

# KLASIFIKASI SPESIES TANAMAN ANTHURIUM BERDASARKAN CITRA DAUN MENGGUNAKAN METODE CONVOLUTIONAL NEURAL NETWORK (CNN)

Alfiansyah Nur Abadi <sup>1</sup>, Agung Toto Wibowo<sup>2</sup>

<sup>1,2</sup>Universitas Telkom, Bandung

alfiansyahnurabadi@students.telkomuniversity.ac.id<sup>1</sup>, agungtoto@telkomuniversity.ac.id<sup>2</sup>

---

## Abstrak

Anthurium merupakan tumbuhan yang berasal dari keluarga Araceae. Diantara beberapa spesies milik araceae, anthurium memiliki jumlah spesies terbesar. Anthurium diperkirakan memiliki lebih dari 1000 spesies dan anggota genus Anthurium. Karena banyaknya jenis dan spesies, dalam tugas akhir ini menggunakan jenis anthurium, yang sering ditemui dan memiliki banyak peminat. Pada tugas akhir ini, kami menggunakan bahasa pemrograman Python dengan metode CNN menggunakan ResNet50 untuk pengolahan citra daun Anthurium. Model yang dibuat untuk mendeteksi dan menentukan spesies dari jenis anthurium. Dalam tugas akhir ini menggunakan dataset yang diambil menggunakan smartphone. Dalam dataset diambil setiap spesies sejumlah 800 data yang sudah diaugmentasi dan data test sejumlah 100 data per spesies.

**Kata Kunci:** Klasifikasi, Convolutional Neural Network (CNN), ResNet50, Anthurium.

---

## Abstract

Anthurium is a plant that comes from the Araceae family. Among several species belonging to the Araceae, anthurium has the largest number of species. Anthurium is estimated to have more than 1000 species and is a member of the genus Anthurium. Due to the many types and species, in this final project, anthurium is used, which is often encountered and has many enthusiasts. In this final project, we use Python programming language with CNN method using ResNet50 for image processing of Anthurium leaves. The model is made to detect and determine the species of the anthurium species. In this final project using a dataset taken using a smartphone. In the dataset, 800 data are taken for each species that have been augmented and the test data is 100 data per species. **Keywords:** Classification, Convolutional Neural Network (CNN), ResNet50, Anthurium.

## 1. Pendahuluan

### 1.1. Latar Belakang

Anthurium termasuk kedalam jenis tanaman dengan salah satu famili terbesar bernama Araceae asli tropis [1]. Anthurium dibagi menjadi dua kelompok yaitu bunga dan daun yang masing-masing terdiri dari lebih dari 500 jenis dan spesies [1]. Beberapa spesies anthurium yang terkenal yaitu *A. amnicola* Dressler, *A. andraeanum* Linden, *A. jenmanii* Engl, *A. crystallinum* Linden, *A. veitchii* Mast. Jenis anthurium daun memiliki bentuk yang unik dan bervariasi merupakan tanaman yang memiliki peminat dan nilai jual yang tinggi karena keunikan dari bentuk daunnya, sehingga daun merupakan hal yang penting untuk memikat mata pecinta tanaman dan menjadikan anthurium berbeda dengan famili tanaman yang lainnya [2].

Ada banyak spesies anthurium yang dapat dijumpai di Indonesia sehingga membuat setiap spesies anthurium memiliki keunikan-keunikan tertentu pada bentuk daunnya [2]. Contohnya pada beberapa daun anthurium memiliki bentuk menyerupai keris dan berbentuk cinta sehingga membuat

pecinta tanaman banyak menyukainya [2]. Karena memiliki beberapa keunikan yang khas anthurium menjadi salah satu tanaman yang memiliki penggemar yang besar sehingga menarik banyak pedagang untuk menjual tanaman anthurium. Karena memiliki banyak jenis dan spesies banyak pedagang atau pecinta tanaman kurang mengetahui pasti nama-nama dari jenis-jenis anthurium yang dijualnya atau dibelinya sehingga sering terjadi salah identifikasi jenis-jenis anthurium dikarenakan ada beberapa jenis anthurium memiliki bentuk-bentuk yang mirip contohnya anthurium hybrid dengan anthurium veitchii sama-sama memiliki bentuk hati [3].

Untuk mengetahui perbedaan-perbedaan jenis anthurium perlu dibuatkan sebuah system yang dapat membedakan dan mengklasifikasikan tanaman anthurium berdasarkan citra daun. Citra daun anthurium dipilih untuk melakukan pengklasifikasian karena daun anthurium memiliki bentuk-bentuk yang unik namun saat tanaman anthurium berusia muda beberapa spesies anthurium memiliki bentuk daun yang hampir sama sehingga diambil lah metode *Convolutional Neural Network* (CNN). CNN merupakan syaraf tiruan yang secara otomatis dapat mengekstrak fitur topologi dari sebuah gambar [4]. CNN banyak digunakan sebagai metode pengenalan data yang berbentuk citra karena hasil akurasi dan kinerjanya yang tidak diragukan lagi pada penyimpanan informasi spasial dari data citra sehingga menghasilkan hasil klasifikasi yang baik. Selain itu pada penelitian ini menggunakan arsitektur ResNet50.

Arsitektur (*Residual neural network*) ResNet mempunyai kemampuan untuk menghindari masalah pengurangan gradien dan ResNet dengan kedalaman 50 lapisan dan memiliki tingkat kedalaman 10 dan 8 kali lebih dalam dari AlexNet dan VGG. Hal ini membuat ResNet50 memiliki akurasi yang lebih baik dibandingkan dengan Alexnet, VGG, MobileNet, LeNet [5]

## 1.2 Topik dan Batasannya

Berdasarkan masalah yang ada diperlukan batasan masalah agar lingkup penelitian menjadi jelas, maka batasan penelitian kali ini yaitu membangun sistem untuk mengklasifikasi tanaman anthurium dengan menggunakan metode CNN dan menganalisa performansi *system* yang dibangun. Sistem yang dibangun hanya dapat melakukan klasifikasi 3 jenis Citra daun tanaman anthurium diantaranya anthurium hybrid, anthurium jemani sawi dan anthurium veitchii. Citra daun yang diambil hanya dari sudut pandang atas atau dari bagian atas daun dengan metode memotret satu-satu dengan bergerak melingkar dari samping kanan ke kiri. Dataset yang diambil seluruhnya menggunakan handphone dengan merek xiaomi mi8 dan iphone 7 dengan teknik pengambilan gambar yang sama.

## 1.3 Tujuan

Tujuan dilakukannya penelitian ini yaitu untuk membangun sistem klasifikasi tanaman anthurium dengan menggunakan metode CNN untuk membedakan lima jenis spesies yang sudah ditentukan dalam batasan yaitu anthurium hybrid, anthurium jemani sawi dan anthurium veitchii. Melakukan pengujian dan menganalisis performansi dari sistem yang dibangun.

## 2. Studi Terkait

Sejauh yang penulis ketahui, penulis menemukan beberapa penelitian terkait dengan klasifikasi tanaman dengan metode *Convolutional Neural Network* (CNN) penulis melihat dari beberapa penelitian yang dilakukan pada tanaman dengan menggunakan citra daun dan dengan berbagai arsitektur yaitu Mobile-Net, ResNet50, LeNet, VGG-16 dan Alexnet [6].

Penelitian tanaman berdasarkan citra daun menggunakan arsitektur Mobile-Net memiliki akurasi yaitu sebesar 86,5%. Penelitian dilakukan dengan menggunakan 2 dataset yaitu F102 dan J30 dengan jumlah citra masing-masing yaitu F102 sebesar 18.200 gambar dan J30 sebesar 1.479 gambar dengan prosentase 80% data test, 10% data validation dan 10% data test [7].

Penelitian selanjutnya berdasarkan citra daun menggunakan arsitektur ResNet50 memiliki akurasi yaitu sebesar 96,53% . Penelitian dilakukan dengan menggunakan 2 dataset yaitu CIFAR-10 dengan CIFAR-100 dengan jumlah citra masing-masing yaitu CIFAR-10 sebesar 60.000 gambar dan CIFAR-100 60.000 gambar dengan prosentase 90% data train dan 10% data test [8].

