

BAB II

DASAR TEORI

2.1 KAJIAN PUSTAKA

Penelitian pada tahun 2019 yang dilakukan oleh Rusma Eko Fiddy Rizarta dan Donny Avianto dengan judul "Pengenalan Citra Rambu Lalu Lintas Menggunakan Ekstraksi Fitur Momen Warna Dan *K-Nearest Neighbor*". Dimana pada penelitian ini menggunakan metode KNN yang termasuk bagian dari *machine learning*. Penelitian ini mengimplementasikan menggunakan metode KNN dan *Euclidean Distance* untuk mengenali citra rambu lalu lintas yang terpasang di jalan. Bertujuan untuk mengurangi tingkat permasalahan saat berlalu lintas di jalan dan mengetahui jenis rambu lalu lintas yang terpasang di jalan. Dengan menggunakan citra gambar sebanyak 36 buah citra dengan 21 citra latih dan 15 citra uji. Hasil dari pengujian total keseluruhan 15 citra uji mendapatkan tingkat akurasi 91,1% [6].

Penelitian yang dipublikasikan oleh Ari Peryanto, dkk pada tahun 2019 pada Jurnal Ilmiah Teknik Informatika dengan judul "Rancang Bangun Klasifikasi Citra Dengan Teknologi *Deep Learning* Berbasis Metode *Convolutional Neural Network*". Penelitian ini menggunakan cabang dari *Machine Learning* yaitu *Deep Learning*, dimana mengimplementasikan yang dilakukan menggunakan library keras dengan Bahasa pemrograman *python*. Data yang digunakan pada penelitian ini diperoleh melalui internet yang terdapat dua kelas data. Kelas yang digunakan pada penelitian ini yaitu kelas data motor dan kelas data mobil. Masing-masing kelas memiliki data yang sama sebanyak 35 data terdiri dari 30 data data training dan 5 data sebagai testing, dengan presentase 85% untuk data training sedangkan 15% untuk data testing. Pada pengklasifikasi ini mendapatkan tingkat akurasi tertinggi sebesar 98,02% [11].

Masih di tahun 2019 Febian Fitra Maulana dan Naim Rochmawati dalam jurnalnya dengan judul "Klasifikasi Citra Buah Menggunakan *Convolutional Neural Network*" melakukan penelitian dengan metode yang sama hanya khusus penelitian ini mengenai klasifikasi citra pada buah. Dataset yang digunakan berasal dari dataset *Fruit-360* didalamnya terdapat citra sayur dan buah-buahan terdiri dari 15 kelas yang dipakai dari 111 kelas buah dan sayur dengan 56.781 citra untuk total data training dan 56.781 citra untuk data testing. Tingkat akurasi pada penelitian ini

mencapai angka 97,97% dari proses testing dengan 345 citra uji[13].

Penelitian yang dilakukan oleh Febri Hafifah, Sayuti Rahman, dan Munjiat Setiani Asih di tahun 2021 memiliki tema yang sama dengan menggunakan metode *Convolutional Neural Network* namun objek pada penelitian ini yaitu jenis kendaraan, yang berjudul “Klasifikasi Jenis Kendaraan Pada Jalan Raya Menggunakan Metode *Convolutional Neural Network*”. Klasifikasi pada penelitian ini memanfaatkan *Existing CNN* yaitu *Googlenet*, digunakan untuk melakukan *training* sedangkan *Shufflenet* untuk melakukan komputasi pada perangkat selular dengan daya 10-150 MFLOP (*Mega Floating-Point Operations Per second*). Dataset yang digunakan dari penelitian sebelumnya dan rekaman dari CCTV pada jalan raya, diambil di daerah Balmera, Semangi, Semarang, Tanjung mulia sebanyak 633 citra gambar sebagai data uji, dan 1479 citra gambar sebagai data latih. Untuk jumlah data keseluruhan berjumlah 2112 dengan 4 *class*, terdiri dari mobil, truk 3 sumbu, dan 4 truk sumbu. Hasil perbandingan tingkat akurasi pada keempat jenis kendaraan tersebut, masing-masing mendapatkan rata-rata 96,84% untuk *shufflenet* yang mengenali jenis kendaraan sedangkan pada *googlenet* mendapatkan rata-rata 95,88% untuk mengenali jenis mobil[3].

Pada tahun sebelumnya tepatnya di tahun 2020 dilakukan penelitian pada yaitu mengenai batasan kecepatan yang dilakukan oleh Nugraheny Wahyu Try dan Fitri Utaminingrum dengan judul “Sistem Pengenalan Rambu Pembatas Kecepatan Menggunakan Metode *Histogram of Oriented Gradients* dan Klasifikasi *K-Nearest Neighbor* Berbasis Raspberry Pi”. Untuk memperoleh ekstraksi fitur karakteristik dari rambu maka diperlukan metode *Histogram Of Oriented Gradients* (HOG) yang kemudian mengklasifikasi pada rambu menggunakan metode *K-Nearest Neighbor* (KNN). Untuk mendapatkan citra dibutuhkan kamera raspberry pi sebagai deteksi dan pengenalan objek. Rambu batas kecepatan maksimum untuk dikenali yakni 20km, 25km, 30km, 40km, dan 50km, sedangkan minimum 20km yang terdiri dari 650 data latih dan 48 data uji. Pada pengujian dilakukan dengan jarak 3m, 5m, 7m, dan 9m. Hasil tingkat akurasi pada penelitian ini diperoleh pada jarak 3m mendapatkan sebesar 76,25%, jarak 5 meter 97,91%, jarak 7 meter sebesar 79,16%, dan jarak 9 meter 83,33%[14].

