

## BAB II

### TINJAUAN PUSTAKA

#### 2.1. Kajian Pustaka

Penelitian terdahulu mengenai klasifikasi objek sudah banyak dilakukan yang Indonesia bahwa setiap pendekatan metode dan objek yang digunakan memiliki nilai akurasi yang berbeda. Salah satu algoritma yang cukup baik yang digunakan untuk klasifikasi objek adalah *Convolutional Neural Network* (CNN). Penggunaan data yang banyak membuat kompleksitas komputasi jaringan yang tinggi dan waktu pelatihan data yang lama.

Beberapa penelitian sebelumnya terkait klasifikasi citra diantaranya adalah penelitian yang dilakukan oleh Rima Dias Ramadhani, Afandi Nur Aziz Thohari, Condro Kartiko, Apri Junaidi, Tri Ginanjar Laksana, dan Novanda Alim Setya Nugraha mengenai Optimasi Akurasi Metode *Convolutional Neural Network* untuk Identifikasi Jenis Sampah. Penelitian ini menggunakan metode *Convolutional Neural Networks* (CNN) dengan melakukan optimasi *hyperparameter* pada arsitektur CNN. Terdapat tiga *hyperparameter* yang digunakan pada penelitian untuk menaikkan nilai akurasi model yaitu *dropout*, *padding*, dan *stride*. Hasil akurasi yang di dapat dari penelitian ini sebesar 67,6% sebelum dilakukan optimasi dan 91,2% setelah dilakukan optimasi [1].

Pada penelitian lain mengenai Pengenalan Sampah Plastik Dengan Model *Convolutional Neural Network* yang dilakukan oleh Irfan Nugraha Pratama, Tatang Rohana dan Tohirin Al Mudzakir. Penelitian ini dilakukan untuk mengklasifikasikan jenis sampah plastik (anorganik) menggunakan algoritma CNN dengan menggunakan model *unsupervised learning*. Penelitian ini mendapatkan nilai akurasi sebesar 80% pada iterasi ke 6000 dengan nilai rata-rata kerugian mencapai 0,03 dengan pembagian dataset untuk pelatihan sebesar 80% dan 20% untuk pengujian.

Penelitian ini menggunakan data berjumlah 300 gambar, dengan 88 gambar botol plastik, 57 gambar gelas plastik dan 61 gambar kantong plastik, serta 94 gambar bukan sampah plastik [6].

Penelitian lain yang dilakukan oleh Widi Hastomo, Sugiyanto dan Sudjiran mengenai *Convolution Neural Network* Arsitektur *MobileNetV2* Untuk Mendeteksi Tumor Otak yang dilakukan yang dibangun menggunakan CNN dengan arsitektur *MobileNet-V2* dengan menggunakan data sebanyak 2870 yang diklasifikasikan menjadi 4 kelas yaitu *glioma*, *meningioma*, *no\_tumor* dan *pituitary*. Pada penelitian ini menghasilkan akurasi yang sangat baik yaitu sebesar 94%, dengan uraian akurasi setiap kelas adalah *glioma* (99%), *meningioma* (85%), *no\_tumor* (99%) dan *pituitary* (96%) [15].

Pada penelitian lain tentang “*Face Mask Detection Using MobileNetV2 in The Era of COVID-19 Pandemic*” oleh Samuel Ady Sanjaya, dan Suryo Adi Rakhmawan. Model yang dibangun menggunakan *MobileNetV2* yang dapat mendeteksi orang yang memakai masker dan tidak memakainya dengan akurasi 96,85% dengan *epoch* 20. Penelitian ini menggunakan 1.916 data dengan masker dan 1.930 tanpa masker. Penelitian ini menggunakan 75% data *training* dan 25% data *testing*. [20].

Pada penelitian lain yang dilakukan oleh Metin Akay, Yong Du, Cheryl L Sershen, Minghua Wu, Ting Y Chen, Shervin Assassi, Chandra Mohan, dan Yasemin M. Akay mengenai “*Deep Learning Classification of Systemic Sclerosis Skin using the MobileNetV2 Model*”. Hasil dari penelitian tersebut didapatkan bahwa arsitektur *MobileNetV2* mencapai akurasi 100% pada set citra pelatihan, 96,8% pada set validasi, dan 95,2% pada set citra pengujian. Dengan menggunakan arsitektur ini waktu yang dibutuhkan untuk melakukan pelatihan kurang dari 5 jam dengan 100 *epoch*, arsitektur jaringan tersebut mudah untuk diimplementasikan, sederhana, murah, dan akurat [19]

Penelitian yang dilakukan oleh Wei-Lung Mao, Wei-Chun Chen, Chien-Tsung Wang, dan Yu-Hao Lin mengenai “*Recycling waste*

*classification using optimized convolutional neural network*". Hasil penelitian didapatkan akurasi tertinggi 99,6%. Dataset yang digunakan yaitu *TrashNet* terdiri dari 2527 gambar yang dibagi menjadi enam kategori. Kategori dataset tersebut antara lain kardus, kaca, logam, kertas, plastik, dan sampah. Penelitian ini menggunakan 90% data *training* dan 10% data *testing*. Penelitian ini menggunakan model *DenseNet121* dan menggunakan GA untuk mengoptimalkan klasifikasi. [9].

