

BAB II LANDASAN TEORI

Penelitian sebelumnya telah mengeksplorasi pengembangan sistem parkir dengan menggunakan algoritma Convolutional Neural Network (CNN). Salah satu penelitian oleh Galahartlambang dkk (2023, no. 114) berjudul "Deteksi Plat Nomor Kendaraan Otomatis dengan Convolutional Neural Network dan OCR pada Tempat Parkir ITB Ahmad Dahlan Lamongan" mengembangkan sistem pengenalan plat nomor kendaraan otomatis menggunakan CNN.

Penelitian lain oleh Felisa dkk (2022, h. 280) berjudul "Perancangan Perangkat Lunak Pengenalan Karakter Plat Nomor Kendaraan Dengan Metode Convolutional Neural Network" menggabungkan CNN dengan Optical Character Recognition (OCR) untuk meningkatkan akurasi pengenalan plat nomor dan memperoleh informasi tambahan dari gambar. Jurnal oleh Mesakh (2021, h. 1) yang berjudul "Sistem Pengenalan Plat Nomor Kendaraan Menggunakan Mask RCNN dan CNN" menggunakan kombinasi metode RCNN dan CNN untuk otomatisasi pengenalan plat nomor.

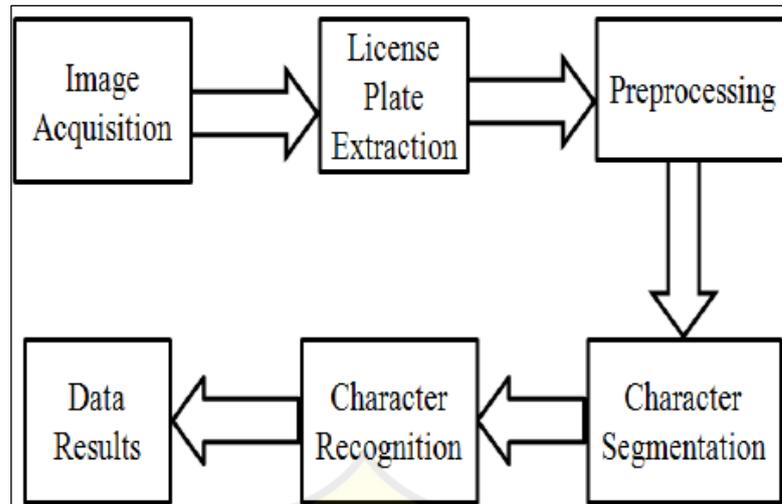
Setiawan dkk (2022, h. 1) dalam penelitiannya berjudul "Deteksi Objek Plat Nomor Kendaraan Dengan Metode CNN" mengembangkan sistem untuk mengurangi kebutuhan pengenalan manual dengan memanfaatkan CNN. Terakhir, Hindarto & Santoso (2021, h. 1) dalam penelitian mereka berjudul "Plat Nomor Kendaraan dengan Convolutional Neural Network" memanfaatkan CNN untuk deteksi objek, yang terbukti efektif dalam pengenalan plat nomor. Penelitian ini bertujuan untuk meningkatkan sistem parkir dengan menambahkan Automatic License Plate Recognition (ALPR) menggunakan CNN, sehingga sistem dapat

memprediksi dan mengklasifikasikan jenis kendaraan berdasarkan plat nomor serta mengenali posisi tanggal dan tahun pada plat tersebut. Hasil penelitian menunjukkan sistem berhasil dalam mengidentifikasi jenis kendaraan dengan baik.

2.1 *Automatic License Plate Recognition (ALPR)*

Sistem Pengenalan Plat Nomor Otomatis (ALPR) adalah teknologi yang memanfaatkan perangkat lunak dan perangkat keras khusus untuk mendeteksi, mengenali, dan mencatat nomor plat kendaraan secara otomatis melalui pemrosesan gambar. Selain itu, ALPR dapat memberikan informasi penting seperti data kendaraan, identitas pemilik, dan status hukum terkait. Teknologi ini diterapkan dalam berbagai industri, termasuk sistem parkir, deteksi lalu lintas kendaraan, dan sistem pembayaran tol berbasis kendaraan (Purwanto & Septiani, 2023, h. 62).