Tabel 2.1 Kajian Pustaka

No	Nama Penelitian	Judul Penelitian	Metode Penelitian	Hasil
1	Rusma Eko Fiddy Rizarta dan Donny Avianto (2019)	Pengenalan Citra Rambu Lalu Lintas Menggunakan Ekstraksi Fitur Momen Warna Dan <i>K-Nearest Neighbor</i>	Ekstraksi Fitur Momen Warna Dan <i>K-Nearest Neighbor</i> (KNN)	Pengenalan Citra Rambu Lalu Lintas Dengan menggunakan citra gambar sebanyak 36 buah citra dengan 21 citra latih dan 15 citra uji. Hasil dari pengujian total keseluruhan 15 citra uji mendapatkan tingkat akurasi 91,1%
2	Ari Peryanto (2019)	Rancang Bangun Klasifikasi Citra Dengan Teknologi <i>Deep Learning</i> Berbasis Metode <i>Convolutional Neural Network</i>	<i>Convolutional Neural Network</i> (CNN)	Hasil dari klasifikasi citra, dengan kelas yang digunakan yaitu kelas data motor dan kelas data mobil. Masing-masing kelas memiliki data yang sama sebanyak 35 data terdiri dari 30 data data training dan 5 data sebagai testing. Pada pengklasifikasi ini mendapatkan tingkat akurasi tertinggi sebesar 98,02%
3	Febian Fitra Maulana dan Naim Rochmawati (2019)	Klasifikasi Citra Buah Menggunakan <i>Convolutional Neural Network</i>	<i>Convolutional Neural Network</i> (CNN)	Hasil dari klasifikasi dengan menggunakan dataset yang digunakan berasal dari dataset <i>Fruit-360</i> didalamnya terdapat citra sayur dan buah-buahan terdiri dari 15 kelas yang dipakai dari 111 kelas buah dan sayur dengan 56.781 citra untuk total data training dan 56.781 citra untuk data testing. Tingkat akurasi pada penelitian ini mencapai angka

				97,97% dari proses testing dengan 345 citra uji.
4	Febri Hafifah, Sayuti Rahman, dan Munjiat Setiani Asih (2021)	Klasifikasi Jenis Kendaraan Pada Jalan Raya Menggunakan Metode <i>Convolutional Neural Network</i>	<i>Convolutional Neural Network</i> (CNN), arsitektur jaringan Googlenet dan <i>Shufflenet</i>	Hasil perbandingan tingkat akurasi pada keempat jenis kendaraan, masing-masing mendapatkan rata-rata 96,84% untuk <i>shufflenet</i> yang mengenali jenis kendaraan sedangkan pada <i>googlenet</i> mendapatkan rata-rata 95,88% untuk mengenali jenis mobil
5	Nugraheny Wahyu Try dan Fitri Utaminingrum (2020)	Sistem Pengenalan Rambu Pembatas Kecepatan Menggunakan Metode <i>Histogram of Oriented Gradients</i> dan Klasifikasi <i>K-Nearest Neighbor</i> Berbasis Raspberry Pi	<i>Histogram of Oriented Gradients</i> (HOG) dan Klasifikasi <i>K-Nearest Neighbor</i> (KNN)	Hasil dari pengenalan dengan pengujian yang dilakukan dengan jarak 3m, 5m, 7m, dan 9m. Hasil tingkat akurasi pada penelitian ini diperoleh pada jarak 3m mendapatkan sebesar 76,25%, jarak 5 meter 97,91%, jarak 7 meter sebesar 79,16%, dan jarak 9 meter 83,33%

2.2 DASAR TEORI

2.2.1 Rambu lalu lintas

Arti dari rambu dalam Kamus Besar Bahasa Indonesia yang berarti tanda atau semboyan. Rambu dapat dikatakan tanda untuk melarang, mengingatkan, menunjukkan dan memberitahukan yang harus diketahui dan diikuti[15]. Mengingat pasal 1 ayat 1 dalam Peraturan menteri perhubungan RI nomor PM 13 tahun 2014 mengenai rambu lalu lintas yang berbunyi, Rambu Lalu Lintas adalah bagian perlengkapan Jalan yang berupa lambang, huruf, angka, kalimat, dan/atau perpaduan yang berfungsi sebagai peringatan, larangan, perintah, atau petunjuk bagi Pengguna Jalan[16]. Rambu lalu lintas memiliki banyak jenis dan kegunaan masing-masing.

