

# **MENGIDENTIFIKASI JENIS TANAMAN BERDASARKAN CITRA DAUN MENGUNAKAN ALGORITMA CONVOLUTIONAL NEURAL NETWORK**

Ikbal Suhardin<sup>1</sup>, Andi Patombongi<sup>2</sup>, Andi Muhammad Islah<sup>3</sup>

STMIK Catur Sakti Kendari

Jl. Drs. H. Abdullah Silondae No. 109 Telp. (0401) 3127274 Kendari

<sup>2</sup>andipatombongi@caturasaki.ac.id, <sup>3</sup>andimuhislah@caturasaki.ac.id

*Setiap tanaman memiliki daun dengan bentuk dan ukuran yang berbeda. Meskipun demikian, banyak manusia belum mengetahui jenis-jenis tanaman seperti daun herbal yang dimana memiliki manfaat bagi manusia dalam kesehatan. Pada penelitian ini digunakan "Supervised Learning" untuk membantu mengenali jenis tanaman berdasarkan daunnya. Pertama-tama sejumlah daun akan difoto, lalu foto tersebut akan diresize menjadi citra baru dengan ukuran tertentu, kemudian dimasukkan ke dalam dataset. Lalu citra akan dikonversi menjadi matriks dimana matriks ini akan dimasukkan ke dalam algoritma CNN (Convolutional Neural Network). Pada algoritma CNN, matriks tersebut akan digunakan untuk mengekstraksi fitur yang ada pada citra menggunakan beberapa filter yang sebelumnya telah ditentukan menggunakan metode konvolusi. Lalu hasil konvolusi tersebut akan digunakan untuk pelatihan menggunakan algoritma feedforward dan backpropagation untuk mendapatkan data weight dan bias yang optimal. Setelah itu dilakukan proses test dimana citra uji akan melalui proses konvolusi. Hasil konvolusi akan diklasifikasi menggunakan algoritma feedforward berdasarkan data weight dan bias yang sudah didapatkan dari proses training sebelumnya. Pengujian dilakukan dengan 5 gambar daun: 200 citra sebagai data training (latih), dan 25 citra sebagai data test (uji)..*

**Kata kunci :** Identifikasi Jenis Daun, Supervised Classification, Deep Learning, CNN

## **I. PENDAHULUAN**

Salah satu problem dalam visi komputer yang telah lama dicari solusinya adalah identifikasi objek pada citra secara umum. Bagaimana menduplikasi kemampuan manusia dalam memahami informasi citra, agar komputer dapat mengenali objek pada citra selayaknya manusia. Proses feature engineering yang digunakan pada umumnya sangat terbatas dimana hanya dapat berlaku pada dataset tertentu saja tanpa kemampuan generalisasi

pada jenis citra apapun. Hal tersebut dikarenakan berbagai perbedaan antar citra antara lain perbedaan sudut pandang, perbedaan skala, perbedaan kondisi pencahayaan, deformasi objek, dan sebagainya.

Dalam sistem ini peneliti ingin membuat program dengan mengenali satu objek citra daun dengan berdasarkan dengan melalui metode convolutional neural network. agar nantinya dapat berguna untuk penelitian selanjutnya dengan objek citra daun lainnya seperti daun herbal yang kebanyakan orang yang belum mengetahui jenis-jenis daun herbal karna keterbatasan manusia yang susah membedakan jenis tanaman di karenakan ada beberapa ciri-ciri daun yang menyerupai dan hampir sama ekstra bentuk maupun warna yang dimana menjadi kesulitan manusia untuk mengetahuinya.

Machine Learning merupakan suatu algoritma yang mengolah data, mempelajari data tersebut, dan menerapkan apa yang telah dipelajari untuk membuat sebuah keputusan. Deep Learning merupakan bagian dari Machine Learning yang di rancang untuk terus menganalisis suatu data dengan struktur logika yang mirip dengan bagaimana manusia mengambil keputusan. Convolutional Neural Network (CNN) merupakan salah satu metode Deep Learning yang digunakan untuk mendeteksi dan mengenali sebuah objek pada citra digital. Kemampuan CNN di klaim sebagai model terbaik untuk memecahkan permasalahan Object Detection dan Object Recognition.

## **II. LANDASAN TEORI**

Cara kerja CNN?

Secara garis besarnya, CNN memanfaatkan proses konvolusi dengan menggerakkan sebuah kernel konvolusi (filter) berukuran tertentu ke sebuah gambar, komputer mendapatkan informasi representatif baru dari hasil perkalian bagian gambar tersebut dengan filter yang digunakan.

Gambar 2.1 Image by Adam Geitgey

Convolutional Neural Network (CNN) adalah salah satu jenis neural network yang biasa digunakan pada data image. CNN bisa digunakan untuk mendeteksi dan mengenali object pada sebuah image. CNN adalah sebuah teknik yang terinspirasi dari cara mamalia — manusia, menghasilkan persepsi visual seperti contoh diatas.

Langkah 1 : Memecah gambar menjadi gambar yang lebih kecil yang tumpang tindih

Gambar 2.2 Image by anak-anak

Dari gambar seorang anak kecil yang menaiki kuda mainan diatas, hasil dari proses konvolusi dapat diilustrasikan sebagai berikut ini

Dengan ini, gambar asli dari seorang anak kecil diatas menjadi 77 gambar yang lebih kecil dengan konvolusi yang sama.

Langkah 2 : Memasukkan setiap gambar yang lebih kecil ke small neural network

Setiap gambar kecil dari hasil konvolusi tersebut kemudian dijadikan input untuk menghasilkan sebuah representasi fitur. Hal ini memberikan CNN kemampuan mengenali sebuah objek, dimanapun posisi objek tersebut muncul pada sebuah gambar.