Proses dalam ALPR umumnya terdiri dari empat langkah utama: deteksi plat nomor, segmentasi karakter, ekstraksi data, dan pengenalan karakter. Untuk mempermudah pemahaman, alur proses dalam ALPR dapat dilihat pada Gambar 2.1 di bawah ini.



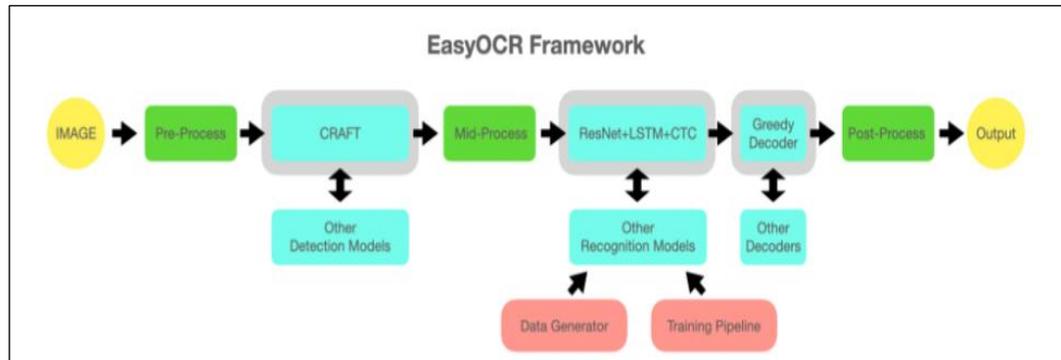
Gambar 2. 1 Alur proses dalam ALPR

2.2 *Optical Character Recognition (OCR)*

Teknologi yang dikenal sebagai Optical Character Recognition (OCR) memungkinkan ekstraksi teks secara otomatis dari gambar atau dokumen yang dipindai. OCR digunakan untuk digitalisasi dokumen, ekstraksi teks, entri data otomatis, dan pengenalan plat nomor. Teknologi ini melibatkan proses deteksi, segmentasi, dan pengenalan karakter dalam gambar (Christoper Nugraha dkk, 2023, h. 606).

2.3 *EasyOCR*

EasyOCR adalah pustaka pemrograman Python yang menyediakan antarmuka mudah untuk tugas-tugas OCR. Pustaka ini menggunakan model yang telah dilatih sebelumnya dan teknik pembelajaran mendalam untuk mengekstraksi teks dari gambar atau citra. EasyOCR mendukung berbagai bahasa dan menunjukkan kinerja yang baik dalam berbagai skenario OCR (Christoper Nugraha dkk, 2023, h. 607). Alur kerja EasyOCR dapat dilihat pada Gambar 2.2 di bawah ini.



Gambar 2. 2 Alur cara kerja EasyOCR

Dari framework yang dijelaskan oleh JaidevAI, pengembang EasyOCR, proses gambar dimulai dengan tahapan pre-processing. Gambar kemudian diteruskan ke model deteksi teks yang bernama Character Region Awareness for Text Detection (CRAFT), yang bertugas mendeteksi area tulisan dan jarak antar huruf. Selanjutnya, gambar yang telah diproses akan masuk ke tahap mid-processing, di mana model pengenalan teks menggunakan CRNN (Convolutional Recurrent Neural Network) yang menggabungkan ResNet, LSTM, dan CTC untuk mengenali karakter.

2.4 Character Region Awareness for Text Detection (CRAFT)

Character Region Awareness for Text Detection (CRAFT) adalah algoritma yang digunakan untuk mendeteksi teks dalam gambar. CRAFT memprediksi dua jenis skor untuk setiap huruf: Region Score dan Affinity Score. Region Score menentukan area dari setiap huruf, membantu dalam pelokalan karakter, sedangkan Affinity Score menunjukkan hubungan antara huruf yang satu dengan huruf berikutnya.

