

BAB I

PENDAHULUAN

1.1 Latar Belakang

Indonesia adalah negara kepulauan yang memiliki tingkat biodiversitas tinggi. Salah satu kekayaan Indonesia yang dimiliki adalah kekayaan hayati yang dimanfaatkan sebagai tanaman obat dalam mengobati suatu penyakit. Kabupaten Rokan Hulu memiliki sumber daya hutan yang cukup tinggi dengan berbagai jenis tanaman obat. Pemanfaatan tanaman yang memiliki kandungan senyawa aktif sebagai bahan baku obat tradisional perlu dijaga dan dilestarikan agar dapat dikembangkan dengan baik sebagai salah satu unsur kebudayaan atau kearifan lokal[1].

Daerah Rokan Hulu masih memiliki banyak hutan dan lahan pertanian, sehingga menjadikannya habitat alami bagi berbagai jenis tanaman yang memiliki manfaat pengobatan tradisional. masyarakat Rokan Hulu, terutama di komunitas adat dan pedesaan, masih menggunakan tanaman obat sebagai bagian dari pengobatan alternatif. Pengetahuan tentang tanaman obat ini telah ditransmisikan secara lisan dari generasi ke generasi dan merupakan komponen kearifan lokal.

Tanaman obat yang tumbuh di Rokan Hulu umumnya berasal dari hutan tropis yang masih cukup terjaga dan sering kali memiliki khasiat unik sesuai kondisi geografis dan ekologis setempat. Tanaman seperti daun sirih, daun kunyit daun kemangi dan daun sirsak digunakan untuk berbagai pengobatan. Klasifikasi terhadap tanaman ini penting untuk memastikan bahwa informasi yang berkaitan dengan jenis, manfaat, dan ciri-ciri tanaman tidak hilang. Selain itu, klasifikasi

memudahkan masyarakat dan peneliti dalam mengidentifikasi dan memanfaatkan tanaman secara tepat.

Klasifikasi merupakan suatu proses untuk menemukan sekumpulan model maupun fungsi yang menjelaskan dan membedakan data ke dalam kelas-kelas tertentu, dengan tujuan dapat menentukan kelas dari suatu objek yang belum diketahui kelasnya[2]. Penggunaan tanaman obat oleh komunitas lokal masih dilakukan dengan cara tradisional dan sering kali tanpa catatan resmi. Proses pengenalan manual terhadap tanaman obat memerlukan keterampilan khusus dan waktu yang cukup banyak, sehingga metode klasifikasi berbasis teknologi menjadi sangat krusial untuk menjamin pelestarian dan pemanfaatan yang berkelanjutan.

Dalam prosesnya, klasifikasi dapat dilakukan dengan banyak cara baik secara manual maupun dengan bantuan teknologi. Klasifikasi yang dilakukan secara manual adalah klasifikasi yang dilakukan oleh manusia tanpa adanya bantuan dari algoritma cerdas komputer. Sedangkan klasifikasi yang dilakukan dengan bantuan teknologi, memiliki beberapa algoritma, di antaranya *Naive Bayes*, *Support Vector Machine*, *Decission Tree*, *Fuzzy*, Jaringan Saraf Tiruan dan *MobileNet*[3].

Namun demikian, penelitian terkait klasifikasi daun tanaman obat di wilayah Riau, khususnya Kabupaten Rokan Hulu, masih tergolong sangat terbatas. Minimnya publikasi ilmiah dan proyek riset di bidang ini menunjukkan adanya celah penelitian yang perlu segera diisi. Selain itu, proses identifikasi tanaman secara manual membutuhkan waktu yang lama dan keahlian khusus dalam mengenali ciri morfologis daun, sehingga kurang efisien jika diterapkan secara luas

di masyarakat. Metode identifikasi tradisional juga memiliki akurasi yang bervariasi dan cenderung subjektif, berbeda dengan pendekatan berbasis *CNN* (*Convolutional Neural Network*) yang telah terbukti mampu memberikan hasil klasifikasi yang lebih cepat, objektif, dan akurat dalam berbagai penelitian pengolahan citra.

Di zaman transformasi digital, penggunaan teknologi yang didasarkan pada kecerdasan buatan, terutama dalam pengolahan gambar digital, bisa menjadi jawaban untuk membantu pelestarian dan pengenalan tanaman obat. Salah satu metode yang cocok adalah menerapkan *deep learning* untuk mengklasifikasikan gambar tanaman secara otomatis. Salah satu jenis arsitektur *Convolutional Neural Network* (*CNN*) yang dikenal efisien dan tidak berat untuk perangkat seluler adalah *MobileNetV2*. Dengan kelebihan dalam hal efisiensi perhitungan dan dimensi yang kecil, arsitektur ini sangat ideal untuk digunakan dalam pengembangan sistem untuk mengklasifikasikan tanaman obat.

MobileNetV2 merupakan *pre-trained* model (model yang sudah dilatih) dalam mengklasifikasikan gambar dengan menggunakan *dataset* yang besar. Dengan menggunakan *pre-trained* model, maka pengembang tidak perlu membangun atau melatih model dari awal sehingga tidak memakan waktu. Pada arsitektur *MobileNetV2* memiliki 2 fitur antara lain *Linear Bottleneck* dan *Shortcut Connection* antar *Bottleneck*. Kedua fitur tersebut menjadikan proses training pada *Neural Network* bisa berjalan secara lebih efisien dan mendapatkan akurasi yang lebih baik[4].

Arsitektur ini sangat cocok untuk diterapkan di lingkungan dengan keterbatasan perangkat keras, seperti pada aplikasi *mobile* atau perangkat *edge computing*. Penggunaan *MobileNetV2* dalam klasifikasi tanaman obat dapat mempermudah proses identifikasi yang cepat dan akurat, bahkan dilakukan langsung dari perangkat *mobile*. Ini memiliki potensi besar untuk dimanfaatkan oleh masyarakat setempat, pelajar, tenaga kesehatan alternatif, dan pemerintah daerah dalam usaha pelestarian, dokumentasi, serta penggunaan tanaman obat di Rokan Hulu. Namun, untuk mencapai tujuan tersebut, diperlukan penelitian yang mendalam tentang cara membangun sistem klasifikasi yang berbasis *MobileNetV2* menggunakan *dataset* lokal tanaman obat di Rokan Hulu.

Berbagai penelitian sebelumnya telah menunjukkan efektivitas *MobileNet* dalam berbagai aplikasi pengolahan citra. Misalnya, penelitian oleh Putra dkk., (2023) berjudul “transfer *learning* untuk klasifikasi penyakit dan hama padi menggunakan *MobileNetV2*”, menunjukkan *accuracy* sebanyak 96%, *precision* sebesar 96%, *sensitivity* sebesar 96%, *specificity* sebesar 99%, dan *f1-score* sebesar 96%[5]. Penelitian lain oleh Zaelani dan Miftahuddin (2022) berjudul “Perbandingan metode *Efficientnet-B3* dan *MobileNetV2* untuk identifikasi jenis buah-buahan menggunakan fitur daun”, menunjukkan akurasi 99% dan jumlah *loss* 0,1046[6].

Berdasarkan masalah yang telah diuraikan, maka judul yang diangkat untuk proposal skripsi ini adalah **“Penerapan Model Arsitektur *MobileNetV2* Berbasis *Transfer Learning* Untuk Klasifikasi Tanaman Obat di Kabupaten Rokan Hulu Melalui Citra Daun”**.

