

Klasifikasi Genus Tanaman Sukulen Menggunakan Convolutional Neural Network

Hamad Fauzi Jessar¹, Agung Toto Wibowo², Ema Rachmawati³

^{1,2,3} Fakultas Informatika, Universitas Telkom, Bandung

¹hamadfauzi@students.telkomuniversity.ac.id, ²agungtoto@telkomuniversity.ac.id,

³emarachmawati@telkomuniversity.ac.id

Abstrak

Tanaman sukulen merupakan jenis tanaman hias yang banyak ditemukan jenisnya di Indonesia. Tanaman sukulen mempunyai banyak jenis genus yang dimana setiap genus mempunyai ciri dan karakteristik yang beragam sehingga sulit untuk mengidentifikasi jenis genus pada tanaman sukulen. Oleh karena itu, penulis membuat sebuah sistem yang dapat mengenali jenis genus tanaman sukulen melalui gambar menggunakan metode *Convolutional Neural Network* (CNN). CNN merupakan salah satu teknik deep learning yang dapat digunakan untuk mengenali objek dua dimensi seperti gambar dan video. CNN memiliki banyak jenis arsitektur jaringan, arsitektur jaringan CNN yang digunakan penulis untuk membangun sistem ini adalah custom arsitektur dan penulis juga menggunakan *k fold cross validation* yang bertujuan untuk memastikan keakuratan akurasi yang dihasilkan oleh model sistem. Penelitian dilakukan penulis dengan membandingkan antara model yang dilatih menggunakan dataset berwarna (RGB) dan model yang dilatih menggunakan dataset *grayscale*. Dari hasil penelitian didapatkan bahwa model yang dilatih menggunakan dataset berwarna mempunyai akurasi testing yang lebih tinggi dibandingkan dengan model yang dilatih menggunakan dataset *grayscale*. Akurasi testing yang dihasilkan model yang dilatih dengan dataset berwarna sebesar 93% sedangkan model yang dilatih menggunakan dataset *grayscale* sebesar 64%.

Kata kunci : convolutional neural network, deep learning, k fold, grayscale, RGB

Abstract

Succulent plants are a type of ornamental plant that are found in many species in Indonesia. Succulent plants have many types of genera, where each genus has various characteristics and characteristics making it difficult to identify the type of genus in succulent plants. Therefore, the authors created a system that can recognize the types of succulent plant genera through images using the Convolutional Neural Network (CNN) method.). CNN is a deep learning technique that can be used to recognize two-dimensional objects such as images and videos. CNN has many types of network architectures, the CNN network architecture used by the author to build this system is a custom architecture and the author also uses k fold cross validation which aims to ensure the accuracy of the accuracy generated by the system model. The research was conducted by the author by comparing the model trained using the color dataset (RGB) and the model trained using the grayscale dataset. From the results of the study, it was found that models trained using color datasets have higher accuracy than models trained using grayscale datasets, namely 93% for models with color datasets while models with grayscale datasets have 64% accuracy.

Keywords: convolutional neural network, deep learning, k fold cross validation, RGB, grayscale

1. Pendahuluan

Latar Belakang

Tanaman sukulen merupakan jenis tanaman yang mempunyai kemampuan menyerap dan menyimpan air pada batang utamanya. Sukulen banyak ditemukan di daerah yang tandus seperti Afrika, Amerika Selatan, dan tempat - tempat lain di dunia. Di Indonesia tanaman sukulen merupakan tanaman yang biasa dijadikan sebagai tanaman hias karena memiliki bentuk dan warna yang indah.

Tanaman sukulen mempunyai banyak jenis genus, yang dimana setiap genus mempunyai ciri dan karakteristik yang beragam, contohnya genus *echeveria* yang memiliki bentuk seperti bunga dan memiliki tekstur

yang mirip dengan lilin, genus *sedum* yang memiliki bentuk lonjong dan penuh warna, dan lain – lain. Selain mempunyai ciri dan karakteristik yang beragam, umumnya tanaman sukulen lebih suka mendapatkan matahari langsung secara penuh. Namun, ada beberapa jenis genus yang lebih baik ditanam di dalam ruangan dan tidak terkena sinar matahari secara penuh, contohnya genus *haworthia*. Oleh karena itu bagi orang awam yang ingin membeli tanaman sukulen perlu mengetahui jenis genus tanaman sukulen yang dibeli.

Oleh karena itu, penulis mencoba membuat sebuah model sistem klasifikasi menggunakan metode *convolutional neural network* yang dapat mengklasifikasikan jenis genus pada tanaman sukulen melalui gambar tanaman sukulen yang nantinya penulis berharap bahwa model klasifikasi yang penulis buat dapat digunakan untuk kepentingan pembelajaran bagi orang yang ingin mengenal tanaman sukulen.

Sistem yang penulis buat juga dimaksudkan untuk menguji metode *convolutional neural network* pada studi kasus tanaman sukulen

Convolutional Neural Network (CNN) merupakan salah satu metode *deep learning* yang mampu belajar secara mandiri untuk melakukan pengenalan objek 2D seperti gambar dan video[1][2]. CNN terbukti mampu menghasilkan akurasi yang cukup besar untuk klasifikasi menggunakan gambar untuk studi kasus tanaman[15][16][17]. CNN mempunyai banyak arsitektur yang bisa digunakan seperti *alexNet*, *VGGNet*, *leNet*, dan lain - lain[3][4]. Pada penelitian ini penulis menggunakan custom arsitektur sehingga dapat lebih bebas dalam melakukan percobaan dan beban komputasi yang dibutuhkan dapat disesuaikan seefektif mungkin.

Topik dan Batasannya

Topik yang dibahas penulis pada tulisan ini yaitu bagaimana mengimplementasikan metode *convolutional neural network* agar dapat mengklasifikasikan jenis – jenis genus tanaman sukulen dan membahas bagaimana mengukur performansi model yang dihasilkan.

batasan masalah yang ditetapkan pada tugas akhir ini yaitu: Pertama, sistem yang akan dibuat hanya dapat mengklasifikasikan lima jenis genus tanaman sukulen yaitu : *Cotyledon*, *Echeveria*, *Hawortia*, *Sedum*, dan *Senecio*. Kedua, gambar yang dapat diklasifikasi harus memiliki resolusi 1 : 1. Ketiga, ada dua sisi angle pengambilan gambar yaitu dari atas dan samping tanaman.

Tujuan

Tujuan yang dicapai dalam tugas akhir ini ialah membuat sebuah model sistem yang dapat mengklasifikasikan genus tanaman sukulen menggunakan *convolutional neural network* dan dapat mengevaluasi performansi model sistem yang dibuat.

2. Studi Terkait

2.1. Penelitian Terdahulu

Sejauh yang penulis ketahui, penulis menemukan beberapa penelitian terkait klasifikasi tanaman menggunakan *Convolutional Neural Network*. Penelitian tersebut dilakukan pada tanaman dengan citra resolusi tinggi, daun herbal dan tanaman toga.