## 2.1 Tanaman Anthurium

Anthurium termasuk kedalam jenis tanaman dengan salah satu famili terbesar dengan nama Araceae yang berasal dari tropis Amerika Tengah [1]. Anthurium terkenal memiliki bentuk daun unik dan indah. Pada beberapa jenis tanaman anthurium memiliki daun yang dapat tumbuh besar dan memiliki pola daun yang dapat membuat pecinta tanaman hias tertarik.

Anthurium umumnya sering ditanam untuk tanaman hias baik diluar maupun didalam ruangan sehingga tanaman anthurium bersifat flexible dalam penempatannya. Tanaman anthurium membutuhkan tanah, air, sinar matahari cukup dan tanah yang gembur jika tanaman anthurium di tanam pada tempat yang kurang cahaya dan air maka tanaman anthurium akan mudah layu dan kering yang dapat mengakibatkan tanaman anthurium mati. Seiring berkembangnya zaman tanaman anthurium dapat ditanam ditempat-tempat gelap (Dalam ruangan) dengan bantuan alat misting dan lampu pembantu [9]. Saat ini tanaman anthurium memiliki banyak sekali jenis dan spesies dikarenakan banyak petani-petani anthurium melakukan cangkok ataupun silang terhadap beberapa jenis atau spesies tanaman anthurium.

## 2.2 Pengolahan Citra Digital (*Image Processing*)

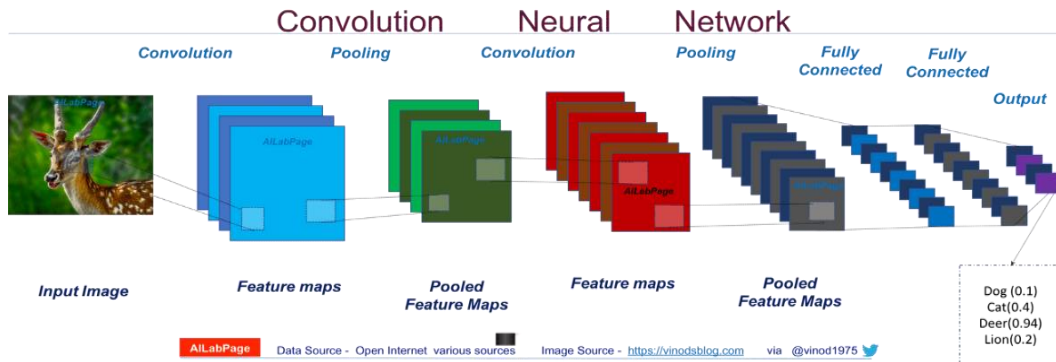
Pengolahan citra merupakan cara sistem dimana melakukan proses dengan masukan (input) gambar dan memiliki output gambar hasil dari modifikasi [10]. Tujuan dilakukannya proses ini adalah untuk melakukan transformasi dan analisa dari suatu gambar [11].

Augmentasi dari gambar merupakan salah satu bagian dari proses pengolahan citra digital, yaitu proses penggandaan pada suatu dataset dengan melakukan proses translasi, transformasi, penambahan/pengurangan noise, rotasi, pembesaran dan flipping dari dataset citra digital [2]. Tujuan dari augmentasi data adalah untuk mengetahui ciri pada sebuah gambar yang optimal yang dimana akan digunakan sebagai bahan analisis dan melakukan proses pengambilan informasi pada suatu objek [12].

## 2.3 Convolutional Neural Network (CNN)

CNN adalah metode yang sering dijadikan metode suatu algoritma untuk melakukan pengolahan citra. Klasifikasi objek pada pengolahan citra menjadi suatu proses yang sering digunakan dalam metode ini. *Convolutional Neural Network* (CNN) memiliki kemampuan yang dapat mengklasifikasi gambar pada dataset yang besar sehingga memiliki hasil akurasi yang baik dan memiliki proses yang ringan dalam melakukan proses pengklasifikasian sehingga metode ini sering digunakan [5].

*Convolutional Neural Network* (CNN) dikenal lebih baik dalam kemampuannya menyimpan informasi spasial dari data citra gambar. Ada beberapa tahap dalam melakukan pengklasifikasian data pada *Convolutional Neural Network* (CNN) yaitu fungsi Aktivasi (softmax) dan fully-connected yang memiliki hasil keluaran berupa klasifikasi. Stuktur CNN terdiri dari Input, *hidden layer*, proses klasifikasi, dan output [6].



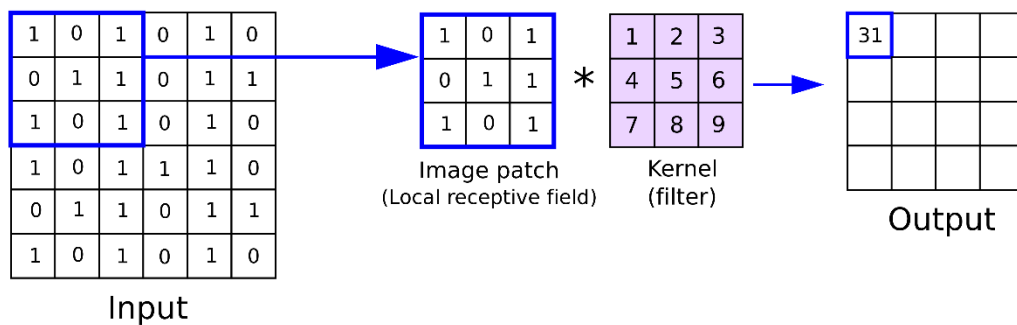
Gambar 1. Tahapan Convolutional Neural Network [2]

**a. Input Layer**

Input Layer merupakan tahapan awal pada *Convolutional Neural Network* (CNN) yaitu tahap memasukan data kedalam sistem yang akan diproses pada tahap selanjutnya atau lapisan *Convolutional Neural Network* (CNN) berikutnya. Input layer berisi data berupa citra gambar yang pada umumnya memiliki 3 bagian yaitu Red, Green, Blue[9].

**b. Convolution Layer**

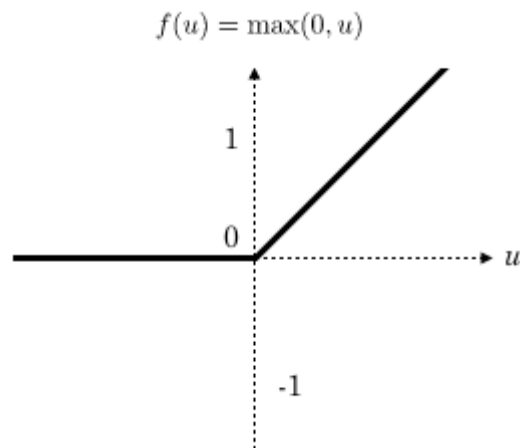
*Convolutional layer* adalah lapisan selanjutnya yang berfungsi untuk melakukan filter atau kernel yang memiliki nilai awal acak. Pada layer ini dilakukan proses konvolusi yang didapat dari layer sebelumnya. Nilai awal acak ini akan di perbarui (*update*) pada saat proses *training* berlangsung. Pada proses ini dikalikan dan bergerak berawal kiri bagian atas ke kanan bagian bawah pada gambar [13].



Gambar 1.2. Ilustrasi Convolution Layer [13]

**c. Fungsi Aktivasi ReLU**

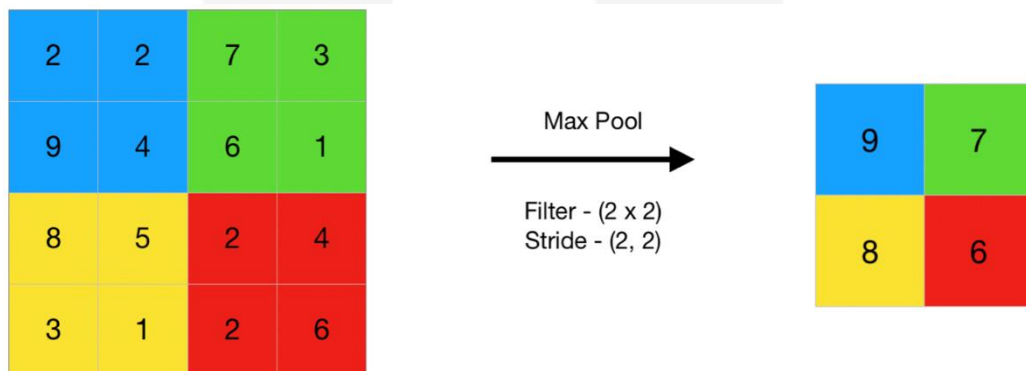
*Rectified Linear Unit* (ReLU) adalah suatu fungsi yang mengeluarkan nilai yang bernilai nol jika  $x < 0$  dan linier dengan kemiringan 1 ketika  $x > 0$ . Pada proses ini akan menerima input data yang kemudian akan di proses sehingga menghasilkan output berupa vektor yang akan diolah lagi pada lapisan-lapisan berikutnya [12]. Fungsi Aktivasi ini dapat bekerja setelah melakukan proses konvolusi dan mendapatkan fitur pada suatu citra gambar. Aktivasi ReLU digunakan untuk mengurangi linearitas yang terjadi pada proses konvolusi sehingga CNN lebih mudah mencapai nilai yang optimum.