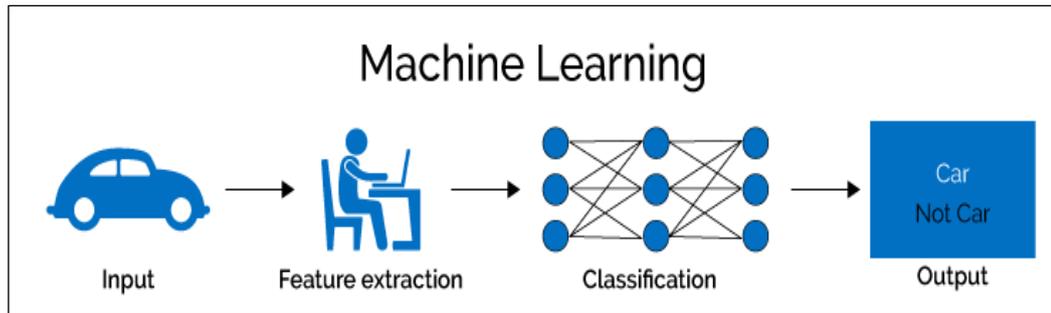
2.5 Artificial Neural Network (ANN)

Jaringan Syaraf Tiruan, atau Artificial Neural Network (ANN), adalah proses yang meniru cara kerja otak manusia dalam memproses informasi berdasarkan sistem saraf biologis manusia, menurut Yuhandri & Mayola (2023, h. 160). Cara kerja ANN adalah sebagai berikut:

1. ANN menerima sinyal input dari sumber eksternal, seperti gambar atau pola, yang kemudian diubah menjadi bentuk vektor.
2. Vektor ini diproses oleh neuron di hidden layer menggunakan fungsi matematis tertentu.
3. Hasil dari hidden layer kemudian diproses di lapisan output untuk menghasilkan output yang diinginkan.
4. Selama proses pembelajaran, ANN belajar dari contoh yang diberikan dan menyesuaikan bobot atau weight pada setiap neuron untuk meminimalkan kesalahan prediksi.

2.6 Machine Learning

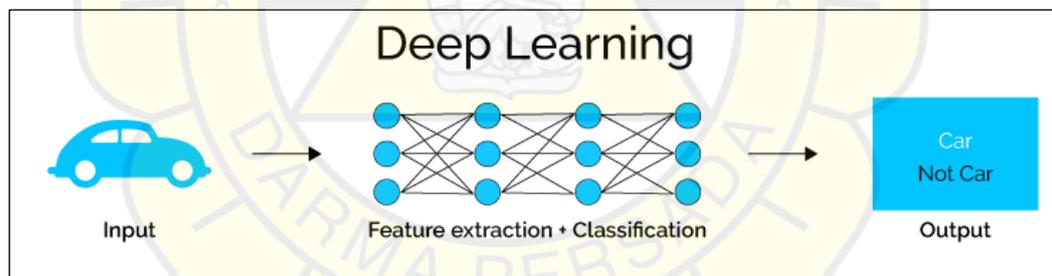
Machine Learning (ML) adalah teknologi yang memungkinkan mesin untuk belajar secara mandiri tanpa instruksi langsung dari pengguna. Machine Learning dikembangkan berdasarkan disiplin ilmu lain seperti statistika, matematika, dan data mining, sehingga mesin dapat menganalisis data dan belajar dari data tersebut tanpa perlu diprogram ulang atau diberikan perintah eksplisit. Tahap-tahap dalam machine learning dapat dilihat pada Gambar 2.2 di bawah ini.



Gambar 2. 3 Tahapan pada Machine Learning

2.7 Deep Neural Network (DNN)

Deep Neural Network (DNN) adalah salah satu model machine learning yang menggunakan Artificial Neural Network (ANN) untuk menangani masalah dengan dataset besar. DNN dapat dianggap sebagai perkembangan dari jaringan syaraf tiruan atau ANN. Dalam DNN, data input diproses secara berurutan melalui setiap lapisan, di mana setiap lapisan melakukan transformasi pada data tersebut dan menghasilkan representasi semantik yang semakin kompleks.