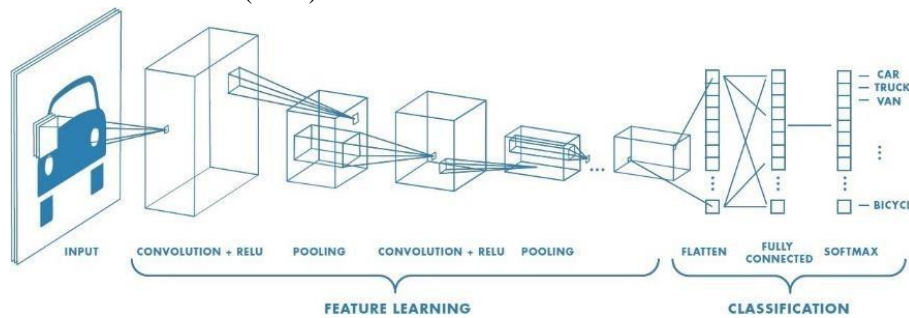
Penelitian tanaman berdasarkan citra resolusi tinggi menghasilkan akurasi validasi sebesar 93% dan akurasi testing sebesar 85%. Penelitian dilakukan terhadap lima kelas jenis tanaman yaitu tanaman padi, bawang merah, kelapa, pisang, dan cabai. Total citra yang digunakan untuk membangun sistem adalah sebanyak 450 data, dengan 350 data digunakan untuk pembelajaran dan 100 data digunakan untuk validasi[15].

Penelitian selanjutnya pada daun herbal menghasilkan akurasi 93,62 untuk akurasi testing secara *offline* dan akurasi 91,04% untuk testing secara *online*. Penelitian dilakukan terhadap tujuh jenis tanaman herbal yaitu binahong, jambu, kunyit, laos, mengkudu, pegagan, dan salam dengan jumlah data sebanyak 3150 data[16].

Penelitian berikutnya pada dilakukan terhadap 10 jenis tanaman toga yaitu teh hijau, tapak dewa, sirsak, semanggi, mengkudu, mahoni, kumis kucing, jambu biji, blimbing wuluh, bayam merah. Penelitian ini menghasilkan akurasi sebesar 75% dengan 300 data pelatihan dan menghasilkan akurasi sebesar 80% dengan data uji. Penelitian menggunakan 1600 data dengan jumlah data training sebanyak 1200, data validasi sebanyak 300 dan 100 data testing[17].

Namun hingga saat ini penulis belum menemukan penelitian terkait klasifikasi tanaman sukulen. Oleh karena itu, penelitian ini akan membahas tentang klasifikasi genus pada tanaman sukulen. Sub bab selanjutnya akan membahas teori – teori yang digunakan penulis pada penelitian kali ini.

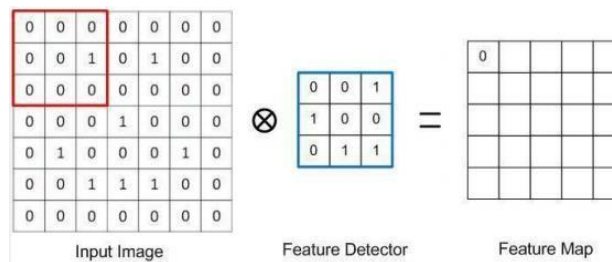
2.2. Convolutional Neural Network (CNN)



Gambar 1 CNN

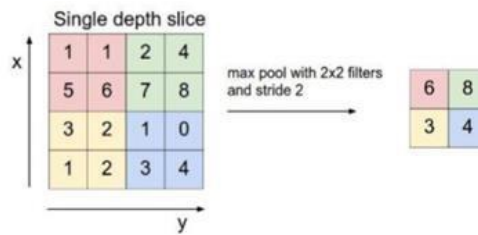
CNN merupakan salah satu metode deep learning yang digunakan untuk mengklasifikasi objek dua dimensi (2D) seperti gambar dan video[2]. Deep learning merupakan metode yang dapat membuat komputer belajar secara mandiri layaknya manusia[6]. Struktur CNN dengan terdiri dari input, proses ekstraksi fitur, proses klasifikasi dan output. Proses ekstraksi dalam CNN terdiri dari beberapa lapisan tersembunyi atau hidden layer, yaitu lapisan konvolusi, fungsi aktivasi (ReLU), pooling, dan flattening[7]. CNN bekerja secara hierarki, sehingga output pada lapisan konvolusi pertama digunakan sebagai input pada lapisan konvolusi selanjutnya. Pada proses klasifikasi terdiri dari fully connected dan fungsi aktivasi softmax yang outputnya berupa hasil klasifikasi.

2.2.1 Konvolusi



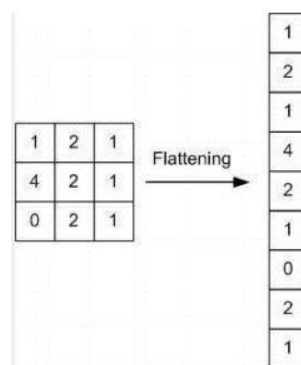
Gambar 2 Konvolusi

Lapisan konvolusi bertujuan untuk mengekstraksi fitur menggunakan kernel atau bisa disebut juga dengan *feature detector* yang ada pada citra input. Kernel ini berisi bobot yang digunakan untuk mendeteksi fitur dari objek seperti tepi, kurva, atau warna. Hasil dari konvolusi disebut *feature map*. Kernel diaplikasikan secara berulang sehingga menghasilkan serangkaian *feature map*. Kumpulan dari *feature map* dinamakan *convolution layer*. Pada tahap konvolusi terdapat parameter yang dapat diubah untuk memodifikasi sifat tiap lapisan, yaitu ukuran kernel, padding, dan stride. Padding adalah penambahan ukuran piksel dengan nilai tertentu disekitar data input agar hasil dari *feature map* tidak terlalu kecil sehingga tidak banyak informasi yang hilang, Nilai padding biasanya nol sehingga disebut dengan zero padding. Stride mengontrol bagaimana filter diterapkan pada data input dengan bergerak sepanjang ukuran piksel yang telah ditentukan[2]. Output dari proses konvolusi ini dijadikan sebagai input untuk lapisan konvolusi selanjutnya. 2.2.1 Pooling



Gambar 3 Ilustrasi Max Pooling

Pooling adalah proses untuk mencari fitur yang sudah didapat pada tahap konvolusi yang bertujuan agar sistem tetap dapat mendeteksi gambar walaupun terjadi perubahan pada gambar seperti posisi dan lain - lain. Terdapat dua macam pooling yang biasa digunakan yaitu max pooling dan average pooling[8]. Nilai yang diambil pada average pooling adalah nilai rata - rata sedangkan pada max pooling adalah nilai tertinggi. 2.2.1 Flattening



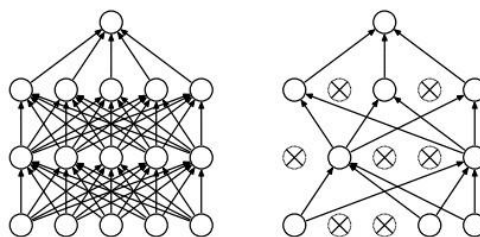
Gambar 4 Ilustrasi Flattening

Flattening merupakan proses mengubah matriks yang dihasilkan pooling layer menjadi satu kolom (vektor tunggal). Vektor tunggal yang dihasilkan akan dijadikan sebagai input pada proses fully connected.

2.2.1 Fully Connected

Lapisan fully connected layer merupakan kumpulan dari proses konvolusi gambar. lapisan ini mendapat input dari lapisan sebelumnya untuk mencari tau kelas klasifikasi mana yang paling cocok untuk input tersebut. 2.3.