1.2 Rumusan Masalah

Berdasarkan latar belakang masalah yang telah diuraikan maka rumusan masalah dalam penelitian ini Bagaimana Menerapkan Model Arsitektur *MobileNetV2* Berbasis *Transfer Learning* Untuk Klasifikasi Tanaman Obat di Kabupaten Rokan Hulu Melalui Citra Daun?

1.3 Tujuan Penelitian

Adapun tujuan penelitian ini adalah Untuk Menerapkan Arsitektur *MobileNetV2* Berbasis *Transfer Learning* Untuk Klasifikasi Tanaman Obat di Kabupaten Rokan Hulu Melalui Citra Daun.

1.4 Batasan Masalah

Agar penelitian ini lebih terfokus dan terarah maka peneliti membatasi masalah yang akan diteliti yaitu :

1. Model *Deep Learning* yang digunakan adalah *MobileNetV2*
2. Tanaman yang akan diteliti berupa 10 tanaman yaitu, daun sirih, daun jambu biji, daun belimbing, daun kemangi, daun kunyit, daun salam, daun pepaya, daun sirsak, lidah buaya, daun kembang sepatu.
3. Adapun jumlah data eksperimen yang digunakan yaitu 1000 *dataset* gambar citra tanaman obat dengan format *jpg* yang berasal dari hasil foto sendiri.
4. *Dataset* terbagi menjadi 2 yaitu data *training* dengan 80% dan data *testing* 20%.

1.5 Manfaat Penelitian

Adapun manfaat dari penelitian ini adalah :

1. Menambah wawasan dalam bidang *deep learning* untuk mengklasifikasikan tanaman obat berdasarkan citra daun
2. Menerapkan model *MobileNetV2* dalam klasifikasi tanaman obat.
3. Menghasilkan model *deep learning* berbasis *MobileNetV2* dalam mengenali tanaman obat di daerah Kabupaten Rokan Hulu berdasarkan citra daun.

1.6 Sistematika Penulisan

Sistematika penulisan dari skripsi ini terdiri dari pokok-pokok permasalahan yang akan dibahas pada masing-masing yang diuraikan menjadi beberapa bagian sebagai berikut :

BAB I PENDAHULUAN

Bab ini berisi latar belakang, rumusan masalah, tujuan penelitian, batasan masalah, manfaat penelitian, dan sistematika penulisan.

BAB II LANDASAN TEORI

Bab ini berisi teori-teori yang digunakan pada penelitian ini.

BAB III METODOLOGI PENELITIAN

Bab ini berisi tahapan-tahapan dalam identifikasi masalah, perumusan masalah, studi literatur, pengumpulan data, studi pustaka, perancangan, implementasi dan pengujian.

BAB IV ANALISA PERANCANGAN

Bab ini berisi Analisa dan perancangan aplikasi

BAB V IMPLEMENTASI DAN PENGUJIAN

Bab ini berisi implementasi dari Analisa dan perancangan dan pengujian pada aplikasi yang berhasil dibangun.

BAB VI PENUTUP

Bab ini berisi rangkuman hasil dari penelitian yang telah dilakukan dan saran-saran untuk penelitian selanjutnya.

BAB II

LANDASAN TEORI

2.1 Klasifikasi

klasifikasi merupakan suatu pekerjaan menilai objek data untuk memasukkannya ke dalam kelas tertentu dari sejumlah kelas yang tersedia. Di dalam klasifikasi, terdapat dua langkah yang dilakukan, yaitu membangun sebuah model yang akan disimpan sebagai memori, dan memanfaatkan model itu untuk melakukan pengenalan, klasifikasi, atau prediksi pada data lainnya, sehingga dapat diidentifikasi kategori mana yang sesuai untuk objek data tersebut berdasarkan model yang sudah ada di dalam memori[7].

2.2 Tanaman Obat

Tanaman obat merupakan tanaman yang sangat populer yang dapat dimanfaatkan sebagai bahan baku obat tradisional dan jamu. Pengobatan menggunakan tanaman obat yang tumbuh di lingkungan sekitar masyarakat maupun dibudidayakan merupakan pengobatan bagi penyakit ringan berdasarkan kepercayaan dan pengalaman yang dialami oleh masyarakat yang kemudian dikembangkan sesuai dengan budaya masyarakat tersebut[8]. Terdapat spesies tanaman yang mengandung senyawa bioaktif dan telah terbukti secara empiris mampu menyembuhkan atau meringankan gejala penyakit.

Beberapa di antaranya adalah sirih, bunga telang, mint dan mengkudu yang cukup sering digunakan dalam pengobatan tradisional di Indonesia. Daun sirih mengandung minyak *atsiri*, senyawa *fenolik*, *flavonoid*, *alkaloid*, *tanin* dan *saponin* yang berkhasiat obat. Penggunaan sirih dalam pengobatan tradisional antara lain

untuk mengobati sariawan, sakit gigi, batuk rejan, nyeri ulu hati, mual, disentri dan luka[9]. Dari segi morfologinya, tanaman obat terdiri dari berbagai jenis dalam penelitian ini peneliti hanya memfokuskan pada 10 jenis tanaman obat saja yaitu sebagai berikut :

1. Daun Sirih



Gambar 2. 1 Daun Sirih

Sirih merupakan salah satu tanaman obat yang potensial dan diketahui secara empiris memiliki khasiat dalam menyembuhkan berbagai penyakit. Daun sirih hijau digunakan oleh masyarakat untuk pengobatan dalam menghentikan pendarahan, gatal-gatal, sariawan dan menyembuhkan penyakit yang disebabkan oleh infeksi bakteri atau pun jamur[10].

2. Daun Jambu Biji



Gambar 2. 2 Daun Jambu Biji

Daun jambu biji sejak lama digunakan untuk pengobatan secara tradisional dan sudah banyak produksi herbal dari sediaan jambu biji. Efek *farmakologis* dari daun jambu biji yaitu anti diare, anti bakteri, anti diabetes, antihipertensi, dan penambah trombosit[11].

3. Daun Salam



Gambar 2. 3 Daun Salam

Daun salam mengandung zat bahan warna, zat samak dan minyak atsiri yang bersifat antibakteri. Zat tanin yang terkandung bersifat menciutkan (*astringent*). Manfaat daun secara tradisional, daun salam digunakan sebagai obat sakit perut. Daun salam juga dapat digunakan untuk menghentikan buang air besar yang berlebihan[12].

4. Daun Kemangi



Gambar 2. 4 Daun Kemangi

Kemangi secara empiris sering digunakan sebagai untuk mengatasi sakit perut, demam dan menghilangkan bau mulut. Selain itu, kemangi juga dapat dimanfaatkan untuk mengatasi penyakit diare. Kandungan senyawa fitokimia pada kemangi diketahui memiliki kemampuan menghambat pertumbuhan bakteri[13].

5. Daun Kunyit



Gambar 2. 5 Daun Kunyit

daun kunyit merupakan sumber senyawa bioaktif yang memiliki efek dalam pencegahan berbagai penyakit seperti kanker dan penuaan dini. ekstrak daun kunyit memiliki efek fungsionalnya seperti memutihkan, kosmetik, kekebalan kulit, anti-inflamasi, dan aktivitas antioksidan. Efek tersebut terutama berasal dari *kurkumin*, senyawa *fenolik* total, dan *flavonoid* pada daun kunyit[14].

6. Daun Belimbing



Gambar 2. 6 Daun Belimbing

Daun belimbing wuluh mengandung golongan senyawa bioaktif berupa flavonoid, tanin, saponin, dan *triterpen*. Metabolit sekunder dalam daun belimbing adalah senyawa bioaktif, dengan berbagai mekanisme kerja yang

memiliki efek sebagai anti oksidasi dan anti terhadap bakteri. Senyawa flavonoid pada umumnya memiliki kegunaan untuk perkembangan dan perlindungan terhadap dampak infeksi dan kerusakan. Selain itu, flavonoid berfungsi mereaksikan protein *ekstraseluler* dan *intraseluler* untuk membentuk senyawa kompleks [15].