Penelitian selanjutnya adalah penelitian yang dilakukan oleh Shanshan Meng dan Wei-Ta Chu mengenai "*A Study of Garbage Classification with Convolutional Neural Networks*". Penelitian ini menggunakan model *support vector machine* (SVM) dengan fitur HOG, *simple convolutional neural network* (CNN), dan CNN dengan *residual block*. Terdapat 6 kategori daur ulang: kardus, kaca, logam, kertas, plastik, dan sampah. Berdasarkan hasil penelitian ini, *simple CNN* dengan atau tanpa *residual block* menunjukkan kinerja yang cukup baik. Dengan hasil akurasi menggunakan SVM+HOG sebesar 47,25%, Simple CNN sebesar 93,75%, ResNet50 sebesar 93,35%, dan HOG+CNN sebesar 93,56%. Nilai *epoch* yang digunakan adalah 40 dengan menggunakan 9095 data *training* dan 1013 data *testing*. [18].

Ringkasan penelitian yang relevan ditunjukkan pada Tabel 2.1

Tabel 2. 1 Penelitian terdahulu

No.	Judul	<i>Comparing</i>	<i>Contrasting</i>	<i>Criticize</i>	<i>Synthesize</i>	<i>Summarize</i>
1.	Optimasi Akurasi Metode <i>Convolutional Neural Network</i> untuk Identifikasi Jenis Sampah [1]	Pada penelitian ini dilakukan optimasi penggunaan metode CNN untuk mendapatkan hasil yang akurat dalam mengidentifikasi jenis sampah. Optimasi dilakukan dengan menambahkan <i>hyperparameter</i> pada arsitektur CNN. <i>Hyperparameter</i> yang digunakan yaitu <i>dropout</i> , <i>padding</i> , dan <i>stride</i> . Data yang digunakan pada penelitian ini diambil dari objek sampah organik dan anorganik dengan masing masing sebanyak 840 data gambar yang didapat	Penelitian ini dilakukan untuk mencari hasil akurasi yang akurat dalam mengidentifikasi jenis sampah. Dilakukan optimasi pada arsitektur CNN dengan menambahkan <i>hyprerparamete</i> . <i>Hyperparameter</i> yang digunakan yaitu <i>dropout</i> , <i>padding</i> dan <i>stride</i> . Penambahan <i>dropout</i> 20% digunakan untuk meningkatkan akurasi dan menghindari <i>overfitting</i> pada saat	Tidak ditunjukkan pengujian pada aplikasi.	Penelitian dilakukan dengan pengumpulan data objek sampah yang diambil dari <i>google image</i> , kemudian dilakukan <i>preprocessing</i> . Setelah itu dilakukan pemodelan CNN, dan kemudian dilakukan pelatihan tanpa optimasi dan optimasi dengan menambahkan <i>hyperparameter</i> . Kemudian dilakukan evaluasi untuk melihat perbandingan hasil dari identifikasi jenis sampah	Penelitian dengan teknik optimasi pada arsitektur CNN dengan menambahkan <i>hyperparameter</i> pada arsitektur CNN menghasilkan kenaikan akurasi sebesar 91,2%, dari nilai akurasi sebelum dilakukan optimasi sebesar 67,6%.

Tabel 2.1 Penelitian terdahulu (Lanjutan)

No.	Judul	<i>Comparing</i>	<i>Contrasting</i>	<i>Criticize</i>	<i>Synthesize</i>	<i>Summarize</i>
		dari <i>Google image</i> . Kemudian dilakukan analisis perbandingan antara sebelum dan sesudah dilakukan optimasi	pelatihan. Penambahan <i>padding</i> dan <i>stride</i> digunakan untuk mempercepat waktu pelatihan.		sebelum di optimasi dan sesudah dioptimasi.	
2.	Pengenalan Sampah Plastik Dengan Model <i>Convolutional Neural Network</i> [6].	Penelitian ini dilakukan untuk mengenali sampah menggunakan model <i>unsupervised learning</i> CNN. Data yang digunakan merupakan data gambar sampah plastik yang diambil menggunakan kamera smartphone. Gambar yang digunakan sebanyak 300 gambar, dengan 88 gambar botol plastik, 57 gambar gelas plastik, 61 gambar kantong plastik serta 94 gambar yang bukan	Peneliti ini dilakukan untuk mengimplementasi metode <i>Convolutional Neural Network</i> (CNN) pada klasifikasi jenis sampah plastik anorganik. Dalam pelatihan model, ada tiga subdivisi <i>regional</i> utama dalam <i>output</i> set data pelatihan. Yaitu, <i>region</i> 82, yang menggunakan <i>mask</i> lebih besar tetapi dapat	Tidak dilakukan augmentasi untuk memperbanyak data.	Penelitian dilakukan dengan menggunakan model <i>unsupervised learning</i> CNN. Data yang digunakan sebanyak 300 data gambar. Kemudian dataset dibagi menjadi 80% data <i>training</i> dan 20% data <i>testing</i> . Kemudian dilakukan <i>training</i> , <i>testing</i> dan evaluasi.	Pada penelitian ini mendapatkan nilai akurasi sebesar 80% pada iterasi ke 6000 dengan nilai rata-rata kerugian mencapai 0,03 dengan pembagian dataset untuk pelatihan sebesar 80% dan 20% untuk pengujian.