Gambar 2.3 neural network

Proses ini dilakukan untuk semua bagian dari masing-masing gambar kecilnya, dengan menggunakan filter yang sama. Dengan kata lain, setiap bagian gambar akan memiliki faktor pengali yang sama, atau dalam konteks neural network disebut sebagai weights sharing. Jika ada sesuatu yang tampak menarik di setiap gambarnya, maka akan ditandai bagian itu sebagai object of interest.

Langkah 3 : Menyimpan hasil dari masing-masing gambar kecil ke dalam array baru Maka akan terlihat seperti ini:

Gambar 2.4 hail output array

Langkah 4 : Downsampling

Pada langkah 3, array masih terlalu besar, maka untuk mengecilkan ukuran array nya digunakan downsampling yang penggunaannya dinamakan max pooling atau mengambil nilai pixel terbesar di setiap pooling kernel. Dengan begitu, sekalipun mengurangi jumlah parameter, informasi terpenting dari bagian tersebut tetap diambil.

Gambar 2.5 Ilustrasi Max Pooling

Langkah 5 : Membuat prediksi

Sejauh ini, kita telah merubah dari gambar yang berukuran besar menjadi array yang cukup kecil. Nah, array merupakan sekelompok angka, jadi dengan menggunakan array kecil itu kita bisa inputkan ke dalam jaringan saraf lain. Jaringan saraf yang paling terakhir akan memutuskan apakah gambarnya cocok atau tidak. Untuk memberikan perbedaan dari langkah konvolusi, maka bisa kita sebut dengan “fully connected” network. Secara garis besarnya, langkah-langkah diatas tampak seperti gambar berikut ini :

Gambar 2.6 Cara kerja CNN

Arsitektur dari CNN dibagi menjadi 2 bagian besar, Feature Extraction Layer dan Fully-Connected Layer (MLP).

Gambar 2.7 Ilustrasi Arsitektur CNN

Feature Extraction Layer

Proses yang terjadi pada bagian ini adalah melakukan “encoding” dari sebuah image menjadi features yang berupa angka-angka yang merepresentasikan image tersebut (Feature Extraction). Feature extraction layer terdiri dari dua bagian yaitu Convolutional Layer dan Pooling Layer. Namun kadang ada beberapa riset/paper yang tidak menggunakan pooling.

b. Convolutional Layer (Conv. Layer)

Gambar 2.9 Ilustrasi proses konvolusi dengan dua filter (W0 dan W1). Image by CS231n[1]

Ruang Warna (RGB)

Ruang Warna RGB biasa diterapkan pada monitor CRT dan kebanyakan sistem grafika komputer. Ruang warna RGB menggunakan tiga komponen dasar yaitu merah (R), hijau (G), dan biru (B). Setiap pixel dibentuk oleh ketiga komponen warna tersebut. Tabel 2.1 menunjukkan contoh warna dan nilai R, G, dan B (Irianto S. Y., 2016).

Table 2.1. warna dan nilai RGB

WARNA	R	G	B
Merah	255	0	0
Hijau	0	255	0
Biru	0	0	255
Hitam	0	0	0
Putih	255	255	255
Kuning	0	255	0

Gambar 2.8 Image RGB

Gambar tersebut menunjukkan RGB (Red, Green, Blue) gambar berukuran 32x32 pixel yang sebenarnya adalah multidimensional array dengan ukuran 32x32 pixel (3 adalah jumlah channel). Convolutional layer terdiri dari neuron yang tersusun sedemikian rupa sehingga membentuk sebuah filter dengan panjang dan tinggi (pixel). Sebagai contoh, layer pertama pada feature extraction layer adalah conv. layer dengan ukuran 5x5x3. Panjang 5 pixel, tinggi 5 pixel, dan tebal/jumlah 3 buah sesuai dengan channel dari gambar tersebut.

Ketiga filter ini akan digeser keseluruhan bagian dari gambar. Setiap pergeseran akan dilakukan operasi “dot” antara input dan nilai dari filter tersebut sehingga menghasilkan sebuah output atau biasa disebut sebagai activation map atau feature map. Proses dari feature map seperti pada gambar berikut.

RGB biasa digunakan karena kemudahan dalam perancangan hardware, tetapi sebenarnya tidak ideal untuk beberapa aplikasi. Mengingat warna merah, hijau, dan biru sesungguhnya terkolerasi erat, sangat sulit untuk beberapa algoritma pemrosesan citra.

Gradient Computer

Setelah proses konversi citra yaitu mengubah gambar dalam bentuk grayscale langkah selanjutnya adalah menghitung nilai gradien setiap piksel dalam gambar.

a. Spatial Orientation Binning

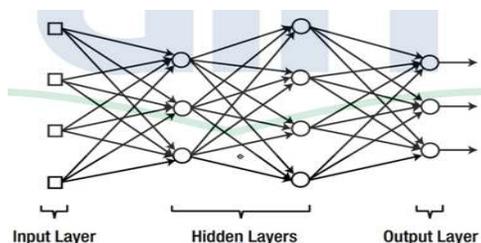
Untuk membuat sebuah histogram dibutuhkan nilai gradien dan nilai tersebut didapat dari nilai tiap piksel dalam sebuah gambar. Gambar kemudian akan dibagi menjadi cells dengan ukuran yang telah ditentukan. Jadi tiap cells dalam gambar akan dibuat histogramnya untuk mengetahui nilai dalam tiap cells karena tiap cell mempunyai nilai yang berbeda. Dalam pembuatan histogramnya diperlukan adanya bin untuk mengetahui nilai gradiennya. Bin akan ditentukan sendiri oleh pengguna. Dalam penelitian sebelumnya bin yang digunakan adalah 4 bin orientation

