# BAB II TINJAUAN PUSTAKA DAN LANDASAN TEORI

## Tinjauan Studi

Hasil pencarian Google Scholar yang berkaitan dengan penelitian pengenalan karakter huruf braille membuat penelitian yang dilakukan menjadi lebih mudah. Berikut adalah beberapa penelitian terdahulu yang memiliki hubungan dalam penelitian yang akan dilakukan.

Penelitian yang telah dilakukan oleh Muhammad Fahmi Herlambang, Asep Nana Hermana, dan Kurnia Ramadhan Putra. “Pengenalan Karakter Huruf Braille dengan Metode *Convolutional Neural Network*” (Herlambang et al., 2021). *Deep Learning* menjadi salah satu teknologi buatan yang dapat membantu mengenali karakter huruf braille. Metode yang digunakan yaitu CNN (*Convolutional Neural Network*) dengan tujuan menguji akurasi pengenalan karakter braille agar melebihi 80%. Penggunaan metode CNN ini mampu mengklasifikasi gambar dengan memproses gambar yang diinput, kemudian mengklasifikasikannya sesuai kategori. Gambar tersebut akan dibuat menjadi *array* berisi nilai pada setiap *pixel* dengan resolusi tinggi\*panjang\*dimensi yang disebut *channel*. Pada *channel* terdiri dari 3 buah citra yang merupakan gambar RGB yang merepresentasikan *Red-Green-Blue* atau 1 lapisan jika gambar *grayscale*. Hasil dari penelitian ini mendapatkan tingkat akurasi 81.54%, nilai presisi 81.54%, nilai *recall* 100%, dan nila *F1 Score* 89.83%.

Penelitian berikutnya oleh Marindo Andriansyah dan Hartanto Junaedi. “Pengenalan Karakter Braille Memanfaatkan *Convolutional Neural Network*” (Andriansyah & Junaedi, 2021). Penelitian ini dilakukan untuk membantu siapa saja yang tidak bisa membaca huruf braille namun bersinggungan langsung dengan orang tunanetra. Dalam penelitian menggunakan CNN (*Convolutional Neural Network*) untuk melatih data sehingga mampu mengenali karakter braille. Tahapan yang dilakukan untuk mengenali karakter braille mulai dari *preprocessing*, *segmentation*, dan *classification*. Dataset yang digunakan merupakan gambar hasil scan dokumen braille yang terdari dari abjad A-Z. Hasil penelitian ini mendapatkan tingkat akurasi *testing* 99.62% dengan nilai *loss* 0.0232.

Penelitian yang dilakukan oleh Kirill Smelyakov, Anastasiya Chupryna, Dmytro Yeremenko, Anton Sakhon, dan Vitalii Polezhai. “*Braille Character Recognition Based on Neural Networks*” (Smelyakov et al., 2018). Penelitian ini menggunakan karakter huruf braille yang digunakan untuk bahasa ukraina (Cyrillic letters) menggunakan algoritma ANN (*Artificial Neural Network*). Pada proses pengenalan karakter braille ini melalui langkah *image acquisition* (akuisisi citra), *image preprocessing* (citra pra-pemrosesan), *image segmentation* (segmentasi citra), *braille dot recognition* (pengenalan titik braille), dan *braille cell* *translation* (translasi sel braille). ANN akan digunakan untuk melakukan training dan testing dalam pengenalan karakter huruf braille. ANN akan mengidentifikasi *Cyrillic letters* dalam sistem representasi huruf braille. Citra yang digunakan pada penelitian ini berukuran 41x60 *pixel*. Hasil dari penelitian ini mendapatkan akurasi *training* mencapai 95.7% dan akurasi *testing* mencapai 95%.

Penelitian yang dilakukan Chaerul Umam dan Lekso Budi Handoko. “*Convolutional Neural Network* (CNN) untuk Identifikasi Karakter Hiragana” (Umam & Handoko, 2021). Penelitian ini menggunakan metode CNN (*Convolutional Neural Network*) untuk melakukan proses *training* dan *testing* dataset. Proses pengenalan karakter hiragana diawali dengan *input* citra kemudian citra dikonversi menjadi citra *grayscale*, lalu dilanjut normalisasi citra yang kemudian akan membagi menjadi citra *training* dan citra *testing*, setelah itu dilakukan *train dataset* menggunakan algoritma CNN (*Convolutional Neural Network*), lalu dilakukan validasi dengan *testing data*. Pada penelitian ini menggunakan citra dengan ukuran 83x84 *pixel*. Pengujian mengguakan *optimizer* adam dengan iterasi *epoch* sebesar 100 mendapatkan hasil akurasi *loss* mencapai 6%, dan akurasinya mencapai 82%.

