

BAB II

TINJAUAN PUSTAKA

2.1 Buah Jeruk

Jeruk adalah tanaman buah tahunan yang berasal dari Asia, dengan Cina sebagai tempat asalnya. Di Indonesia, jeruk telah lama dibudidayakan, termasuk jenis jeruk manis dan keprok yang dibawa oleh Belanda dari Amerika dan Italia. Buah ini kaya vitamin C dan A, sehingga populer di masyarakat [17][18]. Sebagai komoditas hortikultura prioritas, jeruk memberikan keuntungan tinggi bagi petani, baik sebagai buah segar maupun bahan olahan, menjadikannya sumber pendapatan yang menjanjikan [19].

Jeruk menjadi pilihan favorit karena rasanya yang segar dan berbagai manfaat kesehatan. Selain itu, jeruk tidak tergantung musim, karena dapat berbuah dua kali dalam setahun dengan hasil panen yang melimpah, asalkan dirawat dengan baik untuk menghasilkan buah berkualitas. Jeruk memiliki kemampuan untuk tumbuh dan berbuah di berbagai ketinggian, baik di dataran rendah maupun tinggi, serta dapat ditanam di lahan sawah maupun tegalan. Prospek agribisnis jeruk sangat menjanjikan, didukung oleh potensi lahan yang meliputi jutaan hektar untuk perkebunan skala besar dengan memperhatikan kesesuaian agroklimat. Dengan pengelolaan intensif, produksi jeruk dapat meningkat signifikan. Selain itu, potensi pasar yang besar dipengaruhi oleh kenaikan pendapatan, pertumbuhan populasi, dan elastisitas pendapatan terhadap permintaan [20].

Citrus sp atau jeruk, termasuk tanaman angiospermae yang dibudidayakan secara konvensional atau melalui pemuliaan. Teknik fusi protoplas dikembangkan untuk mengatasi ketidakcocokan genetik, menyilangkan tanaman lewat sel somatik. Jeruk hasilnya perlu diuji ketahanan terhadap hama dan penyakit [21].

2.2 Penyakit Buah Jeruk

Salah satu tantangan utama dalam budidaya jeruk adalah serangan hama dan penyakit. Hama tanaman adalah organisme atau hewan yang, dalam jumlah tertentu, dapat menghambat pertumbuhan tanaman dan menyebabkan kerugian secara ekonomi. Penyakit tanaman sebagai ketidakmampuan tanaman untuk

menjalankan fungsi fisiologisnya akibat gangguan terus-menerus yang disebabkan oleh patogen atau faktor lain, sehingga memunculkan gejala penyakit. Sebagian besar patogen yang menyerang tanaman berasal dari kelompok jamur, bakteri, dan virus, yang berdampak pada penurunan hasil produksi [22].

Tanaman jeruk mudah terpapar berbagai penyakit yang dapat mengurangi kualitas dan kuantitas hasil produksi. bahkan menghancurkan perkebunan. Penyakit utama yang menyerang budidaya jeruk termasuk CVPD (*citrus vein phloem degeneration*), disebabkan oleh bakteri *Liberibacter asiatus* virus Tristeza, penyakit blendok akibat jamur *Botryodiplodia theobromae*, serta busuk pangkal batang akibat *Phytophthora citrophthora* [23]. Gambar 2.1 berikut ini merupakan beberapa gambar penyakit buah jeruk yang sering terjadi selama proses budidaya.



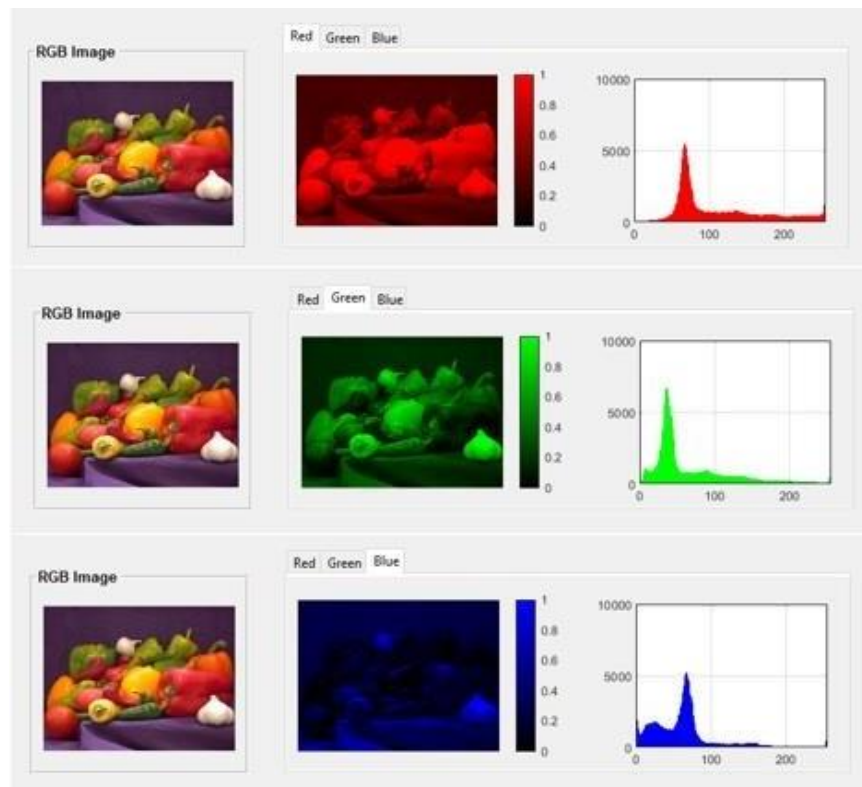
Gambar 2.1 Tanaman Jeruk Yang Terinfeksi Oleh L. Asiaticus [24]