7. Daun pepaya



Gambar 2. 7 Daun Pepaya

Daun pepaya memiliki potensi sebagai *hepatoprotektor*. Pepaya memiliki kandungan antioksidan antara lain adalah *alkaloid*, *flavonoid*, *saponin*, *tanin*, *triterpenoid* dan vitamin E. 7 Ekstrak daun pepaya memiliki aktivitas antioksidan *flavonoid* dan *fenolik* tertinggi dibandingkan bagian lain seperti ekstrak buah mentah, buah matang, dan ekstrak biji pepaya. Kandungan *flavonoid* daun pepaya lebih tinggi daripada daun katuk dan daun kenikir[16].

8. Daun Sirsak



Gambar 2. 8 Daun Sirsak

Daun sirsak memiliki kandungan antara lain: gigantetronin, linoleic acid, muricapentocin, fitosterol, mirisil alcohol, anonol, cacLOURINE, flavonoid, alkaloid, asam lemak, acetogenins, annohexocin, annona muricin, dan gentisic acid. kandungan flavonoid dalam daun sirsak berfungsi sebagai antidiabetic, anti inflamasi dan antioksidan. Selain flavonoid juga terdapat senyawa yang bersifat antidiabetic yaitu tannin[17].

9. Lidah Buaya



Gambar 2. 9 Lidah Buaya

Khasiat dari lidah buaya (*Aloe vera*) cukup beragam, antara lain sebagai anti biotik, anti septik, anti bakteri, anti virus, anti jamur, anti infeksi, anti peradangan dan anti pembengkakan. Keampuhan lidah buaya (*Aloe vera*) terletak pada kandungan nutrisinya, yakni *polisakarida* yang bekerja sama dengan asam-asam *amino esensial* dan *enzim* pemecah protein sehingga dapat mengganti sel yang rusak dan memperbaiki kondisi kulit. Lidah buaya juga mengandung berbagai macam zat di dalam daunnya seperti vitamin, mineral, enzim dan asam amino. Lidah buaya dapat menghambat pertumbuhan organisme penyebab penyakit kulit. Pada uji *in vitro*, diketahui bahwa lidah buaya dapat menghambat pertumbuhan *Dermatophilus Congolensis*[18].

10. Daun Kembang Sepatu



Gambar 2. 10 Daun Kembang Sepatu

Daun kembang sepatu dapat digunakan untuk pengobatan penyakit yang disebabkan bakteri karena mengandung flavonoid, saponin, dan polifenol yang dapat menghambat berkembangnya bakteri. Ekstrak etanol daun bunga sepatu memiliki daya hambat bakteri terhadap pertumbuhan *Escherichia coli* dengan konsentrasi ekstrak sebesar 20% dan 40%⁴. Pengujian Kadar Bunuh Minimal (KBM) menunjukkan ekstrak daun kembang sepatudapat membunuh bakteri *S.aureus* pada konsentrasi 10% [19].

2.3 Citra Digital

Citra adalah suatu bentuk representasi atau gambaran dari suatu objek. Citra dapat berupa foto optik, sinyal video analog seperti gambar pada layar TV, atau format digital yang dapat disimpan di media penyimpanan sebagai *output* dari sistem perekaman data. Citra digital terdiri dari sebuah matriks 2 dimensi dengan kolom dan baris, yang disebut *piksel*. *Piksel* adalah elemen terkecil pada citra yang memiliki nilai untuk merepresentasikan warna pada layar monitor. Terdapat tiga jenis citra digital, yaitu citra *binary* (monokrom), citra skala keabuan (*grayscale*), dan citra warna (*true color*). Citra warna menggunakan tiga nilai *piksel*, yaitu merah, hijau, dan biru, untuk membentuk berbagai macam warna yang dapat

ditampilkan pada layar monitor merah, hijau, dan biru, untuk membentuk berbagai macam warna yang dapat ditampilkan pada layar monitor[20].

2.4 *Machine Learning*

Machine Learning merupakan cabang *artificial intelligence (AI)* yang berfokus di pengolahan data, *Machine learning* termasuk komponen penting di ilmu yang berfokus pada pengembangan data. Melalui penggunaan *statistic*. Algoritma *machine learning* dilatih buat membentuk penjabaran maupun prediksi data dalam pengembangannya[21].

Machine learning merupakan metode berbasis komputer dimana komputer diberikan kemampuan untuk belajar dengan bantuan data tanpa harus terprogram terlebih dahulu. *Machine learning* telah berkembang, andal, dan alat pendukung di bidang medis dalam beberapa tahun terakhir[22].

2.5 *Deep Learning*

Deep Learning merupakan salah satu subdivisi *Machine Learning* yang mengandalkan jaringan saraf tiruan (*ANN*) atau dapat dianggap sebagai kemajuan dari *ANN*. *Deep Learning (DL)* adalah metodologi yang umum digunakan dalam beberapa tahun terakhir untuk mengimplementasikan *Machine Learning (ML)*. Hal ini didasarkan pada jaringan saraf tiruan. *Deep Learning (DL)* adalah teknik jaringan saraf yang menggunakan metode seperti *Restricted Boltzmann Machine (RBM)* untuk meningkatkan proses pembelajaran di jaringan saraf dengan beberapa lapisan, biasanya melebihi 7 lapisan. *Deep Learning* dalam cabang kecerdasan buatan (*AI*) menginstruksikan komputer untuk menganalisis data menggunakan metode yang meniru pengorganisasian otak manusia[23].

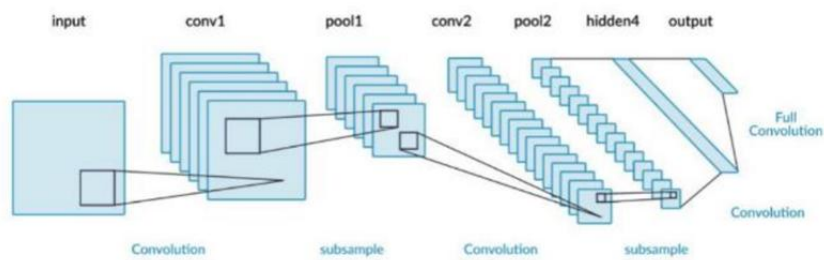
Deep learning adalah bagian dari *machine learning* namun memiliki jaringan tersendiri, disebut *deep* (dalam) karena di dalam struktur algoritmanya memiliki ratusan lapisan jaringan saraf. Salah satu kelebihan *deep learning* dapat membantu menyelesaikan masalah yang cukup kompleks, seperti kemampuan untuk mengenali sebuah citra, mengenali suara, hingga dapat meniru cara kerja otak manusia melalui saraf tiruan yang ada pada algoritmanya[24].