Dropout



Gambar 5 Ilustrasi Dropout

Dropout merupakan teknik yang dapat digunakan untuk mencegah *overfitting*[14] dengan cara menghilangkan beberapa data pada jaringan saraf (*neural network*) yang digunakan pada saat proses training model.

2.4. Genus

Genus merupakan salah satu penamaan pada pengelompokkan makhluk hidup dengan tingkat hierarki di atas spesies dan dibawah famili. Pada penamaan sebuah makhluk hidup genus mewakili kata pertama sedangkan

kata kedua merujuk kepada jenis spesies seperti contohnya *Aloe vera* yang berarti berasal dari genus *Aloe*, *Homo sapiens* yang berasal dari genus *Homo* dan sebagainya.

Pada penelitian tugas akhir ini, ada lima genus tanaman sukulen yang akan penulis gunakan yaitu : 1) genus *cotyledon* yang memiliki ciri berwarna hijau, bercabang – cabang, memiliki bulu pada daun dan batang dan daunnya memiliki gigi coklat keunguan; 2) genus *echeveria* yang memiliki ciri mempunyai daun bundar tebal berwarna putih dengan sedikit merah muda pada bagian ujung daun, dilapisi dengan lapisan serbuk 3) genus *haworthia* yang memiliki ciri daun berbentuk segitiga hijau dengan tonjolan – tonjolan putih di bagian bawah; 4) genus *sedum* yang memiliki ciri daun berwarna hijau terang yang berdaging dan elastis; 5) genus *senecio* dengan ciri daun – daun kecil berbentuk kacang dan berwarna hijau.

2.5. Learning Rate

Learning rate adalah salah satu hyperparameter yang penting untuk disesuaikan karena sangat berpengaruh terhadap pelatihan model. Nilai learning rate yang terlalu besar dapat menyebabkan pelatihan model terhenti di titik minimum sehingga akan mengakibatkan model mempunyai akurasi yang rendah. Menurunkan nilai learning rate dapat meningkatkan akurasi model[18]. Namun, akan ada titik dimana nilai *learning rate* tidak akan meningkatkan akurasi serta tidak membuat model lebih baik[19].

2.6. Fungsi Aktivasi

Fungsi aktivasi adalah fungsi yang digunakan dalam jaringan saraf untuk menghitung nilai w (*weighted*) berdasarkan jumlah input dan bias[9]. Fungsi aktivasi digunakan untuk memutuskan apakah sebuah neuron dapat lanjut atau tidak. Fungsi aktivasi akan memanipulasi data yang disajikan melalui beberapa pemrosesan gradien biasanya gradient descent dan setelahnya menghasilkan output untuk jaringan saraf, yang berisi parameter dalam data. Ada dua fungsi aktivasi yang digunakan untuk membangun sistem ini yaitu fungsi aktivasi *Rectrified Linear Unit* (Relu) dan fungsi aktivasi softmax.

2.6.1 Fungsi Aktivasi Rectrified Linear Unit (Relu)

Fungsi aktivasi Rectified Linear Unit (ReLU) diusulkan oleh Nair dan Hinton 2010, dan sejak itu, telah menjadi yang paling banyak digunakan untuk aplikasi *deep learning* hingga saat ini[10][11]. Relu telah terbukti menjadi fungsi yang paling sukses dan banyak digunakan. Ini menawarkan kinerja yang lebih baik dalam *deep learning* dibandingkan dengan fungsi aktivasi Sigmoid dan tanh[6]. Relu berfungsi untuk mengubah nilai element yang negatif menjadi 0 sehingga tidak ada yang negatif.

2.6.2 Fungsi Softmax.

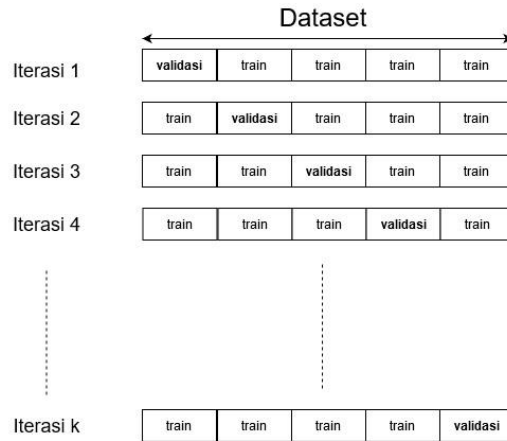
Fungsi Softmax adalah jenis lain dari fungsi aktivasi yang digunakan dalam komputasi saraf. Ini digunakan untuk menghitung probabilitas distribusi dari vektor bilangan real. Fungsi Softmax menghasilkan output yang merupakan kisaran nilai antara 0 dan 1, dengan jumlah probabilitas sama dengan 1 [4].

Fungsi Softmax digunakan dalam model multi-kelas di mana ia mengembalikan probabilitas setiap kelas, dengan kelas target memiliki probabilitas tertinggi. Fungsi Softmax sebagian besar muncul di hampir semua lapisan output dari arsitektur pembelajaran CNN seperti arsitektur *alexnet*, *VGGNet*, *ZFNet* dan lain - lain [4].

2.7. Cross Validation

Cross validation merupakan metode statistik yang digunakan untuk mengevaluasi dan membandingkan algoritma pembelajaran dengan cara membagi data menjadi dua jenis yaitu data training yang digunakan untuk melatih model pembelajaran dan data validasi yang digunakan untuk memvalidasi model pembelajaran. Data train dan data validasi akan disilangkan secara berturut - turut agar setiap data dapat divalidasi dan salah satu jenis cross validation adalah k - fold cross validation [12].