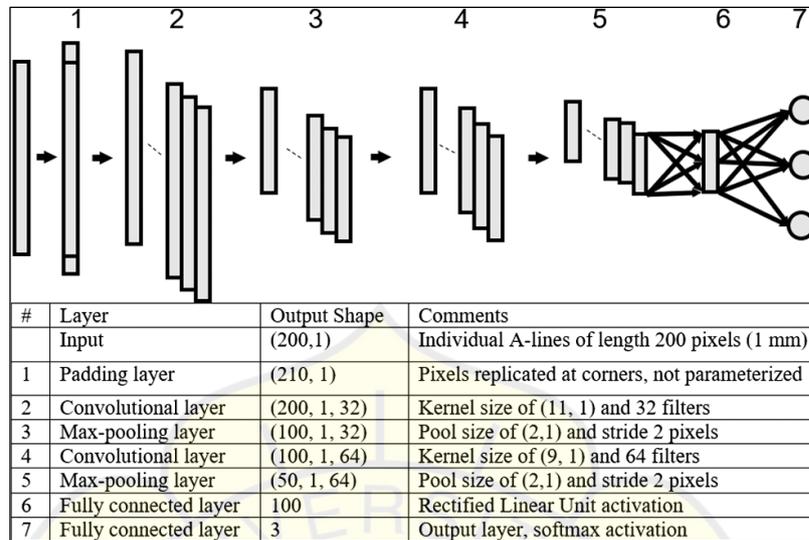


Gambar 2. 4 Tahapan pada Deep Learning

2.8 Convolutional Neural Network (CNN)

Convolutional Neural Network (CNN) adalah salah satu model deep learning yang efektif untuk melatih sistem dengan jumlah data yang besar dan menggabungkan proses ekstraksi fitur dengan klasifikasi. CNN memiliki keunggulan dibandingkan dengan jenis jaringan syaraf tiruan lainnya dalam memproses gambar. Keunggulan ini terletak pada penggunaan convolution layer,

yaitu lapisan yang dirancang untuk mempelajari fitur-fitur kompleks dalam gambar.

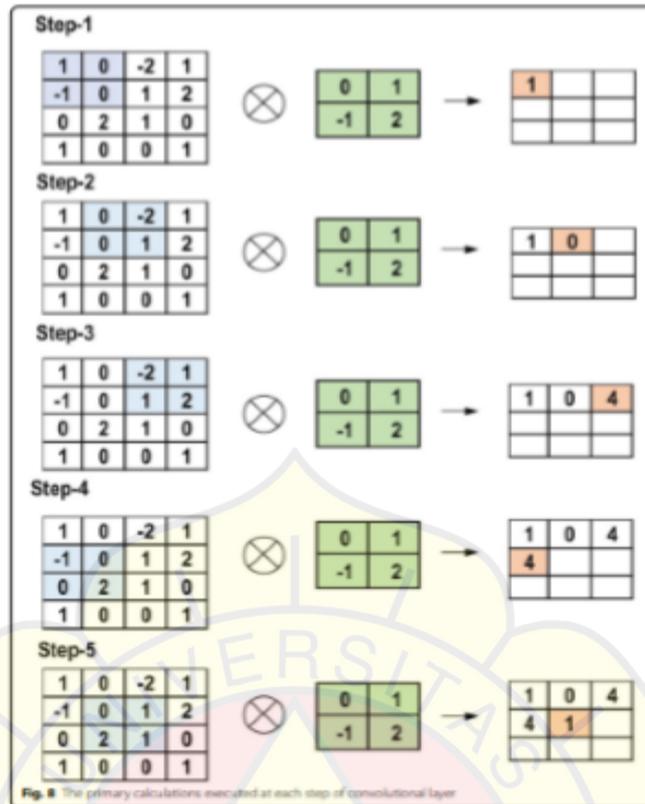


Gambar 2. 5 Arsitektur CNN

Sama halnya dengan yang dilakukan pada penelitian, disebutkan bahwa CNN memiliki beberapa layer utama:

2.9 Convolution Layer

Convolutional Layer adalah lapisan yang terdiri dari beberapa filter konvolusi atau kernel, yang digunakan untuk menghasilkan peta fitur sebagai output. Gambar yang dimasukkan ke dalam proses akan mengalami tahap ekstraksi fitur, di mana gambar tersebut dibagi menjadi beberapa bagian sesuai dengan piksel yang ditentukan oleh parameter tertentu.



