

BAB II

LANDASAN TEORI

Pada bab II akan dibahas mengenai tinjauan pustaka dan teori penunjang dari laporan ini. Adapun teori yang dibahas adalah huruf Braille, *deep learning*, *Convolutional Neural Network*, dan *TensorFlow*.

2.1. Tinjauan Pustaka

Dalam penelitian ini terdapat beberapa pustaka yang berkaitan, antara lain:

- 1) **(Joko Subur, dkk., 2015)** pada jurnal yang berjudul “*Braille Character Recognition Using Artificial Neural Network*”, dimana mereka menggunakan *deep learning* dengan metode *Artificial Neural Network (ANN)* untuk proses pengenalan karakter Braille. Pada penelitian tersebut, dengan pengujian pengenalan Braille pada 10 data gambar yang berbeda dengan akuisisi citra yang memiliki kemiringan yang berbeda dari -1.5 derajat hingga 1.5 derajat. Tingkat keakurasian yang dihasilkan adalah 99% dengan kemiringan -1 derajat hingga 1 derajat. Akan tetapi, tingkat keakurasiannya berkurang jika akuisisi citra memiliki kemiringan lebih dari 1 derajat, dan jika lebih dari 1.5 derajat gambar tidak dapat dikenali [1].
- 2) **(Guanjun Guo, dkk., 2018)** pada jurnal yang berjudul “*A Fast Face Detection Method via Convolutional Neural Network*”, dimana mereka menggunakan metode *Convolutional Neural Network (CNN)* untuk meningkatkan kecepatan dalam proses deteksi wajah berdasarkan *Discriminative Complete Features (DCF)* yang diekstraksi oleh CNN. Dibandingkan dengan metode deteksi wajah canggih menggunakan CNN, metode yang diusulkan melakukan klasifikasi langsung pada DCF, yang secara signifikan dapat meningkatkan efisiensi selama proses deteksi wajah [2].

- 3) (**Aisha Mousa, dkk., 2013**) pada jurnal yang berjudul “*Smart Braille System Recognizer*”, dimana mereka mengusulkan metode pengenalan karakter Braille satu sisi (*single sided Braille character recognition*) yang terdiri dari proses *Pre-processing* atau mengubah citra RGB ke *grayscale*, kemudian terdapat proses *image enhancement*, segmentasi, ekstraksi ciri, dan pengenalan karakter Braille. Dalam proses pengetesan, mereka menggunakan scanner untuk proses akuisisi citra. Hasilnya, metode ini dapat mengenali karakter Braille dengan baik. Akan tetapi, karena proses akuisisi citra masih menggunakan *scanner* maka sistem ini masih belum *real-time* [3].
- 4) (**Pushkar Sathe, dkk., 2019**) pada jurnal yang berjudul “*Waste Segregation using Convolutional Neural Network*”, dimana mereka menggunakan metode *Convolutional Neural Network* (CNN) untuk melakukan segregasi atau pemisahan sampah berdasarkan 4 kategori yaitu, plastik, kertas, kaca, dan kaleng. Sistem ini menggunakan mikrokontroler dan servo motor untuk menggerakkan masing-masing tempat sampah secara otomatis apabila jenis sampah tertentu terdeteksi. Hasil dari penelitian ini cukup baik, dan disarankan untuk menambahkan fitur pada sistem seperti notifikasi sampah akan penuh [4].
- 5) (**Shu Zhan, dkk., 2016**) pada jurnal yang berjudul “*Face Detection using Representation Learning*”, dimana mereka menggunakan metode *Convolutional Neural Network* (CNN) untuk melakukan representasi algoritma deteksi wajah cepat (*fast face detection*), sehingga dapat secara eksplisit menangkap berbagai fitur wajah laten. Dalam penelitian ini, mereka menggunakan CNN multilayer sebagai ekstraktor fitur untuk memperoleh fitur khusus masalah secara otomatis. Untuk memfilter latar terluar dengan cepat, mereka melatih filter latar berdasarkan *Adaboost classifier*. Jadi, bahkan jika CNN sangat kompleks, kecepatan seluruh sistem masih cepat [5].