Secara umum terdapat 4 jenis rambu yang sesuai dengan kebutuhan diantaranya:

- a. Rambu peringatan, rambu ini digunakan untuk peringatan adanya bahaya yang kemungkinan terjadi di jalan yang dapat menimbulkan keselamatan pengguna jalan. Pada dasarnya rambu ini memiliki warna kuning dengan symbol yang berwarna hitam.
- b. Rambu larangan, pada rambu ini larangan untuk bagi pengguna jalan dalam suatu perbuatan atau tindakan yang tidak diperbolehkan dalam berlalu lintas di jalan. Pada dasarnya rambu ini memiliki warna putih dan merah[17].
- c. Rambu perintah, yaitu rambu yang menyatakan perintah yang disampaikan pada pengguna jalan, karena bersifat perintah maka harus dipatuhi. Warna pada rambu ini berwarna biru untuk dasarnya sedangkan putih untuk simbolnya.
- d. Rambu petunjuk, digunakan untuk memandu dan memberikan informasi, seperti penunjuk lokasi atau letak kota, tempat ibadah, lokasi wisata, pom bensin dan lain lain[18].



Gambar 2.1 Rambu lalu lintas[18].

2.2.2 Artificial Intelligence (AI)

Kecerdasan buatan atau *Artificial Intelligence* merupakan bagian dari ilmu pengetahuan komputer yang ditunjukkan dalam perancangan otomatis tingkah laku dalam sistem kecerdasan komputer[19]. Kecerdasan buatan yang

dibuat oleh manusia dalam sistem komputer mendapatkan kemampuan kecerdasan yang dapat menirukan maupun menyerupai beberapa fungsi otak manusia, seperti pengertian bahasa, pengetahuan, pemikiran, pemecahan masalah dan lain sebagainya[20]. Bahkan kecerdasan buatan dapat melebihi manusia dalam mengolah data, menganalisis sehingga mendapatkan tingkat akurasi yang tinggi dalam mengambil keputusan. Cara kerja kecerdasan buatan mulai dari menerima, menyimpan, mengolah, memutuskan, dan mengubah informasi menjadi beragam yang hampir mirip dengan kemampuan manusia dalam memproses informasi, atau sebutan tahapan tersebut disebut *intelligence cycle* pada kecerdasan buatan[21]. Dalam pemecahan masalah kecerdasan buatan memanfaatkan pengetahuan yang maju untuk tugas yang di transfer dari manusia ke dalam komputer. Kecerdasan buatan biasanya dihubungkan dengan ilmu komputer akan tetapi juga terkait dengan bidang lain yang ada di Indonesia saat ini seperti, pendidikan, ekonomi, dan pertahanan nasional, yang dimanfaatkan di industri kesehatan, manufaktur, jasa, dan produk[22]. Kemampuan dalam menggabungkan ilmu pengetahuan dari semua bidang ini pada akhirnya bermanfaat bagi manusia dalam menciptakan suatu kecerdasan buatan[23].

2.2.3 Machine Learning

Secara umum Pembelajaran Mesin (*Machine Learning*) merupakan bagian dari cabang *Artificial Intelligence*. *Machine Learning* adalah ilmu yang memberikan kemampuan program komputer untuk belajar yang dapat mengenali pola-pola di dalam data, dengan tujuan untuk mengubah berbagai macam data menjadi suatu tindakan nyata yang mungkin sesedikit mungkin campur tangan manusia. Tanpa adanya data dalam mengaplikasikan teknik-teknik pada *Machine Learning*, maka algoritma tidak dapat bekerja. Dengan *Machine Learning* dapat menciptakan mesin (komputer) yang belajar dari data data yang ada, kemudian membuat keputusan secara mandiri tanpa perlu diprogram lagi. Terdapat dua istilah yang penting dalam pembangunan model *Machine Learning*, yang pertama data *training* (latih) digunakan untuk melatih algoritma, kedua data *testing* (uji) digunakan untuk melihat performa algoritma yang sudah dilatih sebelumnya ketika mendapatkan data

baru yang belum pernah diketahui[24].

Machine Learning pada dasarnya bertujuan untuk membuat model matematika untuk membantu mesin dalam memahami data. Pada model matematika memiliki berbagai macam variable yang dapat berubah, yang membuat program dibentuk dengan model matematika tersebut memungkinkan program akan otomatis mengikuti perubahan pada data yang berubah.

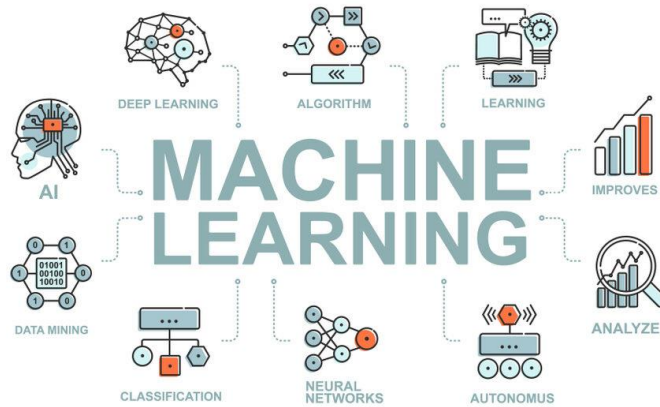
Pada algoritma *Machine Learning* terdapat beberapa katagori berdasarkan klasifikasinya dibagi menjadi dua jenis, diantaranya:

a. *Supervised learning*

Supervised learning yaitu sejenis pembelajaran mesin yang dilatih dibawah pengawasan. Disini maksud dari pengawasan bukan berarti sepenuhnya diawasi oleh manusia, namun manusia harus mempersiapkan hasil atau data yang akan dijadikan masukan dalam proses belajar. Algoritma yang dilatih dilakukan dengan menggunakan data berlabel pada *input* dan *output* yang diinginkan. Hasil dari *output* pada algoritma yang digunakan pada *supervised learning* tidak terdapat campur tangan manusia seperti modifikasi pada program.