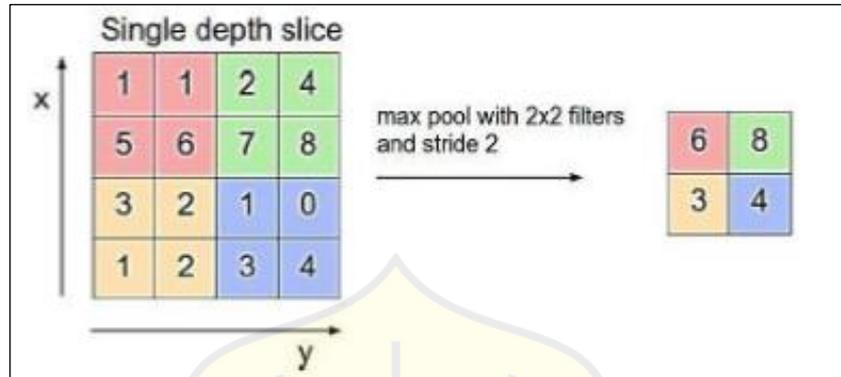
Gambar 2. 6 Proses Convolution Layer

2.10 Subsampling Layer

Subsampling, yang sering disebut sebagai Pooling Layer, adalah tahap kedua dalam ekstraksi fitur setelah convolution. Dalam sebagian besar CNN, metode subsampling yang digunakan adalah max pooling, yang memilih nilai piksel terbesar dari setiap kernel pooling. Dengan cara ini, meskipun jumlah parameter berkurang, informasi penting dari piksel tetap terjaga (Rahmawan dkk, 2023, h. 264).

Metode subsampling yang diterapkan adalah max pooling, yang bekerja dengan membagi output dari lapisan konvolusi menjadi beberapa grid kecil dan memilih nilai maksimum dari setiap grid untuk membentuk matriks citra yang lebih kecil. Nilai maksimum dari setiap grid digunakan untuk menyusun

matriks gambar yang telah disederhanakan, sebagaimana ditunjukkan pada Gambar 2.4 di bawah ini.

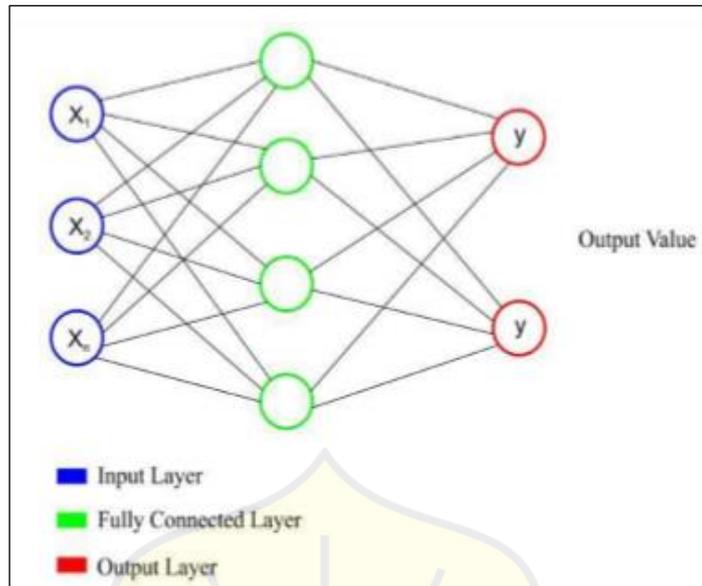


Gambar 2. 7 Operasi Max Pooling

2.11 Fully Connected Layer

Fully connected layer, sebagaimana dijelaskan oleh Nasution (2022, h. 334), adalah bagian dari jaringan neural network di mana gambar yang telah diubah menjadi format satu dimensi (1D) digunakan sebagai input untuk lapisan ini dalam proses prediksi. Lapisan ini umumnya digunakan dalam Multi-Layer Perceptron untuk memproses data agar dapat diklasifikasikan.