- 6) (**Xue-Feng Zhao, dkk., 2019**) pada jurnal yang berjudul “*A Skin-Like Sensor for Intelligent Braille Recognition*”, yang menjelaskan tentang sensor fleksibel bertekstur seperti kulit untuk proses pengenalan karakter huruf braille untuk robot. Pada jurnal ini, untuk pertama kalinya pengenalan braille real-time direalisasikan dengan sensor tactile dan berbagai algoritma keputusan. Jurnal ini sangat baik untuk membantu permasalahan sosial dalam pertukaran informasi bagi tunanetra [6].
- 7) (**Zhenfei Tai, dkk., 2010**) pada jurnal yang berjudul “*Braille Document Recognition using Belief Propagation*”, yang menjelaskan tentang pengenalan dokumen dengan tulisan karakter huruf braille untuk menjaganya dan menyebarkannya ke lebih banyak penyandang tunanetra. Mereka memperkenalkan metode estimasi parameter dokumen braille yang adaptif untuk secara otomatis menentukan sudut rotasi, lekukan, dan jarak pada orientasi horizontal dan vertikal [7].
- 8) (**Xudong Sun, dkk., 2018**) pada jurnal yang berjudul “*Face Detection using Deep Learning: An Improved Faster RCNN Approach*”, yang menjelaskan tentang skema deteksi wajah baru menggunakan *deep learning* dan mencapai kinerja deteksi canggih pada evaluasi tolak ukur deteksi wajah FDDB. Mereka meningkatkan kerangka kerja RCNN yang lebih cepat dengan menggabungkan sejumlah strategi, termasuk penggabungan fitur, penambahan negatif, pelatihan multi-skala, model *pre-training*, dan kalibrasi yang tepat dari parameter-parameter utama [8].

2.2. Landasan Teori

Pada bagian landasan teori terdapat teori-teori yang dapat menunjang proses penelitian, berikut ini adalah teori-teori tersebut.

a) Huruf Braille

Huruf Braille merupakan suatu sistem untuk membaca dan menulis lewat sentuhan. Braille menggunakan karakter yang dibentuk lewat kombinasi enam titik timbul yang diatur dalam sel Braille dalam dua kolom vertikal dengan tiga titik pada kedua kolom dan dirancang sedemikian rupa dan menjadi sebuah sistem penulisan yang dapat digunakan untuk membantu orang tunanetra. Karakter Braille yang sederhana dibentuk oleh satu atau lebih titik dan menempati seluruh sel Braille. Untuk kemudahan, titik-titik pada sel Braille dirujuk dengan angka dan menyesuaikan dengan tombol pada Braillewriter [9].

Ada 3 metode untuk mentranskripsi Braille, yaitu Braillewriter, program pada komputer, dan Braille slate dan stylus. Braillewriter dan komputer bekerja dengan cara yang sama yaitu dengan menggunakan 6 tombol yang sesuai dengan sel Braille. Sementara, jika menggunakan Braille slate titik akan timbul pada sisi kertas yang satunya. Sehingga, penulisannya dilakukan dari kanan ke kiri, jadi saat kertasnya dibalik dapat dibaca dari kiri ke kanan. Maka dari itu, titik 1, 2, dan 3 yang seharusnya ada di sebelah kiri, menjadi di sebelah kanan. Sementara, titik 4, 5, dan 6 yang seharusnya ada di sebelah kanan, menjadi di sebelah kiri. Sehingga, ketika kertasnya dibalik titik 1, 2, dan 3 akan ada di sebelah kiri dan sebaliknya untuk titik 4, 5, dan 6. Karakter huruf braille ditampilkan pada Gambar 2.1. Contoh penulisan Braille ditampilkan pada Gambar 2.2.