b. *Unsubervised learning*

Berbeda dari sebelumnya *unsubervised learning* tidak menggunakan data yang berlabel pada data yang diinputkan. *Unsubervised learning* biasanya membaca data terlebih dahulu yang tidak ada hubunganya dengan pemberian label kemudian program akan memberikan informasi langsung dari data yang sudah ada. Perlu diketahui terdapat dua hal yang perlu dilakukan pada *unsubervised learning* yaitu mentransformasikan *dataset* dan *clustering*. Pada tranformasi *dataset* membuat algoritma program yang dapat mengubah *dataset* yang ada menjadi *dataset* yang mudah dipahami. Pada *clustering* terjadi pemisah data dilakukan oleh algoritma yang menjadi beberapa klaster berdasarkan kesamaan jenis data[25].



Gambar 2.2 *Machine Learning*[26].

2.2.4 *Computer Vision*

Mempelajari suatu cabang ilmu komputer untuk mendapatkan pemahaman tingkat tinggi dari sebuah gambar maupun video digital. Dalam bidang ini berkaitan dengan sistem penglihatan manusia yang sama halnya kemampuan manusia yaitu meniru cara kerja sistem visual (*Human Vision*) manusia yang dapat mengenali objek. Pada penglihatan komputer ini berupaya mengotomatiskan dari penglihatan manusia dalam mengenali objek, yang berkaitan dengan ekstraksi otomatis, analisis serta pemahaman yang berguna dari gambar maupun urutan gambar. Ilmu sebagai penglihatan pada komputer ini memiliki hubungan dengan teori sistem buatan dimana terjadi ekstraksi informasi pada gambar. Untuk mendapatkan data dilakukan pengambilan dalam berbagai bentuk misalnya urutan video, pandangan dari sejumlah kamera atau multi dimensi dari pemeriksa medis[27].

Computer Vision (CV) sendiri adalah upaya manusia membuat perangkat lunak komputer yang dapat mengenali objek di dalam gambar digital. Teknik dari *Computer Vision*, dimana untuk mengestimasi ciri-ciri objek didalam citra, ciri tersebut diukur berkaitan dengan geometri objek.

Terdapat tiga aktivitas dalam proses-proses di dalam *Computer Vision*, diantaranya:

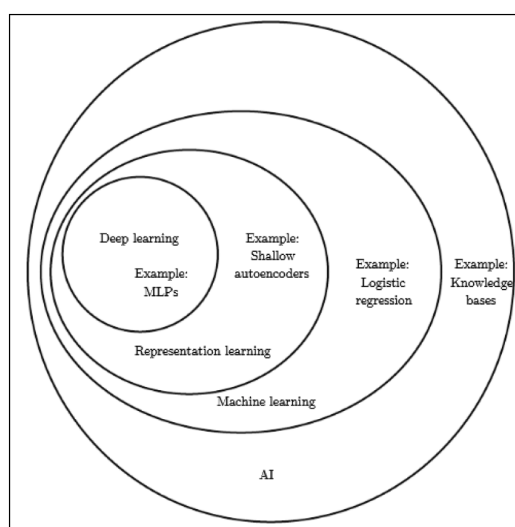
- a. Memperoleh atau mendapatkan citra digital.
- b. Melakukan teknik komputasi sebagai langkah proses atau mengoperasi data citra.
- c. Menganalisis citra dari hasil pemrosesan sebelumnya untuk tujuan tertentu,

contohnya memandu robot, mengontrol peralatan, memantau proses manufaktur, dan lain sebagainya.

Perlu diketahui *Computer Vision* memiliki hubungan dengan pengolahan citra (*Image processing*) sebagai proses awal yang biasanya disebut *preprocessing*. Selain itu terdapat proses untuk menginterpretasi citra pada *Computer Vision* dinamakan pengenalan pola, yang berperan untuk mengenali objek[28].

2.2.5 Deep Learning

Deep Learning dapat diartikan sebagai pembelajaran mesin mendalam merupakan bagian cabang dari *Machine learning* berbasis jaringan syaraf yang memiliki lebih dari dua lapisan (*multi layer*) yang membentuk sebuah tumpukan. Lapisan tersebut merupakan sebuah algoritma yang berfungsi mengklasifikasi perintah yang akan diinputkan sehingga menghasilkan *output*[29]. Dalam *Deep Learning* sebuah komputer dapat belajar mengklasifikasi gambar dan suara secara langsung. Salah satu algoritma *Deep Learning* yang sering digunakan yaitu *Convolutional Neural Network* (CNN), merupakan sebuah pengembangan dari *Multilayer Perceptron* (MPL), digunakan untuk mengolah data dalam bentuk dua dimensi seperti gambar dan suara. Dalam proses komputasi *Deep Learning* menggunakan CPU dan RAM serta memanfaatkan GPU, agar pada saat proses komputasi data yang lebih besar dapat bekerja lebih cepat[30].