Tabel 2.1 Penelitian terdahulu (Lanjutan)

No.	Judul	<i>Comparing</i>	<i>Contrasting</i>	<i>Criticize</i>	<i>Synthesize</i>	<i>Summarize</i>
		objek penelitian. Kemudian dataset dibagi menjadi 80% data <i>training</i> dan 20% data <i>testing</i> .	memprediksi objek lebih kecil, dan <i>region</i> 94, di mana skala prediksi tidak terlalu besar atau kecil tapi menengah, dan menggunakan <i>mask</i> yang sedang. Pada <i>region</i> 106, yang menggunakan <i>mask</i> kecil, dapat memprediksi objek dalam skala yang lebih besar.			
3.	<i>Convolution Neural Network</i> Arsitektur <i>Mobilenet-V2</i> Untuk Mendeteksi Tumor Otak [15]	Penelitian ini dilakukan untuk mendeteksi tumor otak menggunakan metode CNN dengan arsitektur <i>MobilNetV2</i> . Dataset yang digunakan pada penelitian diambil dari hasil <i>CT Scan</i> dengan data sebanyak 2870	Penelitian ini dilakukan untuk mendeteksi tumor otak untuk menekan tingkat kematian penyakit ini. Data yang digunakan pada penelitian ini diambil dari hasil <i>CT Scan</i> dengan data sebanyak 2870	Pada tahap penelitian tidak dilakukan tahap <i>preprocessing</i> .	Penelitian dilakukan menggunakan dataset sebanyak 2870 data gambar yang diklasifikasikan kedalam 4 kelas. Kemudian dilakukan <i>training</i> model menggunakan.	Pada penelitian ini menghasilkan akurasi yang sangat baik yaitu sebesar 94%, dengan uraian akurasi setiap kelas adalah <i>glioma</i> (99%),

Tabel 2.1 Penelitian terdahulu (Lanjutan)

No.	Judul	<i>Comparing</i>	<i>Contrasting</i>	<i>Criticize</i>	<i>Synthesize</i>	<i>Summarize</i>
		yang diklasifikasikan dalam 4 kelas.	data gambar tumor otak yang diklasifikasikan dalam 4 kelas.		arsitektur <i>MobileNetV2</i> , kemudian dilakukan pengujian dan dilakukan evaluasi model menggunakan <i>confusion matrix</i>	<i>meningioma</i> (85%), <i>no_tumor</i> (99%) dan <i>pituaty</i> (96%).
4.	<i>Face Mask Detection Using MobileNetV2 in The Era of COVID-19 Pandemic</i> [20]	Penelitian ini dilakukan untuk mendeteksi penggunaan masker dengan menggunakan <i>MobileNetV2</i> . Data yang digunakan pada penelitian ini adalah 1916 data dengan masker dan 1930 data tanpa masker. Kemudian dilakukan pemotongan gambar sehingga hanya menampilkan objek wajah saja. Hasil dari penelitian di	Penelitian ini dilakukan untuk mendeteksi penggunaan masker untuk membantu pemerintah dalam melakukan beberapa tindakan pencegahan, mitigasi, dan evaluasi program. Pada penelitian ini dilakukan 2 eksperimen yaitu, melakukan pelatihan dengan menggunakan	Pada penelitian ini validasi dinilai berdasarkan persentase orang yang memakai masker di tempat umum menurut kota yang dikorelasikan dengan indeks kewaspadaan COVID-19	Penelitian ini dilakukan dengan pengumpulan data, kemudian dilakukan <i>preprocessing</i> . <i>Preprocessing</i> dilakukan dengan memotong gambar sehingga hanya menampilkan objek wajah, mengubah ukuran gambar menjadi 224 x 224 pixel dan melakukan augmentasi data. Kemudian data dipisah menjadi 3	Penelitian ini menghasilkan akurasi pada model yang dibangun sebesar 96,85% menggunakan 20 <i>epoch</i> , dan ada evaluasi model mendapatkan akurasi <i>F1-Score</i> sebesar 91%.

Tabel 2.1 Penelitian terdahulu (Lanjutan)

No.	Judul	<i>Comparing</i>	<i>Contrasting</i>	<i>Criticize</i>	<i>Synthesize</i>	<i>Summarize</i>
		implementasikan dalam video yang kemudian di dilakukan pengujian pada citra data yang diperoleh dari berbagai sumber di 25 kota di Indonesia.	dataset <i>Kaggle</i> dan dataset <i>Real-World Masked Face</i> (RMFD), dan menerapkan model pada dataset dari 25 kota di Indonesia yang diambil dari CCTV, toko, dan kamera lampu lalu lintas pada tempat tertentu.	sehingga akan ada kemungkinan beban respon pada segmentasi tertentu karena penelitian hanya mencakup penangkapan pada citra di tempat tertentu. Sehingga dibutuhkan pengukuran atau indeks lain yang lebih valid.	yaitu 75% data <i>training</i> dan 25% data <i>testing</i> . Langkah selanjutnya dilakukan perancangan model menggunakan arsitektur MobileNetV2. Kemudian dilakukan evaluasi model dan penerapan model pada dataset yang diambil dari 25 kota di Indonesia	

Tabel 2.1 Penelitian terdahulu (Lanjutan)