2.6 Jaringan Saraf Tiruan

Jaringan saraf tiruan adalah sistem pemroses informasi yang memiliki karakteristik mirip dengan syaraf biologi. Jaringan saraf tiruan mampu melakukan pengenalan kegiatan berbasis data masa lalu. Data masa lalu akan dipelajari oleh jaringan saraf tiruan sehingga mempunyai kemampuan untuk memberikan keputusan terhadap data yang belum pernah dipelajari[25]

2.7 Convolutional Neural Network (CNN)

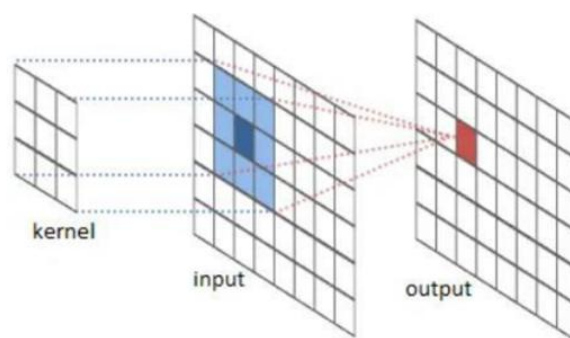
CNN adalah salah satu jaringan saraf tiruan *deep feedforward* yang luas diterapkan pada visi komputer. Metode ini juga dikenal sebagai *ConvNet*. Mirip dengan jaringan syaraf tiruan (*JST*), *CNN* memiliki arsitektur yang terdiri dari *node* atau *neuron* yang saling terhubung. *Neuron* yang saling terhubung satu sama lain dalam sebuah lapisan. arsitektur *CNN* itu sendiri terdiri dari Lapis tersembunyi (*Hidden Layer*) yang berisi *convolutional layers*, *pooling layers*, *normalization layers*, *ReLU layer*, *fully connected layers* dan *loss layer*[26]. Secara umum proses dari *CNN* dapat dilihat pada Gambar di bawah ini



Gambar 2. 19 Proses *Convolutional Neural Network* (CNN)

2.7.1 *Convolutional Layer*

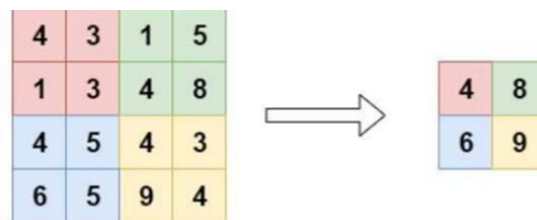
Lapisan yang pertama kali menerima *input* gambar langsung pada arsitektur. Operasi pada lapisan ini sama dengan operasi konvolusi, yaitu melakukan operasi filter kombinasi linier pada wilayah lokal. Filter ini merupakan representasi dari bidang reseptif dari *neuron-neuron* yang terhubung ke dalam lokal (konektivitas lokal) pada citra masukan. Bentuk dari direpresentasikan sebagai volume $B \times K \times L$ atau lapisan Lapisan berukuran $B \times K$ dengan jumlah lapisan sebanyak L . Lapisan konvolusi memiliki *hyperparameter* dan parameter[26].



Gambar 2. 20 *Convolutional layer*

2.7.2 Pooling Layer

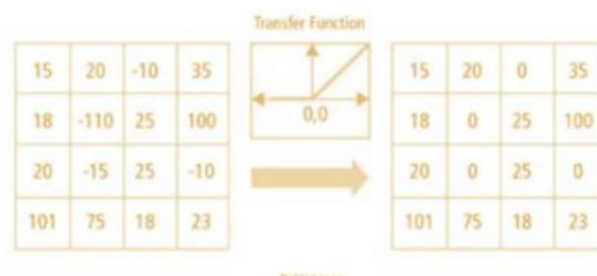
bagian yang digunakan untuk mengurangi ukuran gambar (memeriksa ke bawah) yang berencana untuk bekerja dengan prosedur konvolusi pada lapisan berikut. Interaksi ini harus dimungkinkan dengan beberapa prosedur, termasuk penggabungan maksimal (mengambil nilai terbesar dari sub-gambar), normal *pooling* (mengambil nilai normal dari sub-gambar), dan sebagainya. *pooling* (mengambil nilai normal dari sub gambar), dan seterusnya[26]. Sub gambar yang digunakan dalam kegiatan ini adalah 8x8 yang dapat dilihat pada contoh Gambar



Gambar 2. 21 *Pooling Layer*

2.7.3 ReLu Layer

Lapisan *Redressed Straight Unit (ReLU)* adalah lapisan yang berencana untuk bekerja pada non *linieritas* dari kemampuan pilihan. Kemampuan inisiasi yang digunakan adalah $f(x) = \max(0, x)$. Tujuannya yaitu mengubah semua *input* negatif menjadi nol[26]. Untuk *ReLU layer* sendiri dapat dilihat pada Gambar

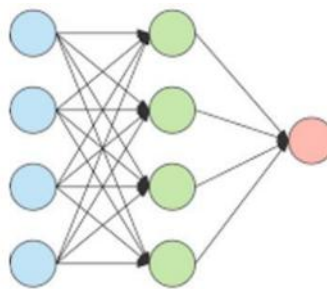


Gambar 2. 22 *Relu Layer*

2.7.4 *Fully Connected Layer*

Pada lapisan ini semua *neuron* sepenuhnya terkait (*fully connected*) seperti pada *multi-facet perceptron (MLP)* dan *multi-facet perceptron (MLP)*. Pada bagian ini terdapat muatan dan kecenderungan yang digunakan untuk mendemonstrasikan informasi yang telah diselesaikan oleh MLP[26].

Berikut *Fully Connected Layer*



Gambar 2. 23 *Fully Connected Layer*

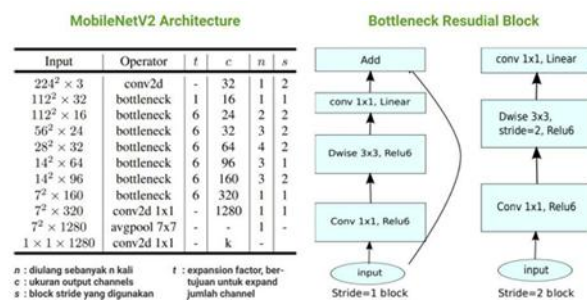
2.8 *MobileNetV2*

MobileNetV2 yang dikembangkan oleh Sandler (Sandler dkk, 2018), merupakan sebuah arsitektur *Convolutional Neural Network (CNN)* berbasis perangkat bergerak yang dirancang untuk mengatasi kebutuhan akan sumber daya komputasi yang terbatas. Arsitektur ini merupakan perbaikan dari arsitektur *MobileNet* sebelumnya. Salah satu perbedaan utama antara *MobileNet* dan arsitektur *CNN* lainnya terletak pada penggunaan lapisan konvolusi atau *convolution layer*. Pada *MobileNetV2*, lapisan konvolusi menggunakan filter dengan ketebalan yang sesuai dengan ketebalan gambar *input*. *MobileNetV2* menggabungkan *depthwise convolution*, *linear bottleneck*, *pointwise convolution* dan *shortcut connections* antar *bottlenecks* untuk meningkatkan efisiensi dan kinerja jaringan[27].