Gambar 2.1 diatas merupakan gambar buah jeruk yang terinfeksi oleh L.Asiaticus. Kualitas buah dari tanaman jeruk yang terinfeksi oleh L. asiaticus menurun, menghasilkan buah yang lebih kecil dengan kandungan vitamin C yang berkurang [24].

2.3 Pengolahan Citra Digital

Pemrosesan citra atau *image processing* adalah teknik untuk mengolah citra menggunakan komputer agar kualitasnya meningkat. Tujuan dari pemrosesan ini adalah untuk mengurangi gangguan atau noise pada citra Sehingga lebih mudah dipahami dan dianalisis [26]. Citra digital adalah representasi dari objek dalam bentuk gambar yang diambil atau direkam, baik dalam bentuk foto, sinyal video analog seperti pada televisi, maupun dalam format digital yang dapat disimpan pada media penyimpanan [25]. Pada pengolahan citra digital kumpulan titik kecil yang dibentuk menjadi citra digital dan titik kecil tersebut merupakan sebuah piksel [27].

Suatu citra atau gambar bisa diartikan fungsikan dengan berukuran M baris dan N kolom, serta juga ada kordinat spasial serta amplitudo pada titik kordinat dan dinamakan sebagai intensitas tingkat keabuan. Koordinat spasial dan *value* amplitudo f seluruhnya diskrit dan berhingga maka citra tersebut merupakan citra *digital* [26]. Citra warna mempunyai 3 jenis bagian warna disetiap pikselnya yakni *red*, *green* dan *blue*. Warna yang berada pada setiap piksel hasil dari gabungan ketiga warna utama yakni merah hijau serta biru yang dikumpulkan pada ruang atau bidang warna. Untuk format *file* grafis itu sendiri *24 bit* dalam menyimpan citra warna dan yang bersumber dari komponen utama yaitu merah hijau biru dengan masing-masingnya bernilai *8 bit* [28] . Gambar 2.2 merupakan contoh RGB pengolahan citra digital.



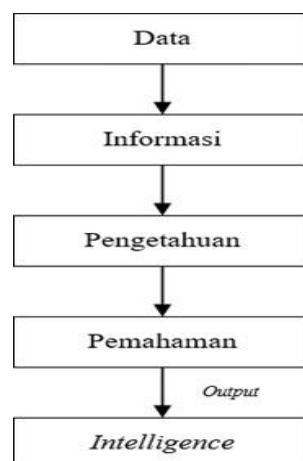
Gambar 2.2 RGB Pengolahan Citra Digital [28]

Gambar 2.2 merupakan contoh gambar dari pengolahan citra digital. Proses citra ini memiliki peran untuk dikenalnya suatu bentuk khusus yang diketahui lewat mesin [26]. Pemrosesan citra digunakan untuk mentrasformasikan citra

menjadi citra lain, seperti *image compression* menjadi proses *preprocessing* dari computer visi [28].

2.4 Kecerdasan Buatan

Kecerdasan Buatan dapat dijelaskan sebagai sistem atau perangkat yang dirancang untuk meniru kemampuan berpikir manusia dan melaksanakan tugas-tugas secara otomatis, yang sebelumnya memerlukan kecerdasan manusia [29]. Tujuan utama AI adalah untuk mengotomatisasi berbagai proses yang melibatkan keputusan atau analisis, yang sering dilakukan oleh manusia. Penggunaan AI kini telah meluas di berbagai sektor, termasuk pendidikan, kesehatan, ekonomi, dan pertanian [30]. Dengan kemajuan teknologi yang pesat, AI telah memberikan dampak signifikan pada kehidupan sehari-hari, menjadikannya bagian penting dalam berbagai aktivitas manusia. Cara kerja AI melibatkan pengumpulan data dengan cepat, pemrosesan berulang, serta penerapan algoritma yang cerdas untuk menghasilkan keputusan atau solusi yang lebih efisien dan akurat. Keberadaan AI semakin meningkatkan efisiensi dan produktivitas di berbagai bidang, mengubah cara kerja dan berinteraksi dalam kehidupan sehari-hari. [31]. Berikut Gambar 2.3 merupakan pola kecerdasan buatan.