No.	Judul	<i>Comparing</i>	<i>Contrasting</i>	<i>Criticize</i>	<i>Synthesize</i>	<i>Summarize</i>
5.	<i>Deep Learning Classification of Systemic Sclerosis Skin using the MobileNetV2 Model</i> [19]	Penelitian ini digunakan untuk mengklasifikasikan karakteristik kulit <i>Systemic sclerosis</i> (SSc) dengan menggunakan arsitektur CNN dengan model pelatihan <i>MobileNetV2</i> . Data yang digunakan sebanyak 1887 data gambar (1041 normal, 423 awal dan 423 tahap akhir SSc) sesudah augmentasi. Kemudian dilakukan perbandingan kinerja dari arsitektur CNN dan arsitektur <i>MobileNetV2</i> .	Penelitian ini digunakan untuk mencari pendekatan metode yang cepat dan handal untuk menilai tingkat keparahan SSc. <i>Training</i> dilakukan dengan 2 arsitektur yaitu arsitektur CNN dan arsitektur <i>MobileNetV2</i> .	Tidak disebutkan jumlah data gambar sebelum dilakukan augmentasi. Tahapan penelitian kurang jelas.	Pada penelitian ini dilakukan dengan mempersiapkan arsitektur yang akan digunakan. Penelitian ini menggunakan arsitektur <i>MobileNetV2</i> . Kemudian dilakukan <i>preprocessing</i> data gambar dengan mengubah ukuran menjadi 128 x 128 <i>pixel</i> dengan format jpg, dan melakukan augmentasi gambar. Kemudian dilakukan pelatihan model. Kemudian dilakukan perbandingan kinerja dari arsitektur CNN dan	Penelitian ini menghasilkan akurasi pada arsitektur <i>MobileNetV2</i> mencapai 100% pada set citra pelatihan, 96,8% pada set citra validasi, dan 95,2% pada set citra pengujian. Waktu pelatihan kurang dari 5 jam. Sedangkan pada arsitektur CNN memperoleh akurasi mencapai 100% pada set citra pelatihan, 87,7% pada set.

Tabel 2.1 Penelitian terdahulu (Lanjutan)

No.	Judul	<i>Comparing</i>	<i>Contrasting</i>	<i>Criticize</i>	<i>Synthesize</i>	<i>Summarize</i>
					arsitektur <i>MobileNetV2</i>	citra validasi, dan 82,9% pada set citra pengujian. Membutuhkan waktu selama 14 jam untuk melatih arsitektur CNN. Dengan demikian klasifikasi kulit SSc dengan menggunakan arsitektur <i>MobileNetV2</i> mendapatkan hasil akurasi yang tinggi dengan waktu pelatihan yang lebih singkat dibandingkan dengan arsitektur CNN

Tabel 2.1 Penelitian terdahulu (Lanjutan)

No.	Judul	<i>Comparing</i>	<i>Contrasting</i>	<i>Criticize</i>	<i>Synthesize</i>	<i>Summarize</i>
6.	<i>Recycling waste classification using optimized convolutional neural network</i> [9]	Pada penelitian ini dilakukan optimasi pada model CNN yaitu DenseNet121. Optimasi dilakukan dengan penambahan data dan menambahkan <i>Genetic algorithm</i> (GA) pada lapisan klasifikasi dari DenseNet121. Dataset yang digunakan dari dataset <i>benchmark</i> , yaitu <i>TrashNet</i> yang terdiri dari 2527 data gambar dengan enam kategori yaitu kardus, kaca, logam, kertas, plastik, dan sampah. Kemudian dilakukan perbandingan kinerja dengan model CNN yang berbeda berdasarkan dataset <i>TrashNet</i> .	Penelitian ini dilakukan untuk mencari hasil akurasi yang lebih akurat dalam mengklasifikasikan sampah. Dilakukan optimasi dengan menggunakan GA untuk mengoptimalkan lapisan <i>DenseNet121</i> . GA digunakan untuk menyempurnakan hyperparameter dari <i>fully-connected layer</i> dari <i>DenseNet121</i>		Penelitian ini dilakukan dengan memilih <i>hyperparameter</i> dari <i>fully-connected layer</i> pada DenseNet sebagai variabel keputusan, kemudian membuat kromosom dan inialisasi populasi. Kemudian dilakukan operasi GA yang meliputi crossover, mutasi, dan perhitungan <i>fitness</i> . Kemudian dilakukan evaluasi kinerja CNN dengan menggunakan <i>confusion matrix</i> .	Penelitian dengan teknik optimasi pada arsitektur CNN dengan menambahkan GA pada lapisan klasifikasi <i>DenseNet121</i> arsitektur CNN menghasilkan kenaikan akurasi sebesar 99.60%.

Tabel 2.1 Penelitian terdahulu (Lanjutan)

No.	Judul	<i>Comparing</i>	<i>Contrasting</i>	<i>Criticize</i>	<i>Synthesize</i>	<i>Summarize</i>
7.	<i>A Study of Garbage Classification with Convolutional Neural Networks</i> [18]	Penelitian ini dilakukan untuk mengidentifikasi objek sampah tunggal dan mengklasifikasikannya kedalam kategori sampah daur ulang. Model yang digunakan yaitu <i>support vector machine</i> dengan fitur HOG, <i>simple CNN</i> , dan CNN dengan <i>residual block</i> . Dataset yang digunakan di ambil dari kaggle dengan total 2527 gambar yang terdiri dari enam kategori yaitu kardus, kaca, logam, kertas, plastic dan sampah. Hasil penelitian kemudian dilakukan perbandingan kinerja.	Penelitian ini dilakukan untuk mencari hasil akurasi yang terbaik dari beberapa model yang digunakan, antara lain <i>support vector machine</i> (SVM) dengan fitur HOG, <i>simple convolutional neural network</i> (CNN), dan CNN dengan residual block. Penelitian ini juga dilakukan optimasi pada model CNN dengan menggunakan pengoptimalan <i>Adam</i> dan <i>Adadelta</i>	Dalam penelitian ini diperlukan variasi kategori sampah pada kehidupan nyata agar ketika mendapatkan gambar objek yang memiliki background tidak polos menghasilkan akurasi yang baik.	Penelitian ini dilakukan augmentasi data kemudian membagi data menjadi 9095 data <i>training</i> dan 1013 data <i>testing</i> . Kemudian dilakukan perancangan dan pelatihan model HOG+SVM, Simple CNN dan Resnet50. Setelah itu dilakukan evaluasi model.	Penelitian ini menghasilkan akurasi pada model SVM +HOG sebesar 47,25%, Simple CNN sebesar 93,74%, Resnet50 sebesar 95,35% dan HOG+CNN sebesar 93,56%. Klasifikasi citra sampah dengan teknik <i>deep learning</i> dengan kombinasi CNN menghasilkan akurasi yang cukup tinggi.