2.7.1 K – Fold Cross Validation

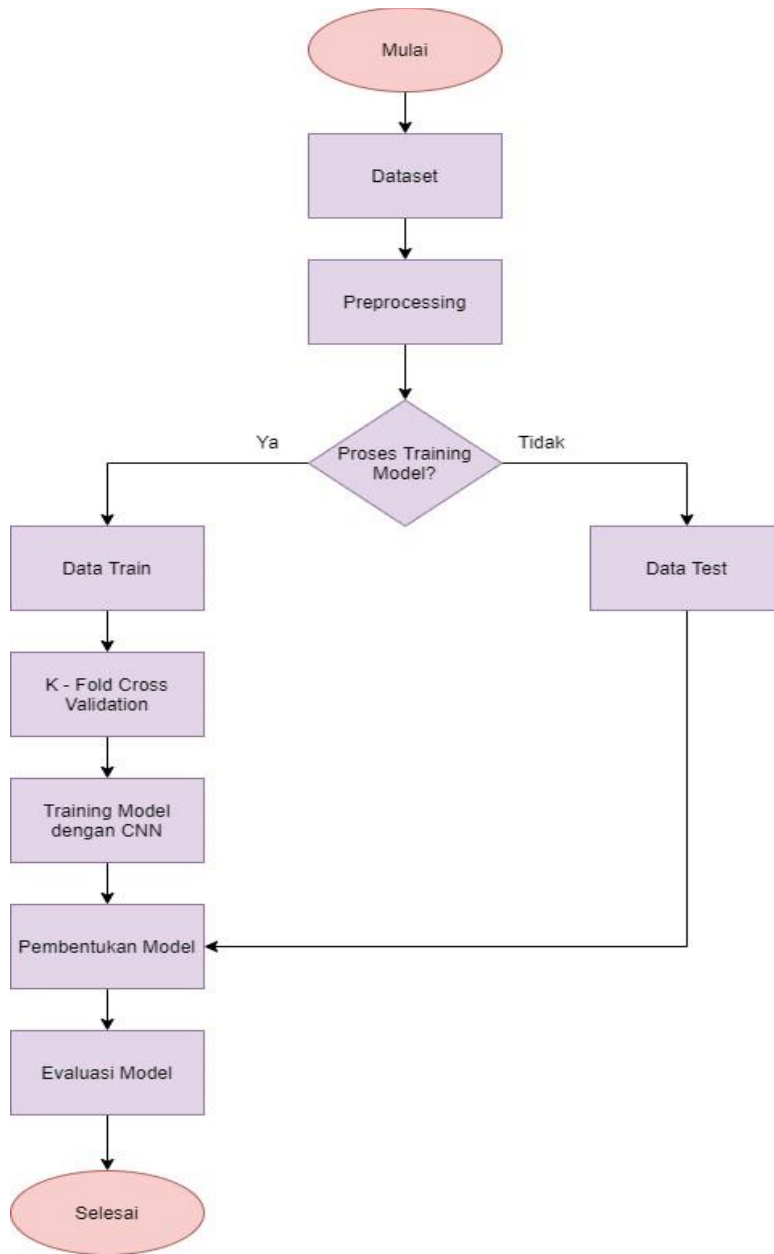


Gambar 6 Ilustrasi K - Fold Cross Validation

Seperti yang bisa dilihat pada Gambar 6, k fold cross validation merupakan metode cross validation dengan cara mempartisi dataset menjadi data validasi dan data train untuk proses pembelajaran dan akan dipartisi ulang pada setiap iterasi dengan jumlah data validasi yang sama namun berbeda data sehingga data validasi pada iterasi sebelumnya akan dijadikan data train dan ada data train pada iterasi sebelumnya yang diubah menjadi data validasi. Hal ini bertujuan agar setiap data yang ada pada dataset dapat di validasi secara menyeluruh sehingga akurasi yang dihasilkan pada saat pembelajaran model akan semakin akurat

3. Sistem yang Dibangun

Pada bagian ini, penulis akan menjelaskan sistem yang dibangun menggunakan flowchart yang dimulai dari menjelaskan dataset yang penulis gunakan, setelah itu melakukan preprocessing dataset, membagi dataset menjadi dua jenis data yaitu data train dan data test, melakukan pelatihan model, menguji model dan menghitung performansi model yang dibuat penulis. Untuk lebih jelasnya lihat pada Gambar 7.








Gambar 7 FlowChart Alur Pembuatan Sistem

3.1. Dataset

Dataset diambil dari Rumah Bunga Rizal Lembang yang terdiri dari 2500 foto tanaman sukulen yang dikumpulkan penulis melalui proses akuisisi citra. Dataset terdiri dari lima jenis genus tanaman sukulen yaitu sebagai berikut :

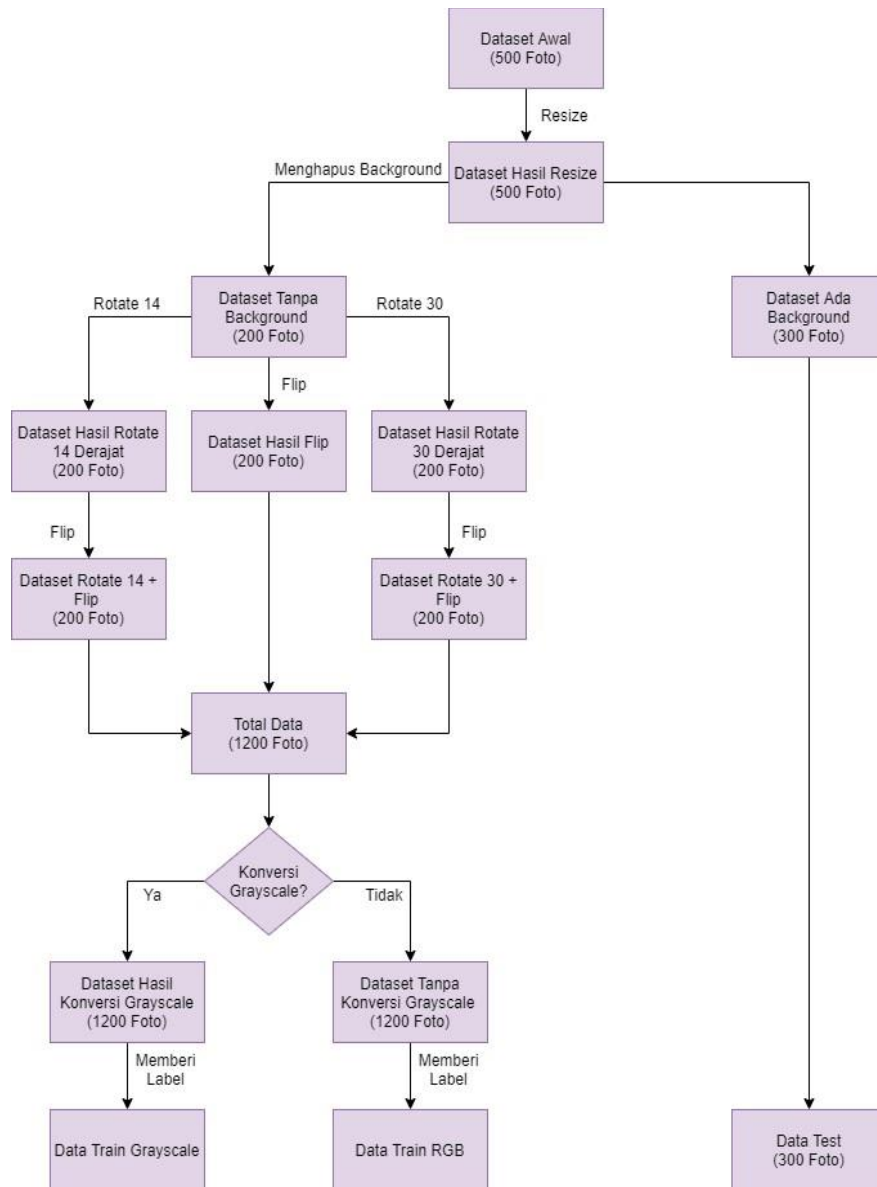
Table 1 Tabel Dataset yang digunakan

Nama Genus	Jumlah Data	Contoh Data
Cotyledon	500	

Echeveria	500	
Hawortia	500	
Sedum	500	
Senecio	500	

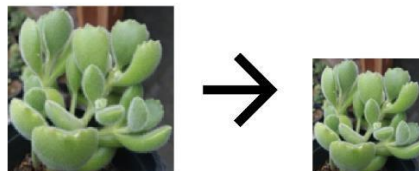
3.2. Preprocessing

Dataset yang terdiri dari 1500 foto dengan komposisi tiap genus sebanyak 500 foto akan dilakukan proses *preprocessing* sehingga diakhir *preprocessing* akan menghasilkan total data sebanyak 7500 foto dengan masing – masing genus sebanyak 1500 foto. Hal ini dilakukan agar jumlah data yang digunakan untuk pelatihan model akan semakin banyak dan bervariasi sehingga hasil performansi model yang didapat juga akan semakin baik. Detail proses *preprocessing* yang dilakukan penulis dapat dilihat pada Gambar 8.