Input	Operator	t (faktor ekspansi)	c (jumlah kanal keluaran)	n (jumlah perulangan blok)	s (stride)
$224^2 \times 3$	Conv2d	-	32	1	2
$112^2 \times 32$	Bottleneck	1	16	1	1
$112^2 \times 16$	Bottleneck	6	24	2	2
$56^2 \times 24$	Bottleneck	6	32	3	2
$28^2 \times 32$	Bottleneck	6	64	4	2
$14^2 \times 64$	Bottleneck	6	96	3	1
$14^2 \times 96$	Bottleneck	6	160	3	2
$7^2 \times 160$	Bottleneck	6	320	1	1
$7^2 \times 320$	Conv2d 1x1	-	1280	1	1
$7^2 \times 1280$	Avgpool 7x7	-	-	1	-
$1 \times 1 \times 1280$	Conv2d 1x1	-	k	-	-

Gambar 2. 24 Arsitektur *MobileNetV2*

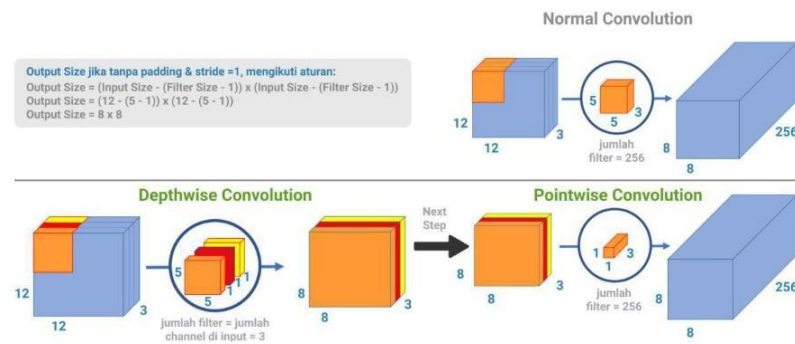
MobileNetV2 adalah modifikasi pertama dari *MobileNet*. Perbedaan mendasar antara arsitektur *MobileNetV2* dan arsitektur *CNN* pada umumnya adalah penggunaan lapisan atau *convolution* layer dengan ketebalan filter yang sesuai dengan ketebalan dari *input image*. Pemilihan metode *0043NN* dengan arsitektur *MobileNetV2* dalam penelitian ini didasari pada efektifitas metode dalam pemanfaatan *resource* yang tidak terlalu berat dan bisa menghasilkan akurasi yang maksimal berdasarkan hasil dari penelitian-penelitian sebelumnya[28] . Model pada penelitian ini disimulasikan di *Google Collaboratory* dengan *Python*, seperti terlihat pada Gambar



Gambar 2. 25 Alur Arsitektur *MobileNetV2*

MobileNetV2 dibuat oleh Google, mempunyai layer khusus yang disebut dengan *depthwise separable convolution*, merupakan sebuah blok yang terdiri dari *depthwise convolution* dan *pointwise convolution*. Tujuan dari layer ini ialah mereduksi komputasi (agar lebih sedikit parameter) sehingga menghasilkan ukuran

model yang lebih kecil [12,13]. Untuk memahami *depthwise separable convolution*, berikut ilustrasi perbandingannya dengan normal *convolution* [30,31]. Terdapat *input image* berukuran 12x12x3 dan 256 *kernel* berukuran 5x5x3.



Gambar 2. 26 Normal Convolution vs Depthwise Separable Convolution

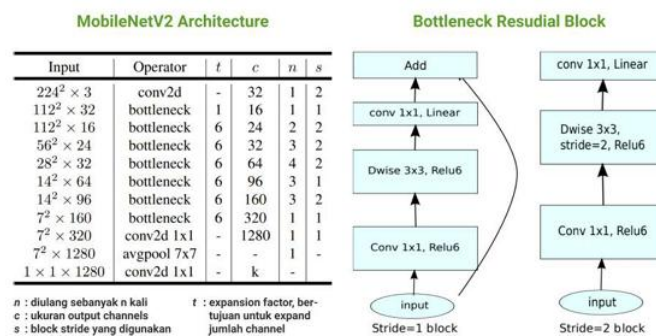
Dalam normal *convolution*, filter melakukan konvolusi terhadap *input image* dan menghasilkan *feature map*(*output image*). Sedangkan dalam *depthwise convolution*, jumlah filter akan sama dengan jumlah *channel* di *input image*, setiap filter akan melakukan konvolusi terhadap masing-masing *channel* di *input image* (atau dengan kata lain, tidak langsung melakukan konvolusi terhadap seluruh *channel* di *input image* seperti yang dilakukan pada normal *convolution*), lalu menghasilkan *feature map* yang memiliki jumlah *channel* yang sama dengan jumlah filter. Tahap selanjutnya, agar *output image* dengan *depthwise convolution* sama dengan *output image* yang dihasilkan dengan normal *convolution*, maka akan dilakukan *pointwise*, dimana *convolution* hasil dari tahap *depthwise convolution* akan di *konvolusi* lagi dengan filter 1x1, filter ini memiliki kedalaman sama dengan jumlah *channel* di *output image* sebelumnya. Berikut perbandingan jumlah komputasi antara normal *convolution* versus *depthwise separable convolution* berdasarkan contoh gambar diatas [20].

1. Normal Convolution

Terdapat 256 filter $5 \times 5 \times 3$ yang bergeser sebanyak 8×8 kali (8×8 yaitu jumlah pergeseran filter 5×5 dari kiri atas ke kanan bawah terhadap *input image* 12×12). Ini berarti $256 \times 5 \times 5 \times 3 \times 8 \times 8 = 1.228.800$ total perkalian yang dilakukan saat proses konvolusi.

2. Depthwise Separable Convolution

Pertama, pada tahap *depthwise convolution* terdapat 3 filter $5 \times 5 \times 1$ yang bergeser sebanyak 8×8 kali. Ini berarti $3 \times 5 \times 5 \times 1 \times 8 \times 8 = 4.800$. Selanjutnya, pada tahap *pointwise convolution* terdapat 256 filter $1 \times 1 \times 3$ yang bergeser sebanyak 8×8 kali. Ini berarti $256 \times 1 \times 1 \times 3 \times 8 \times 8 = 49.152$. Maka total perkalian yang dilakukan saat proses konvolusi di *depthwise separable convolution* ini yaitu $4.800 + 49.152 = 52.952$. Dapat dilihat bahwa, jumlah 52.952 hanya kurang lebih 22,7% nya saja dari 1.228.800 di normal convolution.



Gambar 2. 27 Arsitektur MobileNetV2

Perkembangan lebih lanjut dari *MobileNet* yaitu *MobileNetV2*, dimana ditambahkan fitur baru yaitu *bottleneck* yang merupakan sebuah blok dalam arsitektur yang di dalamnya menggunakan *depthwise separable convolution*[20].

2.9 *Python*

Python adalah bahasa pemrograman yang populer dan sering digunakan oleh berbagai kalangan, seperti akademisi dan masyarakat umum, untuk berbagai tujuan, seperti membuat aplikasi desktop, website, game, melakukan analisis, serta membangun proyek kecerdasan buatan[29].

Python merupakan bahasa pemrograman yang berorientasi objek dinamis, dapat digunakan untuk bermacam macam pengembangan perangkat lunak. *Python* menyediakan dukungan yang kuat untuk integrasi dengan bahasa pemrograman lain dan alat-alat bantu lainnya. *Python* hadir dengan pustaka-pustaka standar yang dapat diperluas serta dapat dipelajari hanya dalam beberapa hari. Bahasa pemrograman yang interpretatif multiguna dengan filosofi perancangan yang berfokus pada tingkat keterbacaan kode. *Python* diklaim sebagai bahasa yang menggabungkan kapabilitas, kemampuan, dengan sintaksis kode yang sangat jelas, dan dilengkapi dengan fungsionalitas pustaka standar yang besar serta komprehensif[30].

Ada banyak *library* dan. *library-library* ini memiliki fitur yang memadai untuk mempermudah proses implementasi dan pengembangan proyek *CNN* yaitu sebagai berikut:

1. *OpenCV (Open Source Computer Vision)*

adalah *library* dari fungsi pemrograman untuk *realtime* visi komputer .

OpenCV menggunakan lisensi BSD dan bersifat gratis baik untuk penggunaan akademis maupun komersial. *OpenCV* dapat digunakan dalam

bahasa pemrograman C, C++, *Python*, *Java*, dan sebagainya. *OpenCV* dapat digunakan pada sistem operasi *Windows*, *Linux*, *Android*, *iOS* dan *Mac OS*. *OpenCV* memiliki lebih dari 2500 algoritma yang telah dioptimalkan [30].