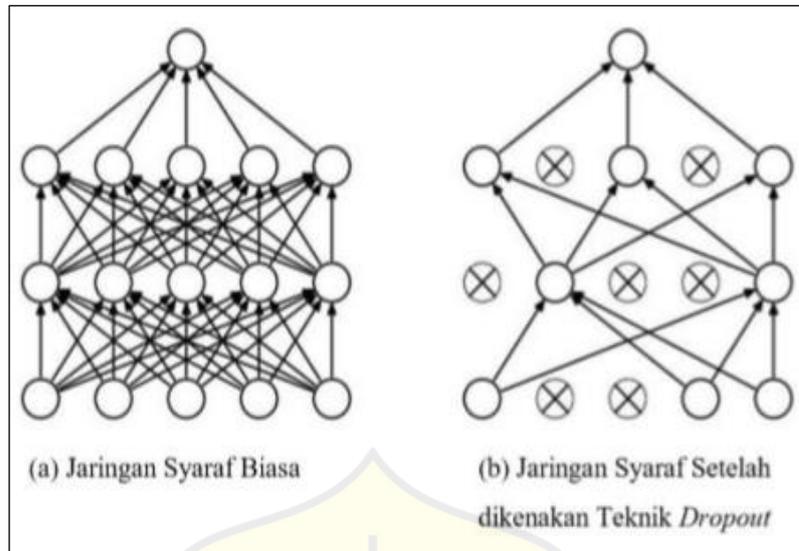
Perbedaan utama antara fully connected layer dan convolution layer adalah bahwa neuron pada convolution layer hanya terhubung ke area tertentu dari input. Sebaliknya, neuron dalam fully connected layer terhubung ke seluruh input. Meskipun kedua lapisan ini menggunakan produk dot, fungsinya tidak terlalu berbeda. Penggambaran tentang fully connected layer dapat dilihat pada Gambar berikut.



Gambar 2. 8 Fully Connected Layer

2.12 Dropout

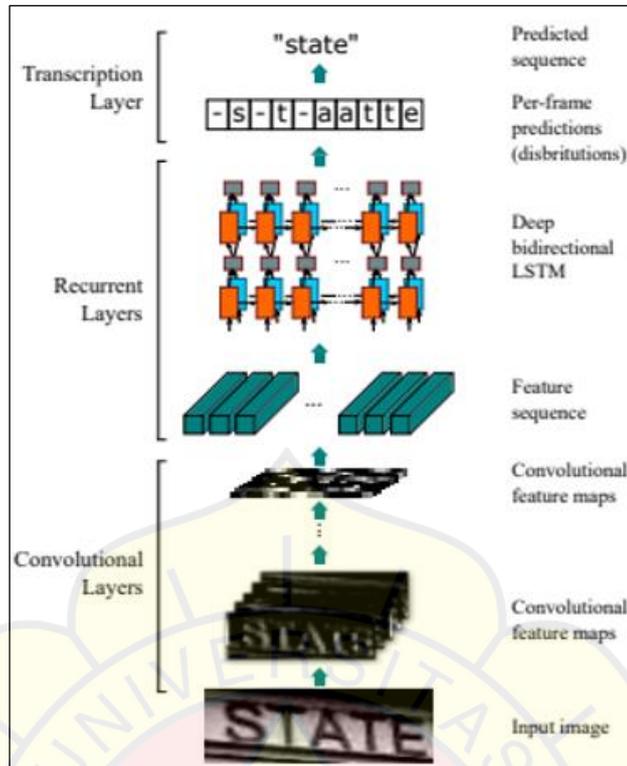
Dropout adalah teknik yang digunakan untuk mengurangi overfitting sekaligus mempercepat proses pelatihan model. Teknik ini melibatkan penghilangan neuron secara acak dari lapisan tersembunyi (hidden layer) dan lapisan yang dapat dilihat (visible layer) dalam jaringan. Dengan menghapus neuron tertentu dari sistem, dropout mencegah neuron tersebut berkontribusi pada proses pelatihan untuk sementara waktu, yang membantu meningkatkan generalisasi model. Neuron yang dihapus dipilih secara acak.



Gambar 2. 9 Tahap *Dropout* CNN

2.13 Convolutional Recurrent Neural Network (CRNN)

Convolutional Recurrent Neural Network (CRNN) adalah model neural network yang mengintegrasikan CNN, RNN, dan CTC untuk memprediksi kata-kata dalam citra yang diberikan. CRNN memadukan keunggulan dari CNN dan RNN, sehingga proses klasifikasi tidak hanya terbatas pada huruf individual, tetapi juga mencakup keseluruhan kata. Arsitektur CRNN dapat dilihat pada Gambar berikut ini.



Gambar 2. 10 Arsitektur CRNN

2.14 Residual Network

Residual Network atau ResNet menggunakan banyak jenis layer yang menjadikan ResNet ini unggul dibandingkan dengan arsitektur lainnya. Arsitektur ResNet ini merupakan pemenang kompetisi ILSVRC dan mencapai nilai kesalahan lima besar hanya 3,6% yang menjadikan ResNet sebagai pengklasifikasi pertama yang mencapai kinerja klasifikasi manusia Gunawan & Al-Rivan (2023, h. 130).