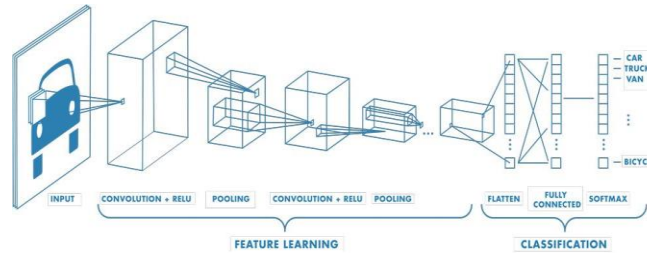
Gambar 2.3 Diagram venn *Deep Learning*[31].

2.2.6 Convolutional Neural Network

Convolutional Neural Network (CNN) adalah salah satu pengembang dari jaringan syaraf tiruan yang biasanya digunakan untuk mendeteksi dan mengenali pada objek yang berupa data gambar[29]. CNN merupakan salah satu jenis dari metode *Deep Neural Network*, dimana didalamnya terdapat tingkat jaringan serta memiliki banyak penerapan yang dilakukan didalam gambar atau citra. Prinsip kerja dari metode CNN setiap neuronya disajikan dalam bentuk dengan ukuran dua dimensi, berbeda dengan MPL yang setiap neuronya memiliki ukuran satu dimensi[32]. CNN dirancang untuk mengklasifikasi data yang berlabel dengan menggunakan metode *supervised learning*, cara kerjanya dengan mengelompokkan suatu data ke data yang sudah ada yang terdiri dari data yang dilatih dan data yang terdapat variabel. Dalam CNN terdapat lapisan yang jumlahnya mencapai puluhan hingga ratusan lapisan, dimana masing-masing lapisan memiliki peran yang sama yakni belajar mendeteksi macam gambar. Lapisan tersebut didalamnya terdapat susunan 3 dimensi (lebar, tinggi, kedalam). Lebar dan tinggi mengacu pada ukuran lapisan, sementara itu kedalam merupakan jumlah lapisannya.

Lapisan pada CNN umumnya dibagi menjadi dua tipe, diantaranya lapisan ekstraksi fitur (*feature extraction layer*) yang lokasinya berada di awal arsitektur dengan tersusunnya atas beberapa lapisan serta setiap lapisan tersusun atas neuron yang terhubung oleh daerah local (*local region*) dari lapisan sebelumnya. Pada lapisan ekstraksi fitur ini memiliki beberapa jenis lapisan, yaitu lapisan yang pertama dinamakan *convolutional layer* dan lapisan kedua adalah *pooling layer*. Pada masing-masing lapisan ditetapkan fungsi aktivasi dengan posisinya di antara jenis lapisan pertama dan jenis lapisan kedua. *Input* gambar akan melewati secara langsung dan diproses pada lapisan ini, kemudian *output* yang dihasilkan pada lapisan ini berupa vektor kemudian untuk diolah pada lapisan berikutnya. Setelah lapisan ekstraksi fitur terdapat lapisan klasifikasi (*classification layer*), yang tersusun beberapa lapisan serta pada lapisan tersusun atas neuron yang terhubung secara penuh dengan lapisan yang lain atau bias disebut *fully connected*.

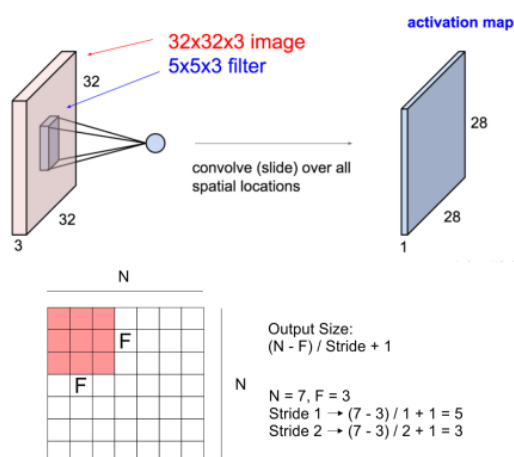
Lapisan ini menerima *input* dari lapisan sebelumnya, data yang berupa vektor akan ditransformasikan seperti *Multi Neural Networks* dan akan ditambahkan di beberapa *hidden layer*[33].



Gambar 2.4 *Convolutional Neural Network*[34].