⠁	⠃	⠉	⠇	⠑	⠋	⠍	⠊	⠎	⠚
a	b	c	d	e	f	g	h	i	j
⠅	⠆	⠎	⠏	⠒	⠗	⠕	⠖	⠓	⠔
k	l	m	n	o	p	q	r	s	t
⠥	⠦	⠩	⠨	⠬	⠭				
u	v	w	x	y	z				

Gambar 2. 1. Karakter Huruf Braille

(Sumber: B. Bhanushali, dkk. 2018)



Gambar 2. 2. Contoh Penulisan Karakter Huruf Braille

(Sumber: Natt Róžańska, 2013)

b) *Deep Learning*

Deep Learning adalah salah satu fungsi kecerdasan buatan yang meniru cara kerja otak manusia dalam memproses data dan menciptakan pola untuk digunakan dalam pengambilan keputusan. *Deep Learning* merupakan bagian dari *Machine Learning*. *Deep Learning* terdiri dari banyak kelas atau jenis, beberapa yang paling sering digunakan adalah *Convolutional Neural Network* (CNN), *Multilayer Perceptrons* (MLP), dan *Recurrent Neural Network* (RNN).

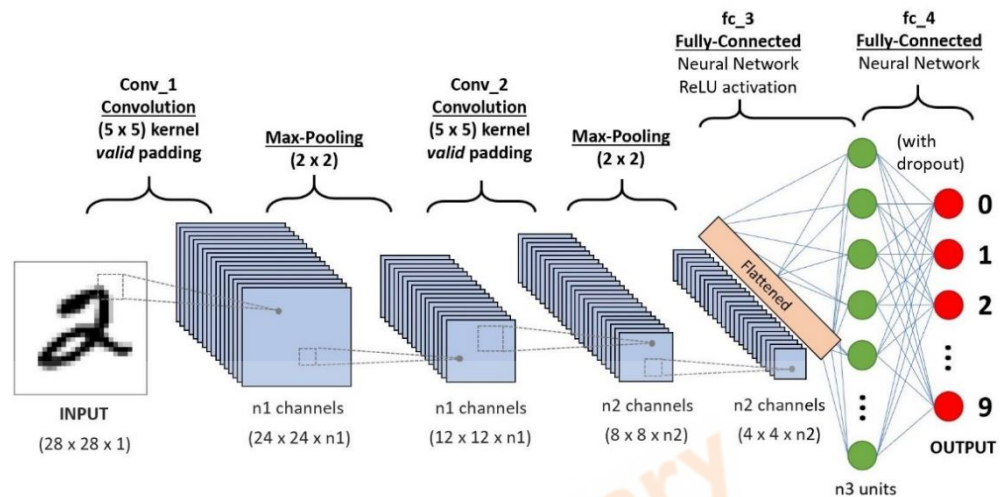
Keunggulan *deep learning* adalah memiliki performa terbaik dalam menyelesaikan masalah kompleks, mengurangi kebutuhan untuk rekayasa ciri, dan memiliki arsitektur yang mampu beradaptasi terhadap permasalahan baru dengan mudah. Sementara, kekurangan dari *deep learning* adalah membutuhkan banyak data, proses training yang memakan waktu, dan *overfitting* [10][11].

c) *Convolutional Neural Network*

Convolutional Neural Network (CNN) merupakan salah satu kelas dari *deep learning* yang mampu melakukan pengenalan gambar dan klasifikasi gambar. Metode CNN merupakan suatu kelas pada *neural network* yang berspesialisasi dalam memproses data yang memiliki topologi seperti *grid*, misalnya gambar. Metode CNN dapat digunakan dalam pengenalan wajah, analisis dokumen, klasifikasi gambar, klasifikasi video, dsb.