**Gambar 1.3. Fungsi Aktifasi ReLU [12]**

#### d. Pooling Layer

*Pooling Layer* adalah proses selanjutnya yang dimana memiliki tujuan untuk meminimalkan ukuran sebuah data citra gambar agar fitur yang akan di ambil memiliki ciri fitur yang jelas. Pada *Convolutional Neural Network* (CNN) banyak digunakan subsampling Max pooling. Cara kerja *Max pooling* yaitu dengan membagi hasil keluaran dari proses tahap convolution layer menjadi beberapa bagian yang selanjutnya mengambil nilai tertinggi dari setiap bagian gridnya untuk menyusun matriks [12]. Proses ini dilakukan untuk mengurangi beban komputasi pada saat melakukan proses training dan untuk membantu sistem jika ditemukan kasus gambar terjadi perubahan posisi. Cara untuk menentukan pooling layer yang ada 2 cara yaitu nilai tertinggi yang diambil dan nilai rata-rata yang diambil.



**Gambar 1.4. Fungsi Pooling Layer [11]**

#### e. Flatten Layer

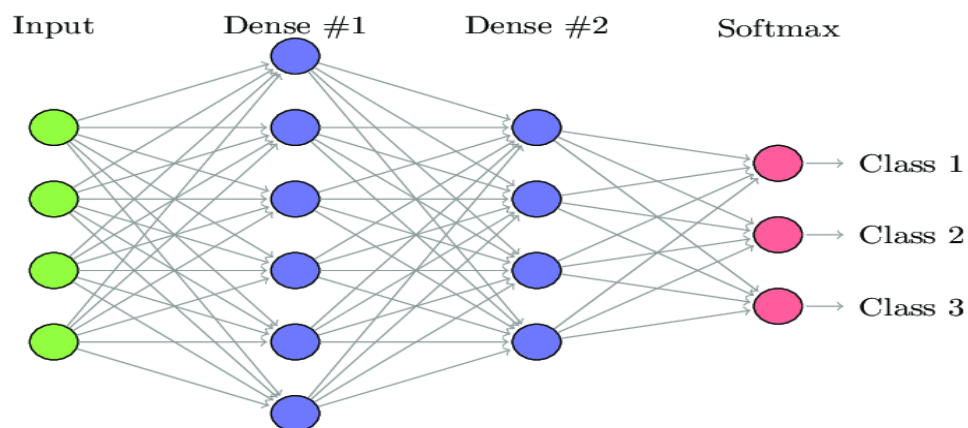
*Flatten Layer* adalah proses terjadinya perubahan bentuk matrix dari proses sebelumnya menjadi sebuah vektor (*array satu dimensi*) proses ini dilakukan untuk fitur yang berhasil di kumpulkan dapat masuk ke proses selanjutnya yaitu proses *input Fully-Connected Layer* atau layer klasifikasi.



Gambar 1.5. Fungsi Flatten Layer [8]

**f. Fully Connected Layer**

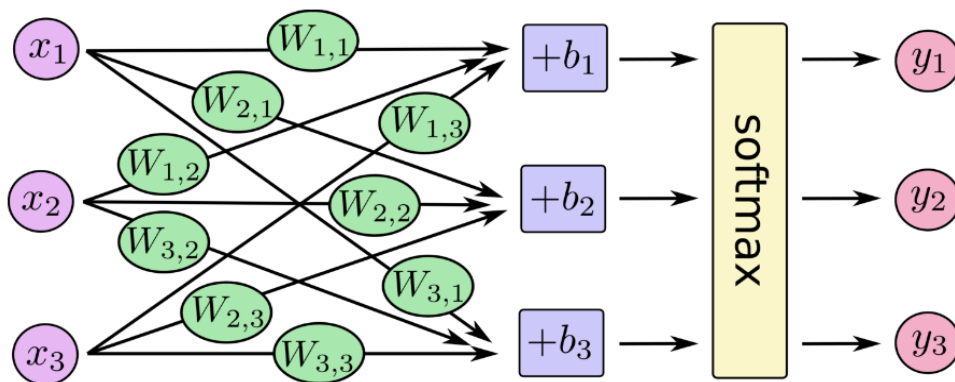
Lapisan ini adalah proses yang mengubah dimensi data, yang bertujuan untuk membuat data dapat diklasifikasikan secara linier. Pada *Fully Connected Layer* adalah proses dimana mencari klasifikasi sebuah kelas yang paling cocok dari input pada input layer dengan menghasilkan probabilitas tertinggi. Pada proses ini terdiri dari 3 bagian yaitu input, lapisan tersembunyi dan output. Pada proses ini node/neuron dari beberapa layer yang ada saling terhubung dan pada output layer aktivasi yang sering digunakan yaitu softmax untuk gambar yang hanya memiliki 1 objek dalam 1 gambar.



Gambar 1.6. Fungsi Fully Connected Layer [10]

**g. Fungsi Aktifasi Softmax**

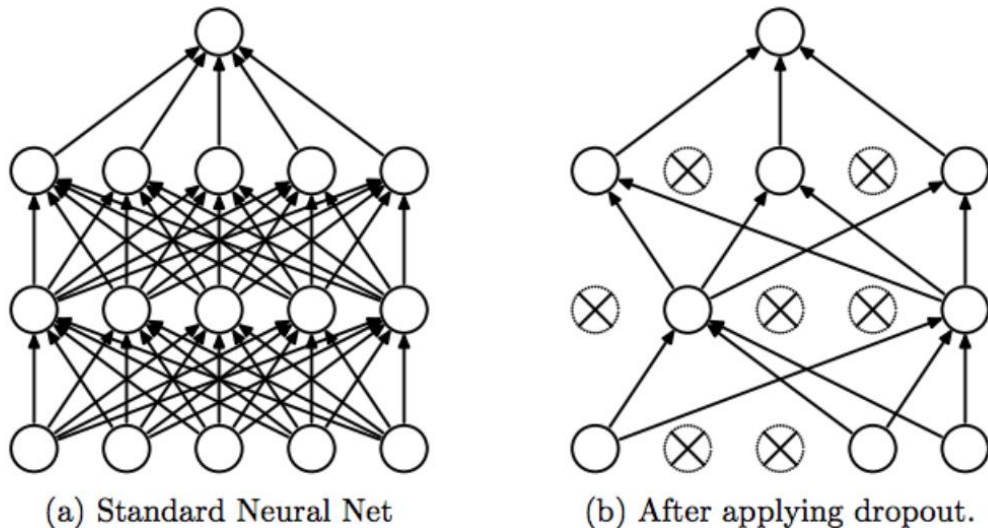
Fungsi Aktivasi *Softmax* biasanya digunakan pada kasus klasifikasi lebih dari 2 kelas (*multi-class*), proses ini berlangsung pada *output layer* pada proses *fully connected layer*. Tujuan dari proses aktivasi softmax untuk menghitung probabilitas pada setiap kelas dan menentukan kelas untuk *input* citra yang diproses pada proses sebelumnya. Fungsi aktivasi softmax menghasilkan nilai antara 0 dan 1.