2. *TensorFlow*

TensorFlow adalah *open source library* untuk *machine learning* yang di *release* oleh *Google* yang mendukung beberapa bahasa pemrograman. Dalam proses *Transfer Learning*, *Tensorflow* berperan untuk memproses *MobileNetV2* Model untuk di *training* ulang menggunakan data yang baru dan kemudian menghasilkan *classifier* dengan komputasi yang cepat dan akurasi yang baik. *Tensorflow* dapat digunakan pada semua sistem operasi [30].

3. Keras

Keras Merupakan *high-level API* dari *Tensorflow*, dikembangkan oleh *François Chollet*, seorang *engineer* di *Google*. Dirancang untuk manusia, bukan mesin (lihat: <https://keras.io/>). Artinya mudah dipelajari dan digunakan. Sehingga para peneliti bisa lebih fokus pada eksperimen yang dilakukan bukan pada bagaimana cara menggunakan *library* tersebut. Dengan kata lain, Keras membungkus *Tensorflow* menjadi blok-blok yang lebih mudah digunakan, terlebih untuk para pemula.

4. *Numpy*

Numpy merupakan salah satu pustaka pendukung untuk bahasa pemrograman *python*. *Numpy* memberikan dukungan untuk *python* dalam hal matriks atau himpunan dengan multidimensi yang cukup besar. *Numpy* merupakan *library* yang lebih berfokus kepada *scientific computing*[31].

1. *Pandas*

Pandas adalah pustaka *Python* yang dirancang untuk manipulasi dan analisis data. Diperkenalkan pada tahun 2008 oleh Wes McKinney, *Pandas* menawarkan struktur data yang fleksibel dan alat analisis yang kuat, memungkinkan pengguna untuk bekerja dengan data terstruktur dengan mudah [32].

2.10 Evaluasi Model Klasifikasi

1. *Confusion Matrix*

Salah satu cara untuk menilai kinerja suatu model klasifikasi adalah melalui pengukuran akurasi model tersebut. Ada beberapa istilah dasar yang menjadi dasar dalam menghitung nilai akurasi, seperti *true positive (TP)*, *true negative (TN)*, *false positive (FP)*, dan *false negative (FN)*. Istilah-istilah tersebut sering dijelaskan dalam bentuk matriks yang dikenal sebagai *confusion matrix* [31]

		Actual Values	
		Positive (1)	Negative (0)
Predicted Values	Positive (1)	TP	FP
	Negative (0)	FN	TN

Gambar 2. 28 Confusion Matrix

Dimana :

1. TP adalah *True Positif*, yaitu jumlah data positif yang terklasifikasi dengan benar oleh sistem.
2. TN adalah *True Negatif*, yaitu jumlah data negatif yang terklasifikasi dengan benar oleh sistem.
3. FN adalah *False Negatif*, yaitu jumlah data negatif namun terklasifikasi salah oleh sistem.
4. FP adalah *False Positif*, yaitu jumlah data positif namun terklasifikasi salah oleh sistem [33].

Akurasi dalam konteks klasifikasi merujuk pada persentase kebenaran dalam pengelompokan data setelah diuji pada hasil klasifikasi. Penghitungan akurasi dengan menggunakan *confusion matrix* dapat dijelaskan pada Persamaan 2 [31].

$$akurasi = \frac{TP+TN}{TP+FP+FN+TN} \quad (1)$$

Confusion matrix membantu dalam mengevaluasi performa model dengan menyajikan informasi tentang sejauh mana kemampuan model untuk

mengidentifikasi dan mengklasifikasikan data. Dari elemen-elemen yang terdapat di dalamnya, kita dapat menghitung beberapa matrik evaluasi krusial, termasuk akurasi, presisi, *recall* (sensitivitas), *F1-score*, dan lainnya [31].

Akurasi, presisi, *recall*, dan *F1-score* adalah metode-metode pengujian dalam evaluasi performa sistem. Akurasi mengukur tingkat kesamaan antara nilai prediksi dan nilai sebenarnya, mencerminkan jumlah data yang diklasifikasikan secara benar. Presisi membandingkan jumlah informasi relevan yang diperoleh oleh sistem dengan total informasi yang diambil, memperhitungkan baik informasi relevan maupun tidak relevan. *Recall*, sebaliknya, membandingkan jumlah informasi relevan yang diperoleh dengan total informasi relevan yang tersedia, fokus pada rasio benar positif dari seluruh data yang benar positif. *F1-score*, sebagai ukuran statistik, menggabungkan presisi dan *recall* untuk menciptakan keseimbangan, berguna ketika distribusi kelas tidak seimbang[31].

2. Accuracy

Accuracy adalah rasio yang memiliki prediksi nilai benar (nilai positif dan nilai negatif) berdasarkan keseluruhan data. Akurasi dapat menggambarkan keakuratan model klasifikasi yang digunakan. Nilai akurasi dapat diperoleh menggunakan persamaan berikut ini[34].

$$akurasi = \frac{TP+TN}{TP+FP+FN+TN} \quad (2)$$

3. *Precision*

Precision adalah rasio yang memiliki prediksi nilai benar positif jika dibandingkan dengan keseluruhan hasil yang diprediksi positif. *Precision* dapat menggambarkan keakuratan data yang diinginkan dengan hasil prediksi yang diperoleh model klasifikasi. Nilai *Precision* dapat diperoleh menggunakan persamaan berikut ini [34].

$$precision = \frac{TP}{TP+FP} \quad (3)$$

4. *Recall*

Recall adalah rasio prediksi benar positif dibandingkan dengan keseluruhan data yang benar positif. *Recall* menggambarkan hasil dari model klasifikasi yang digunakan dalam menemukan kembali sebuah informasi. Nilai *Recall* dapat diperoleh menggunakan persamaan berikut ini [34].

$$recall = \frac{TP}{TP+FN} \quad (4)$$

5. *F1 Score*

F1 Score adalah perhitungan kombinasi dari nilai *precision* dan nilai *recall* yang kemudian hasilnya disebut sebagai nilai pengukuran. *F1 Score* dapat diperoleh menggunakan persamaan berikut ini [34].

$$F1\ Score = 2 \times \frac{Precision \times Recall}{Precision + Recall} \quad (5)$$

2.11 Google Colab

Google Colab (singkatan dari *Google Colaboratory*) adalah platform berbasis *cloud computing* yang disediakan oleh *Google*. Ini memungkinkan pengguna untuk mengeksekusi kode *Python* dalam lingkungan berbasis *cloud* tanpa perlu menginstal atau mengatur lingkungan lokal mereka sendiri. *Google Colab* sering digunakan oleh para ilmuwan data, peneliti, dan pengembang untuk melakukan berbagai jenis pekerjaan, termasuk pemrosesan data, pengembangan model kecerdasan buatan, analisis data, dan pelatihan model mesin[35].

Resource Colab tidak dijamin dan sifatnya terbatas, serta batas penggunaannya terkadang berfluktuasi. Hal ini diperlukan agar *Colab* dapat menyediakan *resource* secara gratis. Pengguna yang ingin memiliki akses lebih andal ke *resource* yang lebih baik dapat menggunakan *Colab Pro*. Memperkenalkan *Colab Pro* merupakan langkah pertama yang *Google* ambil untuk melayani pengguna yang ingin melakukan lebih banyak hal di *Colab*. Tujuan jangka panjang pihak *Google* adalah untuk terus menyediakan versi gratis *Colab*, dan di saat yang bersamaan berkembang secara berkelanjutan untuk memenuhi kebutuhan pengguna *Google*[36].