Berdasarkan beberapa penelitian sebelumnya mengenai klasifikasi objek menggunakan algoritma *Convolutional Neural Network* (CNN) mendapatkan akurasi yang tinggi. Maka, Algoritma CNN sudah terbukti sangat baik digunakan untuk klasifikasi objek citra. Penggunaan arsitektur *MobileNetV2* juga dapat meningkatkan akurasi dan mengatasi kompleksitas komputasi jaringan yang tinggi dan waktu pelatihan data yang lama [19]. Dengan demikian penelitian ini akan menerapkan algoritma CNN untuk klasifikasi sampah menggunakan arsitektur *MobileNetV2*.

## **2.2. Dasar Teori**

### **2.2.1. Sampah**

Sampah merupakan barang yang tidak memiliki nilai seperti barang rusak, atau barang yang tidak digunakan lagi yang sebagian besar berasal dari rumah tangga [1]. Berdasarkan jenisnya, sampah diklasifikasikan menjadi 3 jenis yaitu sampah organik, sampah anorganik dan sampah berbahaya. Sampah organik merupakan sampah yang bisa terurai dengan mudah secara alami sedangkan sampah anorganik adalah sampah yang sulit terurai secara alami dan membutuhkan waktu yang lama, jika sampah anorganik tertimbun dalam tanah dalam waktu lama, akan menyebabkan rusaknya lapisan tanah. Sampah Berbahaya adalah sampah yang mudah terbakar, mencemari lingkungan dan membahayakan kesehatan manusia. [4].

Sampah organik merupakan sampah yang mengandung senyawa organik dan dapat terurai oleh mikroorganisme seperti sisa makanan, kayu, ranting pohon, kayu dan daun – daun kering dan sebagainya, yang ditunjukkan pada gambar 2.1. Selain itu, sampah organik biasanya berwarna hijau atau coklat, dan bentuknya tidak beraturan.



Gambar 2. 1 Sampah Organik [21]

Sampah anorganik mengandung zat-zat yang bersifat anorganik dan sulit diuraikan oleh mikroorganisme. Sampah anorganik yang tertimbun dalam tanah dapat menyebabkan pencemaran tanah, yaitu rusaknya lapisan tanah. Sampah anorganik misalnya karton, kaca, logam, plastik, dan kertas yang ditunjukkan pada gambar 2.2. Selain itu, sampah anorganik biasanya berwarna putih atau biru, bentuk padat atau lebih solid [1][3].



Gambar 2. 2 Sampah Anorganik [22]

Sampah berbahaya atau B3 merupakan sampah yang mudah terbakar, membahayakan kesehatan manusia dan mencemari lingkungan [4]. Sampah berbahaya berasal dari sisa usaha dan/atau kegiatan yang mengandung B3. Sampah B3 dihasilkan dari

kegiatan baik dari sektor industri, pelayanan kesehatan, pariwisata atau dari domestik rumah tangga [23]. Salah satu sampah B3 adalah batu baterai (gambar 2.3) yang memiliki sifat korosif, dapat menyebabkan iritasi pada kulit dan pengkaratan pada baja [24].



Gambar 2. 3 Batu baterai[25]

### 2.2.2. *Machine Learning*

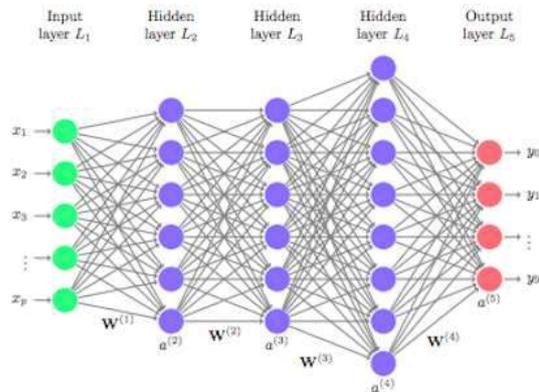
*Machine Learning* adalah bidang kecerdasan buatan (*Artificial Intelligence*) yang dapat belajar dari data, tanpa mengikuti instruksi yang diprogram secara eksplisit. *Machine Learning* berfokus pada pengembangan program komputer yang dapat mengajarkan dirinya sendiri untuk tumbuh dan berubah dengan data baru [7]. Prinsip dasar *machine learning* adalah menggunakan data untuk membuat model statistik dan membuat prediksi masa depan berdasarkan data *input* masa lalu atau mempelajari pola yang terdapat di dalam data. Kemampuan modifikasi serta adaptasi dalam merespon perubahan data merupakan keunggulan dari *machine learning* [26].

Secara umum, algoritma *machine learning* terbagi dalam tiga kategori yaitu, *Supervised learning*, *unsupervised learning* dan *Reinforcement learning*. Pada algoritma *Supervised learning* digunakan untuk mempelajari *mapping function* antara input dengan output. Berbagai kemungkinan *output* sudah diketahui dan

data yang digunakan untuk pelatihan sudah diberi label jawaban yang benar. Sedangkan pada *unsupervised learning* menggunakan kumpulan data pelatihan yang tidak berlabel untuk memodelkan struktur data dan membuatnya lebih subjektif. *Reinforcement learning* merupakan metode yang dipengaruhi oleh feedback dari lingkungan dengan teknik learning yang berulang – ulang (*iterative*) dan menyesuaikan (*adaptive*), yang dipercaya mendekati manusia belajar [27].