2.15 Long Short Term Memory (LSTM)

Salah satu jenis Recurrent Neural Network (RNN) yang paling umum adalah Long Short Term Memory (LSTM), yang dirancang untuk mengatasi masalah ketergantungan jangka panjang dan sangat cocok untuk data

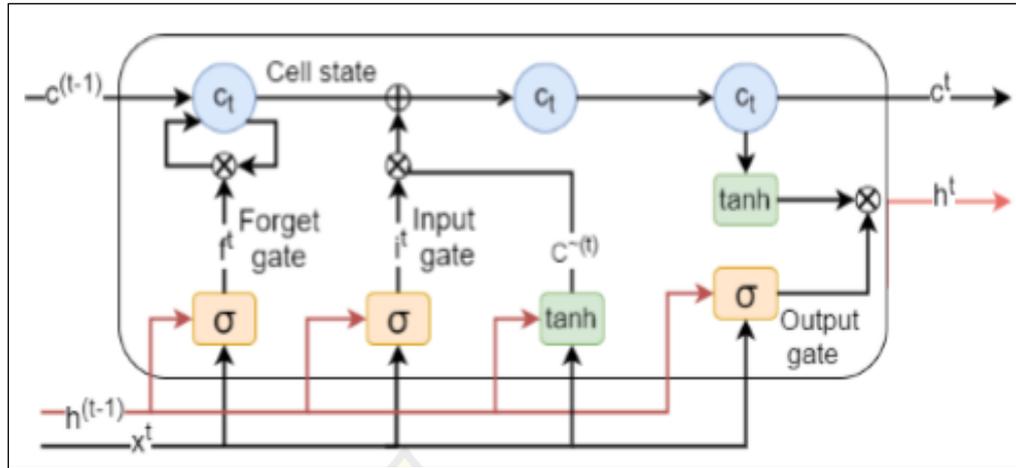
berurutan waktu. Model LSTM mengelola data melalui struktur gate untuk mempertahankan dan memperbarui kondisi dari memory cell. Struktur gate ini terdiri dari tiga komponen: forget gate, input gate, dan output gate (Rahmawati dkk, 2023, h. 69).

Pada forget gate, input yang diterima oleh memory cell diproses melalui transformasi dari keluaran h_{t-1} dan informasi eksternal sebagai input. Matriks weight w_{fw} dan bias b_{fb} digunakan, dengan fungsi aktivasi sigmoid (σ) diterapkan.

Pada lapisan pertama input gate, nilai dari state sebelumnya dan data baru diolah untuk menghasilkan nilai antara 0 dan 1 menggunakan fungsi sigmoid. Kemudian, lapisan kedua menggunakan fungsi aktivasi tan hiperbolik untuk membuat vektor nilai baru yang akan disimpan dalam memory cell.

Pembaruan cell state dilakukan dengan mengalikan nilai dari forget gate dengan cell state sebelumnya, kemudian menambahkan hasil perkalian antara input gate dan nilai tanh dari cell state kandidat. Output gate menghasilkan informasi keluaran menggunakan fungsi aktivasi sigmoid.

Setelah memperoleh nilai dari output gate, cell state diproses melalui fungsi tan hiperbolik dan hasilnya dikalikan dengan output gate dari sigmoid layer. Nilai output akhir cell dihitung menggunakan persamaan h_t dan rumus MAPE disediakan setelahnya. Gambaran susunan dari LSTM dapat dilihat pada Gambar berikut.



Gambar 2. 11 Susunan cell LSTM

2.16 Connectionist Temporal Classification (CTC)

Connectionist Temporal Classification (CTC) adalah sebuah neural network yang digunakan untuk menangani masalah urutan, seperti dalam pengenalan tanda tangan dan suara. CTC mengatasi masalah di mana satu huruf mungkin memerlukan lebih dari satu time-step dalam gambar dengan menggabungkan huruf-huruf yang berulang menjadi satu karakter tertentu. Dalam EasyOCR, CTC berfungsi untuk mendekode output dari LSTM dan menyusunnya menjadi kata dengan probabilitas tertinggi.