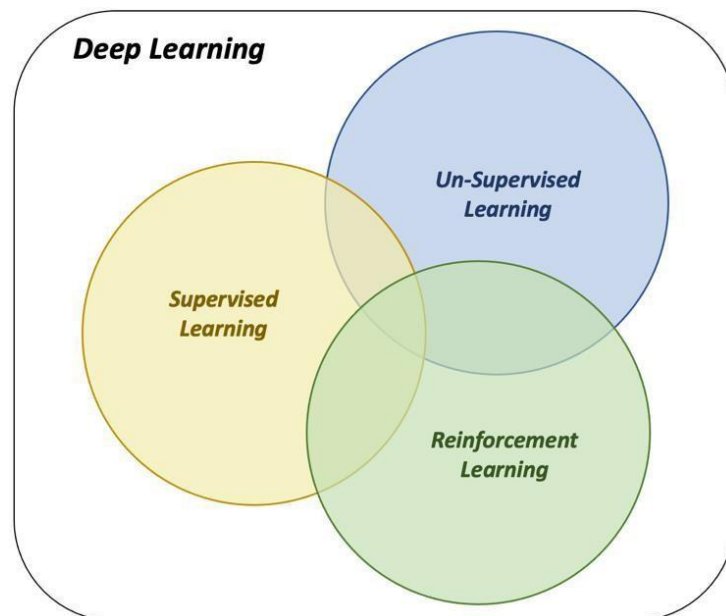
Gambar 2.3 Pola Kecerdasan Buatan [29]

Kecerdasan Buatan (AI) adalah bidang ilmu dan teknik yang bertujuan untuk mengembangkan mesin yang memiliki kemampuan cerdas, terutama dalam membuat program atau aplikasi komputer yang pintar. AI merupakan upaya untuk

menciptakan komputer, robot, atau aplikasi yang dapat beroperasi secara cerdas, mirip seperti cara kerja manusia [32].

2.5 Deep Learning

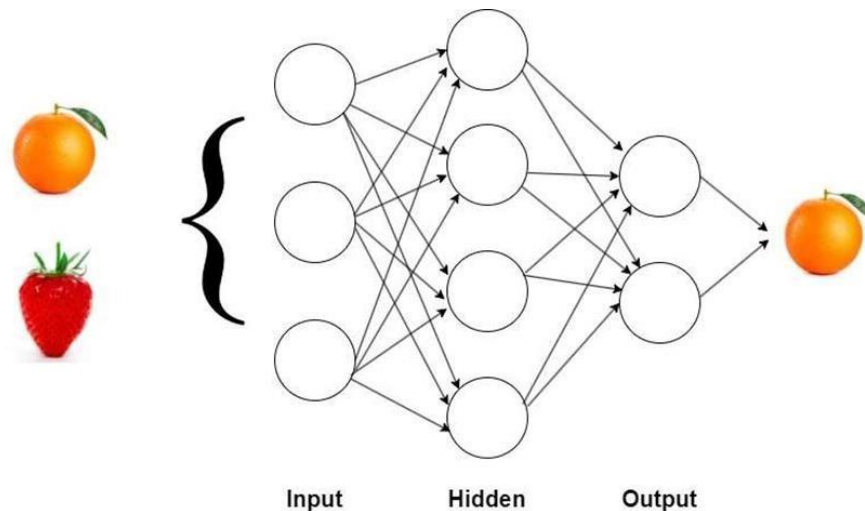
Deep learning (DL) merupakan bagian dari Machine Learning yang terinspirasi oleh struktur korteks otak manusia dengan memanfaatkan jaringan syaraf buatan yang memiliki banyak lapisan tersembunyi. Metode ini merupakan pengembangan dari *Convolutional Neural Networks (CNN)* yang memperbaiki teknik sebelumnya. Dengan model Deep Learning, sejumlah parameter yang tidak perlu dapat dikurangi, Selain itu deformasi gambar input seperti translasi, rotasi, dan skala dapat diatasi [29]. DL memanfaatkan beberapa lapisan di antara input layer dan output layer, yang memungkinkan pemrosesan non-linear melalui beberapa tahap. Hasil dari proses ini dapat digunakan untuk pembelajaran fitur (feature learning) dan klasifikasi pola (pattern classification) [30]. Gambar 2.4 Merupakan Kategori tiga kategori yang ada pada *Deep Learning*.