### **2.2.3. Deep Learning**

*Deep Learning* merupakan cabang *machine learning* berbasis Jaringan Syaraf Tiruan atau *Artificial Neural Network (ANN)* [28]. *Deep Learning* memiliki banyak lapisan yang membentuk tumpukan, lapisan tersebut adalah sebuah algoritma pemodelan abstraksi tingkat tinggi pada data, dengan fungsi transformasi *non-linear* yang berlapis dan mendalam, lapisan pada *deep learning* dapat mencapai hingga ratusan lapisan [29]. *Deep learning* memiliki 3 lapisan yaitu *input layer*, *hidden layer* dan *output layer* ditunjukkan pada gambar 2.4. Pada *input layer*, vektor input  $x = (x_1, x_2, \dots, x_p)$  diberikan ke sistem. Di lapisan output, merupakan hasil perhitungan dari fungsi aktivasi yang ada pada hidden layer. *Hidden Layer* terletak di antara lapisan input dan lapisan output [30]. Algoritma pada *deep learning* dapat digunakan pada pembelajaran terarah (*supervised learning*), pembelajaran tak terarah (*unsupervised learning*) dan semi terarah (*semi-supervised learning*), yang dapat digunakan juga di berbagai aplikasi, seperti pengenalan citra maupun suara, mengklasifikasi teks, dan lain – lain [31].

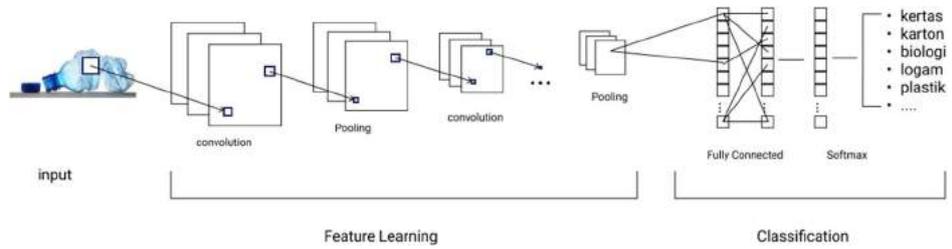


Gambar 2. 4 Layer Pada *Deep Learning* [30]

#### 2.2.4. Convolutional Neural Network (CNN)

*Convolutional Neural Network* (CNN) merupakan salah satu penerapan *Deep Learning*. Dilihat dari arsitekturnya, CNN termasuk ke dalam kelas *deep feedforward Artificial Intelligence* [7]. *Convolutional Neural Network* adalah neural network yang didesain untuk mengolah data dua dimensi [11]. Metode CNN terdiri dari dua metode yaitu klasifikasi yang menggunakan *feedforward* dan tahap pembelajarannya menggunakan *backpropagation*. Prinsip kerja dari metode ini mempunyai kesamaan pada metode MLP, tetapi di dalam metode CNN setiap neuronnya disajikan dalam bentuk dua dimensi yang mana tidak sama seperti pada metode MLP yang setiap neuron hanya memiliki ukuran satu dimensi [32].

CNN mempunyai beberapa layer yang digunakan untuk melakukan filter pada setiap prosesnya [33]. Arsitektur CNN terdiri dari, satu lapis *input layer*, satu lapis *output layer*, dan beberapa *hidden layers*. Lapis tersembunyi pada umumnya berisi *convolutional layers*, *pooling layers*, *ReLU layer*, *normalization layers*, *fully connected layers*, dan *loss layer*. Semua lapisan tersebut disusun secara bertumpuk, seperti *sandwich* [7].

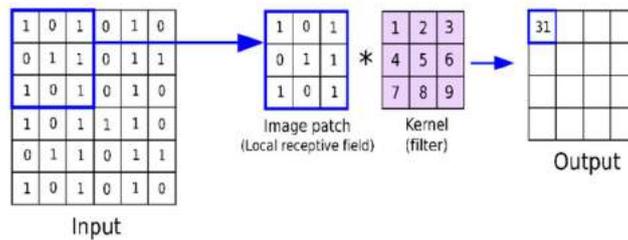


Gambar 2. 5 Arsitektur CNN

Berikut ini penjelasan mengenai *layer* pada arsitektur CNN:

### 2.2.4.1. Convolutional layer

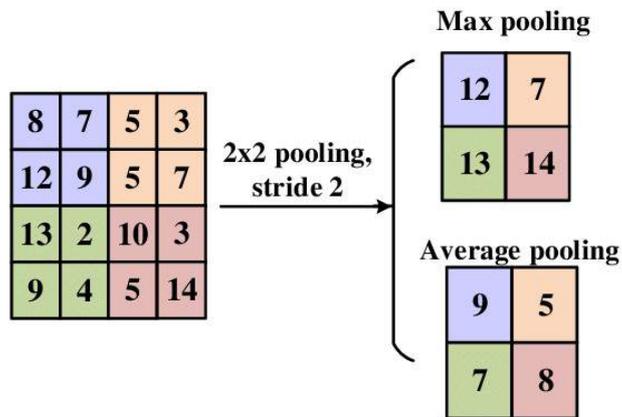
*Convolutional Layer* merupakan lapisan pertama yang menerima input pada arsitektur. Lapisan ini merupakan bagian yang menggabungkan filter linier dan area lokal untuk melakukan operasi konvolusi. Bentuk dari lapisan ini adalah sebuah filter dengan panjang (*pixel*), lebar (*pixel*) dan tebal tergantung pada *channel image* data yang diinputkan. Ketiga filter ini akan bergeser ke seluruh bagian gambar. Pergeseran tersebut akan melakukan operasi “dot” antara input dan nilai filter, yang menghasilkan *output* yang disebut *activation map (feature map)* [29]. Proses *convolutional layer* ditunjukkan pada gambar 2.6.