2.2.7 Convolutional Layer

Convolutional layer (lapisan konvolusi) merupakan bagian blok bangunan inti dari CNN. Sebagian besar komputasi dilakukan pada lapisan ini. Semua data akan mengalami proses konvolusi apabila data menyentuh pada lapisan ini. Pada umumnya, *Convolutional layer* dalam arsitektur CNN menggunakan lebih dari satu *filter*. Setiap *filter* akan dikonversi ke semua bagian data masukan kemudian menghasilkan sebuah *feature map* 2D. Pada filter memiliki panjang, tinggi, dan tebal yang disesuaikan oleh *channel* data masukan. Lapisan konvolusi mengalami yang namanya kompleksitas melalui optimalisasi pada *outputnya*. Hal tersebut dikarenakan oleh *hiperparameters* yang harus ditentukan oleh pengguna diantaranya, *depth*, *stride* dan *pengaturan zero padding*[34].



Gambar 2.5 *Convolutional layer*[34].

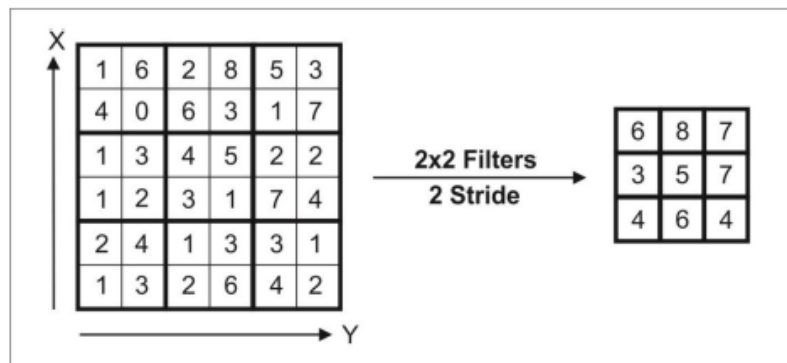
Pada gambar 2.5 terdapat persamaan pada *Convolutional layer* untuk mencari nilai *output (activation map)* yaitu:

$$(N-F)/S + 1 \quad (1)$$

Dimana nilai N merupakan ukuran tinggi dan lebar gambar, nilai F merupakan ukuran filter, dan nilai S besar pergeseran filter pada setiap proses komputasi.

2.2.8 Pooling Layer

Pooling Layer merupakan tahap selanjutnya dengan menerima dari hasil *output* pada layer konvolusi sebelumnya. Pada dasarnya konsepnya sama seperti *convolution layer*. Layer ini terdiri dari *filter* yang memiliki ukuran dan *stride/langkah* tertentu selanjutnya akan bergeser kesemua area *feature map*. Metode yang digunakan pada *pooling layer* menggunakan *maxpooling*. Pada *maxpooling* *output* dari *convolutional layer* akan dibagi dengan menggeser *filter* menjadi beberapa grid memilih nilai maximal seperti pada gambar 2.6 dalam suatu area tertentu dari setiap *grid*. Ukuran *input* matik pada gambar yang dihasilkan lebih kecil dari ukuran aslinya, guna untuk mengurangi dimensi data, yang mempengaruhi jumlah parameter pada langkah berikutnya[29].

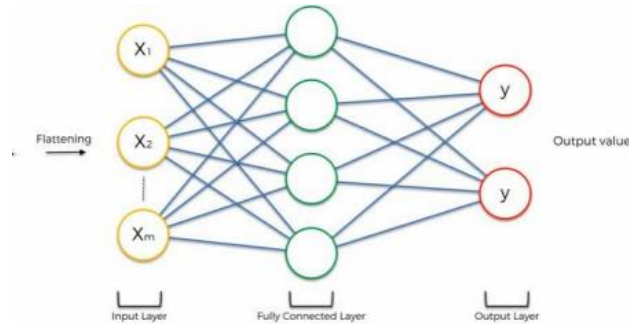


Gambar 2.6 Operasi *Maxpooling*[29].

2.2.9 Fully Connected Layer

Pada lapisan yang terhubung secara penuh atau biasa disebut *fully connected layer* ini memiliki Fungsi sendiri untuk menjadikan multidimensional *array* ke dalam vektor. Dimana neuron memiliki koneksi penuh ke semua aktivasi dalam lapisan sebelumnya, serta neuron tersebut harus ditransformasikan menjadi data satu dimensi dahulu sebelum diinputkan ke dalam disebut *fully connected layer*[35]. Istilah dari perubahan ukuran dimensi ini disebut *flatten*. Sama halnya *neural network* pada umumnya,

dimana *fully connected layer* ini juga terdapat *node* yang saling terhubung dan memiliki fungsi aktivasi. *Output* pada layer ini merupakan sebuah prediksi yang berdasarkan data dari sebelum dimasukan[36].



Gambar 2.7 *fully connected layer*[36].

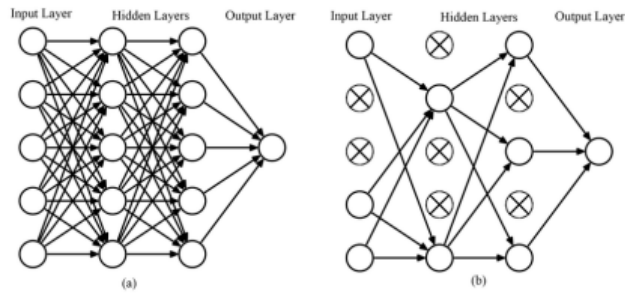
Terdapat persamaan pada *fully connected layer* dapat dituliskan berikut ini:

$$Z_r = \sum_{c=1}^j X_c W_{cr} + b_r \quad (2)$$

Persamaan pada *fully connected layer* variabel r yaitu $1, 2, 3, \dots, R$. Variabel R sendiri merupakan jumlah neuron yang keluar. Sedangkan variabel J yaitu jumlah neuron yang masuk. Penjelasan W_{cr} yaitu nilai neuron input ke- c yang akan menuju neuron *output* ke- r , untuk b_r yaitu bias untuk nilai neuron output ke- r [37].