Gambar 8 Pre-processing dataset tiap genus

3.2.1 *Resize*



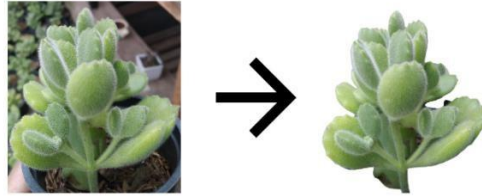
Gambar 9 Ilustrasi Resize

Resize bertujuan untuk menyamakan ukuran gambar yang ada pada dataset sebelum semua data yang ada pada dataset dijadikan sebagai input training model agar dapat memastikan bahwa output model adalah vektor $1 \times 1 \times N[14]$. Ukuran semua gambar akan diubah menjadi $128 \times 128 \times 3$.

3.2.2 Membagi Menjadi Data Train dan Data Test

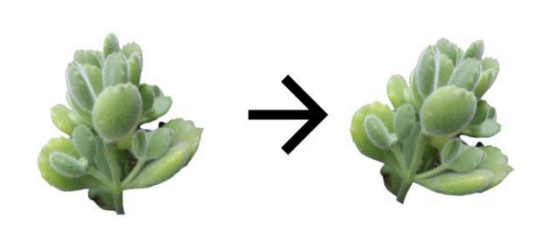
Setelah ukuran semua gambar yang ada pada dataset diubah menjadi 128x128x3, dataset akan dibagi menjadi dua jenis data yaitu data train dan data test dengan pembagian data 2 : 3 sehingga penulis akan mendapatkan 300 data test dan 200 data train.

3.2.3 Menghapus Background



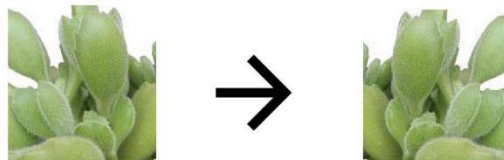
Gambar 10 Ilustrasi Menghapus Background

Setelah data dibagi menjadi dua jenis yaitu data train dan data test, data yang ada pada data train akan dilakukan penghapusan background yang bertujuan untuk mengurangi noise sehingga data yang digunakan untuk proses pembelajaran model akan semakin baik. 3.2.3 *Rotate*



Gambar 11 Ilustrasi Rotate

Setelah menghapus seluruh background gambar yang ada pada data train, data yang ada pada data train akan di rotate 14 derajat ke arah kanan dan rotate 30 derajat ke arah kanan. Rotate bertujuan membuat data semakin variatif dan membuat model dapat mengidentifikasi data baru dengan lebih akurat. 3.2.4 *Flip*



Gambar 12 Ilustrasi Flip

Pada tahap ini, penulis sudah mempunyai tiga jenis data yang ada pada data train yaitu : 1) data hasil menghapus background (200 foto); 2) data hasil rotate 14 derajat; 3) data hasil rotate 30 derajat. Dari ketiga jenis data tersebut akan dilakukan flip dengan tujuan menambah variasi data agar model dapat mengidentifikasi data baru dengan lebih akurat.

3.2.5 *Labeling*

Di tahap ini, data yang ada pada data train berjumlah 1200 foto yang berasal dari data hasil hapus background sebanyak 200 foto + data hasil rotate 14 derajat 200 foto + data hasil rotate 30 derajat sebanyak 200 foto + data hasil hapus dan flip 200 foto + data hasil rotate 14 derajat + flip 200 foto + data hasil rotate 30 dan flip 200 foto dan data test berjumlah 300 foto. Setelah terkumpul 1200 data train dan 300 data test selanjutnya akan diberi label untuk semua data agar data dapat dikenali model pada saat proses pelatihan dan proses uji.

3.2.5 *Grayscale*

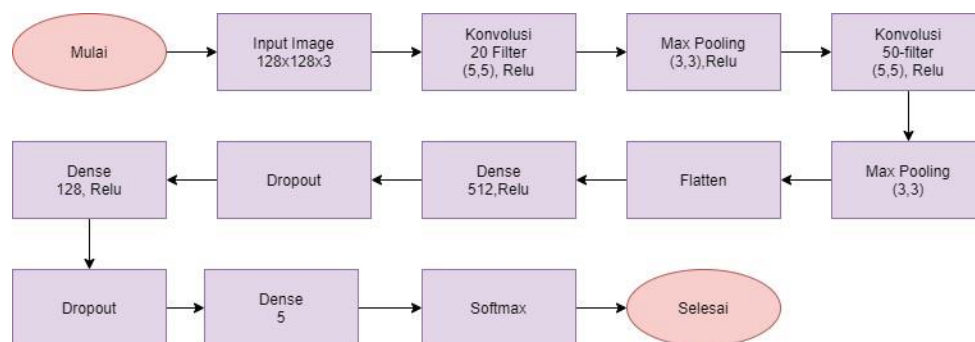
Dikarenakan penulis membuat dua model pada skenario pengujian yaitu model yang akan dilatih menggunakan dataset berwarna dan model yang dilatih menggunakan dataset *grayscale*. Oleh karena itu, akan ada data yang dikonversi menjadi *grayscale* untuk melakukan proses pelatihan model.

3.3. K – Fold Cross Validation

Setelah semua dataset dilakukan proses preprocessing, sebelum data train digunakan untuk melatih model. Data train akan dipecah menjadi data train dan data validasi menggunakan k – fold cross validation dengan nilai $k = 5$. Hal tersebut dilakukan agar seluruh data pada data train dapat tervalidasi dengan baik sehingga akurasi yang dihasilkan model akan semakin akurat.

3.4. Klasifikasi CNN

Pada tahap ini setiap data yang ada pada data train dan data validasi akan di masukkan ke dalam sistem lalu data tersebut akan di training menggunakan metode *convolutional neural network* untuk melakukan proses pembelajaran model sistem. Arsitektur CNN yang digunakan pada proses training model dapat dilihat pada Gambar 13.



Gambar 13 Arsitektur CNN

Seperti yang terlihat pada Gambar 13, input gambar yang digunakan pada proses training berukuran $128 \times 128 \times 3$ akan melewati dua proses konvolusi dengan filter 20 pada konvolusi pertama sehingga diakhir akan menghasilkan 20 *feature map* dan 50 pada konvolusi kedua dengan ukuran *feature detector* 5×5 , model juga akan melewati dua max pooling dengan ukuran matrix pooling 3×3 , model juga akan melewati tiga lapisan *hidden layer* dengan jumlah node 512, 128 dan 5. Diakhir akan ada fungsi aktivasi *softmax* untuk mengeluarkan probabilitas setiap kelas dengan probabilitas tertinggi merupakan kelas hasil klasifikasi.