Gambar 1.7. Fungsi Softmax [9]

**h. Dropout**

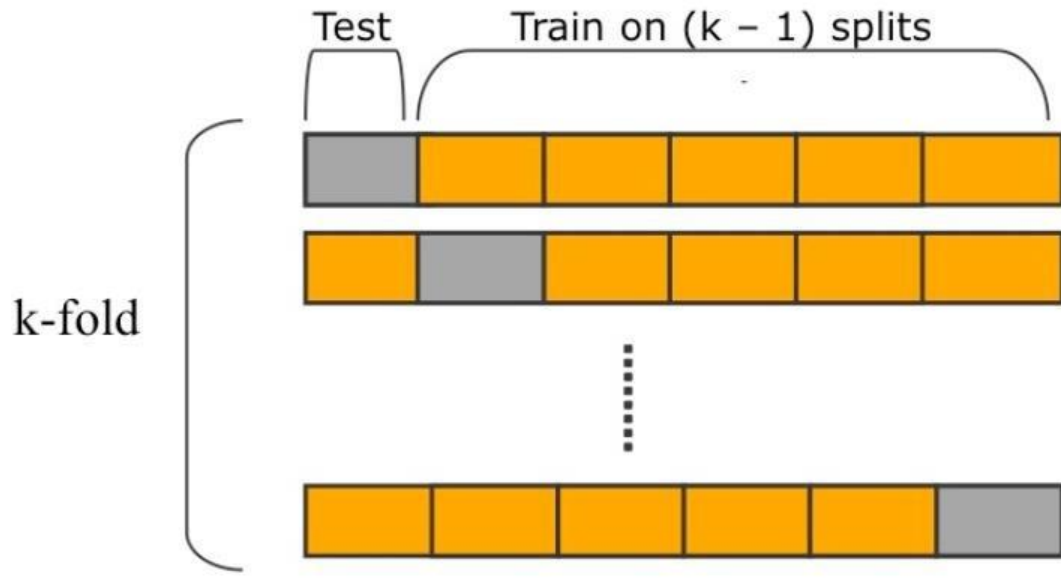
*Dropout* adalah teknik untuk meminimalisir terjadinya *overfitting* pada saat melakukan proses training, yaitu kondisi dimana beberapa pixel akan dibuang secara acak selama proses *training* model berjalan. *Dropout* memiliki tujuan untuk mengurangi terjadinya *overfitting* dan mengurangi atau mempercepat proses training.



Gambar 1.8. Fungsi Dropout [7]

**2.4 Cross Validation K-Fold**

*Cross validation* (K-Fold) merupakan teknik statistik yang sering digunakan dengan tujuan mengevaluasi dan membandingkan model yang akan dibangun. Cara kerja K-Fold yaitu dengan membagi dataset menjadi 2 jenis subset yaitu data untuk *train* dan data untuk validasi. Data train dan validation akan disilangkan sebanyak K perulangan, dan setiap perulangan data akan secara terurut tervalidasi sehingga model yang dihasilkan akan berjumlah K model.



Gambar 2. Ilustrasi K-Fold [6]

## 2.5 Hyperparameter

*Hyperparameter* adalah teknik *eksperimental* parameter yang memiliki nilai ditentukan oleh perancang model CNN sesuai dengan kasus yang akan diteliti. Hyperparameter bertujuan untuk mengontrol proses training agar mendapat nilai akurasi yang lebih baik. Berikut Hyperparameter yang dapat disesuaikan :

### a. Input Shape

Sebelum melakukan input shape ada baiknya citra yang akan digunakan tidak *blur*, gelap, atau gangguan noise lainnya sebelum di augmentasi sebelum model CNN mempelajari features pada citra gambar hal ini bertujuan agar model CNN tidak mengalami overfitting. Selain itu ukuran gambar yang terlalu kecil bisa menjadi faktor CNN tidak bisa mempelajari features pada citra dengan baik.

### b. Batch Size

Pada saat pelatihan data (Training) jika dilakukan sekaligus akan memberatkan proses pelatihan secara komputasi, apalagi jika data yang dilatih memiliki bobot yang besar hingga ribuan apalagi data yang di proses merupakan data RGB. Untuk mengatasi hal ini batch size berfungsi untuk memecah dataset menjadi beberapa bagian kecil [6].

### c. Epoch

Epoch adalah suatu representasi angka yang menunjukkan satu putaran penuh saat proses pelatihan (*training*) terhadap seluruh *dataset*. Semakin tinggi nilai epoch maka nilai skor akurasi akan semakin tinggi dan nilai loss akan semakin rendah. Proses ini di hentikan apabila nilai loss dan nilai akurasi tidak mengalami naik dan turun perubahan secara signifikan.

### d. Jumlah Filter & Filter Size

Jumlah filter pada *Convolutional Layer* bisa disesuaikan, secara umum banyak peneliti yang menggunakan nilai  $2^n$  (misalnya: 8/16/32/64/128/256) sebagai acuan penggunaan angka pada filter.



e. Optimizer

Optimizer adalah Sebuah metode yang biasanya digunakan untuk meng-*update* bobot (*weight*). Hal ini bertujuan untuk menurunkan nilai *loss*.

f. Learning Rate

Learning Rate merupakan parameter yang mengontrol cepat atau lambatnya proses model dalam mempelajari data saat pelatihan (*training*). Hal ini sangat berpengaruh terhadap hasil dari skor akurasi pada model yang dilatih.

g. Jumlah Hidden Layer & Jumlah Node/Neuron

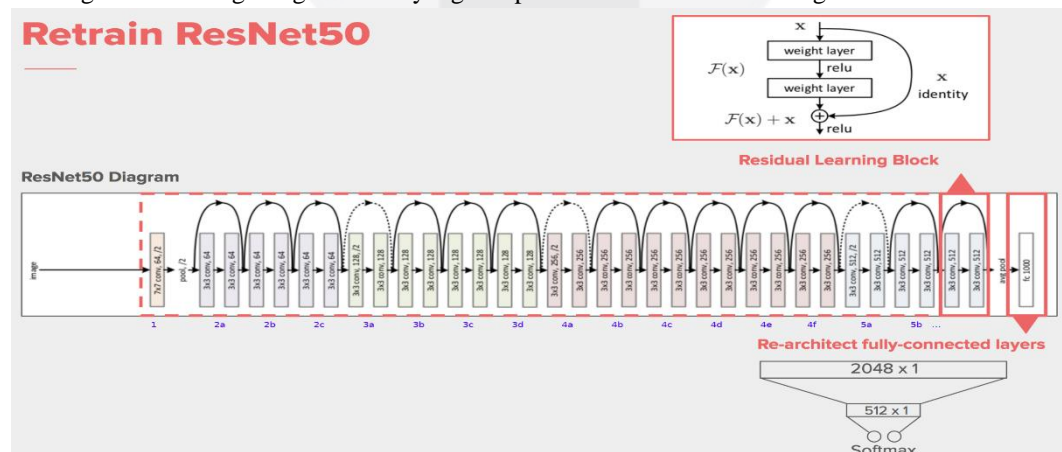
Jumlah *Hidden Layer* pada tahap *Fully Connected* bisa disesuaikan namun biasanya para ahli menggunakan hidden layer dimulai dari 2 *hidden layer* [5]. Kemudian untuk *neuron* pada setiap *Hidden Layer* bisa disesuaikan juga, para ahli biasanya menggunakan nilai  $2^n$  (misalnya: 8/16/32/64/128/256)[5]. Namun jika jumlah *neuron* yang digunakan semakin banyak, maka waktu pelatihan akan semakin lama dan ukuran model CNN akan besar juga.

## 2.6 Evaluasi Model

Untuk mengevaluasi model yang telah dibangun untuk mengetahui keakuratan model dalam mengenali data baru peneliti menggunakan *Confusion Matrix*. Hal ini bertujuan untuk melihat rata-rata akurasi model selama iterasi fold yang dilakukan. Untuk mengetahui berapa akurasi model maka digunakannya sejumlah data lapangan sebagai data testing pada model terbaik yang dibangun untuk mengenali objek.