### Deep Learning

Deep learning memungkinkan model komputasi yang terdiri atas beberapa lapisan pemrosesan untuk mempelajari representasi data dengan berbagai tingkat abstraksi. Deep learning menemukan struktur rumit dalam kumpulan data besar dengan menggunakan algoritma backpropagation untuk menunjukkan bagaimana mesin harus mengubah parameter internal yang digunakan untuk menghitung representasi di setiap lapisan dari representasi di lapisan sebelumnya (LeCun et al., 2015). Deep learning terdiri atas:

#### Neural Network

Istilah neural networks pertama kali digunakan oleh McCulloch & Pitts (1990) dalam percobaan untuk menemukan representasi matematis dari pemrosesan informasi dalam sistem biologis. Jaringan saraf (neural networks) merupakan jaringan dari node (simpul), yang meniru struktur neuron otak dari makhluk hidup. Node menghitung jumlah nilai bobot dari masukan dan memprosesnya pada lapisan tersembunyi, lalu mengeluarkan hasil dari fungsi pengaktifan dengan nilai bobot. Neural networks telah dikembangkan dari arsitektur sederhana menjadi struktur yang semakin kompleks. Awalnya, pelopor neural networks memiliki arsitektur yang sangat sederhana dengan hanya lapisan input dan output, yang disebut jaringan saraf single-layer. Ketika lapisan tersembunyi (hidden layer) ditambahkan ke jaringan saraf single-layer, maka akan menghasilkan jaringan saraf multi-layer. Oleh karena itu, jaringan saraf multi-layer terdiri atas lapisan input, lapisan tersembunyi, dan lapisan output seperti pada Gambar 2.2 (Kim, 2017). Untuk mendapatkan neuron tujuan ( $y$ ) maka nilai yang ada pada neuron ( $x$ ) dikalkulasi dengan bobot ( $w$ ) dan ditambahkan dengan bias ( $b$ ) lalu diaktivasi dengan fungsi ( $g$ ), yang akan menentukan neuron selanjutnya ( $y$ ).



Gambar 2.10 Struktur dari neural networks

#### Fungsi Aktivasi

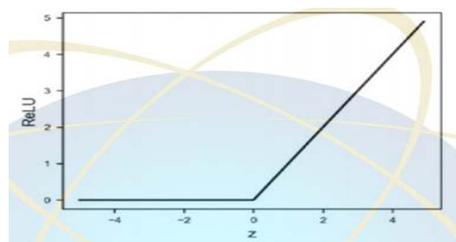
Fungsi aktivasi adalah sebuah fungsi yang berguna untuk menentukan aktif tidaknya neuron di dalam neural networks.

##### 1. Softmax

Fungsi Softmax adalah fungsi eksponensial yang dinormalisasi untuk mengubah vektor asli  $D$ -dimensi dengan nilai riil yang berubah-ubah menjadi vektor probabilitas  $D$ -dimensi dengan nilai riil dalam kisaran  $[0,1]$ . Fungsi Softmax biasanya diterapkan ke bidang pembelajaran mesin, seperti regresi logistik, jaringan saraf tiruan, pembelajaran penguatan. Fungsi Softmax dapat digunakan untuk menghitung nilai dari probabilitas untuk semua label. Rumus dari softmax dapat dilihat pada

persamaan (2.2) dimana nilai probabilitas ( $S$ ) pada kelas ke ( $y$ ) diambil dari neuron pada layer klasifikasi terakhir yang berupa angka eksponensial ( $e$ ) yang dibagi jumlah nilai eksponensial itu sendiri. Hasil dari label yang ada mengubahnya, akan diambil sebuah vektor nilai yang memiliki nilai riil dan mengubahnya menjadi vektor dengan nilai dengan kisaran angka nol dan satu. Jika semua hasil dijumlah maka akan bernilai satu Rectified Linear Unit (ReLU)

ReLU menggunakan fungsi  $f(z) = \max(0, z)$ , yang artinya jika output positif maka akan menghasilkan nilai yang sama, jika tidak maka akan menghasilkan nilai 0. ReLU tidak hanya meningkatkan kinerja secara signifikan tetapi juga membantu mengurangi jumlah perhitungan selama fase pelatihan. Hal ini terjadi akibat dari nilai 0 dalam output ketika nilai  $z$  negatif, sehingga menonaktifkan neuron (Moolayil, 2019). Hasil dari fungsi ditampilkan dalam Gambar 2.11.

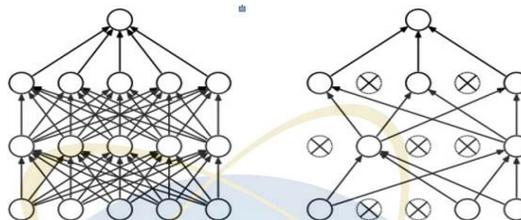


Gambar 2.11 Fungsi aktivasi ReLU

#### DropOut

DropOut adalah salah satu teknik yang digunakan untuk menghindari terjadinya overfitting dalam model. Dalam metode ini, aktivasi beberapa neuron yang dipilih secara acak dalam jaringan diambil sebagai nol selama pelatihan. Neuron yang dipilih diubah dalam setiap iterasi pelatihan. Proses pembelajaran menjadi lebih andal dan overfitting dikurangi dengan metode ini.