Penelitian yang dilakukan Bi-Min Hsu. “*Braille Recognition for Reducing Asymmetric Communication between the Blind and Non-Blind*” (Hsu, 2020). Penelitian ini mengenalkan dataset braille yang terdiri dari 37 karakter yang sesuai dengan 71 kelas karakter dan menggunakan metode yang menggabungkan *Ratio Character Segmentation Algorithm* (RSCA) dan *Convolutional Neural Network* (CNN) untuk mengkonversi braille. Citra yang digunakan memiliki dimensi 28x28 *pixel* yang termasuk citra RGB. Cara pengenalan huruf braille pada penelitian ini yaitu dengan *image acquisition* (akuisisi citra) dengan mengambil citra melalui kamera hp ataupun citra yang tersimpan di hp, lalu *image preprocessing* (pra-processing citra) dengan mengatur kemiringan citra lalu pemotongan spasi pada citra, dilanjutkan dengan *image segmentation* (segmentasi citra) menggunakan algoritma RCSA (*Ratio Character Segmentation Algorithm*), langkah terakhir yaitu *image recognition* (pengenalan citra) dengan memasukkan simbol braille ke dalam model CNN. Pada penelitian ini 80% dataset masuk ke dalam data *traini*ng dan 20% data *testing* dan *validation*. Hasil penelitian mendapatkan akurasi prediksi dan klasifikasi citra mencapai 98.73%.

Tabel 1.1 State of the art

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **No** | **Tahun** | **Judul** | **Peneliti** | **Metode** | **Ringkasan** |
| 1 | 2021 | Pengenalan Karakter Huruf Braille dengan Metode *Convolutional Neural Network* | Muhammad Fahmi Herlambang, Asep Nana Hermana, dan Kurnia Ramadhan Putra | *Convolutional Neural Network* (CNN) | Penggunaan metode CNN ini mampu mengklasifikasi gambar dengan memproses gambar yang diinput, kemudian mengklasifikasikannya sesuai kategori. Hasil dari penelitian ini mendapatkan tingkat akurasi 81.54%, nilai presisi 81.54%, nilai *recall* 100%, dan nila *F1 Score* 89.83%. |
| 2 | 2021 | Pengenalan Karakter Braille Memanfaatkan *Convolutional Neural Network* | Marindo Andriansyah, Hartanto Junaedi | *Convolutional Neural Network* (CNN) | Tahapan yang dilakukan untuk mengenali karakter braille mulai dari *preprocessing,* *segmentation,* dan *classification.* Dataset yang digunakan merupakan gambar hasil *scan* dokumen braille yang terdari dari abjad A-Z. Hasil penelitian ini mendapatkan tingkat akurasi testing 99.62% dengan nilai *loss* 0.0232 |
| 3 | 2021 | *Convolutional Neural Network* (CNN) untuk Identifikasi Karakter Hiragana | Chaerul Umam, Lekso Budi Handoko. | *Convolutional Neural Network* (CNN) | Penggunakan algoritma CNN untuk melakukan *training,* validasi dan *testing data*. Pada penelitian ini menggunakan citra dengan ukuran 83x84 *pixel*. Pengujian mengguakan *optimizer* adam dengan iterasi *epoch* sebesar 100 mendapatkan hasil akurasi *loss* mencapai 6%, dan akurasinya mencapai 82%. |
| 4 | 2020 | *Braille Recognition for Reducing Asymmetric Communication between the Blind and Non-Blind* | Bi-Min Hsu | *Convolutional Neural Network* (CNN) | Pada penelitian ini 80% dataset masuk ke dalam *data training* dan 20% *data testing* dan *validation*. Hasil penelitian mendapatkan akurasi prediksi dan klasifikasi citra mencapai 98.73%. |
| 5 | 2018 | *Braille Character Recognition Based on Neural Networks* | Kirill Smelyakov, Anastasiya Chupryna, Dmytro Yeremenko, Anton Sakhon, dan Vitalii Polezhai. | *Artificial Neural Network* (ANN) | ANN akan mengidentifikasi *Cyrillic letters* dalam sistem representasi huruf braille. Citra yang digunakan pada penelitian ini berukuran 41x60 *pixel*. Hasil dari penelitian ini mendapatkan akurasi *training* mencapai 95.7% dan akurasi *testing* mencapai 95%. |