## 2.7 ResNet50

Pada tahun 2015, He et al mengusulkan arsitektur ResNet. ResNet memiliki kekurangan yaitu informasi sisa selalu diteruskan dan pintasan identitas tidak pernah ditutup [5]. Tautan sisa (koneksi pintas) mempercepat konvergensi jaringan dalam, sehingga memberi ResNet kemampuan untuk menghindari masalah pengurangan gradien [5]. ResNet50 memiliki keunggulan yaitu memiliki kedalaman lapisan sedalam 50 jumlah lapisan ini lebih dalam dibandingkan dengan arsitektur alexnet, lenet dan VGG. Selain memiliki kedalaman yang lebih dalam ResNet50 memiliki keunggulan lain yaitu memiliki tingkat komputasi yang lebih rendah dibandingkan dengan VGG [8]. Dalam ResNet memiliki blok residual yang menjadi fundamental dari arsitektur ResNet yang berfungsi untuk mengurangi redudan yang ada pada saat melakukan training.

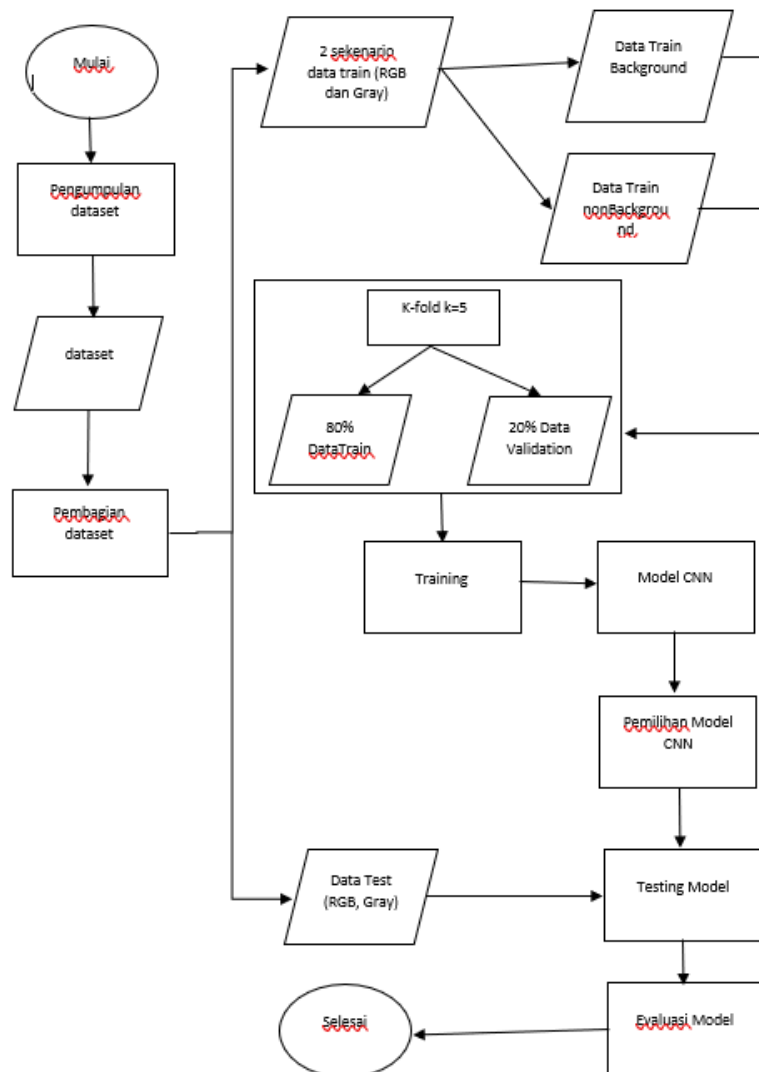


Gambar 3. Ilustrasi ResNet50 dan Residual Block [7]

### 3. Sistem yang Dibangun

#### 3.1 Deskripsi Umum

Gambaran sistem yang dibangun pada penelitian kali ini adalah mengklasifikasi tanaman anthurium berdasarkan citra daun dengan menggunakan metode CNN arsitektur ResNet50. Penelitian kali ini digunakan K-Fold K = 5 kemudian dropout 0,3 dan menggunakan learning rate 0,0001. Topik yang diangkat pada penelitian ini yaitu mengklasifikasikan 3 spesies tanaman anthurium. Adapun spesies yang diklasifikasikan yaitu anthurium hybrid, anthurium jemani sawi, anthurium veitchii dan anthurium gelombang cinta. Skenario yang digunakan pada penelitian ini yaitu gambar citra menggunakan background, gambar citra tidak menggunakan background dan dengan dataset dirubah menjadi hitam putih. Berikut gambaran dari sistem yang dibuat pada penelitian ini menggunakan flowchart :



Gambar 4. Deskripsi Umum

#### 3.2 Pengumpulan Dataset

Pada tahap pengumpulan dataset penulis mengumpulkan dataset berupa foto tanaman anthurium yang banyak dipasarkan dan tempat budidaya yang berlokasi di Lembang. Dalam penelitian ini

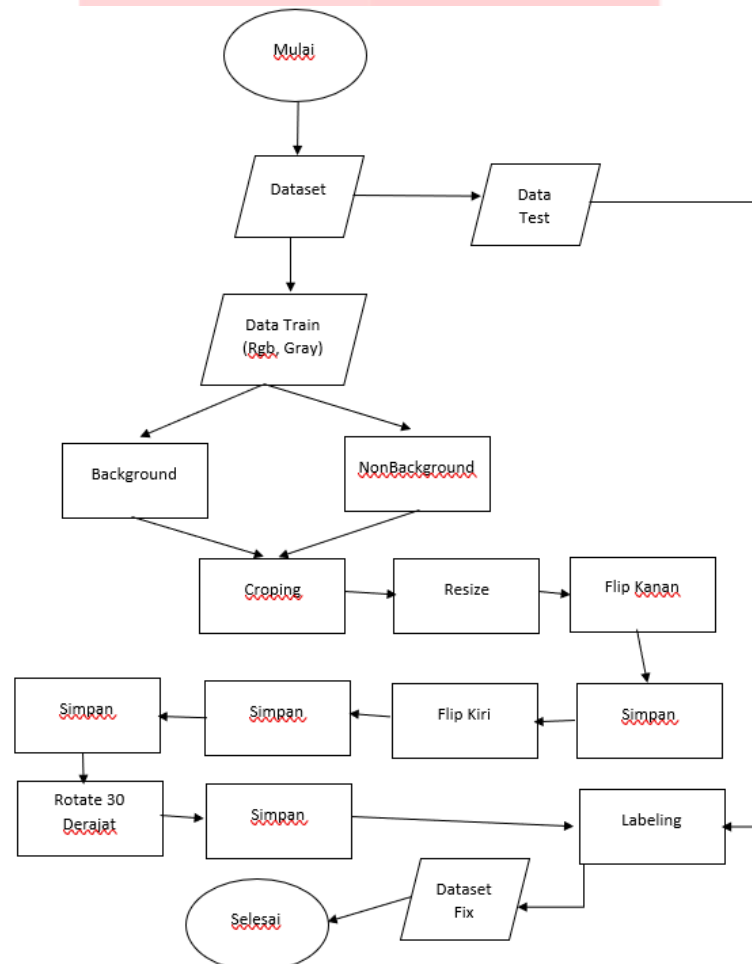
penulis berhasil mengumpulkan 3 spesies tanaman anthurium yaitu anthurium hybrid, anthurium jemani sawi, anthurium veitchii dan anthurium gelombang cinta yang berjumlah 300 foto per spesies. Dataset diambil dengan sudut pandang atas daun sehingga hanya muka daun saja yang terlihat dan dataset diambil dengan cara memotret satu persatu dari kanan ke kiri.

### 3.3 Pembagian Dataset

Pada tahap ini penulis melakukan pembagian dataset yang telah dikumpulkan sebanyak 300 foto per spesies menjadi 200 foto untuk train dan 100 foto untuk test.

### 3.4 Preprocessing dan Augmentasi

Pada tahap ini penulis melakukan proses augmentasi dataset. Data train dilakukan cropping agar pas posisi citra daun. Setelah dicrop data train diresize untuk mempersingkat penyimpanan. Setelah resize data train dilakukan augmentasi yang berjumlah 200 foto. Foto diaugmentasi dengan cara putar kekanan, putar kekiri dan putar 30 derajat sehingga menghasilkan 800 foto data train per spesies. Berikut flow dari preprocessing dan augmentasi :



**Gambar 5. Preprocessing dan Augmentasi**

#### 3.4.1 *Resize*

Tahap resize dilakukan agar gambar asli diubah ukuran pixelnya menjadi lebih kecil agar pada proses komputasi dalam pencarian dapat berjalan lebih cepat.