Istilah “DropOut” mengacu pada pemutusan neuron (tersembunyi dan terlihat) dalam neural network. Dengan mengeluarkan unit (neural) untuk sementara menghapusnya dari jaringan (network), bersama dengan semua koneksi masuk dan keluarnya, seperti yang ditunjukkan pada Gambar 2.12. Pemilihan unit yang dijatuhkan secara acak (Srivastava et al., 2014).



Gambar 2.12 kiri adalah Sebelah neural networks biasa, sebelah kanan setelah melakukan DropOut

#### 4. Loss Function

Goodfellow et al. (2016) menjelaskan loss function atau cost function adalah metode untuk mengevaluasi seberapa baik algoritma dalam memodelkan data yang diberikan. Jika hasil prediksi menyimpang terlalu banyak dari hasil aktual, loss function akan memiliki nilai dalam jumlah yang sangat besar. Secara bertahap, dengan bantuan

beberapa fungsi pengoptimalan, loss function belajar untuk mengurangi kesalahan dalam prediksi.

5. Backpropagation

Menurut Cilimkovic (2015) backpropagation merupakan algoritma untuk mencari nilai minimum dari loss function dalam bobot (weight) menggunakan teknik yang disebut aturan delta atau gradient descent. Bobot yang meminimalkan loss function kemudian dianggap sebagai solusi untuk masalah pembelajaran.

Algoritma dapat dibagi kedalam empat langkah berikut:

6. feed forward

Feed forward di buat untuk mengatasi adanya gangguan (disturbance) perbedaan antara keduanya terletak pada dari sisi mana gangguan tersebut terjadi. Pada cascade control, penamnaan control kedua (slave/inner control) bertujuan untuk mengatasi gangguan (disturbance) yang terjadi pada manipulated variabel, sedangkan feedforward control di gunakan untuk mengatasi gangguan pada beban (load).

a. Perhitungan feed-forward

b. Backpropagation ke lapisan output

Algoritma dihentikan ketika nilai fungsi kesalahan menjadi cukup kecil

Lapisan Convolutional Neural Network

Convolutional Neural Network (CNN) terdiri atas tiga lapis (layer) yaitu lapis masukan (input layer), lapis keluaran (output layer), dan beberapa lapis tersembunyi (hidden layers). Lapis tersembunyi (hidden layer) umumnya berisi convolutional layers, pooling layers, normalization layers, ReLU layer, fully connected layers, dan serta loss layer (Alom et al., 2018).

a. Convolutional Layer

Lapisan konvolusional (convolutional layer) merupakan lapisan inti CNN, pada lapisan ini sebagian besar proses komputasi dilakukan. Tujuan utama konvolusi dalam kaitannya dengan ConvNet adalah untuk mengekstraksi fitur dari gambar yang dimasukkan (Karim, 2018). Lapisan konvolusi terdiri atas struktur dengan sejumlah filter dengan ukuran tetap yang memungkinkan fungsi kompleks diterapkan pada gambar yang telah dimasukkan. Proses ini dilakukan dengan cara menggeser filter di atas gambar. Setiap filter memiliki bobot dan nilai bias yang sama di seluruh gambar selama proses ini. Proses ini disebut mekanisme pembagian nilai berat dan mekanisme ini memberikan kemampuan untuk mewakili fitur yang sama pada seluruh gambar (Sarigül, Ozyildirim, & Avci, 2019). Lapisan konvolusi menghasilkan gambar baru yang disebut peta fitur. Peta fitur menonjolkan fitur unik dari gambar asli. Lapisan konvolusi beroperasi dengan cara yang sangat berbeda dibandingkan dengan lapisan jaringan saraf lainnya. Pada Gambar 2.6 menunjukkan proses dari lapisan konvolusi, di mana tanda \* menunjukkan operasi konvolusi, dan tanda  $\phi$  adalah fungsi aktivasi. Ikon dengan skala abu-abu (greyscale) di antara operator ini menunjukkan filter konvolusi. Lapisan konvolusi menghasilkan jumlah peta fitur yang sama dengan filter konvolusi. Karena itu, misalnya, jika lapisan konvolusi berisi empat filter, itu akan menghasilkan empat peta fitur (Kim, 2017).

Terdapat dua komponen penting lainnya dalam convolutional layer:

1. Strides

Stride adalah jumlah yang digunakan oleh filter untuk menggeser gambar. Jika stride bernilai 1 maka filter akan berpindah 1 piksel secara horizontal dan vertikal. Hingga konvolusi menjadi sama dengan lebar dan kedalaman dari gambar yang dimasukkan. Stride bernilai 2 membuat lapisan convolutional dengan setengah lebar dan tinggi gambar. Jika filter meluas di luar dari ukuran gambar, maka dapat mengabaikan nilai yang tidak diketahui ini atau menggantinya dengan nilai nol (Karim, 2018)

2. Padding

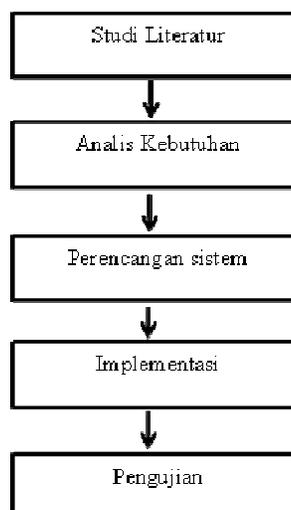
Menurut Karim (2018) padding merupakan operasi untuk menambah ukuran dari data yang input. Dalam data satu dimensi, cukup menambahkan array dengan konstanta; dalam data dua dimensi, data dapat ditambahkan dengan mengelilingi matriks dengan konstanta. Dalam n-dimensional, data dikelilingi hypercube n-dimensional dengan konstanta. Penggunaan padding bertujuan untuk memanipulasi dimensi output dari peta fitur. Penggunaan padding dapat untuk mengatur dimensi output agar tetap sama seperti dimensi input atau setidaknya tidak berkurang drastis sehingga dapat dilakukan ekstraksi feature yang lebih mendalam. Dalam sebagian besar kasus, nilai konstanta ini adalah nol dan disebut zero padding:

- Valid padding: Hanya menjatuhkan nilai dari kolom paling kanan (atau baris paling bawah)
- Same Padding: Mencoba untuk meratakan kiri dan kanan data, tetapi jika jumlah kolom yang akan ditambahkan adalah ganjil, maka akan menambahkan kolom tambahan ke kanan.