Berdasarkan penelitian terdahulu, dapat disimpulkan bahwa hasil dari klasifikasi berkisaran 80% - 100%. Pada penelitian ini akan dilakukan pengenalan karakter huruf braille menggunakan metode *Convolutional Neural Network* (CNN) dengan tujuan membantu non tunanetra mampu memahami huruf braille sehingga memudahkan untuk berkomunikasi dengan tunanetra. Harapan dari penelitian ini yaitu menjadi platform dalam pengembangan penelitian selanjutnya menggunakan algoritma *Convolutional Neural Network* (CNN).

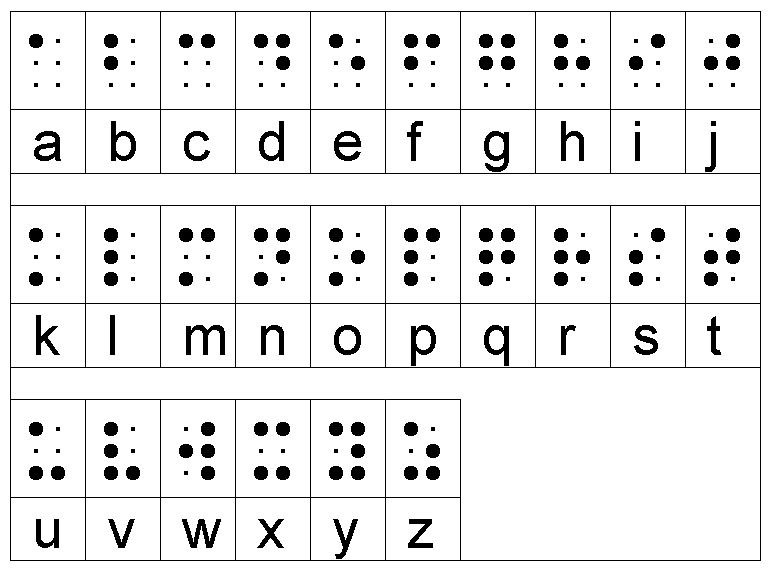
## Tinjauan Pustaka

Pada bagian tinjauan pustaka terdapat teori-teori yang dapat menunjang proses penelitian, berikut merupakan teori-teori tersebut.

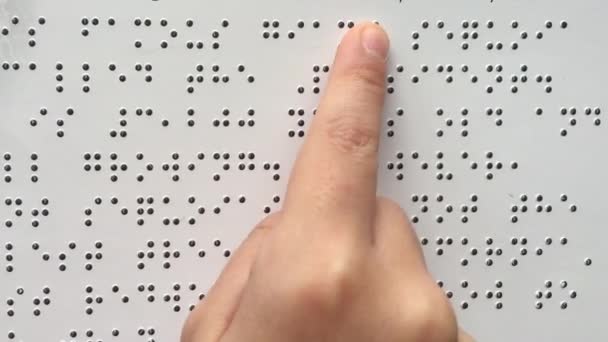
### Huruf Braille

Huruf braille merupakan salah satu alat bantu bagi para tunanetra dalam membaca dan menulis melalui sentuhan. Huruf braille pertama kali diciptakan oleh seorang berkebangsaan Perancis, Louis Braille, yang mengalami kebutaan akibat tertusuk jarum. Huruf braille berbentuk 6 titik timbul, di mana 2 titik secara horizontal dan 3 titik secara vertikal. Titik-titik tersebut terletak pada sel yang bernama sel braille.

Huruf braille umumnya berbentuk titik timbul yang dapat dibaca dari arah kiri ke kanan. Cara membaca huruf braille ini perlu dilatih kepekaan tangan dalam mengenali masing-masing karakter huruf. Selain itu juga diharuskan hafal dengan masing-masing karakter huruf agar dapat membacanya. Karena hal ini proses agar bisa membaca tulisan braille membutuhkan waktu yang relatif lama.



Gambar 2.1 Karakter Huruf Braille



Gambar 2.2 Membaca Huruf Braille

### Pengenalan Karakter

Pengenalan karater termasuk sistem pengenalan pola. Sistem ini mampu mengenali karakter atau tulisan teks pada sebuah citra. Pengenalan pola sendiri merupakan sebuah sistem yang mampu untuk membaca atau mengenali citra masukan sesuai dengan citra yang telah ditentukan. Pengaplikasiannya sendiri seperti pengenalan sidik jari, teks, tulisan, hingga wajah seseorang.