3.5. Evaluasi Model

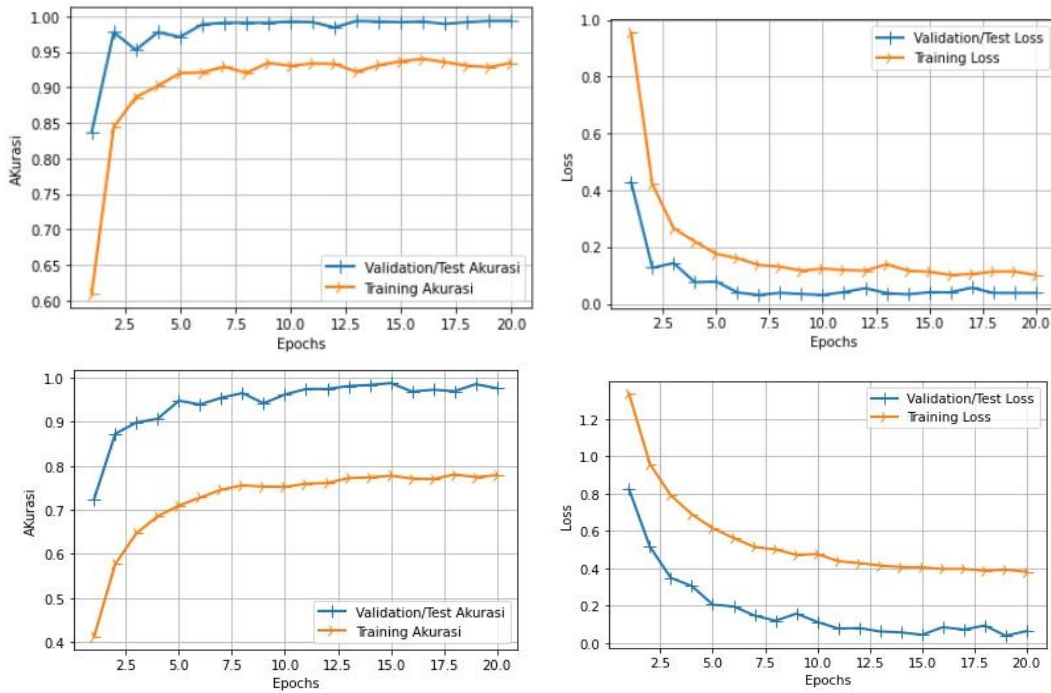
Evaluasi model akan dilakukan dengan cara melihat *confusion matrix* dari masing - masing model dan juga akan melihat rata – rata akurasi yang dihasilkan model selama iterasi fold pada k – fold cross validation. Model dengan nilai akurasi testing terbaik juga akan diuji menggunakan 50 data dari internet untuk melihat keakuratan akurasi model yang dihasilkan dari *confusion matrix*.

4. Evaluasi

4.1. Skenario Pengujian

Ada beberapa pengujian yang dilakukan penulis. Pertama, penulis mencari tau nilai epoch terbaik. Kedua, penulis melakukan berbagai percobaan nilai *learning rate* untuk mencari tau nilai *learning rate* terbaik. Ketiga, penulis melakukan berbagai percobaan nilai *dropout* untuk mencari tau nilai *dropout* terbaik. Percobaan dilakukan menggunakan dua model yang dilatih menggunakan dataset berwarna (RGB) dan dataset *grayscale*.

4.1.1 Percobaan Epoch

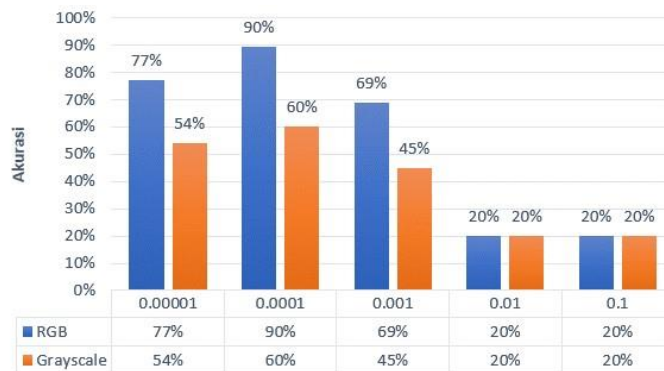


Gambar 14 Grafik Akurasi (kiri) dan Loss (kanan) Model RGB (Atas) dan Model Grayscale (Bawah)

Epoch digunakan untuk memantau proses pelatihan, semakin tinggi epoch maka hasil akurasi akan semakin tinggi dan nilai loss akan semakin rendah. Namun, akan ada suatu titik dimana nilai loss dan nilai akurasi tidak berubah secara signifikan sehingga nilai *epoch* tidak perlu ditambah.

Seperti yang terlihat pada Gambar 14, Nilai akurasi dan nilai loss model tidak mengalami perubahan yang signifikan. Oleh karena itu, menggunakan epoch 20 sudah cukup untuk pelatihan model.

4.1.1 Percobaan Learning Rate

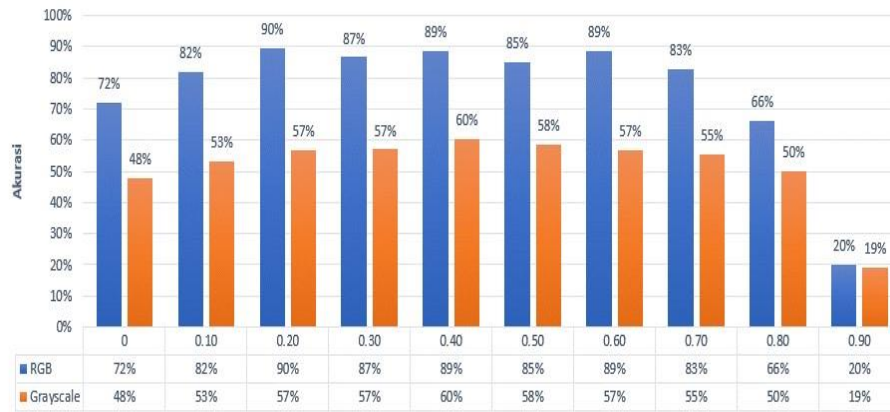


Gambar 15 Diagram Rata - Rata Akurasi Data Testing Learning Rate

Learning rate merupakan parameter penting yang berpengaruh terhadap kecepatan pelatihan dan performansi model. Oleh karena itu, penulis membuat lima percobaan nilai learning rate untuk menentukan nilai learning rate terbaik pada model RGB dan model *grayscale*. Hasil percobaan yang dilakukan penulis dapat dilihat pada Gambar 15.

Seperti yang terlihat pada Gambar 15, dari empat percobaan nilai learning rate yang dilakukan, nilai learning rate yang menghasilkan akurasi tertinggi adalah 0,0001 untuk model dengan dataset RGB maupun *grayscale* dengan nilai rata – rata akurasi sebesar 90% untuk model dengan dataset RGB dan 60% untuk model dengan dataset *grayscale* sedangkan nilai dengan akurasi terkecil yaitu learning rate 0,01 dan 0,1 dengan rata -rata akurasi 20% untuk model dengan dataset RGB maupun model dengan dataset *grayscale*.

4.1.2 Percobaan Dropout



Gambar 16 Diagram Rata - Rata Akurasi Data Testing Menggunakan Dropout

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*. Percobaan penulis dapat dilihat pada Gambar 16.

Seperti yang terlihat pada Gambar 16, dari 9 percobaan nilai dropout yang dilakukan penulis, nilai dropout yang menghasilkan akurasi tertinggi adalah 0,2 untuk model dengan dataset RGB dengan akurasi rata – rata 90% dan 0,4 untuk model dengan dataset *grayscale* dengan akurasi rata – rata sebesar 60% sedangkan nilai dropout yang menghasilkan akurasi terkecil adalah 0,9 untuk dataset RGB dengan nilai rata – rata akurasi sebesar 20% dan 0,9 untuk dataset *grayscale* dengan nilai – rata – rata akurasi sebesar 19%.