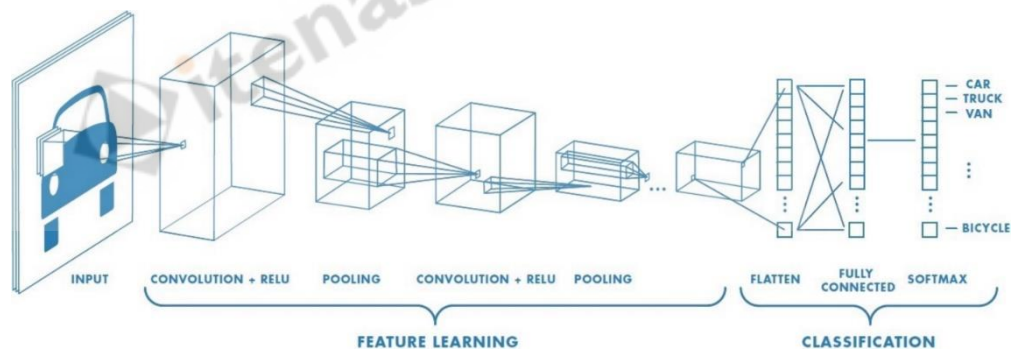
Metode CNN mengklasifikasi gambar dengan memproses gambar yang diinput, kemudian mengklasifikasikannya pada kategori tertentu misalnya pada citra manusia, terdapat wajah, mata, bibir, hidung, tangan, dll. Gambar akan dibuat menjadi *array* berisi nilai pada setiap pixel dengan resolusi tinggi*panjang*dimensi yang disebut *channel*. Dimana, *channel* ini biasanya terdiri dari 3 buah yang berarti citra merupakan gambar RGB dengan masing-masing lapisan (*channel*) merepresentasikan *Red-Green-*

Blue atau 1 lapisan jika gambar *grayscale*. Akan tetapi, jumlah lapisan juga bisa melebihi 3, bahkan hingga ratusan yang merepresentasikan berbagai warna lainnya dengan arsitektur RGB.



Gambar 2. 3. Ilustrasi Arsitektur Convolutional Neural Network

(Sumber: Sumit Saha, 2018)



Gambar 2. 4. Ilustrasi Arsitektur Convolutional Neural Network

(Sumber: Sumit Saha, 2018)

Seperti yang ditampilkan pada Gambar 2.3 dan Gambar 2.4, Arsitektur CNN dibagi menjadi dua bagian besar, yaitu *Feature Extraction Layer* dan *Fully Connected Layer*. Pada *Feature Extraction Layer* terjadi “*encoding*” dari sebuah *image* menjadi *feature map* yang berupa angka-angka yang merepresentasikan citra tersebut (*Feature Extraction*).

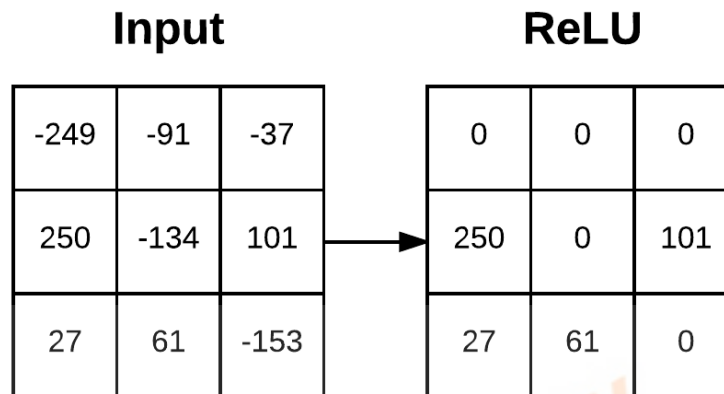
Feature Extraction Layer terdiri dari *convolution layer* dan *pooling layer*. *Convolution layer* merupakan bagian utama pada metode CNN yang menjadi pembeda dengan *neural network* lainnya. *Convolution layer* adalah lapisan pertama yang mengekstraksi ciri dari citra yang dimasukkan. Proses Konvolusi (*Convolution*) menjaga hubungan antara *pixel* dengan mempelajari ciri citra menggunakan operasi matematis antara matriks citra dengan filter atau *kernel*. *Kernel* adalah sebuah operator yang diterapkan ke seluruh citra untuk mendapatkan nilai *array* dari sebuah citra. *Kernel* adalah matriks biasanya berukuran 3*3 atau 5*5 yang berisi nilai acak antara -1 dan 1. Contoh operasi matematis pada proses konvolusi digambarkan pada Gambar 2.5.

