penelitian tentang metode untuk mendeteksi penyakit pada daun tanaman, maka dibutuhkan rangkuman tentang metode-metode yang digunakan untuk mendeteksi penyakit pada daun tanaman.

Solusi yang ditawarkan adalah pertama-tama citra *RGB* di akuisisi lalu dimasukkan ke color space setelah itu dilakukan segmentasi untuk mendapatkan

segmen yang berguna dan dikalkulasi fitur dari teksturnya, terakhir adalah mengkonfigurasi neural network untuk mengenali fitur.

Hasil yang didapat adalah teknik-teknik yang digunakan pada penelitian ini digunakan untuk menganalisis daun tanaman yang sehat dan yang sakit, tantangan pada teknik ini adalah memvisualisasikan data latar elakang pada citra hasil, optimisasi teknik untuk penyakit pada daun tanaman yang lebih spesifik, dan mengotomatisasi proses monitoring pada daun tanaman dengan kondisi-kondisi yang asli berada di dunia nyata.

Saran yang dianjurkan untuk peneliti selanjutnya adalah algoritme *BPNN*, *SVM*, *K-means clustering*, dan *SGDM* masih banyak kekurangan dan dapat ditingkatkan pada penelitian yang akan datang.

Permasalahan yang diangkat pada penelitian [23] yaitu algoritme yang digunakan selama ini kurang cocok untuk digunakan pada tanaman sayuran yang berada pada rumah kaca karena memiliki karakteristik tersendiri. Solusi yang diajukan adalah pertama dilakukan *data preprocessing* lalu dilakukan *morphological filtering* yang setelah itu dilakukan *watershed segmentation* sampai nilai *solidity* mencapai nilai maksimum.

Hasil yang didapatkan pada penelitian [23] adalah:

- Mengolah gambar penyakit tanaman timun terhadap latar belakang yang kompleks dengan 3 kali menjalankan algoritme *marked-watershed algorithm*, daun yang menjadi sasaran ter-ekstraksi

dengan efisien dari gangguan pada latar belakang seperti daun yang bertumpuk dan elemen yang bukan merupakan sasaran.

- Algoritme ini mengenalkan *neighborhood grayscale information*, yang akan menghilangkan permasalahan kurangnya *penggunaan image pixel spatial information* oleh *FCM*, dan meningkatkan kapasitas *noise-filtering* pada algoritme tersebut.
- Dengan menggunakan *space grayscale information* yang berbasis metode *weighted neighborhood gray values* dimana akan memperbaiki segmentasi dari titik ujung tanaman timun dan memiliki hasil yang telah meningkat.
- Jarak abu-abu antara *pixel* dan pusat *cluster* diperbaharui, yang mengakibatkan perubahan yang signifikan ke *fuzzy membership* setelah 1 iterasi. Algoritme ini dapat mencapai *convergence* dalam beberapa iterasi saja, mengurangi waktu yang diperlukan untuk menjalankan algoritme ini dan meningkatkan efisiensi dan efektivitas dari algoritme.

Penelitian [14] memiliki permasalahan bahwa penyakit pada tanaman merupakan hal yang sangat penting untuk meningkatkan kualitas hidup manusia, namun pada nyatanya permasalahan deteksi penyakit pada tanaman walaupun sudah menggunakan berbagai metode tetap saja yang menjadi permasalahan adalah *dataset*-nya. Oleh karena itu, diajukan metode *deep learning* untuk menghasilkan *dataset* tersendiri.

Solusi yang ditawarkan adalah pertama-tama dilakukan akuisisi citra dari penangkapan gambar dengan menggunakan kamera digital lalu dilakukan *data preprocessing*, setelah itu dilakukan *LSTM* untuk ekstraksi fitur secara rekursif. Selanjutnya algoritme akan dibandingkan dengan algoritme *random forest* dan *KNN*.

Hasil yang didapat pada penelitian [14] adalah *LSTM* memiliki akurasi kedua tertinggi yaitu 95.79%, *KNN* memiliki akurasi yang paling baik dengan 98.19% dan *random forest* memiliki akurasi yang paling buruk yaitu 94.95%.

Penelitian [12] memiliki permasalahan bahwa algoritme klasik *deep learning* untuk klasifikasi penyakit tanaman yaitu *CNN* masih memiliki kelemahan yaitu dapat mengikutsertakan gambar *background* pada citra yang tidak relevan.

Solusi yang ditawarkan adalah menggunakan algoritme *RNN* untuk klasifikasi penyakit pada tanaman dan didapat hasil dari 4 *set test* yang dilakukan model *RNN* yang diajukan secara garis besar lebih unggul dibandingkan dengan algoritme lainnya.

Permasalahan pada penelitian [15] adalah tidak ada penelitian yang berhasil segmentasi citra *four chambered view left ventricle* karena struktur yang kompleks dan gerakan dinding yang bervariasi.

Solusi yang ditawarkan adalah menggunakan *dense RNN* untuk segmentasi citra *four chambered view left ventricle* dari gambar *MRI* dan didapat hasil bahwa akurasi dari gambar pertama telah meningkat sangat signifikan dan mencapai 92.13% pada *IoU metrics*.

Berikut ini adalah rangkuman dari penelitian terkait dalam bentuk tabel yang disajikan pada Tabel 2.2: Rangkuman Penelitian Terkait di bawah ini.