#### 3.4.2 Cropping

Tahap crop atau potong dilakukan bertujuan untuk menyeleksi bagian-bagian yang penting saja pada gambar asli agar pada saat proses klasifikasi hanya tertuju pada objek yang diinginkan.

#### 3.4.3 Flipping

Pada proses ini penulis melakukan operasi *flipping* yaitu suatu perubahan orientasi citra secara Horizontal dengan tujuan menambah variatif suatu data.

#### 3.4.4 Rotate

Proses rotasi, yaitu suatu operasi memutar suatu citra terhadap titik pusatnya. Hal ini dilakukan untuk menambah data agar semakin variatif sehingga model dapat mempelajari data baru dengan mudah.

#### 3.4.5 Flipping + Rotate

Setelah dicerminkan terhadap citra itu sendiri kemudian akan dilakukan rotasi terhadap arah jarum jam atau berlawanan jarum jam.

#### 3.4.6 Data Test / Data Lapangan

Sama halnya dengan data Train, data Testing lapangan perlu dilakukan pemilihan data dari data yang blur namun bedanya pada data Train dilakukan proses Augmentasi data hanya dengan melakukan operasi *Resize* saja, dan data Testing hanya berjumlah 84 dalam setiap spesiesnya. Hal ini dikarenakan data Testing adalah data yang akan diujikan kepada data Train nanti untuk menentukan seberapa akurat data yang telah dipelajari dengan data baru yang tidak dilakukan pelatihan (data Train).

### 3.5 K-Fold Cross Validation

Pada tahap ini data train setelah di preprocessing dan augmentasi selanjutnya data train dilakukan k-fold sebelum masuk kedalam training sistem yang dibangun. K-Fold yang digunakan dengan  $K = 5$  sehingga data train menjadi 5 model berbeda dengan 80% data train dan 20% data validation.

### 3.6 Hitam Putih

Dikarenakan penulis membuat dua model pada skenario pengujian yaitu model yang akan dilatih menggunakan dataset berwarna dan model yang dilatih menggunakan dataset hitam putih. Oleh karena itu, akan ada data yang dikonversi menjadi hitam putih untuk melakukan proses pelatihan model.

### 3.7 Training

Pada tahap ini sudah mulai dilakukan training kedalam sistem yang dibangun dengan metode *Convolutional Neural Network* (CNN) menggunakan arsitektur ResNet50. Training dilakukan sebanyak 10 Epoch nilai tertinggi yang dihasilkan akan diambil dari lima fold yang digunakan. Skenario yang digunakan dalam testing yaitu dengan skenario menggunakan background dan tanpa background.

### 3.8 Testing

Tahap selanjutnya yaitu testing dalam skenario yang digunakan yaitu skenario menggunakan background dan tanpa background dari hasil akurasi tertinggi yang disimpan akan dilakukan testing dengan data test yang sudah disiapkan dengan skenario yang sama ditambahkan dengan skenario testing menggunakan data dari internet dan akan menghasilkan output akurasi dari sistem yang dibuat. Dalam testing background bukan menjadi hal yang unik dikarenakan latar belakang

merupakan hal yang random dan selalu berbeda-beda setiap gambar sehingga hanya citra daun yang menjadi fitur penting dalam pendeteksian. Pada tugas akhir ini menggunakan background yang beragam pada datatest.

#### 4. Evaluasi

##### 4.1 Sekenario Pengujian

Ada beberapa pengujian yang dilakukan oleh penulis. Pertama, penulis melakukan berbagai percobaan nilai *learning rate* untuk mencari tau nilai *learning rate* terbaik. Kedua, penulis melakukan percobaan nilai *dropout* untuk mencari tau nilai *dropout* yang terbaik. Percobaan dilakukan menggunakan dua model yang dilatih menggunakan dataset berwarna (RGB) dan dataset hitam putih yang dimana masing-masing memiliki dataset menggunakan background dan tidak menggunakan background. Semua sekenario menggunakan epoch sebanyak 10 kali.

##### 4.1.1 Percobaan Learning Rate dengan Score Rata-Rata Akurasi Terhadap Testing

Tabel 1. Tabel Percobaan Learning Rate

Learning rate	RGB Background	RGB Non Background	HitamPutih Background	HitamPutih Non Background
0.00001	80.35%	78.45%	81,67%	70,89%
0.0001	83,46%	79,26%	84,46%	76,39%
0.003	82,20%	67,16%	81,86%	74,56%
0.03	54,46%	50,15%	76,65%	60,08%
0.1	38,59%	35,29%	37,06%	50,15%

*Learning rate* merupakan parameter penting yang berpengaruh terhadap kecepatan pelatihan dan performansi model. Oleh karena itu, penulis membuat empat percobaan nilai *learning rate* untuk menentukan nilai *learning rate* terbaik pada model RGB dan model *grayscale* yang masing-masing memiliki dataset dengan background dan tanpa background. Berikut hasil percobaan pada Tabel 1. diatas.

Pada Tabel 1, dari empat percobaan nilai *learning rate* yang telah dilakukan, nilai *learning rate* tertinggi akurasi adalah 0,0001 untuk model dengan dataset RGB maupun *grayscale* dengan nilai rata - rata akurasi diatas 79% untuk model dengan dataset RGB dan diatas 70% untuk model dengan dataset *grayscale* sedangkan nilai dengan akurasi terkecil yaitu *learning rate* 0,03 dan 0,1 dengan rata -rata akurasi 30% - 50% untuk model dengan dataset RGB maupun model dengan dataset *grayscale*.

##### 4.1.2 Percobaan Dropout dengan Score Rata-Rata Akurasi Terhadap Testing

Tabel 2. Tabel Percobaan Dropout

Dropout	RGB Background	RGB Non Background	HitamPutih Background	HitamPutih Non Background
0.1	84,33%	62,26%	80,57%	80,22%
0.2	84,83%	80,46%	85,23%	83,3%
0.3	80,86 %	62,59%	83,89%	81,56%
0.4	76,26%	62,99%	70,4%	68,5%
0.5	74,59%	60,86%	62,6%	59,7%

Dropout merupakan teknik yang dapat digunakan untuk mencegah terjadinya *overfitting* dan juga selain dapat mencegah terjadinya *overfitting*, *dropout* juga dapat meningkatkan akurasi testing yang dihasilkan. Oleh karena itu, penulis melakukan berbagai percobaan untuk melihat nilai dropout terbaik untuk model RGB dan model *Grayscale*. Hasil percobaan ada pada Tabel 2.

Pada Tabel 2, dari 5 percobaan nilai dropout yang dilakukan penulis, nilai dropout yang menghasilkan akurasi tertinggi adalah 0,2 untuk model dengan dataset RGB dan *grayscale* dengan akurasi rata – rata 80% sedangkan nilai dropout yang menghasilkan akurasi terkecil adalah 0,5 untuk dataset RGB dan *grayscale* dengan nilai rata – rata akurasi sebesar 60%.

Pada Tabel 2, menggunakan *dropout* mempunyai nilai akurasi yang lebih tinggi. Rata – rata akurasi testing tanpa menggunakan *dropout* adalah 83% sedangkan rata – rata akurasi testing tertinggi menggunakan dropout adalah 84,83%.

## 4.2 Hasil Pengujian

Pada tahap ini, penulis membuat model sesuai dengan rancangan sistem yang dibuat dengan arsitektur CNN yang ada pada Gambar 13 dengan K-Fold  $k = 5$ . Nilai  $k = 5$  diambil karena  $k$  yang sering digunakan pada penelitian adalah 5 dan 10 yang dimana jika semakin besar nilai  $k$ -nya maka komputasi yang dibutuhkan atau dikerjakan akan semakin berat oleh karena itu agar komputasi tidak terlalu berat penulis memilih nilai  $k = 5$ . Pengujian dilakukan penulis menggunakan hyperparameter sebagai berikut :

1. Epoch : 10
2. Learning Rate : 0,0001
3. Dropout : 0,2 untuk RGB dan 0,4 untuk Hitam Putih
4. Fungsi Aktivasi : ReLU
5. Ukuran Gambar : 224 x 224 x 3

### 4.2.1 Hasil Score Akurasi Pengujian

**Tabel 3. Tabel Skor Akurasi Testing**

Skenario	Training	Validasi	Testing
RGB, BG	89,51%	90,79%	83,13%
RGB, NBG	80,92%	84,91%	81,69%
Gray, BG	88,41%	90,87%	82,39%
Gray, NBG	81,51%	83,91%	64,6%

Seperti yang terlihat pada Tabel 3, akurasi training, validasi dan testing model RGB dengan background berada di rentang 83% – 95%. Sedangkan akurasi training, validasi dan testing model *grayscale* 64% – 83%.