Gambar 2. 6 Proses Convolutional Layer [34]

### 2.2.4.2. Pooling Layer

Pooling layer merupakan tahapan setelah *Convolutional Layer*. *Pooling Layer* terdiri dari filter dengan ukuran dan *stride* tertentu. Pada setiap pergeseran yang dilakukan tergantung pada jumlah *stride* yang akan digeser pada seluruh area *activation map*. *Pooling layer* yang paling umum digunakan adalah *max pooling* dan *average pooling*. Sebagai contoh (gambar 2.7), apabila menggunakan *Max Pooling 2x2* dengan *Stride 2*, maka pada setiap pergeseran filter, nilai yang diambil adalah nilai yang terbesar. Sedangkan *Average Pooling* akan mengambil nilai rata-rata [11].



Gambar 2. 7 Proses *Max Pooling* dan *Average Pooling* [35]

### 2.2.4.3. Aktivasi ReLU

Rectified Linear Unit (ReLU) adalah fungsi aktivasi untuk memberikan kemampuan network agar dapat melakukan tugas-tugas non-linear [36]. Pada layer ini tidak akan mempengaruhi bidang reseptif convolutional layer [37]. Fungsi ReLU ditunjukkan rumus (2,1).

$$f(x) = \max(0, x) \quad (2, 1)$$

Jika Fungsi ReLU dijabarkan seperti rumus (2,2).

$$f(x) = \begin{cases} x, & x > 0 \\ 0, & x \leq 0 \end{cases} \quad (2, 2)$$

Keterangan:

- $f(x)$  = fungsi ReLU
- Jika nilai  $x$  lebih besar 0, maka nilai  $x$  akan tetap
- Jika nilai  $x$  lebih kecil atau sama dengan 0, maka nilai  $x$  akan dinaikkan menjadi 0

#### 2.2.4.4. Aktivasi *Softmax*

Aktivasi *Softmax* adalah bentuk lain dari algoritma *Logistic Regression* yang dapat digunakan untuk mengklasifikasi tiga kelas atau lebih. Standar klasifikasi yang umum yang digunakan adalah klasifikasi kelas biner [28]. Fungsi *softmax* ditunjukkan pada rumus (2,3):

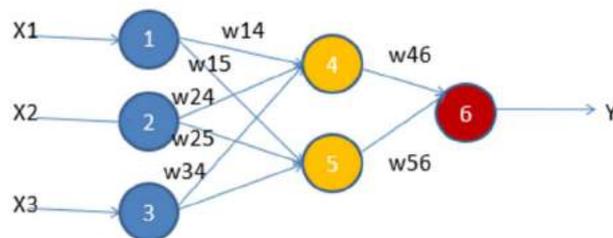
$$f_j(Z) = \frac{e^{z_j}}{\sum_k e^{z_k}} \quad (2,3)$$

Notasi  $f_j$  menunjukkan hasil fungsi untuk setiap elemen ke- $j$  dalam vektor keluaran kelas. Argumen  $z$  merupakan hipotesis yang diberikan oleh model pelatihan sehingga dapat diklasifikasikan oleh fungsi *Softmax*. *Softmax* digunakan untuk menghitung probabilitas untuk semua label. Vektor dengan nilai riil diambil dari label yang ada akan diubah menjadi vektor

dengan nilai antara nol dan satu yang jika semua nilai dijumlah akan bernilai satu [28].

#### 2.2.4.5. Fully Connected Layer

*Fully connected layer* adalah untuk mentransformasikan dimensi data sehingga dapat diklasifikasikan secara linier [13]. *Fully connected layer* menerima *input* dari hasil *output pooling layer* dalam bentuk *feature map*. Karena *feature map* masih dalam bentuk array multidimensional maka akan dilakukan *reshape feature map* dan menghasilkan vektor sebanyak *n*-dimensi, dimana *n* adalah jumlah kelas output yang harus dipilih program[29]. Untuk lebih jelasnya dapat dilihat proses *Fully connected layer* pada Gambar 2.8.



Gambar 2. 8 Proses *Fully Connected Layer*[13]

#### 2.2.5. Transfer Learning

*Neural network* memiliki struktur hirarki yang baik, dengan fitur umum dan khusus yang secara bertahap diperoleh saat jaringan semakin dalam. Metode pembelajaran mesin tradisional membutuhkan distribusi yang sama dari data domain sumber dan data domain target. *Transfer learning* merupakan teknik yang memanfaatkan kesamaan antar domain target dan domain sumber

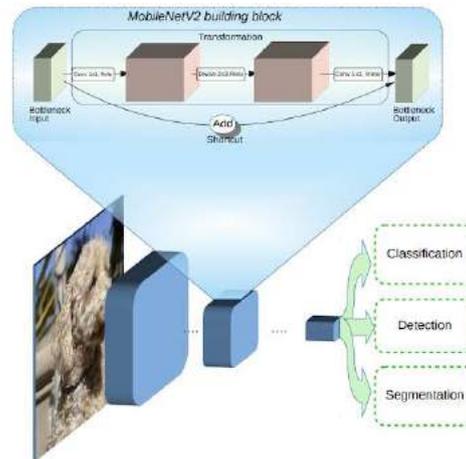
dan menggunakan domain sumber sebagai pola dasar untuk mempercepat pembelajaran di domain target [14]. Metode ini tidak hanya mengurangi biaya pelatihan model, tetapi juga meningkatkan efek pembelajaran mesin secara signifikan. Oleh karena itu, transfer learning dapat membantu menangani beberapa skenario aplikasi baru, yang memungkinkan pembelajaran mesin saat tidak tersedia cukup data berlabel [38].