Tujuan dari pengenalan karakter ini sendiri untuk mendukung perkembanga teknologi yang semakin pesat dan berbentuk digital. Ketika citra masukan berkualitas baik maka citra pengeluarannya dapat berjalan dengan baik. Kualitas citra sendiri ditentukan oleh komposisi susunan karakter pada citra atau kecilnya *noise* yang terdapat dalam citra.

Pengenalan karakter disebut juga *Optical Character Recognition* (OMR). Secara umum prinsip kerja OMR sebagai berikut.

1. *Input* citra yang berisi karakter teks yang akan dilakukan pengenalan. Citra ini bisa berupa foto, gambar, dokumen, dll.
2. *File* citra tersebut diproses menggunakan aplikasi yang mampu mengenali pola pada citra tersebut.
3. *Output* dari citra tersebut berupa data atau karakter yang siap diolah pada langkah selanjutnya.

### *Artificial Intelligence*

*Artificial Intelligence* merupakan pemrograman yang dilakukan pada mesin di mana nantinya mesin akan melakukan pekerjaan yang lebih baik daripada manusia. Adanya *artificial intelligence* menjadi semakin banyak pula inovasi-inovasi baru yang dikembangan oleh manusia. Tujuan dari *artificial intelligence* sendiri selain membuat mesin menjadi ceras, juga membuat mesin menjadi lebih bermanfaat. Ruang lingkup dari *artificial intelligence* sendiri di antaranya sistem pakar (*expert system*), pengolahan bahasa alami (*natural language processing*), pengenalan suara (*voice recognition*), robotika dan sistem sensor, *computer vision, intellegent computer aided instruction, game playing.*

### *Machine Learning*

*Machine Learning* atau dikenal dengan pembelajaran mesin merupakan pembelajaran mesin tanpa perlu diprogram secara eksplisit. *Machine Learning* disingkat menjadi ML. ML merupakan kecerdasan buatan yang mempelajari tentang banya data. ML dibutuhkan untuk melakukan kerja cerdas dan cepat dalam menyelesaikan berbagai permasalahan. Adanya ML ini membuat banyak peneliti berpikir bahwa ML menjadi kemajuan menuju *Artificial Intelligence* (AI) yang dilakukan oleh manusia.

*Machine Learning* sendiri mempelajari tentang algoritma dan model statistik. Dikarenakan memerluka data, membuat ML ini bergantung pada pola dan kesimpulan. Untuk memperoleh pola dan kesimpulan tersebut, ML melakukan data sampel atau yang disebut training data. Sehingga ML merupakan teknologi yang mampu menganalisa pola, dan memprediksi suatu hal dalam waktu yang singkat. Metode pembelajaran ML sebagai berikut.

1. *Supervised Learning* (Pembelajaran Terarah)

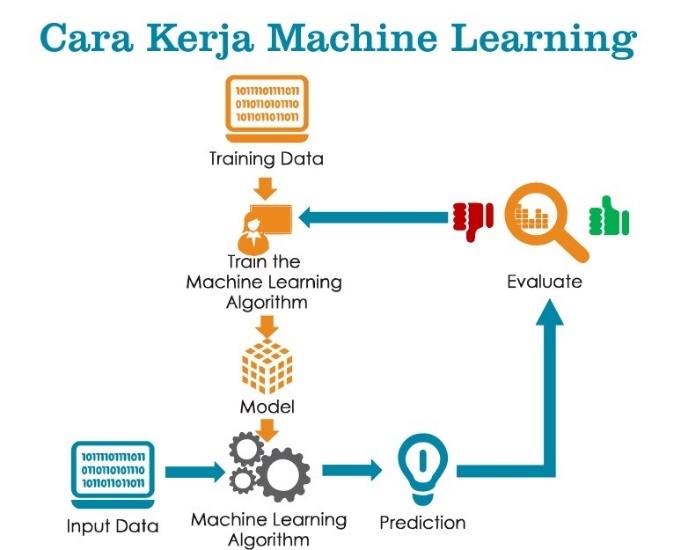
Metode ini dilakukan dengan pemberian label pada dataset yang digunakan oleh *Machine Learning* yang kemudian diklasifikasikan untuk melihat tingkat akurasinya. Pada metode ini nantinya *Machine Learning* akan mempelajari hubungan dan ketergantunga antar data.