2.12 Penelitian Terkait

Tabel 2. 1 Penelitian Terkait

No	Nama	Tahun	Judul	Hasil
1	Ariya Purnama, Jamaludin Indra, Santi Arum Puspita Lestari,	2025	Deteksi Pelanggaran Penggunaan Helm dengan Metode	Model yang dikembangkan mencapai:

	Sutan Faisal		SSD dan Arsitektur <i>MobileNetV2</i>	1. Akurasi maksimum sebesar 100% pada data pelatihan. 2. Akurasi deteksi 95% untuk pengendara yang mengenakan helm. 3. Akurasi deteksi 98% untuk pengendara yang tidak mengenakan helm. 4. <i>Mean Average Precision (mAP)</i> sebesar 99% pada ambang batas <i>IoU</i> 0.5 (mAP50). 5. <i>Mean Average Precision (mAP)</i> sebesar 66% pada ambang batas <i>IoU</i> 0.75 (mAP75). Hasil ini menunjukkan efektivitas sistem dalam mendeteksi pelanggaran penggunaan helm oleh pengendara sepeda motor.
--	--------------	--	--	--

2	Mohamad Ansori Pratama	2025	Implementasi Arsitektur <i>MobileNetV2</i> untuk Deteksi Penyakit Antraknosa dan Busuk Buah pada Cabai Rawit	Model <i>MobileNetV2</i> yang diterapkan dalam penelitian ini menunjukkan hasil akurasi tinggi: 1. Akurasi pelatihan mencapai 97,36% pada <i>dataset</i> pelatihan. 2. Akurasi validasi mencapai 100% pada <i>dataset</i> validasi. 3. Grafik akurasi dan <i>loss</i> menunjukkan performa model yang stabil dengan <i>overfitting</i> minimal. 3. Penelitian ini diharapkan dapat membantu petani dalam mendeteksi penyakit secara cepat dan meningkatkan produktivitas cabai rawit di Indonesia.
---	---------------------------	------	---	---

3	Nagala Wangsa Kencana, Rusydi Umar, Murinto	2025	Implementasi <i>Transfer Learning</i> Untuk Klasifikasi Jenis Ras Ayam Menggunakan Arsitektur <i>MobileNetV2</i>	Model <i>MobileNetV2</i> yang dikembangkan mencapai: 1. Akurasi pengujian sebesar 95% dengan <i>loss</i> 0,378. Penelitian ini menggunakan <i>dataset</i> yang terdiri dari 1800 citra untuk pelatihan dan 300 citra untuk pengujian, dengan 12 kelas ras ayam. Hasil menunjukkan bahwa sistem klasifikasi jenis ras ayam yang dibangun dapat digunakan secara efektif untuk membantu identifikasi ras ayam berdasarkan citra.
4	Bayu Wijayanto, Rizqi Maulana Mahendra,	2025	Identifikasi Jenis Ikan Cupang Menggunakan	Hasil pengujian menunjukkan akurasi tertinggi

	Mohammad Ibnu Salam		Metode <i>CNN</i> Dengan Arsitektur <i>MobileNetV2</i> Berbasis <i>Mobile</i>	untuk jenis ikan cupang: 1. Ikan Cupang <i>Halfmoon</i>: 95% 2. Ikan Cupang <i>Crowntail</i>: 94% 3. Ikan Cupang <i>Plakat</i>: 91% Implementasi berbasis <i>mobile</i> memungkinkan pengguna untuk melakukan identifikasi ikan cupang secara cepat dan praktis melalui aplikasi yang dirancang, diharapkan dapat membantu penggemar ikan cupang dan pedagang dalam mengenali jenis ikan dengan lebih mudah dan efisien.
5	Raihan Fahlevi, Chaerur Rozikin	2025	Identifikasi Isyarat Tangan BISINDO dengan Algoritma <i>CNN</i> dan <i>Transfer Learning</i>	Model yang dikembangkan mampu mengenali huruf BISINDO dengan:

			Menggunakan MobileNetV2	1. Akurasi validasi sebesar 95% 2. Akurasi pengujian sebesar 93,33% Hasil ini menunjukkan bahwa penggunaan <i>transfer learning</i> dengan <i>MobileNetV2</i> efektif dalam membangun sistem pengenalan huruf BISINDO secara cepat dan akurat, serta diharapkan dapat menjembatani komunikasi antara pengguna BISINDO dan masyarakat umum secara lebih efisien.
6	Dimas Febri Kuncoro, Rian Ardianto	2024	Analisis Sel Darah Putih dengan Pendekatan Bioinformatika menggunakan	Model klasifikasi sel darah putih yang diimplementasikan menggunakan <i>Convolutional</i>

			Arsitektur MobileNetV2	<i>Neural Network (CNN)</i> dengan arsitektur <i>MobileNetV2</i> mencapai tingkat akurasi sebesar 94.8% setelah proses pelatihan sebanyak 15 <i>epoch</i> . Model ini menunjukkan performa yang baik dalam mengklasifikasikan berbagai jenis sel darah putih, dengan evaluasi yang dilakukan menggunakan metrik standar seperti akurasi, <i>presisi</i> , <i>recall</i> , dan <i>F1-score</i> .
7	Devfris Dhimas Permana Putra Galang Kurnia Anaga Wahyu Tia Fitriyana	2024	Implementasi Algoritma <i>Convolutional Neural Network</i> Arsitektur <i>MobileNetV2</i> Untuk Klasifikasi	1. Model <i>CNN</i> dengan arsitektur <i>MobileNetV2</i> mencapai akurasi sebesar 42,68% dalam klasifikasi ekspresi wajah.

			Ekspresi Wajah Pada <i>Dataset</i> FER	2. Penelitian menyarankan eksplorasi lanjutan dengan teknik <i>augmentasi</i> data dan dropout untuk meningkatkan akurasi model. 3. Disarankan juga untuk menggunakan data latih yang lebih besar untuk hasil yang lebih baik.
8	Putri Sarah Fransisca, Nurhafifah Matondang	2023	Deteksi Citra Digital Penyakit Cacar Monyet menggunakan Algoritma <i>Convolutional Neural Network</i> dengan <i>Arsitektur MobileNetV2</i>	Model yang dikembangkan menggunakan algoritma <i>Convolutional Neural Network</i> (CNN) dengan arsitektur <i>MobileNetV2</i> menunjukkan hasil sebagai berikut: 1. Akurasi pada data uji dengan Adam <i>optimizer</i>: 94%

				2. Akurasi pada data latih dengan Adam <i>optimizer</i>: 92% 3. <i>Loss function</i> dengan Adam <i>optimizer</i>: 27% 4. Akurasi pada data uji dengan <i>RMSprop optimizer</i>: 97% 5. Akurasi pada data latih dengan <i>RMSprop optimizer</i>: 97% 6. <i>Loss function</i> dengan <i>RMSprop</i>
--	--	--	--	---