Gambar 2.4 Kategori Dalam *Deep learning* [30]

Deep learning (DL) merupakan cabang dari Machine Learning yang menggunakan beberapa lapisan pemrosesan non-linear untuk ekstraksi fitur dan transformasi data. Setiap lapisan menggunakan keluaran dari lapisan sebelumnya sebagai input. DL dapat diterapkan pada algoritma supervised atau unsupervised,

yang memungkinkan untuk digunakan dalam analisis pola tanpa pengawasan (unsupervised) dan klasifikasi dengan pengawasan (supervised). Proses ini memanfaatkan struktur bertingkat yang memungkinkan model untuk mengenali pola lebih kompleks dan melakukan prediksi dengan akurasi yang tinggi pada berbagai aplikasi, termasuk pengolahan citra dan analisis data [30]. Berikut Gambar 2.5 merupakan tahapan pada *Deep Learning*.



Gambar 2.5 Tahapan *Deep learning* [28]

Deep learning sangat efektif dalam tugas visi komputer, terutama dalam klasifikasi objek pada citra. Salah satu metode yang populer untuk klasifikasi ini adalah *Convolutional Neural Networks* (CNN) [31]. Perkembangan *Deep learning* saat ini telah didorong oleh banyaknya library dan API yang tersedia. *TensorFlow*, misalnya, adalah sebuah library yang memungkinkan ekspresi algoritma pembelajaran mesin dan eksekusi perintah dengan memanfaatkan informasi tentang objek yang ada, sehingga dapat membedakan dan mengklasifikasikan objek-objek tersebut. TensorFlow juga mendukung pelatihan model menggunakan *Central Processing Unit* (CPU) dan *Graphic Processing Unit* (GPU) [32].

2.6 *Convolutional Neural Network* (CNN)

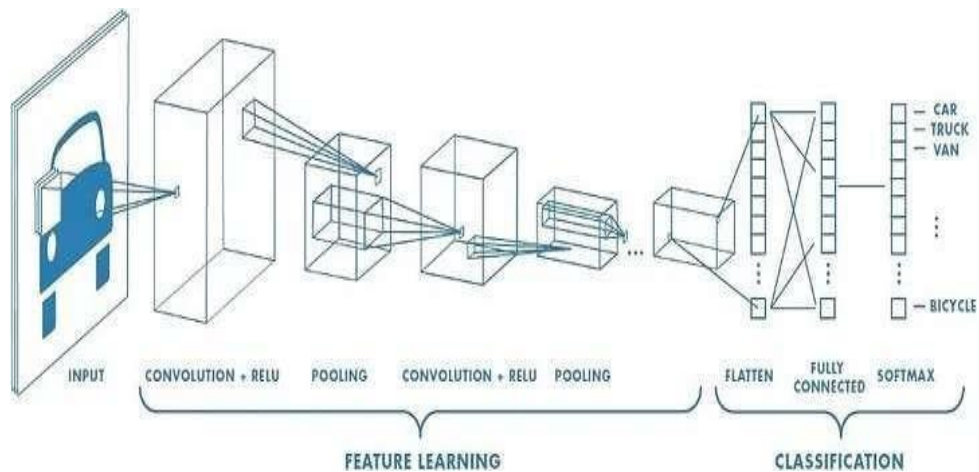
Deep learning telah memperkenalkan berbagai arsitektur yang dapat diterapkan dalam berbagai kasus, salah satunya adalah *Convolutional Neural Networks* (CNN) [33]. CNN pertama kali dikembangkan oleh Kunihiro Fukushima,

seorang peneliti dari NHK Broadcasting Science Research Laboratories di Tokyo, Jepang, dengan nama *Neo Cognitron*. CNN adalah pengembangan dari *Multilayer Perceptron* (MLP) yang dirancang khusus untuk memproses data dua dimensi, seperti gambar, dan lebih efisien dalam mengenali pola-pola kompleks [34]. CNN adalah salah satu jenis Deep Learning yang paling populer dan telah mencapai terobosan besar dalam dekade terakhir di berbagai bidang, mulai dari pengenalan pola dalam pemrosesan gambar, deteksi objek, hingga pengenalan wajah [33]. CNN diimplementasikan sebagai jaringan multi-layer yang dilatih untuk melakukan keputusan klasifikasi. Pendekatan-pendekatan dibangun untuk menjalankan tugas-tugas berbeda secara simultan seperti ekstraksi fitur, reduksi dimensi data, dan klasifikasi. Selain itu, *Convolutional* berfokus pada memisahkan dan mendeteksi pola-pola dalam gambar [35].

Convolutional Neural Networks (CNN) adalah jenis algoritma *deep learning* yang banyak digunakan dalam pemrosesan citra, penglihatan komputer, dan pengenalan pola. CNN memiliki struktur hierarkis yang terdiri dari beberapa lapisan, masing-masing melakukan transformasi tertentu. Lapisan pertama, yaitu lapisan konvolusional, berfungsi untuk mengekstraksi fitur dari gambar input dengan menerapkan filter kecil atau kernel. Kernel ini membantu mendeteksi fitur dasar seperti tepi, tekstur, dan bentuk. Melalui operasi konvolusi, CNN dapat mempelajari hubungan spasial antar piksel yang penting untuk memahami konten gambar. Setelah lapisan konvolusional, lapisan pooling digunakan untuk mengurangi dimensi spasial gambar, sehingga meningkatkan efisiensi komputasi tanpa kehilangan fitur penting. Selanjutnya, ada lapisan fully connected yang meratakan fitur yang telah dipooled dan menghubungkannya ke lapisan output akhir. Lapisan ini membantu model untuk membuat keputusan berdasarkan fitur yang diekstraksi, sering kali digunakan dalam tugas klasifikasi.