1. *Unsupervised Learning* (Pembelajaran Tak Terarah)

Metode ini dilakukan tanpa pengawasan, sehingga prosesnya dilakukan pada data yang belum berlabel. Dengan metode ini diharapkan *Machine Learning* mampu mengidentifikasi pola tersebut dan mengetahui relasinya tanpa adanya bantuan.

1. *Reinforcement Learning*

Metode ini dilakukan dengan tipe penguatan pembelajaran. Di mana algoritma pada metode ini memiliki kemapuan untuk memberikan reward saat model semakin baik dan error saat model semakin buruk.



Gambar 2.3 Cara Kerja *Machine Learning*

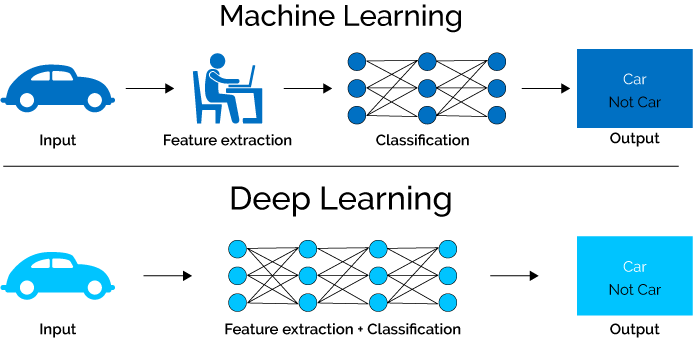
Pada gambar di atas terdapat alur bagaimana cara kerja *Machine Learning* yaitu dengan memasukkan (input) data menggunakan algoritma *Machine Learning* yang kemudian menghasilkan sebuah model. Di mana nantinya model ini yang akan melakukan prediksi terkait data tersebut. Hasil dari *Machine Learning* merupakan hasil pegolahan algoritam tersebut.

Kelebihan dari adanya *Machine Learning* ini yaitu menjadi peluang bagi pengusaha maupun praktisi teknologi dalam menciptakan teknologi baru dan inovatif, adanya *Machine Learning* juga membuat semua pekerjaan yang dilakukan manusia menjadi lebih mudah dan cepat.

Kelebihan ini juga menimbulkan kekurangan dari adanya macine learning yaitu menjadi semakin kecil lapangan pekerjaan dikarenakan hampir semua pekerjaan dapat dilakukan oleh mesin, semakin mudah dengan adanya *Machine Learning* membuat banyak manusia menjadi malas dan ketergantungan.

### *Deep Learning*

*Deep Learning* merupakan cabang ilmu dari *Machine Learning* yang berbasis Jaringan Tiruan Syaraf (JST). Dalam *Deep Learning*, sebuah komputer mampu untuk mendeteksi gambar maupun suara. Salah satu algoritma yang terkenal dari *Deep Learning* yaitu *Convolutional Neural Network* (CNN). CNN dirancang untuk mengolah data bentuk 2 dimensi, yaitu gambar atau suara. CNN juga belajar langsung dari citra, sehingga mengurangi beban dari pemrograman.



Gambar 2.4 Perbedaan *Machine Learning* dan *Deep Learning*

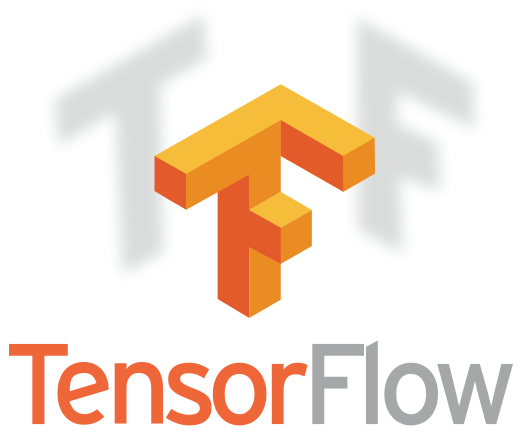
Berikut beberapa perbedaan dari *Machine Learning* dan *Deep Learning* yang disajikan dalam bentuk tabel di bawah ini.