Metode *transfer learning* yang sering dilakukan diantaranya adalah melatih semua parameter setelah memuat bobot pra-pelatihan, hanya menggunakan beberapa parameter lapisan terakhir yang dilatih setelah memuat bobot pra-pelatihan, menambahkan lapisan yang terhubung penuh berdasarkan jaringan asli, atau hanya menggunakan lapisan yang terhubung penuh terakhir yang dilatih setelah memuat bobot pra-pelatihan [38].

#### **2.2.6. MobileNet V2**

Arsitektur *MobileNet* adalah jaringan saraf *convolutional* ringan yang diusulkan oleh tim Google pada tahun 2017, dengan fokus pada perangkat seluler atau tertanam [14]. *MobileNetV2* merupakan pengembangan dari *MobileNetV1*, menggunakan teknik konvolusi kedalaman terpisah atau *depthwise separable convolution* (DSP) dengan memperkenalkan *inverted residuals* dan *linear bottlenecks* [15]. Arsitektur *MobileNetV2* ditunjukkan pada gambar 2.9. Pada bagian input dan output antara model dalam komponen bottleneck berfungsi merubah konsep dari rendah menjadi tinggi. Secara bersamaan, lapisan dalam merangkum fungsi model untuk mengubah input dari konsep tingkat yang lebih rendah (piksel) ke deskriptor tingkat yang lebih tinggi (klasifikasi gambar). Jadi, seperti koneksi residual dalam arsitektur CNN

tradisional, diantara *bottleneck* akan membuat proses pelatihan lebih cepat dan dengan akurasi yang lebih baik [39].



Gambar 2. 9 Arsitektur *MobileNetV2* [39]

### 2.2.7. Preprocessing

Tahap *preprocessing* adalah tahap sebelum pelatihan dan pengujian data [20]. Pada tahap ini dilakukan standarisasi dataset agar semua citra menjadi rata. Pada tahap ini dapat menggunakan library python tambahan yaitu *ImageDataGenerator*. [40]. *Preprocessing* yang dapat dilakukan diantaranya pembagian dataset, mengubah ukuran gambar, dan augmentasi data. Pembagian dataset dilakukan untuk memisahkan data yang akan digunakan untuk membangun model dan pengujian terhadap model yang sudah dibangun [29]. Pengubahan ukuran piksel citra adalah langkah pra-pemrosesan penting dalam visi komputer untuk mengefektifkan model pelatihan. Semakin kecil ukuran gambar, semakin baik model akan berjalan [20]. Augmentasi merupakan proses mengubah atau memodifikasi gambar. Augmentasi dapat memperbanyak data, sehingga model dapat melakukan generalisasi dengan baik [41]. Serangkaian metode augmentasi seperti *flipping*, *cropping*, *enlarging* dan *rescaling* [19].

### 2.2.8. Confusion Matrix

*Confusion matrix* merupakan sebuah perhitungan untuk mengukur hasil akurasi untuk mengetahui performa algoritma klasifikasi yang digunakan. Hasil akhir dari matrik ini adalah tingkat akurasi, presisi, *recall*, *F1-Score* dengan satuan persen (%) [1]. Akurasi menggambarkan nilai keakuratan model dalam mengklasifikasikan dengan benar, menggunakan rasio prediksi benar (positif dan negatif) dibandingkan dengan seluruh data [42]. Presisi merupakan tingkat keakuratan antara data yang diprediksi oleh model dengan data yang diminta, menggunakan rasio prediksi benar positif dibandingkan dengan hasil prediksi positif secara keseluruhan. *Recall* menggambarkan keberhasilan model dalam menemukan kembali sebuah informasi, menggunakan rasio prediksi benar positif dibandingkan dengan keseluruhan data yang positif dan benar [43]. *F1-Score* adalah *harmonic mean* dari *precision* dan *recall*, secara representasi jika *F1-Score* memiliki skor yang baik berarti model klasifikasi memiliki *precision* dan *recall* yang baik [44]. Pada pengukuran kinerja menggunakan *confusion matrix*, seperti yang terlihat pada Tabel 2.2

Tabel 2. 2 *Confusion Matrix*

Klasifikasi		Prediksi	
		Positif	Negatif
Aktual	Positif	True Positive (TP)	True Negative (TN)
	Negatif	False Positive (FP)	False Negative (FN)

Dari tabel 2.2 dapat dihitung tingkat akurasi, presisi dan recall dari sebuah model algoritma dengan menggunakan persamaan (2,3), (2,4), (2,5) , (2,6).

$$Akurasi = \frac{TP+TN}{TP+TN+FP+F} * 100\% \quad (2,3)$$

$$Presisi = \frac{TP}{FP+TP} * 100\% \quad (2,4)$$

$$Recall = \frac{TP}{TP+FN} * 100\% \quad (2,5)$$

$$F1 - Score = \frac{2 \times Presisi \times Recal}{Presisi + Recal} * 100\% \quad (2,6)$$

Keterangan:

True Positive (TP): Jumlah data kelas positif terklasifikasi benar

False Positive (FP): Jumlah data kelas negatif terklasifikasi salah

True Negative (TN): Jumlah data kelas negatif terklasifikasi benar

False Negative (FN): Jumlah data kelas positif terklasifikasi salah