Keunggulan CNN terletak pada kemampuannya untuk secara otomatis mempelajari fitur hierarkis tanpa memerlukan ekstraksi fitur manual. Lapisan-lapisan pada CNN dilatih menggunakan backpropagation, di mana jaringan menyesuaikan bobotnya untuk meminimalkan kesalahan prediksi. CNN sangat efektif untuk tugas seperti pengenalan gambar, deteksi objek, dan segmentasi. Selain itu, CNN dapat menangani *Dataset* besar dan struktur gambar yang

kompleks, menjadikannya sangat penting di berbagai bidang seperti pencitraan medis, mobil otonom, dan pengenalan wajah. Dengan adanya framework *deep learning* seperti TensorFlow dan PyTorch, implementasi dan penerapan CNN menjadi lebih mudah dan praktis dalam aplikasi dunia nyata [39]. Gambar 2.6 merupakan Tahapan pada CNN.



Gambar 2.6 Tahapan CNN [39]

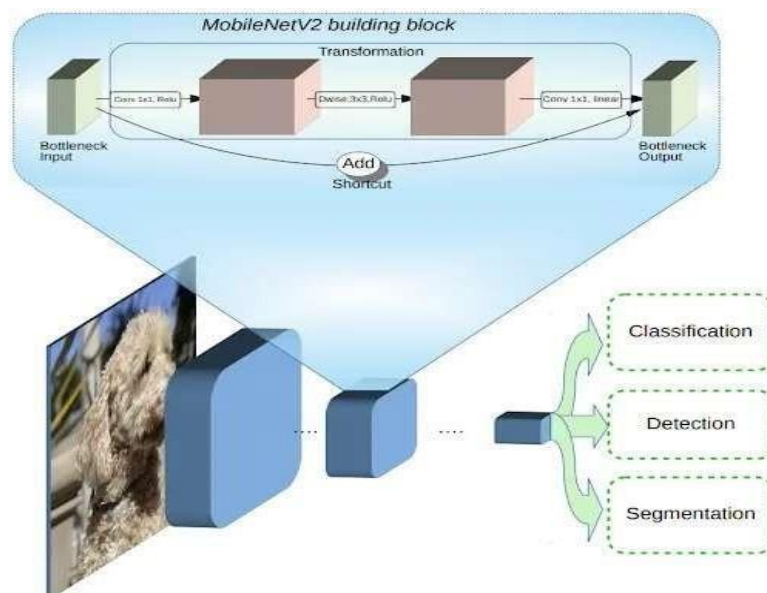
CNN berfungsi untuk membuat *feature map* dari input gambar sehingga gambar lebih dapat dikenali oleh computer [40]. CNN terdiri dari beberapa lapisan yang berbeda, termasuk lapisan input data, lapisan konvolusi, lapisan aktivasi *ReLU*, lapisan penyatuan, dan lapisan yang terhubung sepenuhnya. Lapisan-lapisan yang disebutkan di atas digunakan untuk mencapai ekstraksi fitur dan klasifikasi secara efektif [41].

2.7 MobileNet V2

MobileNetV2 adalah salah satu model *deep learning* terbaru yang sangat efisien, dirancang untuk mengatasi masalah penggunaan sumber daya komputasi yang besar tanpa mengorbankan akurasi. Berbeda dengan arsitektur CNN tradisional, MobileNetV2 menggunakan dua jenis konvolusi, yaitu *depthwise convolution* dan *pointwise convolution*. Penggunaan keduanya membuat MobileNetV2 lebih ringan dan lebih cepat dalam proses komputasi, memungkinkan model ini bekerja lebih baik pada perangkat dengan keterbatasan sumber daya, seperti perangkat mobile atau edge devices. Beberapa penelitian menunjukkan

bahwa MobileNetV2 dapat memberikan hasil klasifikasi gambar dengan tingkat akurasi yang sangat baik meskipun menggunakan lebih sedikit sumber daya dibandingkan dengan model CNN lainnya [42][43][44]. Keunggulan ini menjadikan MobileNetV2 sebagai pilihan utama dalam berbagai aplikasi pengolahan citra, terutama yang memerlukan efisiensi tinggi [45]. *MobileNetV2* memiliki modul modul lapisan baru, yaitu *inverted residual* dengan *linear bottleneck*, yang secara signifikan mengurangi memori yang dibutuhkan untuk pemrosesan [21].