Tabel 2.1 Perbedaan *Machine Learning* dan *Deep Learning*

|  |  |  |
| --- | --- | --- |
| **Perbedaan** | ***Machine Learning*** | ***Deep Learning*** |
| Fundamental | Sub bidang dari AI yang mana mampu mengenali pola berdasarkan data agar bisa berjalan otomatis tanpa harus diprogram ke level lanjut. ML melatih dan menemukan hasil yang paling akurat. | Sub bidang dar ML yang berkaitan dengan *neural network*. Sehingga disebut jaringan tiruan syaraf (JST). Dibuat seperti ML namun lebih ke level algoritma, yang mana algoritma tersebut mereplikasi kinerja otak manusia. |
| Algoritma | Naive Bayes Classifier, SVM (Support Vector Machine), KNN (K-Nearest Neighbor), Neural Network, Hierarchical Clustering, Non- Hierarchical Clustering, K-Means Clustering | DNN (*Deep Neural Network*), ANN (*Artificial Neural Network*), CNN (*Convolutional Neural Network*) |
| Cakupan | Cakupan lebih luas | Kurang luas karena bagian dari *Machine Learning* |
| Data | Membutuhkan data yang cukup | Membutuhkan data lebih banyak karena jika datanya sedikit hasilnya tidak terlalu bagus |
| Proses | Menggunakan algoritma untuk menggali data, belajar menggunakan data tersebut, dan mengambil keputusan berdasarkan data | Menyusun algoritma berlapis-lapis untuk menciptakan jaringan syaraf tiruan yang menyerupai otak manusia, sehingga dapat mempelajari dan membuat keputusan cerdas sendiri |
| Contoh Implementasi | Strategi digital marketing, penerjemah tulisan tangan menjadi teks, pengecekan terjemahan bahasa | Siri dari Apple, Google Assistant dari Google, AlphaGo dari Google |

Dapat disimpulkan bahwa *Deep Learning* tidak lebih dari *Machine Learning* karena merupakan bagian (sub bagian) dari *Machine Learning*. Namun *Deep Learning* melakukan penelitian lebih mendalam dari pada *Machine Learning*. *Deep Learning* sendiri dalam mengolah data membutuhkan banyak data dalam proses pengolahannya dibanding dengan *Machine Learning*.

### TensorFlow



Gambar 2.5 Logo TensorFlow

Diciptakan oleh Google Brain, tensorflow menjadi salah satu library yang terbuka dan gratis yang digunakan dalam membuat *Machine Learning*. Tensorlow banyak menggabungkan model dan algoritma *Machine Learning* termasuk *Deep Learning*. Tensorflow mampu melatih dan menjalankan neural network untuk keperluan klasfikasi, dengan tensorflow juga bisa melakukan deteksi tulisan selanjutnya dengan NLP (Natural Language Processing).

Tensorflow menyediakan semua untuk programmer melalui bahasa Python. Hal ini dikarenakan Python mudah untuk dipelajari dan digunakan, serta menyediakan cara mudah dalam mengekspresikan abstraksi tingkat tinggi dapat digabungkan bersama.

### Keras



Gambar 2.6 Logo Keras

Keras merupakan library yang bertujuan menyederhanakan implementasi dari algoritma *Deep Learning* di atas tensorflow. Penciptaan Keras dirancang untuk manusia bukan mesin. Keras digunakan untuk pengembangan dan pengevaluasian model *Deep Learning*. Selain itu Keras menawarkan API yang sederhana dan konsisten. Fungsi Keras yaitu mengurangi jumlah tindakan dalam implementasi kode serta menjelaskan kesalahan pengguna sehingga dapat ditindaklanjuti.

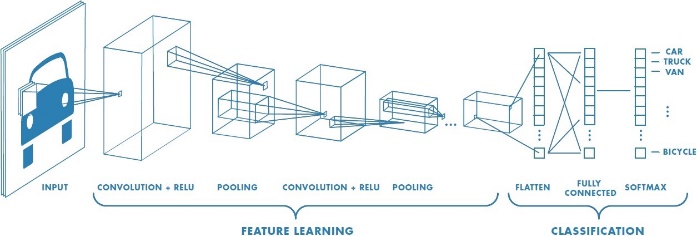
### Python



Gambar 2.7 Logo Python

Python merupakan bahasa pemrograman interpretatif multiguna, di mana bahasa ini mudah dipahami oleh pengguna. Hal ini dikarenakan python lebih menekankan pada keterbacaan kode sehingga lebih mudah memahami syntax. Penggunaan python pada *Machine Learning* yaitu karena bahasa python merupakan bahasa pemrograman tingkat tinggi karena mudah dibaca dan ditulis bahkan oleh orang awam sekalipun. Library yang digunakan python dalam *Machine Learning* di antaranya Numpy, Pandas, Matplotlib, dan Scikit Learn.