Akurasi testing dengan 4 sekenario pada table 3 memiliki nilai yang cukup tinggi dikarenakan dataset yang diambil dalam keadaan yang hampir sama dan menggunakan augmentasi yang berfariatif sehingga mendapatkan hasil yang baik.

**4.2.2 Hasil Score Loss Pengujian**

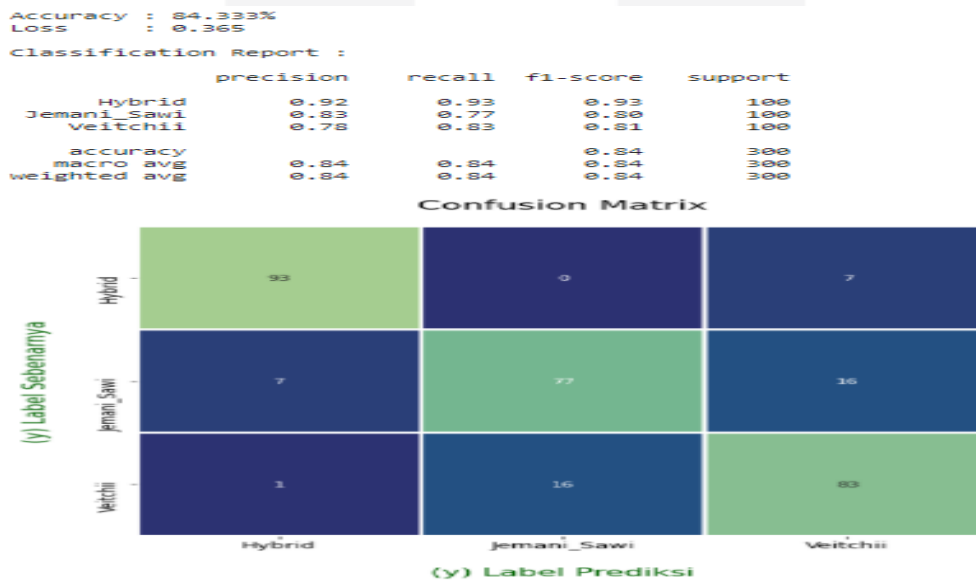
**Tabel 4. Tabel Loss Testing**

Loss	Training	Validasi	Testing
RGB, BG	0,27	0,22	0,39
RGB, NBG	0,43	0,39	0,56
Gray, BG	0,30	0,26	0,41
Gray, NBG	0,43	0,39	0,87

Seperti yang terlihat pada Tabel 4, model RGB dengan background mempunyai nilai loss terendah yaitu 0,39 untuk loss tertinggi terdapat pada sekenario gray tanpa background yaitu 0,87. Model *grayscale* dengan background mempunyai nilai loss lebih rendah dibandingkan dengan gray tanpa background yaitu 0,41.

Model Rgb dengan background juga mempunyai grafik nilai loss yang lebih kecil dibandingkan dengan model *grayscale* dengan background yang menandakan bahwa model RGB lebih baik dibandingkan dengan model *grayscale* dikarenakan semakin kecil nilai loss maka model akan semakin baik.

**4.2.3 Hasil Testing Model Terbaik RGB Background**

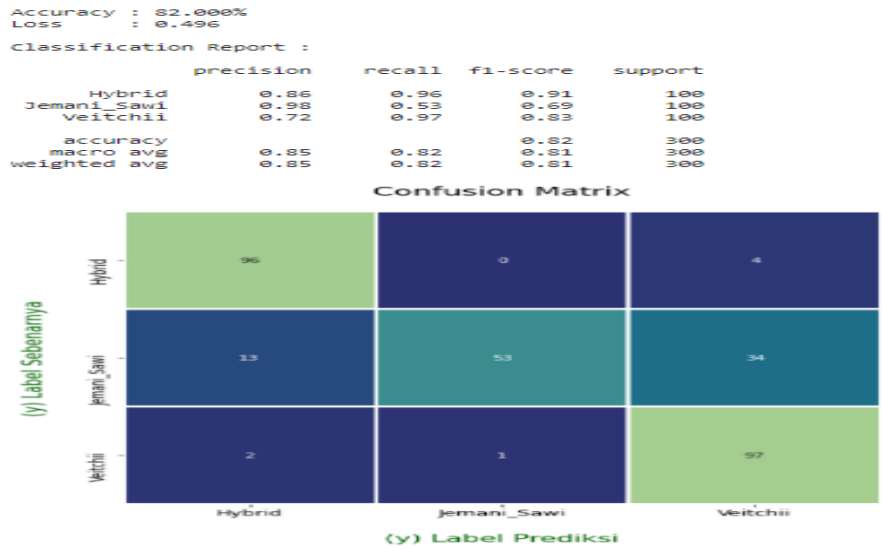


**Gambar 6. Model Terbaik RGB Background**

Berdasarkan *classification report* yang ada di Gambar 6, model memiliki akurasi sebesar 84,33% dengan nilai akurasi tertinggi didapat dari spesies *veitchii* yaitu sebesar 88% dan nilai akurasi terendah didapat dari spesies *jemani sawi* yaitu sebesar 77%.

Berdasarkan *confusion matrix* yang ada pada Gambar 6, model rgb dengan background salah mengklasifikasikan data sebanyak 7 data dari 100 gambar untuk spesies anthurium hybrid, 26 data dari 100 gambar untuk spesies jemani sawi dan 19 data dari 100 gambar untuk spesies veitchii.

#### 4.2.4 Hasil Testing Model Terbaik RGB Non Background

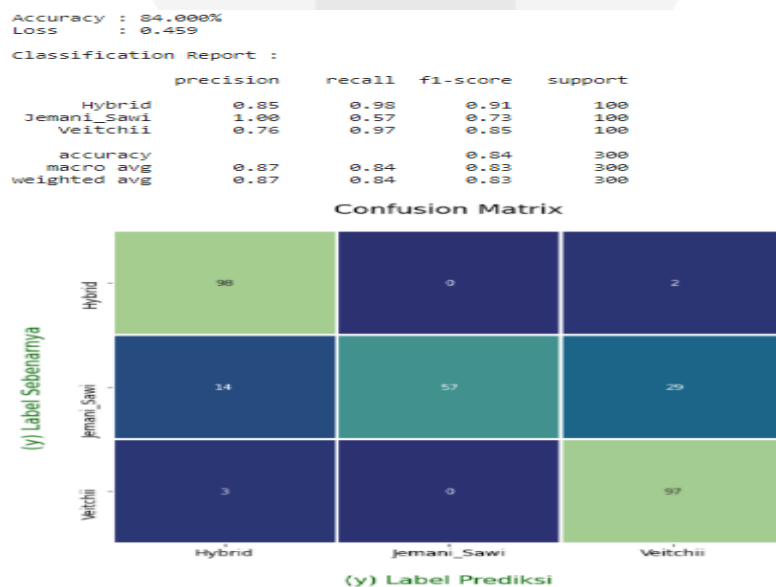


**Gambar 7. Model Terbaik RGB Tanpa Background**

Berdasarkan *classification report* yang ada di Gambar 7, model memiliki akurasi sebesar 82% dengan nilai akurasi tertinggi didapat dari spesies veitchii yaitu sebesar 97% dan nilai akurasi terendah didapat dari spesies jemani sawi yaitu sebesar 53%.

Berdasarkan *confusion matrix* yang ada pada Gambar 7, model rgb tanpa background salah mengklasifikasikan data sebanyak 4 data dari 100 gambar untuk spesies anthurium hybrid, 53 data dari 100 gambar untuk spesies jemani sawi dan 3 data dari 100 gambar untuk spesies veitchii.