2.2.10 Dropout

Definisi dari *droupout* yaitu salah satu teknik regulasi (*regulazation*) jaringan syaraf (*neural network*) dimana beberapa neuron akan dipilih secara acak dan tidak digunakan selama proses pelatihan model jaringan. Neuron tersebut akan dinonaktifkan secara acak, hal ini neuron diberhentikan sementara oleh model jaringan. *Dropout* berperan sebagai proses mencegahnya terjadinya *overviting*, kemudian *dropout* juga akan mengurangi jumlah parameter yang digunakan pada model akhirnya membuat poses pelatihan semakin cepat. Menonaktifkan neuron berarti menghilangkan sementara dari sel syaraf atau jaringan tertentu, kemudian dengan diberikannya nilai probabilitasnya yakni nol dan satu. Nilai ini dapat ditentukan ketika memberikan *dropout* pada *hidden layer*[38].



Gambar 2.8 Perbedaan jaringan syaraf sebelum dan sesudah menggunakan *dropout*[38].

2.2.11 Python

Python merupakan salah bahasa pemrograman yang diciptakan oleh Guido van Rossum yang sering digunakan sebagai bahasa skripting dan pemrograman Web. Bahasa *python* dapat dikatakan Bahasa yang bisa menggabungkan kapabilitas, kemampuan, dengan sintaksis kode yang sangat jelas. Bahasa *python* sebagai Bahasa pemograman yang dinamis dilengkapi dengan manajemen memori otomatis. Pada Bahasa ini umumnya digunakan Bahasa skrip meskipun dalam praktiknya dalam penggunaan bahasa ini dalam pemanfaatannya cakupannya lebih luas, misalnya *python* digunakan untuk pengembangan dalam keperluan di dalam perangkat lunak yang dapat beroperasi dalam berbagai system. Bahasa *python* tergolong bahasa tingkat tinggi selain bahasa pemrograman lain seperti pascal, c++, perl,java, dan lain sebagainya[39].

Bahasa *python* memiliki berbagai fungsi dalam bidang seperti, membuat website, aplikasi jaringan, aplikasi bidang robotika, sampai dengan aplikasi kecerdasan buatan. Modul yang akan digunakan dalam pengembangan pada aplikasi tergolong cukup banyak yang digunakan sesuai dengan kebutuhan. Program yang dikembangkan oleh *python* dapat dijalankan dari berbagai system operasi baik *Windows*, *Linux*, *Mac OS*, *Unix*, selain itu dapat dijalankan pada system operasi berbasis *mobile* yaitu *Android* dan *IOS*. *Python* dapat diinstal dengan menggunakan paket distribusi *Anaconda (Anaconda Distribution)* yang dapat digunakan tanpa memerlukan biaya, walaupun di dalam *Anaconda* terdapat *package* lain selain *Python*. Dengan *Anaconda* didalamnya terdapat *tools* yang digunakan untuk menjalankan *python* seperti *jupyter notebook* yang

digunakan sebagai tool dalam pembuatan program *machine learning*[40].



Gambar 2.9 Logo *python*[41].

Bebera contoh *library* pada *python* yang sering digunakan, diantaranya:

a. *Tensorflow*

Tensorflow merupakan perpustakaan (*library open source*) perangkat lunak yang dikembangkan oleh tim Google Brain untuk komputasi pada *machine learning* dalam skala besar, yang mengkombinasikan banyak model pada algoritma *machine learning* dan algoritma *deep learning*. Peran *Python* di dalam *tensorflow* menyediakan API *front-end* untuk membangun aplikasi dengan *framework*, bersamaan dengan menjalankan aplikasi tersebut dengan performa yang tinggi. *Tensorflow* bias melatih maupun menjalankan jaringan syaraf tiruan untuk melakukan sebuah klasifikasi dalam tulisan tangan dan pengenalan pada gambar[42].



Gambar 2.10 Logo *Tensorflow*[43].

b. *Numpy*

Numpy merupakan salah satu *library* pada *python* yang dilengkapi dengan sumber daya yang dapat membantu para. *Data Scientist* dengan merubah *python* menjadi sebuah alat analisi dan pemodelan ilmiah yang kuat. Disisi lain *numpy* kependekan dari *Numerical Python*, memiliki kemampuan membangun objek *N-dimensional array* yang menyerupai dengan *list* pada *python*. Pada *library* ini berfokus pada *scientific computing*. *Numpy array* memiliki keunggulan yang dapat memudahkan melakukan komputas pada

aljabar linear terletak pada operasi pada vektor (1 *dimension array*) dan matriks (2 *dimension array*) serta dalam penggunaan memori relatif lebih sedikit dibandingkan dengan *runtime* yang sangat cepat dengan *list* pada *python*[44].