### *Convolutional Neural Network*



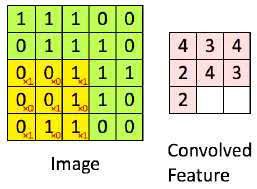
Gambar 2.8 Konsep *Convolutional Neural Network*

*Convolutional Neural Network* merupakan salah satu jenis dari neural network yang biasanya digunakan untuk memproses data bentuk gambar (image). *Convolutional Neural Network* memiliki banyak layer yang difungsikan untuk melakukan filter pada gambar. *Convolutional Neural Network* digunakan untuk mengklasifikasikan data yang sudah terlabel menggunakan metode supervised learning. CNN memiliki lapisan-lapisan yang tersusun oleh *neuron* 3 dimensi (lebar, tinggi, kedalaman).

Secara umum tipe lapisan pada *Convolutional Neural Network* dibedakan menjadi 2, yaitu pada lapisan pertama adalah lapisan ekstraksi fitur (feature extraction layer) di mana letaknya pada awal arsitektur yang mana setiap lapisan tersusun neuron yang terkoneksi dengan lapisan sebelumnya. Lapisan pertama adalah convolutional layer sedangkan lapisan kedua adalah pooling layer. Sedangkan lapisan kedua merupakan lapisan klasifikasi (classification layer) di mana setiap lapisan tersusun atas neuron yang terkoneksi secara penuh (fully connected) dengan lapisan sebelumnya. Layer ini menerima input dari hasil keluaran layer ekstraksi fitur gambar berupa vektor. Hasil keluaran berupa akurasi kelas untuk klasifikasi.

### Operasi Konvolusi

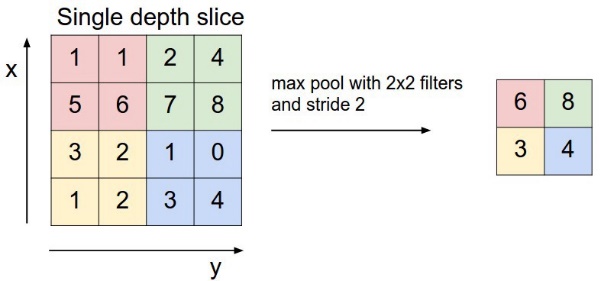
Operasi konvolusi melakukan operasi pada output dari layer sebelumnya. Layer tersebut yang mandasari CNN. Konvolusi berarti mengaplikasikan sebuah fungsi pada output pada fungsi lain secara berulang. Tujuan dilakukannya konvolusi yaitu untuk mengekstrasi fitur dari citra input. Filter yang terdapat pada operasi konvolusi memiliki panjang, tinggi (pixel), dan tebal sesuai dengan channel. Setiap filter akan mengalami pergeseran dan operasi dot antara data masukan dan nilai dari filter.



Gambar 2.9 Operasi Konvolusi

### Pooling Layer

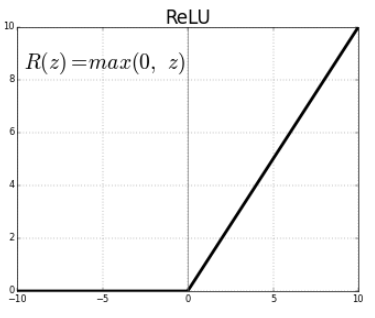
Pooling layer biasanya dilakukan setelah operasi konvolusi. Terdiri dari sebuah filter daengan ukuran dan stride tertentu. Setiap pergeserannya akan ditentukan oleh jumlah stride yang akan digeser pada seluruh feature map atau activation map. Pooling layer yang biasanya digunakan yaitu Max Pooling dan Average Pooling. Pooling layer biasanya disisipkan secara teratur setelah beberapa lapisan konvolusi.