III. METODE PENELITIAN

Tahapan Penelitian

Metode yang digunakan melalui beberapa tahapan kerangka kerja agar penelitian ini lebih terstruktur dan teratur. Gambar 3.1 merupakan diagram blok penelitian



Gambar 3.1. Tahapan Penelitian

Pelaksanaan penelitian ini, dibagi menjadi tiga tahapan, yaitu tahap persiapan, tahap pelaksanaan, dan tahap

laporan akhir dan presentasi. Adapun uraiannya sebagai berikut :

3.1.1. Studi Literatur

Penulis melakkukan Studi literatur dalam penelitian ini dengan mencari referensi teori yang relefan dengan kasus atau permasalahan yang ditemukan yang bertujuan agar dapat membantu dalam penyelesaian masalah. Langkah ini dilakukan untuk mempelajari berbagai sumber atau teori buku, jurnal dan internet yang berhubungan dengan penelitian.

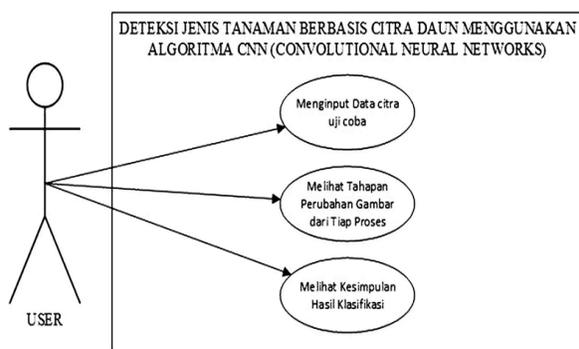
3.1.2. Analisis kebutuhan

Tujuan dari tahap analisis untuk memahami dengan sesungguhnya kebutuhan dari sistem dan menentukan kebutuhan atau kondisi yang harus dipenuhi oleh sistem.

3.1.2.1. Kebutuhan Pengguna

Kebutuhan pengguna digambarkan dalam diagram use case yang terlihat pada Gambar 3.2.

1) Use Case System



Pada Gambar Use Case System menunjukkan hal yang dapat dilakukan oleh user di dalam sistem. Hal-hal tersebut antara lain:

Bahan dan Peralatan yang Digunakan

Perangkat lunak yang digunakan dalam pengimplemetnasian identifikasi citra pada dataset Caltech-101 adalah sebagai berikut:

Tabel 3.1 Bahan dan Peralatan

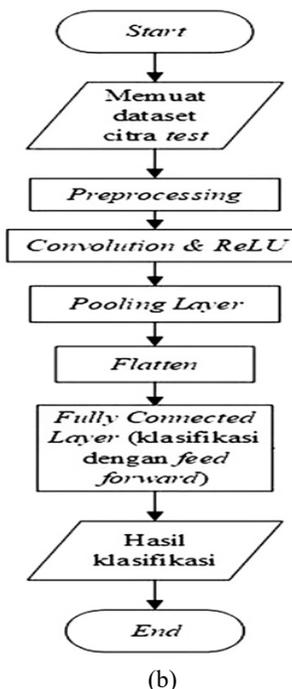
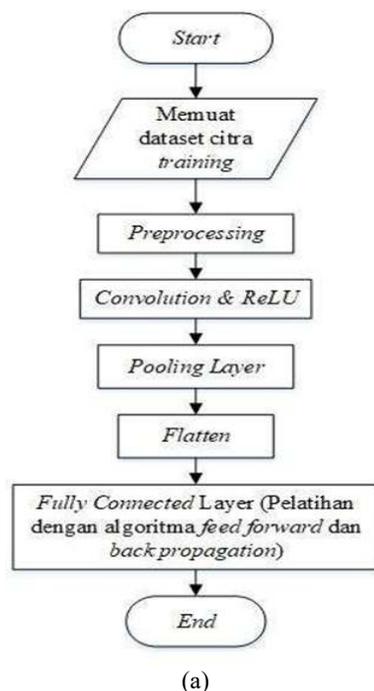
Perangkat Keras	Komputer visi dan hanphone
Perangkat Lunak	Matlab R2020a Windows 7 Ultimate-32 bit

Data Masukan

Data masukan pada sistem ini berupa citra dengan telah dirubah menjadi gray scale data citra yang digunakan adalah dataset trainingBR-250. trainingBR-250 adalah dataset dengan data citra yang terdiri dari 250 kelas objek dan 1 kelas background. Jumlah data disetiap folder antara 5 data citra 40 sampai data citra. Jumlah data citra disetiap kelas rata-rata 40 data citra.