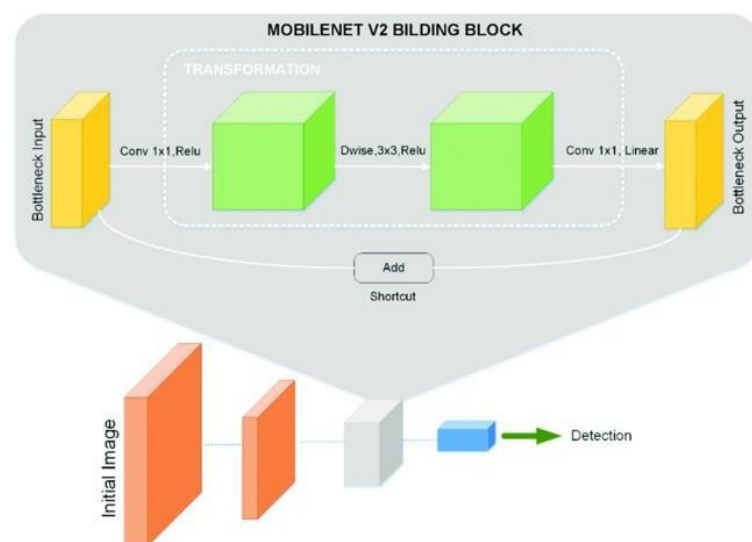
MobileNet adalah arsitektur CNN yang dirancang untuk mengatasi tantangan penggunaan sumber daya komputasi yang besar. Model ini telah dilatih menggunakan *Dataset* ImageNet untuk tugas deteksi dan klasifikasi gambar, dan diimplementasikan dengan menggunakan library TensorFlow [42]. Perbedaan utama antara MobileNet dan arsitektur CNN lainnya adalah penerapan lapisan konvolusi yang menggunakan filter yang lebih tipis, disesuaikan dengan ketebalan gambar input. MobileNet membagi konvolusi menjadi dua jenis, yaitu *depthwise convolution* dan *pointwise convolution*, yang memungkinkan model ini untuk bekerja lebih efisien tanpa mengorbankan kinerja [42]. Gambar 2.7 merupakan Tahapan pada *MobileNetV2*.



Gambar 2.7 Tahapan MobileNetV2 [44]

MobileNet adalah salah satu model yang diuji di TensorFlow untuk mempercepat penelitian di berbagai bidang. Model ini terus berkembang sejak

Inception v1 hingga Inception v3, dengan perbaikan pada struktur jaringan. Dalam penelitian ini, MobileNet terbukti efektif dalam klasifikasi gambar [44]. MobileNetV2, khususnya, memiliki kemampuan untuk mempelajari fitur yang lebih kompleks karena menggunakan lebih banyak lapisan konvolusi dengan ukuran filter yang lebih kecil, dibandingkan dengan model sebelumnya seperti AlexNet atau GoogleNet, yang memanfaatkan filter yang lebih besar untuk proses pembelajaran. Model ini lebih efisien dalam penggunaan sumber daya komputasi. *MobileNet* dioptimalkan untuk menjadi kecil dan efisien [41]. Gambar 2.8 merupakan arsitektur *MobileNetV2*.



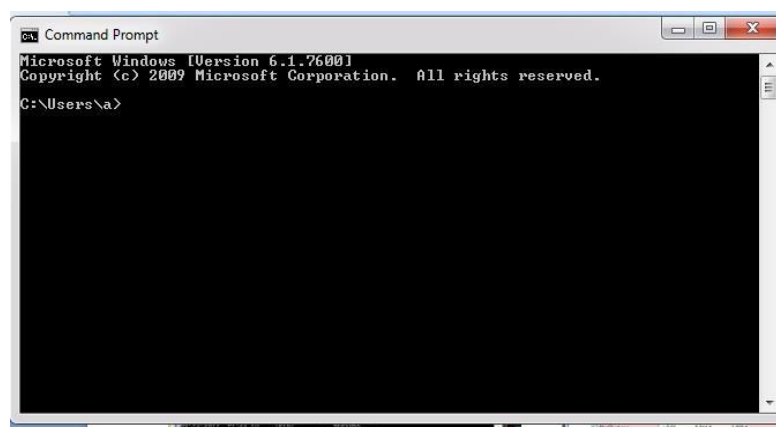
Gambar 2.8 Arsitektur MobileNetV2 [41]

Struktur utama *MobileNetV2* didasarkan pada versi sebelumnya, yaitu *MobileNetV1* dengan menambahkan *inverted residual* dengan modul *bottleneck linier* [41]. Arsitektur MobileNet didasarkan pada konvolusi depthwise yang dapat dipisahkan. Konvolusi 2D standar memproses semua saluran input secara langsung untuk menghasilkan satu saluran output dengan melakukan konvolusi dalam dimensi kedalaman saluran. Sedangkan pada konvolusi kedalaman, gambar input dan filter dipisahkan ke dalam saluran yang berbeda, kemudian setiap saluran input

dikonvolusi dengan saluran filter yang sesuai. Setelah saluran output yang telah difilter dihasilkan, saluran-saluran tersebut kemudian digabungkan kembali [21].