131	162	232	84	91	207
104	-1	0	+1	237	109
243	-2	0	+2	135	26
185	-1	0	+1	61	225
157	124	25	14	102	108
5	155	16	218	232	249

Gambar 2. 5. Contoh Operasi Matematis Proses Konvolusi

(Sumber: Rosebrock, Adrian.)

Hasil konvolusi matriks citra dengan *filter (kernel)* tersebut disebut *feature map*. Adapun, ReLU atau *Rectified Linear Unit* yang berfungsi untuk mengubah nilai negatif pada *feature map* menjadi positif, seperti ditampilkan pada Gambar 2.6.



Gambar 2. 6. Contoh Penerapan Aktivasi ReLu

(Sumber: Rosebrock, Adrian.)

Pooling layer atau proses *Max-Pooling* merupakan lapisan yang berfungsi untuk mengurangi ukuran spasial untuk mengurangi jumlah parameter dan perhitungan ketika ukuran citra terlalu besar. *Pooling layer* beroperasi pada setiap *feature map* secara independen.

Fully Connected Layer merupakan bagian dimana hasil *feature map* dari *feature extraction layer* yang berbentuk *array* multi-dimensi diubah menjadi vektor agar dapat dibentuk menjadi *fully connected layer* seperti sebuah jaringan syaraf (*neural network*), proses ini disebut sebagai *Flatten*. Kemudian, semua ciri yang sudah terbentuk menjadi jaringan syaraf dikombinasikan untuk membuat suatu model. Lalu, dengan fungsi aktivasi seperti *softmax* atau sigmoid digunakan untuk mengklasifikasikan output misalnya, mobil, wajah, kucing, macan, dll.

Singkatnya, cara kerja metode CNN yaitu, CNN akan melatih dan menguji setiap gambar melalui serangkaian proses. Dimulai dari pemecahan gambar menjadi gambar yang lebih kecil, kemudian memasukkan setiap gambar yang lebih kecil ke *neural network* yang lebih kecil, menyimpan hasil dari masing-masing gambar kecil ke dalam array baru, *downsampling* atau mengurangi ukuran spasial untuk mengurangi jumlah parameter dan perhitungan ketika ukuran citra terlalu besar, dan membuat prediksi.

Kelebihan dari metode CNN adalah dapat secara otomatis mengekstraksi ciri penting dari setiap citra tanpa bantuan manusia, selain itu metode CNN juga lebih efisien dibandingkan metode *neural network* lainnya terutama untuk memori dan kompleksitas. Sementara, kekurangan dari metode CNN adalah membutuhkan banyak data latih, proses pelatihan (*training*) yang memakan waktu, dan *overfitting*. *Overfitting* dapat terjadi dikarenakan terlalu banyak data latih maka algoritma kehilangan kemampuan untuk menggeneralisasi [12][13][14][15][16][21].

d) Keras

Keras merupakan *library* jaringan syaraf (*neural network*) *open-source* yang ditulis di Bahasa pemrograman *python*. Keras mampu berjalan pada *Tensor Processing Unit* (TPU) atau GPU yang besar, dan model Keras dapat diekspor untuk dijalankan pada *browser* atau *smartphone* [17].

e) TensorFlow

TensorFlow merupakan *library* perangkat lunak *open-source* yang biasa digunakan untuk *dataflow* dan membedakan pemrograman dari berbagai tugas. TensorFlow menawarkan berbagai variasi dari *toolkit* yang berbeda yang memungkinkan untuk membangun model pada tingkat abstraksi yang kita inginkan [18].

f) *Optimizer Adam*

Optimizer Adam (Adaptive Moment Estimation) merupakan sebuah metode yang berfungsi untuk mengurangi loss dengan mengubah atribut pada jaringan syaraf (*neural network*) seperti *weight* dan *learning rate*. Kelebihan dari *optimizer Adam* yaitu cukup efisien secara komputasi dan cocok untuk permasalahan dengan data atau *parameter* yang besar [19].