Tabel 2.2: Rangkuman Penelitian Terkait

No	Peneliti	Judul Penelitian	Tahun	Metode dan Hasil Penelitian	Kelebihan	Kelemahan
1.	Nurhidayati, Syaiful Anam, Agus Widodo [17]	Automatic Detection of Corn Leaf Disease by Using Fuzzy C-Means Algorithm and Morphological Operation	2019	Menggunakan <i>FCM</i> ditambah teknik <i>Morphology</i> untuk deteksi otomatis penyakit pada daun jagung. Hasilnya Memiliki performa yang baik	Sukses Menggabungkan <i>FCM</i> dan <i>Morphology</i>	Data uji sedikit
2.	A. Kumari, S.Meena kshi, S.	Plant Leaf Disease Detection Using	2018	Menggunakan metode <i>FCM</i> untuk deteksi penyakit pada	<i>FCM</i> dapat mendeteksi berbagai macam	- Data sedikit -Data tidak real

	Abinaya [7]	Fuzzy C- Means Clustering Algorithm		daun tanaman dan didapat hasil <i>FCM</i> memiliki akurasi yang lebih tinggi	penyakit pada daun tanaman	
3.	I Made Budi Adnyan a,I Ketut Gede Darma Putra, danI Putu Agung Bayupat i[21]	Segmenta si Citra Berbasis Clustering Menggun akan Algoritma Fuzzy C- Means	2015	Mengkombinasi <i>Fuzzy C-Means</i> dengan <i>Statistical Region Merging</i> untuk melakukan segmentasi citra dan didapat hasil Metode <i>FCMSRM</i> yang dimodifikasi dapat melakukan segmentasi citra	<i>FCM</i> dengan <i>Statistical Region Merging</i> dapat digunakan untuk men- segmentasi citra	Jumlah citra yang digunakan untuk pengujian sangat sedikit.
4.	Ms. Kiran R. Gavhale , Prof. Ujwalla Gawand e[22]	An Overview of the Research on Plant Leaves Disease detection using	2014	Citra <i>RGB</i> dilakukan segmentasi lalu mengkonfigurasi <i>neural network</i> untuk mengenali fitur. Didapat hasil Bahwa teknik- teknik yang	Membuat rangkuman dalam hal pendeteksian penyakit pada daun tanaman.	Banyak algoritme yang terkenal dan tidak di rangkum ke dalam paper ini

		Image Processing Techniques		digunakan pada penelitian ini digunakan untuk menganalisis daun tanaman yang sehat dan yang sakit.		
5.	Xuebing Bai, Xinxing Li, Zetian Fu, Xiongjie Lv, Lingxian Zhang [23]	A fuzzy clustering segmentation method based on neighborhood grayscale information for defining cucumber leaf spot disease images	2017	Segmentasi dengan menggunakan <i>fuzzy clustering</i> yang berbasis <i>neighbourhood grayscale information</i> Hasil: -Daun ter-ekstraksi dengan baik - Berhasil mengkombinasikan <i>neighborhood grayscale information</i> dan <i>FCM</i> .	Penulis menemukan metode baru untuk ekstraksi penyakit pada daun timun dengan background citra yang kompleks	Terlalu sedikit citra yang di segmentasi pada penelitian ini.

				-Meningkatkan performa algoritme <i>FCM</i> .		
6.	S. Gnanasravanan, B.Tharani, Mona Sahu[14]	Long short-term memory recurrent neural networks for plant disease identification	2021	Identifikasi penyakit pada tanaman dengan metode <i>LSTM</i> , didapat hasil bahwa segmentasi berhasil dilakukan dan memiliki nilai akurasi yang baik	Berhasil menggunakan <i>LSTM</i> untuk identifikasi penyakit pada tanaman	Akurasi masih kalah dengan algoritme <i>KNN</i>
7.	Sue Han Lee, Hervé Goëau, Pierre Bonnet, Alexis Joly[12]	Attention-based recurrent neural network for plant disease classification	2020	Menggunakan algoritme <i>attention based RNN</i> untuk klasifikasi penyakit pada tanaman. Didapat hasil bahwa <i>attention-based RNN</i> memiliki kemampuan generalisasi yang	Berhasil melakukan implementasi <i>attention based RNN</i> untuk klasifikasi penyakit pada tanaman	Pada data <i>seen crops</i> dari <i>PlantVillage dataset</i> algoritme <i>RNN</i> yang diajukan memiliki akurasi sedikit dibawah algoritme <i>CNN</i> yang diimplementasi

				lebih tinggi dibandingkan dengan <i>CNN</i>		pada penelitian ini.
8.	Yu Wang, Wanjun Zhang[15]	A dense RNN for sequential four-chambered view left ventricle wall segmentation and cardiac state estimation	2021	Menggunakan metode <i>dense RNN</i> untuk segmentasi <i>four-chamber view left ventricle</i> pada dinding jantung. Didapat hasil bahwa gambar <i>frame</i> pertama mengalami peningkatan akurasi yang signifikan dan secara keseluruhan mencapai nilai <i>IoU</i> 92.13%	Berhasil untuk segmentasi <i>four-chamber view left ventricle</i> pada dinding jantung	