#### 4.2.5 Hasil Testing Model Terbaik Gray Background



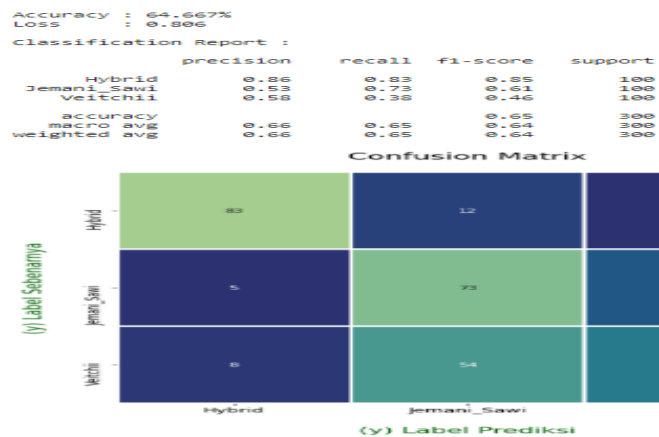
**Gambar 8. Model Terbaik Gray Background**



Berdasarkan *classification report* yang ada di Gambar 8, model memiliki akurasi sebesar 84% dengan nilai akurasi tertinggi didapat dari spesies *veitchii* yaitu sebesar 98% dan nilai akurasi terendah didapat dari spesies *jemani sawi* yaitu sebesar 57%.

Berdasarkan *confussion matrix* yang ada pada Gambar 8, model *grayscale* dengan background salah mengklasifikasikan data sebanyak 2 data dari 100 gambar untuk spesies *anthurium Hybrid*, 43 data dari 100 gambar untuk spesies *jemani sawi* dan 3 data dari 100 gambar untuk spesies *veitchii*.

#### 4.2.6 Hasil Testing Model Terbaik Gray Non Background



Gambar 9. Model Terbaik Gray Tanpa Background

Berdasarkan *classification report* yang ada di Gambar 9, model memiliki akurasi sebesar 64,66% dengan nilai akurasi tertinggi didapat dari spesies Hybrid yaitu sebesar 83,00% dan akurasi terendah yaitu spesies *veitchii* yaitu sebesar 54%.

Berdasarkan *confussion matrix* yang ada pada Gambar 9, model *grayscale* salah mengklasifikasikan data sebanyak 17 data dari 100 gambar untuk spesies hybrid, 27 data dari 100 gambar untuk spesies *jemani sawi* dan 62 data dari 100 gambar untuk spesies *veitchii*. Dalam pengujian penulis menemukan bahwa mengapa akurasinya sangat rendah dikarenakan ada kemiripan antara *jemani sawi* dengan *veitchii* sehingga membuat akurasi rendah.

#### 4.3 Pengaruh Warna pada Dataset

Dari hasil pengujian yang dihasilkan penulis, model yang dilatih menggunakan dataset RGB mempunyai rata – rata akurasi testing yang lebih besar dibandingkan dengan model yang dilatih dengan dataset *grayscale* dengan akurasi sebesar 84,33% untuk model dengan dataset RGB dengan background dan 64,6% untuk model dengan dataset *grayscale* yang membuktikan bahwa melatih model menggunakan dataset berwarna dapat menambah keakuratan model dalam mengklasifikasi objek.

#### 4.4 Pengaruh Background pada Dataset

Dari hasil pengujian yang dihasilkan penulis, model yang dilatih menggunakan dataset RGB dengan background mempunyai rata – rata akurasi testing yang lebih besar dibandingkan dengan model yang dilatih dengan dataset *grayscale* dengan akurasi sebesar 84,33% untuk model dengan dataset RGB dengan background dan 82% untuk model dengan dataset *grayscale* dengan background yang membuktikan bahwa melatih model menggunakan dataset berwarna dapat menambah keakuratan model dalam mengklasifikasi objek.

## 5. Kesimpulan

Dari penelitian yang sudah dilakukan ditarik kesimpulan bahwa CNN terbukti mampu mengklasifikasikan dengan baik untuk studi kasus spesies tanaman anthurium dengan akurasi yang cukup besar yaitu 84,33% untuk 300 data testing dan membuktikan bahwa melatih model dataset RGB background menghasilkan akurasi yang lebih baik dibandingkan dengan melatih model dengan dataset *grayscale* tanpa background, *grayscale* dengan background dan berwarna tanpa background.

Saran untuk penelitian selanjutnya, menambah jumlah dataset sehingga dapat mengklasifikasikan jenis-jenis lainnya, mencari tau hyperparameter yang lebih baik sehingga dapat menghasilkan performansi model yang lebih baik dan juga harapannya model yang telah dibuat dapat diimplementasikan pada sistem berbasis android ataupun website sehingga dapat digunakan secara umum.



## REFERENSI

- [1] A. Soleimanipour, G. R. Chegini, and J. Massah, "Classification of anthurium flowers using combination of PCA, LDA and support vector machine," *Agric. Eng. Int. CIGR J.*, vol. 20, no. 1, pp. 219–228, 2018.
- [2] M. Toğaçar, B. Ergen, and Z. Cömert, "Classification of flower species by using features extracted from the intersection of feature selection methods in convolutional neural network models," *Meas. J. Int. Meas. Confed.*, vol. 158, 2020, doi: 10.1016/j.measurement.2020.107703.
- [3] S. M. Omer, R. M. Hasan, and B. N. Anwer, "An Image Dataset Construction for Flower Recognition Using Convolutional Neural Network," *Sci. J. Univ. Zakho*, vol. 8, no. 3, pp. 112–117, 2020, doi: 10.25271/sjuoz.2020.8.3.747.
- [4] M. V.D. Prasad *et al.*, "An efficient classification of flower images with convolutional neural networks," *Int. J. Eng. Technol.*, vol. 7, no. 1.1, pp. 384–391, 2018, doi: 10.14419/ijet.v7i1.1.9857.
- [5] B. R. Mete and T. Ensari, "Flower Classification with Deep CNN and Machine Learning Algorithms," *3rd Int. Symp. Multidiscip. Stud. Innov. Technol. ISMSIT 2019 - Proc.*, pp. 6–10, 2019, doi: 10.1109/ISMSIT.2019.8932908.
- [6] T. Thanh *et al.*, "Flower species identification using deep convolutional neural networks," *Reg. Conf. Comput. Inf. Eng. 2016 (RCCIE 2016)*, no. October, p. 6, 2016.
- [7] N. B. Linsangan and R. S. Pangantihon, "FPGA-based plant identification through leaf veins," *ACM Int. Conf. Proceeding Ser.*, pp. 100–104, 2018, doi: 10.1145/3301879.3301905.
- [8] S. Targ, D. Almeida, and K. Lyman, "Resnet in Resnet: Generalizing Residual Architectures," pp. 1–7, 2016, [Online]. Available: <http://arxiv.org/abs/1603.08029>.
- [9] E. Cengil and A. Cinar, "Multiple classification of flower images using transfer learning," *2019 Int. Conf. Artif. Intell. Data Process. Symp. IDAP 2019*, 2019, doi: 10.1109/IDAP.2019.8875953.
- [10] Y. Sun, L. Zhu, G. Wang, and F. Zhao, "Multi-Input Convolutional Neural Network for Flower Grading," *J. Electr. Comput. Eng.*, vol. 2017, 2017, doi: 10.1155/2017/9240407.
- [11] E. López-Jiménez, J. I. Vasquez-Gomez, M. A. Sanchez-Acevedo, J. C. Herrera-Lozada, and A. V. Uriarte-Arcia, "Columnar cactus recognition in aerial images using a deep learning approach," *Ecol. Inform.*, vol. 52, pp. 131–138, 2019, doi: 10.1016/j.ecoinf.2019.05.005.
- [12] A. Jahanbakhshi, M. Momeny, M. Mahmoudi, and Y. D. Zhang, "Classification of sour lemons based on apparent defects using stochastic pooling mechanism in deep convolutional neural networks," *Sci. Hortic. (Amsterdam)*, vol. 263, no. July 2019, p. 109133, 2020, doi: 10.1016/j.scienta.2019.109133.
- [13] I. Patel and S. Patel, "learning for flower species detection and classification," no. January, 2020.