Perancangan Sistem

Sistem deteksi dan pengenalan jenis daun yang dikembangkan pada penelitian ini dapat dimodelkan dengan flowchart yang terdapat pada Gambar berikut :



Gambar 3.2 Flowchart Sistem (a) Training dan (b) Test

Pada gambar 3.2, terdapat dua alur proses, alur proses (a) adalah alur proses sistem data training dan (b) data test. Pada permulaan dari kedua alur proses, baik citra uji dan citra test akan dimuat untuk masuk ke tahap Preprocessing. Pada tahap ini dilakukan resize gambar pada citra test maupun citra training. Setelah citra

melewati tahap preprocessing, citra akan melalui proses convolution, dimana pada tahap ini citra akan direpresentasikan dalam bentuk matriks 2 dimensi yang mana akan dilakukan perhitungan dot product dengan filter yang telah ditentukan. Setelah proses convolution selesai dilakukan, tiap nilai elemen dari hasil Convolution akan menggunakan fungsi aktivasi ReLU (Rectified Linear Unit). Setelah proses Convolution dan ReLU, hasil matriks dari proses sebelumnya akan dilakukan pooling, dimana jenis pooling yang digunakan di tahap ini adalah max pooling untuk mengambil nilai terbesar dari area matriks dan mengumpulkannya menjadi suatu matriks baru dengan ukuran yang lebih kecil. Setelah itu, dilakukan proses flatten, dimana matriks 2 dimensi yang dihasilkan di tahap pooling akan diubah menjadi satu baris array, lalu data citra training yang di-flatten akan disimpan ke notepad yang akan diakses oleh proses training sedangkan data citra test yang di-flatten akan langsung di lewatkan ke proses citra selanjutnya.

Pada proses (a), hasil dari proses flatten pada citra training akan dilanjutkan ke proses Fully Connected Layer. Pada tahap ini data flatten pada citra pelatihan yang disimpan di notepad akan di- training dengan algoritma CNN (Convolutional Neural Network) yang mana proses pelatihan meliputi tahap feed forward untuk menentukan nilai neuron berdasarkan target yang dituju dan backpropagation untuk menentukan weight dan bias berdasarkan jumlah iterasi pelatihan (epoch) dan learning rate yang mana merupakan parameter tambahan yang diperlukan agar neuron yang dihasilkan pada tahap feed forward mendekati target dengan error seminim mungkin, hasil pelatihan berupa nilai weight (w) dan bias (b) yang akan digunakan untuk mengklasifikasikan tanaman.

Pada proses (b), hasil dari proses flatten akan citra test akan dilakukan proses klasifikasi, sebelum melakukan proses klasifikasi akan dilakukan pengecekan apakah data training sudah dilatih atau belum. Setelah melakukan pengecekan, hasil dari proses flatten citra

test akan diklasifikasi menggunakan feed forward. Hasil akhir akan ditampilkan pada sistem. Penjelasan ringkas mengenai tahapan yang dilakukan akan dijelaskan sebagai berikut:

#### 1. Memuat dataset citra

Disiapkan sebuah dataset yang terdiri dari dataset citra training dan kemudian citra ini direpresentasikan ke dalam bentuk matriks 3 channel (Red, Green, Blue).

#### 2. Preprocessing

Tahap ini bertujuan untuk memperkecil (rezise) tiap citra yang ada menjadi ukuran tertentu agar mirip dan memudahkan proses selanjutnya.

#### 3. Convolution & ReLU

Citra yang telah di-processing dari tahap sebelumnya akan dilakukan proses convolution. Dimana pada tahap ini,

citra akan digambarkan dalam bentuk matriks yang terdiri dari angka 0 sampai 255. Lalu, citra akan di konvolusi (convolution) dengan beberapa filter yang mana hasilnya akan diteruskan pada proses ReLU yang dilakukan dengan mengubah setiap elemen citra konvolusi yang bernilai dibawah 0 (minus/negatif) diubah atau dimutlakan menjadi 0 (nol).

#### 4. Pooling layer

Membagikan nilai output dari convolution layer menjadi beberapa kotak kecil, kemudian mengambil nilai maksimal dari setiap kotak untuk menyusun matriks citra yang telah ditentukan untuk mengurangi dimensi fitur matriks.

#### 5. Flatten

Mengubah nilai masukan menjadi sebuah nilai array hasil pooling. Setiap hasil dari proses Pooling Layer akan diubah ke dalam array satu dimensi.

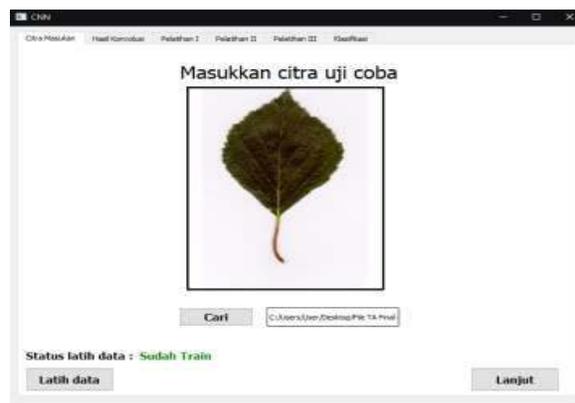
#### 6. Fully Connected Layer

Proses dimana matriks yang sudah di-flatten dimasukkan untuk melewati jaringan neuron agar dapat memprediksi probabilitas keluaran Nilai flatten dari proses akan digunakan untuk melatih neural network.

#### Implementasi dan Pengujian Implementasi

Pada tahap implementasi sistem, yang dibuat sesuai dengan rancangan sistem yang diperlukan dilapangan. Setelah sistem dianalisis dan didesain secara rinci, maka akan menuju tahap implementasi. Implementasi merupakan tahap meletakkan sistem sehingga siap untuk dioperasikan. Implementasi bertujuan untuk mengkonfirmasi modul-modul perancangan, sehingga pengguna dapat memberikan masukan kepada pembangun sistem.