2.8 *Command Prompt*

Command Prompt, atau yang dikenal dengan nama `cmd.exe`, adalah aplikasi baris perintah yang digunakan pada sistem operasi Windows untuk menjalankan perintah secara langsung. Fungsinya sangat penting untuk mengotomatiskan berbagai tugas menggunakan skrip atau file batch, serta untuk menjalankan fungsi administratif dan pemecahan masalah. Di Windows, *Command Prompt* sering kali disebut sebagai Windows Command Processor dan memfasilitasi pengguna untuk mengakses berbagai perintah yang tidak tersedia dalam antarmuka grafis. Selain itu, *Command Prompt* juga membantu dalam memecahkan masalah teknis yang sering terjadi di sistem. Pada sistem operasi Linux, peran ini dipegang oleh pengguna dengan hak akses root yang memberikan kontrol penuh. Tampilan awal *Command Prompt* dapat diakses melalui aplikasi di menu Start, dan pengguna dapat memasukkan perintah untuk melakukan berbagai operasi sistem yang diperlukan.



Gambar 2.9 Tampilan *Command Prompt* [46]

Command prompt dapat membaca sekumpulan instruksi dari sebuah file. Dalam hal ini shell `cmd` bertindak sebagai penerjemah bahasa dan isi file dapat dianggap sebagai program yang sebenarnya. Ketika mengeksekusi program batch ini, tidak ada fase kompilasi perantara. Program-program ini biasanya dibaca, ditafsirkan dan dieksekusi baris demi baris. Karena tidak ada kompilasi, tidak ada produksi file yang dapat dieksekusi terpisah. Karena alasan ini, program-program

ini disebut *skrip batch* atau *skrip shell* [42]. *Command prompt* merupakan salah satu cara untuk mengakses dan mengontrol berbagai aspek dari sistem operasi *Windows* tanpa menggunakan antarmuka grafis pengguna (GUI) [44]. *Command prompt* adalah antarmuka baris perintah yang dapat digunakan untuk menjalankan berbagai perintah dan skrip, termasuk yang terkait dengan *machine learning*.

2.9 Efektivitas Sistem

Performansi subset fitur didapati lewat penilaian *accuracy*, *precision*, *recall* serta *F1-Score*. Ketiga metrik evaluasi ini dihitung dengan *confusion matrix* pada Tabel 2.1.

Tabel 2.1 *Confusion Matrix*

Kelas Sebenarnya	Hasil klasifikasi	
	Positif	Negatif
Positif	TP	FN
Negatif	FP	TN

Tabel 2.1 merupakan tampilan dari *confusion matrix* yang dimana memiliki keterangan *True Positive* (TP), *True Negative* (TN), *False Positive* (FP) dan *False Negative* (FN) didapati lewat tabel *confusion matrix*. TN ialah banyaknya lulus dengan tepat serta berhasilnya diprediksi dengan tepat waktu. TP ialah total lulus tidak tepat yang berhasil diperkirakan lulus tidak tepat. FP yaitu banyaknya lulus tepat waktu yang diperkirakan sebaliknya. Sedangkan FN adalah lulus tidak tepat waktu yang sudah dideteksi sebaliknya, Tabel 2.1 menunjukkan bagian yang ada pada *confusion matrix* dengan pembagian pada hasil klasifikasi dan kelas sebenarnya, *Specificity* atau *presisi* ialah ketrampilan jenis saat mengenali semua informasi lulus tidak tepat waktu dengan benar. *Accuracy* menyebutkan bahwa ketrampilan model saat menjalankan deteksi lulusan tepat serta tidak tepat waktu dengan baik. dan persamaan yang digunakan untuk menentukan nilai efektivitas sistem yaitu Persamaan (2.1) yang merupakan persamaan untuk mencari nilai *accuracy*.

$$\text{Accuracy} = \frac{\text{total TP}}{\text{total semua data}} \quad (2.1)$$

Berdasarkan Persamaan (2.1) *Accuracy* mengatakan ketrampilan model saat deteksi lulusan tepat serta tidak tepat waktu, dan untuk mencari nilai *precision* memakai Persamaan (2.2).

$$Precision = \frac{TP}{TP + FP} \quad (2.2)$$

Berdasarkan Persamaan (2.2) *Specificity* atau *presisi* yaitu ketrampilan mengetahui semua informasi lulus tidak tepat waktu dengan benar, untuk mencari nilai *sensitivity* atau *recall* memakai persamaan (2.3).