				<i>optimizer</i>: 52% Hasil ini menunjukkan bahwa Adam <i>optimizer</i> lebih efektif dalam menyempurnakan parameter model untuk mengoptimalkan deteksi gambar cacar monyet selama pelatihan.
9	Nindo Syafi'al Arief Wahyu Anggara Putra Dieky Septhian Rastra Pratama	2024	implementasi <i>CNN</i> Arsitektur <i>MobileNetV2</i> Untuk Klasifikasi Tulisan Aksara Jawa	Model <i>CNN</i> <i>MobileNetV2</i> yang dikembangkan mencapai: 1. Akurasi sebesar 87,41% tanpa <i>fine-tune</i>. 2. Presisi sebesar 86,95% tanpa <i>fine-tune</i>. 3. Akurasi setelah <i>fine-tune</i> mencapai 91,04% dengan presisi 89,64%. Penelitian ini menunjukkan bahwa model dapat

				digunakan untuk klasifikasi tulisan aksara Jawa dan berkontribusi terhadap digitalisasi naskah kuno berhuruf Jawa.
10	Ahmad Zidan Hibatullaha, Muhammad Fadhilah Rahman, Anggraini Puspita Sari	2024	Pemanfaatan Metode <i>Convolutional Neural Network (CNN)</i> Dengan Arsitektur <i>MobileNetV2</i> Untuk Penilaian Kelayakan Rumah	Sistem penilaian kelayakan rumah yang dikembangkan menggunakan metode <i>CNN</i> menunjukkan: 1. Peningkatan efisiensi dan akurasi dalam proses penilaian kelayakan rumah untuk bantuan sosial. 2. Kemampuan untuk menganalisis foto rumah yang diunggah dan memberikan penilaian berdasarkan kriteria yang telah ditetapkan.

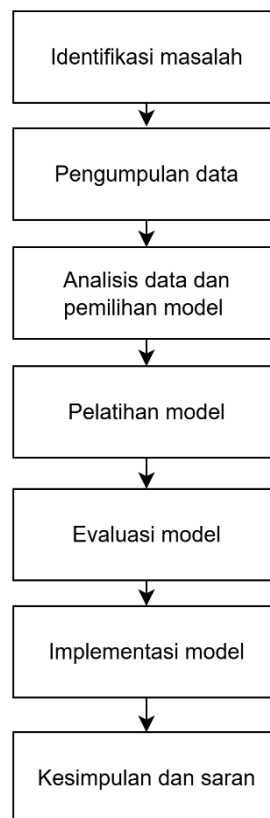
				Diharapkan sistem ini dapat meningkatkan efektivitas bantuan sosial dan pembangunan masyarakat, serta mengurangi beban kerja administratif.
--	--	--	--	---

BAB III

METODOLOGI PENELITIAN

3.1 Kerangka Penelitian

Penelitian ini dilakukan secara bertahap dengan pendekatan yang terstruktur dan sistematis. Setiap tahapan saling berkaitan untuk memastikan proses penelitian berjalan dengan baik dan menghasilkan model klasifikasi yang bagus. Metode yang digunakan digambarkan secara rinci dalam skema yang menggambarkan alur penelitian secara jelas dan teratur. Adapun tahapan lengkap dari penelitian ini dapat dilihat pada Gambar berikut :



Gambar 3. 1 Tahapan Penelitian

3.2 Identifikasi Masalah

Identifikasi tanaman obat secara manual di Kabupaten Rokan Hulu masih menghadapi berbagai kendala, seperti ketergantungan pada keahlian individu, potensi kesalahan dalam pengenalan, serta keterbatasan efisiensi waktu. Seiring dengan perkembangan teknologi pengolahan citra dan pembelajaran mesin, diperlukan sebuah pendekatan berbasis *deep learning* yang mampu mengotomatisasi proses klasifikasi tanaman obat secara lebih akurat dan efisien. Oleh karena itu, penelitian ini mengkaji penerapan arsitektur *MobileNetV2* berbasis *transfer learning* sebagai solusi terhadap permasalahan tersebut.

3.3 Pengumpulan Data

Pengumpulan data dalam penelitian ini dilakukan dengan menghimpun citra daun dari 10 jenis tanaman obat Di Kabupaten Rokan Hulu. Data citra tersebut diperoleh melalui data dari *kaggle*. Setiap gambar kemudian diklasifikasikan dan diberi label sesuai dengan jenis tanaman obatnya. Data yang telah terkumpul selanjutnya digunakan sebagai *input* dalam proses pelatihan dan pengujian model klasifikasi berbasis arsitektur *MobileNetV2*.

3.4 Analisis Data Dan Pemilihan Model

Pada tahap ini, *dataset* dianalisis dan model dipilih untuk memahami distribusi data serta menentukan arsitektur yang paling sesuai. Dengan menganalisis karakteristik *dataset* secara mendalam, pemilihan model dapat dilakukan dengan lebih tepat sehingga hasil klasifikasi menjadi lebih akurat dan dapat diterapkan dalam kondisi nyata. Setelah mempertimbangkan jumlah *dataset* yang digunakan serta kebutuhan komputasi sumber daya, model yang dipilih untuk penerapan ini adalah *MobileNetV2*.

3.5 Pelatihan Model

Proses pelatihan model dilakukan dengan mengimplementasikan *MobileNetV2* menggunakan *TensorFlow* atau *keras*. *Hyperparameter* seperti *learning rate*, *batch size*, jumlah *epoch*, dan jenis *optimizer* disesuaikan agar model dapat mencapai performa terbaik.

3.6 Evaluasi Model

Evaluasi model dalam penelitian ini dilakukan untuk mengukur tingkat performa dan akurasi dari arsitektur *MobileNetV2* dalam mengklasifikasikan citra daun tanaman obat di Kabupaten Rokan Hulu. Untuk memperoleh hasil evaluasi yang komprehensif, digunakan beberapa metrik pengukuran, antara lain akurasi untuk menilai proporsi prediksi yang benar, presisi untuk mengukur ketepatan prediksi positif, *recall* untuk menilai sensitivitas model dalam mendeteksi kelas yang relevan, serta *F1-score* sebagai rata-rata harmonis dari presisi dan *recall*. Selain itu, digunakan *confusion matrix* untuk memberikan gambaran visual mengenai distribusi klasifikasi antara kelas yang diprediksi dan kelas aktual. Evaluasi ini bertujuan untuk mengetahui sejauh mana model mampu menggeneralisasi data baru dan memastikan efektivitasnya dalam tugas klasifikasi citra daun tanaman obat.

3.7 Implementasi Model

Pada tahap ini, model *deep learning* telah dilatih mulai diterapkan dan diuji menggunakan *dataset* yang tersedia. Proses ini dilakukan untuk menilai bagaimana model bekerja dalam kondisi nyata serta memastikan bahwa hasil klasifikasinya sesuai dengan yang diharapkan. Implementasi dilakukan menggunakan *Google*

Colaboratory sebagai platform utama, di mana model dijalankan dan diuji. Dalam penelitian ini, perangkat keras yang digunakan untuk menjalankan model meliputi:

1. Perangkat keras (*hardware*) antara lain:

Processor : 11th Gen Intel(R) Core(TM) i3-1115G4 @
3.00GHz (4 CPU), ~3.0GHz

Memori : 4.00 GB

System Type : Windows 11 *Home Single Language* 64-bit (10.0,
Build 22631)

Hardisk : 500 GB

2. Perangkat Lunak (*software*) antara lain:

Sistem Operasi : *Windows 11*

Bahasa pemrograman : *Python*

Platform Eksperimen: *Google Colaboratory* sebagai lingkungan utama untuk melatih, menguji dan mengevaluasi model.

3.8 Kesimpulan dan Saran

Tahap akhir, penelitian akan menyimpulkan hasil eksperimen dan saran dari performa model yang telah dikembangkan. Kesimpulan diambil berdasarkan evaluasi yang telah dilakukan untuk menentukan apakah sistem klasifikasi Tanaman obat dengan arsitektur *MobileNetV2* Berbasis *Transfer Learning* cukup akurat dan dapat diimplementasikan secara efektif