Sebelum dilakukan proses klasifikasi menggunakan Convolutional Neural Network, pertama sekali, user akan diminta untuk memulai pelatihan jika



Gambar 3.3 Tampilan Input Citra Masuk

data pelatihan tidak ada status pelatihan dapat diketahui melalui pesan status yang ada disamping tombol “Latih data”.

Pada form Deteksi dan Pengenalan Jenis Daun, proses – proses tersebut akan di klik pada tombol yang tersedia pada Gambar 4. Penjelasan mengenai tombol dan proses yang terjadi ketika tombol di klik adalah sebagai berikut: Tombol “Cari”, tombol yang digunakan untuk memasukkan citra berwarna objek daun yang sudah ada di dalam data test. Cara memasukkan citra tersebut dapat dilakukan dengan cara menekan tombol Open. Tombol “Latih Data”, tombol yang digunakan untuk men-training data daun yang belum di training. Status latih data akan berwarna merah jika data daun belum di training dan akan berwarna hijau jika data daun sudah di training dan juga tombol “Lanjut” tidak berfungsi. Tombol “Lanjut”, tombol yang digunakan untuk masuk ke tab hasil konvolusi. Gambar yang diambil dari data test hanya gambar yang memiliki background monochrome (satu warna). Gambar yang bisa di test adalah gambar yang sudah ada di dalam data test (uji).

#### 4.1.3.2. Pengujian

Peneliti melakukan beberapa pengujian berdasarkan perancangan program yang telah dibuat.

1. Pengujian koneksi antara computer dan kamera
2. Pengujian pengambilan data daun pada tumbuhan.
3. Pengujian resolusi gambar atau citra daun
  - Resolusi Video, Beberapa resolusi video yang akan diuji adalah 720 x 480 dan 480 x 480.

## VI. HASIL DAN PEMBAHASAN

Pengujian yang dilakukan pada sistem ini merupakan pengujian terhadap perangkat keras dan juga perangkat lunak yang sudah saling terhubung, tujuan dari pengujian ini adalah untuk mengetahui apakah sistem sudah berjalan dengan baik dan sesuai yang diharapkan.

### Preprocessing Data

Tahapan preprocessing terhadap data citra dilakukan dengan melakukan pengumpulan data (gathering), mengubah ukuran pixel pada citra (resize), dan melakukan augmentasi pada citra dari keseluruhan citra training.

#### Pengumpulan data citra

##### Data training

Data pertama adalah data training yang terdiri atas 5 jenis daun (daun cabai, bayam putih, bayam merah, kangkung, daun mimjangan,) dan masing-masing jenis berjumlah sekitar 40 gambar, sehingga total terdapat 200 data gambar.

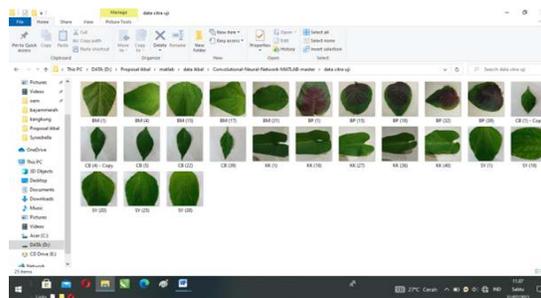
##### Pelabelan data

folder training dan folder testing. Folder training berfungsi untuk menaruh data untuk diproses pada proses pembelajaran, sedangkan folder testing berfungsi untuk memvalidasi data pada proses training. Pada setiap subfolder diberi nama daun dan disisi dengan data sesuai namanya seperti pada Gambar 4.3. Perbandingan jumlah

data pada folder training berjumlah 50% dan folder testing atau sebesar 20% dari jumlah data.

### a. Input Layer

Tahapan pertama dalam melakukan pelatihan terhadap model adalah memasukkan data citra daun ke input layer, pada layer ini citra gambar dikonversi kedalam matriks tiga dimensi, dengan ukuran panjang×lebar×3 channel RGB (Red, Green, Blue). Pada penelitian ini nilai RGB pada setiap piksel dinormalisasi menjadi rentang 0-1 untuk mempermudah proses komputasi dengan cara membagi setiap nilai RGB setiap piksel dengan 337.



Gambar 4.1 Data uji coba

### Proses Training daun

Hasil yang didapatkan untuk proses pelatihan terhadap dataset training menghasilkan nilai Accuracy dan nilai loss, dengan nilai Accuracy tertinggi pada nilai loss training hingga 0.5 pada nilai Accuracy. Semakin tinggi nilai Accuracy maka semakin baik kinerja model yang didapatkan. Adapun nilai loss semakin rendah nilai loss maka semakin buruk kinerja model yang di dapatkan.

### Proses training

Pada proses ini akan dilakukan training terhadap data daun yang sudah peneliti kumpulkan sebelumnya. Grafik training data pertama adalah sebagai berikut :



Gambar 4.2 Model untuk dataset training 1

### 1. training 1

Akurasi pada dataset training 1 mendapatkan hasil akurasi tertinggi 57,03 untuk hasil pada validasi dan pada hasil training. Pelatihan dengan dapat menghasilkan akurasi tertinggi pada training

pelatihan pertama 1 dengan proses training pembelajaran itu masih kurang baik dengan hasilnya

**Keterangan :**

Training on single CPU.  
Initializing input data normalization.