Gambar 2.11 Logo Numpy[45].

c. *Keras*

Keras adalah sebuah *library* pada jaringan syaraf tiruan *deep learning* yang memiliki tingkat yang tinggi dengan penyusunan menggunakan bahasa *python* serta mampu di atas *tensorflow*, CTNK dan *Theano*. Pada *library* ini menyiapkan sebuah fitur yang dapat digunakan untuk mempermudah pengembangan lebih dalam mengenai *deep learning* dan mendukung pada algoritma *convolutional neural network*, yang dapat dikembangkan kemungkinan besar pengujian yang cepat terhadap CPU (*Central Processing Unit*) dan GPU (*Graphic Processing Unit*)[46].



Gambar 2.12 Logo Keras[47].

d. *Pandas*

Nama lain dari *pandas* yaitu *Python for Data Analysis* merupakan

library yang berlisensi BSD dan *open source* yang menyiapkan struktur data beserta analisis data yang tujuannya mempermudah penggunaannya dan kinerja yang tinggi dalam bahasa pemrograman *python*. Dalam pengertian menurut Rohman, *pandas* adalah sebuah *library python* yang dapat memproses dalam menganalisis data seperti memanipulasi data, persiapan data, serta membersihkan data. Kemampuan dari *pandas* biasanya mengacu pada membaca dan membersihkan data mentah ke dalam suatu format yang sesuai sebelum melakukan analisis. Hal lain dari *pandas* untuk menyesuaikan data dalam perbandingan serta penggabungan pada data set, penanganan data yang hilang, dan lain sebagainya[46].



Gambar 2.13 Logo *pandas*[48].

e. *Scikit-Learn*

Library yang satu ini merupakan sebuah *library* yang sangat populer untuk keperluan machine learning yang menyediakan banyak modul untuk pembuatan model, baik *supervised* maupun *unsupervised learning* yang digunakan secara gratis. *Scikit-Learn* dilengkapi banyak alat yang dapat digunakan untuk melakukan *fitting* model, *preprocessing*, model *selection* dan *evaluation*, *clustering*, *classification* dan lain sebagainya[49].



Gambar 2.14 Logo *Scikit-Learn*[50].

2.2.12 Fungsi Aktivasi *Rectifier Linear Unit* (ReLU)

Rectifier Linear Unit (ReLU) merupakan fungsi aktivasi yang sering digunakan untuk klasifikasi pada CNN. Fungsi aktivasi ini memiliki perhitungan sederhana. Fungsi aktivasi memiliki tugas untuk menentukan kapan *neuron* akan diaktifkan berdasarkan nilai bobot *input*. Fungsi aktivasi ReLU memiliki persamaan matematika, berikut ini:

$$re(x) = \max(0, x) \quad (3)$$

Pada persamaan tersebut, variabel $re(x)$ yaitu fungsi aktivasi itu sendiri, kemudian nilai x fungsi aktivasi merupakan *input*. Sedangkan $\max(0, x)$ yaitu fungsi nilai \max dari variabel 0 dan x [51].

2.2.13 *Confusion Matrix*

Confusion Matrix merupakan suatu metode yang digunakan untuk mengukur performa dari suatu klasifikasi model yang akan dibuat dengan nilai akurasi dari suatu model. Terdapat beberapa istilah dalam *confusion matrix* yang akan dipakai untuk mencari nilai akurasi diantaranya, *true positive* (TP), *true negative* (TN), *false positive* (FP), dan *false negative* (FN).

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

Gambar 2.15 *Confusion Matrix*[37].

Gambar 2.15 menunjukkan rangkuman dari istilah istilah dalam suatu *Confusion Matrix*. Nilai akurasi dalam klasifikasi yaitu presentase ketepatan dari data yang telah di uji pada suatu klasifikasi[37]. Persamaan untuk mengetahui nilai akurasi dari *confusion matrix* sebagai berikut:

$$\text{Akurasi} = \frac{TP+TN}{TP+FP+FN+TN} \quad (4)$$

2.2.14 *Holdout Validation*

Holdout merupakan salah satu cara metode validasi untuk menghasilkan tingkat akurasi pada sebuah *deep learning*, dimana menyediakan sejumlah data yang akan digunakan sebagai data *test*, dan data *train*. Pada *holdout* terdapat proses dimana data *train* dan *test* akan diacak, yang kemungkinan terjadi pada proses ini mengalami *overrepresented* pada klasifikasi. Menandakan bahwa klasifikasi ini dominan dibandingkan dengan klasifikasi lainnya, yang mengakibatkan data tersebut yang terbentuk menjadi representatif. Oleh sebab itu perlunya prosedur *stratification holdout*, yang pada dasarnya pada prosedur ini dapat dijamin bahwa disetiap klasifikasi akan terwakili pada data *train* dan *test* yang terbentuk secara proporsional. Pada validasi ini terdapat prosedur yang dapat dinamakan *repeated holdout*, dimana melakukan proses pengulangan terhadap semua proses *train* dan *test* dari data yang teracak, yang selanjutnya hasil akan dirata-rata[52].