Gambar 2.10 *Pooling Layer*

### Aktivasi ReLu

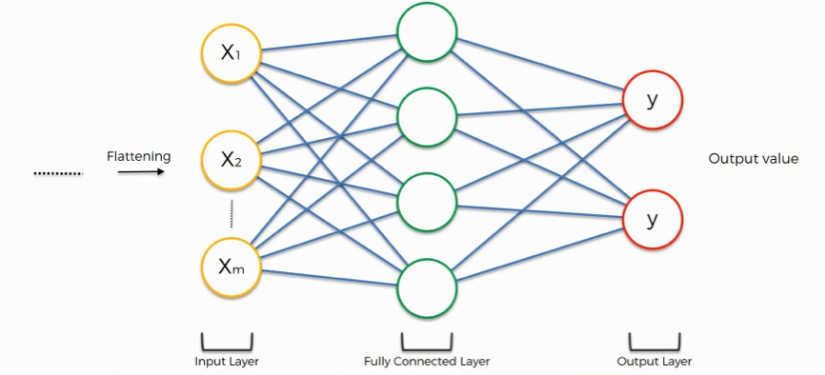
Aktivasi ReLu (Rectified Linear Unit) merupakan lapisan dalam CNN, di mana mengaplikasikan fungsi yang mana melakukan thresholding dengan nilai 0 terhadap nilai pixel pada input data. Aktivasi ini membuat seluruh nilai pixel kurang dari nol pada suatu citra akan dijadikan 0.



Gambar 2.11 Aktivasi ReLu

### Fully-Connected Layer

Layer ini biasa digunakan dalam MLP yang bertujuan untuk transformasi pada dimensi data agar dapat diklasifikasi secara linear. Feature map yang dihasilkan dari feature extraction masih berbentuk multidimensional array, sehingga harus dilakukan flatten atau reshape sehingga bisa digunakan sebagai input dari Fully-Connected Layer. Fully-Connected Layer memiliki beberapa hidden layer, activation function, output laye, dan loss function.



Gambar 2.12 Fully-Connected Layer

### Aktivasi Softmax

Aktivasi softmax merupakan bentuk lain dari logistic regression yang digunakan untuk mengklasifikasi lebih dari dua kelas. Standar yang biasa dilakukan adalah tugas klasifikasi kelas biner. Softmax memungkinkan menghitung peluang untuk semua label. Dari label yang ada akan diambil sebuah vektor nilai bernilai riil dan merubahnya menjadi vektoe dengan nilai antara 0 dan 1 yang jika semua dijumlah akan bernilai 1.

Persamaan pada softmax sebagai berikut.

Notasi fj menunjukkan hasil fungsi untuk setiap elemen ke-j pada vektor keluaran kelas. Argumen Z merupakan hipotesis yang diberikan oleh model pelatihan sehingga bisa diklasifikasi oleh Softmax.

## Kerangka Pemikiran

Berkembangnya teknologi yang mampu menggunakan mesin untuk melakukan pekerjaan manusia. Adanya teknologi *Machine Learning* sangat membantu meringankan pekerjaan manusia. Dalam penelitian yang akan meakukan pengenalan karakter huruf braille menggunakan teknologi *Machine Learning* sehingga dapat membantu orang non tunanetra mampu mengenali buku yang bertuliskan braille.

Berikut merupakan gambaran pola pikir atau konsep yang dilakukan penulis dalam menyusun metode penelitian. Dimulai dari latar belakang hingga pengujian.

Tabel 3.1 Kerangka Pemikiran

|  |  |  |
| --- | --- | --- |
| **Latar Belakang** | | |
| Huruf braille menjadi salah satu alat bantu membaca dan menulis oleh orang tunanetra. Bagi non tunanetra, mengenal karakter huruf braille menjadi suatu hal yang penting saat akan belajar membaca huruf braille. | | |
| **Rumusan Masalah** | | |
| Tingkat akurasi dari penelitian pengenalan karakter huruf braille sehingga meminilimasir terjadinya *error* atau salah prediksi. | | |
| **Tujuan** | | |
| Membantu orang non tunanetra dalam belajar huruf braille. | | |
| **Eksperimen** | | |
| **Data** | **Algoritma** | **Tools** |
| Citra 28x28 pixel dalam citra BW Scale huruf braille setiap karakter dari A-Z dari *website:* <https://www.kaggle.com/datasets/shanks0465/braille-character-dataset?select=Dataset+Description.txt>  Terdapat 1.560 citra karakter abjad A-Z. | Pada penelitian ini menggunakan algoritma *Convolutional Neural Network* (CNN). | Library:   * TensorFlow * Keras * Numpy * Pandas * Matplotlib   Software:   * Python * Google Cloud   TBC |
| **Hasil** | | |