Epoch	Iteration	Time Elapsed (hh:mm:ss)	Mini-batch Accuracy	Mini-batch Loss	Base Learning Rate
1	1	00:00:13	17.97%	2.0402	1.0000e-04
50	50	00:01:50	27.34%	1.7332	1.0000e-04
100	100	00:03:23	29.69%	1.6347	1.0000e-04

Proses pembelajaran :

```
imageSize = [64 64 3];
options = trainingOptions('sgdm', ...
'MaxEpochs',100,...
```

- Hasil :
- Accuracy training = 29.69 sangat buruk
- Loss training = 1,6347 sangat buruk

**V. KESIMPULAN DAN SARAN**

**a. Kesimpulan**

Program identifikasi citra daun pada penelitian ini menggunakan algoritma yaitu convolutional neural network dimana CNN bekerja sebagai untuk mendeteksi dan mengenali object pada sebuah image. kemudian akan mengklasifikasi objek future daun. Pada algoritma CNN yang berfungsi sebagai mengklasifikasi objek future daun yang peneliti sudah membuat folder masing-masing jenis daun.

1. Berdasarkan hasil testing terhadap data yang sudah ditraining memiliki persentasi akurasi deteksi lebih tinggi dari pada data testing yang dilakukan tidak diambil dari data training, tetapi walaupun tidak memiliki tingkat akurasi yang tinggi data testing yang tidak di training masih dapat terdeteksi. Jadi kesimpulannya, semakin banyak data training maka semakin tinggi tingkat akurasi deteksi yang dihasilkan.

2. Pada proses testing dilakukan percobaan dengan beberapa variasi angka imageSize, dan MaxEpochs. Maka disimpulkan angka imageSize 200x200x3 ke atas memiliki akurasi paling tinggi. Selanjutnya angka MaxEpochs yang dimana proses lama belajar agar mendapatkan angka testing yang paling akurat yaitu 80%/dan 90%. Di mana angka ini adalah keyakinan bahwa future yang testing adalah benar.

3. Proses pengujian terhadap pengaruh resolusi dimana pada semua pengujian yang dilakukan menggunakan ukuran citra dengan kualitas pixel 200 x 200 menghasilkan akurasi yang bagus, sedangkan jika di perbesar maka akan mengurangi akurasi deteksi. Jadi disimpulkan bahwa semakin besar ukuran gambar maka semakin tidak akurat.

4. Berdasarkan hasil pengujian pengaruh pengambilan data gambar tidak bagus (buram). Maka banyak hasil testing yang sudah di training itu banyak yang eror dibandingkan benar dari beberapa uji coba gambar . Pengujian dilakukan dengan beberapa data gambar yang sudah ditraing dan tidak ditraining dan disimpulkan data gambar yang tidak training memiliki akurasi rendah begitupun sebaliknya, data gambar yang dtraing memiliki akurasi yang tinggi.

**Saran**

Diharapkan kedepannya, terutama untuk penelitian ini dapat dikembangkan lebih baik lagi, maka penulis memberikan saran kepada peneliti selajutnya yang ingin melanjutkan penelitian ini, adalah sebagai berikut :

1. Mengembangkan program yang lebih dinamis dalam menentukan imagesize future, agar gambar lebih jernihselama proses training daun.
2. Menggunakan PC yang memiliki spesifikasi lebih tinggi dari yang saat ini digunakan.
3. Mengembangkan penelitian dari metode CNN menjadiimplementasi.

**REFERENSI**

[1] Janahi, L., Griffiths, M., Al-Ammal, H., 2015. A conceptual model for IT Governance: A case study research, in: International Conference on Computer Vision and Image Analysis Applications. Presented at the 2015 International Conference on Computer Vision and Image Analysis Applications (ICCVIA), IEEE, Sousse, Tunisia, pp. 1–11. <https://doi.org/10.1109/ICCVIA.2015.7351894>

[2] Johnstone, D., Huff, S., Hope, B., 2006. IT Projects: Conflict, Governance, and Systems Thinking, in: Proceedings of the 39th Annual Hawaii International Conference on System Sciences (HICSS'06). Presented at the Proceedings of the 39th Annual Hawaii International Conference on System Sciences (HICSS'06), IEEE, Kauia, HI, USA, pp. 197b– 197b. <https://doi.org/10.1109/HICSS.2006.235>

[3] Pesantez-Aviles, F., Calle-Lopez, D., Robles-Bykbaev, V., Rodas-Tobar, M., Vasquez-Vasquez, C., 2017. A Recommender System Based on Data Mining Techniques to Support the Automatic Assignment of Courses to Teachers in Higher Education. IEEE, pp. 231–236. <https://doi.org/10.1109/INCISCOS.2017.22>

[4] Carlson, K. M. et al. (2012) 'Committed carbon emissions, deforestation, and community land conversion from oil palm plantation expansion in West Kalimantan, Indonesia', Proceedings of the National Academy of Sciences, 109(19), pp. 7559–7564.

[5] Prabowo, D. A. and Abdullah, D. (2018) 'Deteksi dan perhitungan objek berdasarkan warna menggunakan Color Object Tracking', Pseudocode, 5(2), pp. 85–91.

[6] Saputra, A. K. (2016) 'Aplikasi Deteksi Objek Menggunakan Histogram Of Oriented Gradient Untuk Modul Sistem Cerdas Pada Robot Nao', Universitas Bandar Lampung, Lampung.

[7] Sari, D. F. and Swanjaya, D. (2020) 'Implementasi Convolutional Neural Network Untuk Identifikasi Penyakit Daun Gambas', in Prosiding SEMNAS INOTEK (Seminar Nasional Inovasi Teknologi), pp. 137–142