$$Recall = \frac{TP}{TP + FN} \quad (2.3)$$

Berdasarkan Persamaan (2.3) *sensitivity* atau *recall* dirincikan dengan tingkat kemampuan model untuk mengetahui semua informasi lulus tepat waktu dengan tepat, untuk mencari nilai *F1-score* menggunakan Persamaan (2.4).

$$F1-score = 2 \times \frac{Precision \cdot Recall}{Precision + Recall} \quad (2.4)$$

Berdasarkan Persamaan (2.4) *F1-score* yaitu metrik evaluasi yang digunakan dalam klasifikasi untuk mengukur keseimbangan antara *precision* dan *recall*. Nilai *F1-score* berkisar antara 0 hingga 1, hasil 1 menunjukkan hasil yang sempurna dan 0 menunjukkan hasil yang sangat buruk.

2.10 Kajian Pustaka

Penelitian ini mengajukan metode berbasis Convolutional Neural Network (CNN) untuk mendeteksi penyakit mata melalui citra fundus. Pendekatan yang digunakan mengadopsi transfer learning dengan dua komponen utama: base model dan head model. MobilenetV2 dipilih sebagai base model berdasarkan hasil eksperimen karena menunjukkan efisiensi tinggi dan akurasi yang unggul dibandingkan model CNN lain seperti ResNet50V2, InceptionV3, InceptionResNetV2, VGG16, dan VGG19. Uji coba dilakukan pada *Dataset* citra fundus yang berisi 601 gambar dengan klasifikasi ke dalam dua kategori, yaitu "Normal" dan "Abnormal". Hasil menunjukkan bahwa metode ini memberikan kinerja optimal meskipun *Dataset* relatif kecil, dengan akurasi mencapai 72% [26]. Penelitian ini menguji arsitektur CNN seperti ResNet 101, AlexNet, dan VGG-16-19 untuk mendeteksi parasit penyebab malaria. Hasilnya menunjukkan bahwa ResNet 101 unggul dengan akurasi 95,92%, membuktikan efektivitasnya dalam klasifikasi penyakit berbasis citra medis [21].

Penelitian ini bertujuan untuk mendeteksi malaria menggunakan algoritma pre-trained models seperti EfficientNet B0 dan MobileNetV2. Berdasarkan hasil pengujian, kedua algoritma ini dapat diandalkan untuk prediksi malaria. Namun, MobileNetV2 menunjukkan performa yang unggul dengan rata-rata akurasi mencapai 97,27%. Algoritma ini memiliki keunggulan dalam efisiensi pemrosesan

dan akurasi prediksi yang tinggi, menjadikannya pilihan yang lebih baik dibandingkan metode lainnya. Penelitian ini menegaskan potensi penggunaan MobileNetV2 dalam diagnosis malaria secara akurat dan efisien [41].

Penelitian ini bertujuan mendeteksi penyakit pada otak dengan memanfaatkan arsitektur CNN, khususnya MobileNetV2. klasifikasi menggunakan pemrosesan gambar tradisional dilakukan secara bertahap hingga akhirnya mengadopsi arsitektur MobileNetV2 yang dikenal memiliki akurasi tinggi. Arsitektur ini unggul dalam efisiensi dengan ukuran model dan jumlah parameter pelatihan yang kecil dan memberikan performa yang baik. Tahapan penelitian meliputi pengumpulan data, pemrosesan awal data (data preprocessing), klasifikasi, evaluasi, hingga implementasi. MobileNetV2 menunjukkan kemampuannya yang optimal dalam klasifikasi jenis tumor otak dengan menghasilkan akurasi sebesar 88,64%. Hal ini membuktikan keunggulan utama MobileNetV2 mencakup efisiensi komputasi dan akurasi tinggi meskipun menggunakan model yang relatif ringan. Dengan pendekatan ini, penelitian memberikan kontribusi signifikan terhadap penerapan teknologi berbasis CNN dalam bidang kesehatan, khususnya untuk mendeteksi penyakit otak dengan tingkat akurasi yang sangat memadai [50].

Penelitian ini melakukan klasifikasi penyakit pada daun teh dengan menggunakan gabungan tiga arsitektur CNN, yaitu GoogleNet, Xception, dan Inception ResNet V2. Ketiga arsitektur tersebut memanfaatkan *Dataset* yang sama untuk pengujian dan validasi, dengan menerapkan fungsi aktivasi ReLU serta Batch Size sebesar 10. *Dataset* yang digunakan mencakup 4.272 citra, yang dibagi dalam proporsi data training sebesar 80%, testing 10%, dan validasi 10%. Pendekatan ini berhasil menghasilkan akurasi akhir sebesar 89,64%. Penelitian ini menunjukkan efektivitas kombinasi arsitektur CNN dalam meningkatkan performa klasifikasi, khususnya pada penyakit daun teh, dengan memanfaatkan metode pengolahan data dan algoritma berbasis *deep learning* [43].