Seperti yang terlihat juga pada Gambar 16, menggunakan *dropout* mempunyai rata – rata akurasi yang lebih tinggi dibandingkan dengan tidak menggunakan *dropout*. Rata – rata akurasi testing tanpa menggunakan *dropout* adalah 72% sedangkan rata – rata akurasi testing tertinggi menggunakan dropout adalah 90%.

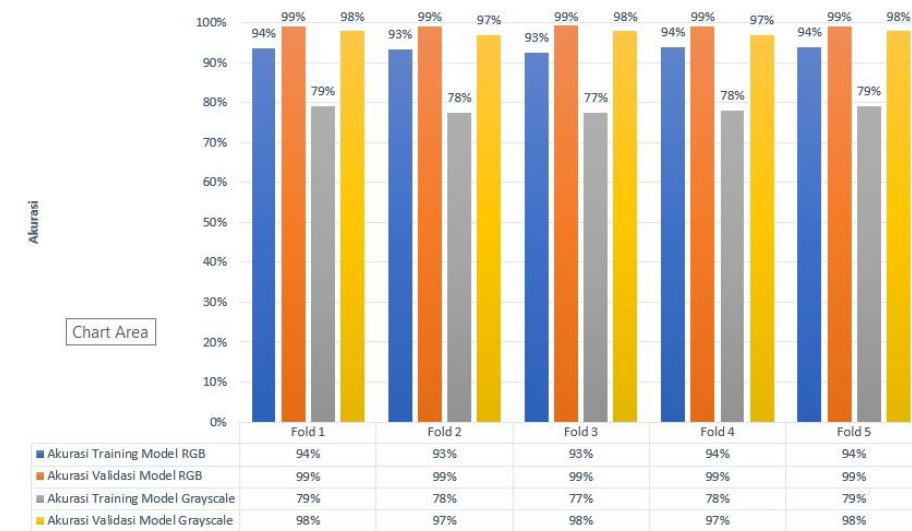
4.2. Hasil Pengujian

Pada tahap ini, penulis membuat model sesuai dengan rancangan arsitektur CNN yang ada pada Gambar 13 dengan menggunakan *K – Fold Cross – Validation* dengan nilai $k = 5$. Nilai $k = 5$ diambil karena nilai k yang biasa digunakan pada penelitian adalah 5 dan 10 yang dimana semakin besar nilai k maka komputasi yang dibutuhkan akan semakin berat oleh karena itu agar komputasi tidak terlalu berat dipilihlah nilai $k = 5$. Pengujian dilakukan penulis menggunakan parameter sebagai berikut :

1. Epoch : 20
2. Learning Rate : 0,0001
3. Dropout :
 - i. Model dengan dataset RGB : 0,2
 - ii. Model dengan dataset grayscale : 0,4
4. Fungsi Aktivasi : Relu
5. Ukuran Gambar : 128x128x3

Epoch 20 diambil karena berdasarkan percobaan penulis yang ada pada Gambar 13. Nilai *dropout* diambil berdasarkan percobaan penulis yang ada pada Gambar 15. Nilai *learning rate* diambil berdasarkan percobaan yang dilakukan penulis pada Gambar 14. Fungsi Relu dipakai mengacu pada sub bab 2.5.1.

4.2.1 Hasil Akurasi Training dan Akurasi Validasi

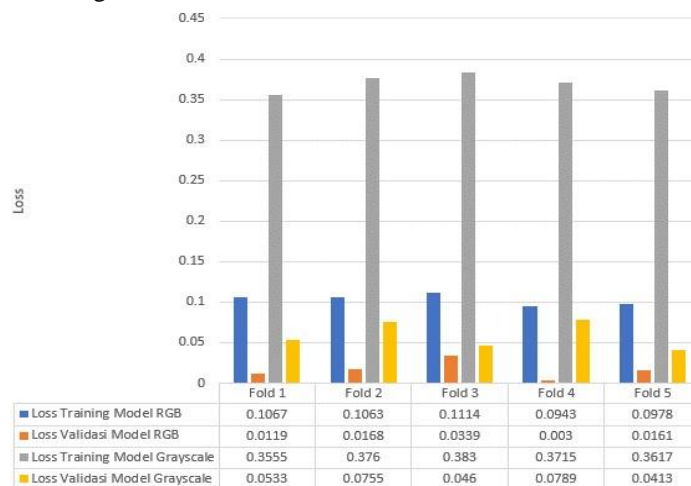


Gambar 17 Diagram Akurasi Training dan Validasi Model

Seperti yang terlihat pada Gambar 17, akurasi training model RGB berada di rentang 93% – 94% dan akurasi validasi model RGB berada pada 99% sedangkan akurasi training model *grayscale* 77% – 79% dan akurasi validasi model RGB berada di rentang 97% – 98%.

Akurasi validasi model RGB dan model *Grayscale* memiliki nilai yang sangat tinggi dikarenakan data yang digunakan untuk melakukan validasi merupakan data yang tidak memiliki *noise* sehingga model dapat mengklasifikasikan dengan baik data tersebut.

4.2.2 Hasil Loss Training dan Loss Validasi



Gambar 18 Diagram Loss Training dan Validasi Model

Seperti yang terlihat pada Gambar 18, model RGB mempunyai nilai loss tertinggi pada fold 3 yaitu 0,0339 untuk loss validasi dan loss 0,1114 untuk loss training. Model *grayscale* mempunyai nilai loss tertinggi juga pada fold 3 yaitu 0,383 untuk loss training dan 0,046 untuk loss validasi.

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

4.2.3 Hasil Testing Model RGB

	precision	recall	f1-score	support
Cotyledon	1.00	0.95	0.98	300
Echeveria	0.87	0.94	0.90	300
Hawortia	0.99	0.92	0.95	300
Sedum	0.89	0.86	0.88	300
Senecio	0.94	1.00	0.97	300
accuracy			0.93	1500
macro avg	0.94	0.93	0.93	1500
weighted avg	0.94	0.93	0.93	1500

Gambar 19 *Confussion Matrix* dan *Clasification Report* Model RGB

Berdasarkan *classification report* yang ada di Gambar 19, model 1 memiliki akurasi sebesar 93% dengan nilai akurasi tertinggi didapat dari genus senecio yaitu sebesar 100% dan nilai akurasi terendah didapat dari genus sedum dengan akurasi 86%

Berdasarkan *confussion matrix* yang ada pada Gambar 19, model 1 dapat mengklasifikasikan dengan benar 286 dari 300 gambar untuk genus cotyledon, 282 dari 300 gambar untuk genus echeveria, 275 dari 300 gambar untuk genus hawortia, 259 dari 300 gambar untuk genus sedum, dan 300 dari 300 gambar untuk genus senecio.

4.2.4 Hasil Testing Model Grayscale

	precision	recall	f1-score	support
Cotyledon	0.56	0.08	0.14	300
Echeveria	0.67	0.52	0.58	300
Hawortia	0.92	0.84	0.88	300
Sedum	0.37	0.76	0.50	300
Senecio	0.89	0.99	0.94	300
accuracy			0.64	1500
macro avg	0.68	0.64	0.61	1500
weighted avg	0.68	0.64	0.61	1500

Gambar 20 *Confussion Matrix* dan *Clasification Report* Model Grayscale

Berdasarkan *classification report* yang ada di Gambar 20, model memiliki akurasi sebesar 64% dengan nilai akurasi tertinggi didapat dari genus senecio yaitu sebesar 99% dan nilai akurasi terendah didapat dari genus cotyledon yaitu sebesar 8%.

Berdasarkan *confussion matrix* yang ada pada Gambar 20, model *grayscale* salah mengklasifikasikan data sebanyak 275 data dari 300 gambar untuk genus cotyledon, 145 data dari 300 gambar untuk genus echeveria, 49 data dari 300 gambar untuk genus hawortia, 72 data dari 300 gambar untuk genus sedum, dan 2 data dari 300 gambar untuk genus senecio.

Seperti yang terlihat juga pada Gambar 20, ada banyak gambar yang salah diidentifikasi oleh model sistem, hal ini kemungkinan dikarenakan adanya kemiripan bentuk dari gambar tersebut dengan jenis lain sehingga menyebabkan terjadinya kesalahan. Contoh foto yang memiliki kemiripan bentuk dapat dilihat pada Gambar 21.



Gambar 21 *Kemiripan Bentuk*

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 93% untuk model dengan dataset RGB dan 64% untuk model dengan dataset *grayscale* yang membuktikan bahwa melatih model menggunakan dataset berwarna dapat menambah keakuratan model dalam mengklasifikasi objek

4.4 Hasil Pengujian Menggunakan data baru dari internet

Untuk membuktikan akurasi model yang dilatih menggunakan dataset berwarna yang merupakan model dengan akurasi terbaik, penulis menguji kembali model dengan data yang didapat dari internet. Berikut hasil pengujian nya :

Table 2 Tabel Uji Dengan Data Internet

Genus	Jumlah Foto	Jumlah Benar	Jumlah Salah
Cotyledon	10	9	1
Echeveria	10	10	0
Hawortia	10	9	1
Sedum	10	6	4
Senecio	10	10	0
Total	50	45	6

Seperti yang terlihat di Tabel 2 dari pengujian dengan 50 data baru yang diambil dari internet pada model , model berhasil memprediksi dengan benar sebanyak 44 gambar dan salah memprediksi sebanyak 6 gambar. Akurasi uji data baru adalah sebesar $(44/50)*100 = 88\%$ yang berarti tidak jauh berbeda dengan akurasi testing model sehingga membuktikan bahwa akurasi testing model sudah cukup akurat.

5. Kesimpulan

Dari hasil penelitian yang telah dilakukan dapat ditarik kesimpulan bahwa metode CNN terbukti mampu mengklasifikasikan dengan baik untuk studi kasus genus tanaman sukulen dengan akurasi yang cukup besar yaitu 93% untuk 1500 data testing dan 88% untuk 50 data dari internet dan juga membuktikan bahwa melatih model dengan dataset berwarna dapat menghasilkan akurasi yang lebih baik dibandingkan dengan melatih model dengan dataset *grayscale*.

Saran untuk penelitian selanjutnya, menambah jumlah dataset sehingga dapat mengklasifikasikan genus – genus 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]. Khan . A , Sohail . A , Zahoora . U , and Qureshi . A . S , “A Survey of the Recent Architectures of Deep Convolutional Neural Network” , in Computer Vision and Pattern Recognition, 2019.
- [2]. A. Saad , M.A. Tareq , and Z.A. Saad , “Understanding of a Convolutional Neural Network, in The International Conference on Engineering and Technology, 2017.
- [3]. Nwankpa . C . E , Ijomah .W , Gachagan . A , and Mars . S , “Activation Functions : Comparison of Trends in Practice and Research for Deep Learning”, in Computer Vision and Science Recognition, 2018.

- [4]. Bayot . R . K . O , “A Survey on Object Classification using Convolutional Neural Network”.
- [5]. K. Alex , S. Ilya , and H. E. Geoffrey , “ImageNet Clasification with Deep Convolutional Neural Network”. [6]. M.D. Zeiler, M. Ranzato, R.Monga, M. Mao, K. Yang,Q. V. Le, and G.E. Hnton, “On rectified linear units for speech processing” in International Conference on Acoustics , Speech and Signal Processing. IEEE , 2013, pp. 35173521.
- [7]. J. Turian, J. Berhstra , and Y. Bengio, “Quadratic featur and deep architectures fot chunking“ ,in Proceedings of Human Language Technologies: The 2009 Annual Conference of the North American Chapter of the Association for Computational Linguistics,, vol. Companion Volume:.,2009,pp. 245-248.
- [8]. Bejiga, M.B., Zeggada, A. Nouffidj, A., and Melgani.F.(2017). A Convolutional Neural Network approach for assisting avalanche search and rescue operations with UAV imagery. Remote Sensing, 9(2).
- [9]. P. Ramachandran, B. Zoph, and Q. V. Le , “Searching for Activation Functions,” ArXiv, 2017.
- [10]. I. Sarirotul and N. Agung , “Implementasi Deep Learning Pada Identifikasi Jenis Tumbuhan Berdasarkan Citra Daun Menggunakan Convolutional Neural Network”, Jurnal Sistem & Teknologi Indonesia , Vol 3 , No 2 , 2018 .
- [11]. A. N. Erlyna and Harintaka , “Implementasi Metode Convolutional Neural Network untuk Klasifikasi Tanaman Pada Citra Resolusi Tinggi,”2008
- [12]. R. Payam, T. Lei, and L.Huan , “Cross Validation”
- [13]. Z. Liang, Z. Yali, W. Shengjin, and T. Qi, “Good Practice in CNN Feature Transfer”,2016.
- [14]. S. Nitish, H. Geoffrey, S. Ilya, and S. Ruslan, “Dropout : A Simple Way to Prevent Neural Networks from Overfitting”, Journal of Machine Learning Research, 1929 – 1958, 2014.
- [15]. A. N. Erlyna dan Harintaka, “ IMPLEMENTASI METODE CONVOLUTIONAL NEURAL NETWORK UNTUK KLASIFIKASI TANAMAN PADA CITRA RESOLUSI TINGGI”, 2018.
- [16]. Haryono, A. Khairul, S. Azmi, “Autentikasi Daun Herbal Menggunakan Convolutional Neural Network dan Raspberry Pi”, Jurnal Nasional Teknik Elektro dan Teknologi Informasi, Vol. 9, No 3, 2020.
- [17]. A. F. Slamet, “IMPLEMENTASI DEEP LEARNING UNTUK KLASIFIKASI TANAMAN TOGA BERDASARKAN CIRI DAUN BERBASIS ANDROID”, Ubiquitous : Computers and its application journal, Vol 2, No 2, 2019.
- [18]. W. Yanzhao, L. Ling, B. Juhyun, C. Ka-Ho, I. Arun, P. Calton, W. Wenqi, Y. Lei, and Z. Qi, “Demystifying Learning Rate Policies for High Accuracy Training of Deep Neural Network”, 2019.
- [19]. Wilson, D. & Martinez, Tony. 2001. The need for small learning rates on large problems. 1. 115 - 119 vol.1. 10.1109/IJCNN.2001.939002.