2.17 Data Training, Data Testing dan Data Validation

Data yang digunakan dalam pengujian klasifikasi dibagi menjadi tiga kategori: data training, data testing, dan data validation. Data training adalah data yang telah ada sebelumnya dan sesuai dengan fakta. Data testing adalah data yang digunakan untuk mengukur seberapa baik pengklasifikasi dalam melakukan klasifikasi yang benar (Baiq Nurul Azmi dkk, 2023, h. 284). Data

validation adalah data baru yang digunakan untuk mengevaluasi dan mengukur akurasi model yang telah dibangun.

Model klasifikasi yang menggunakan CNN dibangun berdasarkan data training, kemudian kinerjanya diukur menggunakan data testing, dan dievaluasi dengan data validation. Biasanya, pembagian data training, testing, dan validation menggunakan proporsi diskrit, seperti 80:10:10, yang berarti 80% untuk data training, 10% untuk data testing, dan 10% untuk data validation.

2.18 Confusion Matrix

Confusion Matrix adalah metode yang umum digunakan untuk menghitung akurasi dalam data mining, biasanya digambarkan dalam bentuk tabel yang menunjukkan jumlah data testing yang diklasifikasikan dengan benar dan yang salah. Dalam confusion matrix, evaluasi model dilakukan dengan menghitung nilai-nilai seperti True Positive (TP), True Negative (TN), False Positive (FP), dan False Negative (FN). TP adalah jumlah kasus di mana model memprediksi dengan benar bahwa contoh termasuk dalam kelas positif. TN adalah jumlah kasus di mana model memprediksi dengan benar bahwa contoh termasuk dalam kelas negatif. FP adalah jumlah kasus di mana model salah memprediksi kelas positif sebagai negatif, dan FN adalah jumlah kasus di mana model salah memprediksi kelas negatif sebagai positif.

Contoh hasil confusion matrix dapat dilihat pada tabel berikut:

Tabel 2. 1 Contoh Confusion Matrix

<i>Actual Class/Predicted Class</i>	<i>Positive</i>	<i>Negative</i>
<i>Positive</i>	TP	FP
<i>Negative</i>	FN	TN

Actual Class adalah kelas yang sebenarnya pada test set, sedangkan **predicted class** adalah kelas yang diprediksi oleh model classifier yang telah dibangun. Menurut Hakim dkk (2023, h. 207), akurasi merupakan rasio perbandingan antara jumlah prediksi yang benar dengan total keseluruhan data. Berikut adalah rumus untuk menghitung nilai akurasi:

$$\text{Akurasi} = \frac{\text{TP} + \text{TN}}{\text{TP} + \text{TN} + \text{FP} + \text{FN}}$$

Selain akurasi, kita juga dapat menilai kinerja model klasifikasi dengan metrik seperti Recall (Sensitivitas), Precision, dan F1-score, yang semuanya dapat dihitung menggunakan hasil dari confusion matrix. Recall adalah metrik evaluasi yang digunakan untuk mengukur seberapa baik model dalam mengklasifikasikan semua data positif dengan benar.

$$\text{recall} = \frac{\text{TP}}{\text{TP} + \text{FN}}$$

Precision yaitu matriks evaluasi yang cara kerjanya menghitung keakuratan model dalam memprediksi jumlah positif yang benar dari semua prediksi positif yang telah dilakukan

$$\text{precision} = \frac{-b \pm \sqrt{b^2 - 4ac}}{2a}$$

F1-Score adalah gabungan dari Recall dan Precision yang digunakan untuk mengukur kinerja model klasifikasi. Metrik evaluasi ini memberikan informasi yang lebih lengkap dibandingkan akurasi, karena mampu memberikan hasil yang baik bahkan pada dataset yang tidak seimbang (Uly dkk, 2023, h. 62).

$$F1 - Score = 2 \frac{(recall * precission)}{(recall + precission)}$$

2.19 Split Data

Split data merupakan suatu proses membagi dataset yang akan *digunakan* dalam penelitian (Putri dkk 2023, h. 4). Dataset akan dibagi menjadi tiga subset atau bagia, yaitu 80% *data training*, 10% *data testing*, dan 10% *data validation*.

2.20 Tools

2.21.1 Bahasa Pemrograman Python

Bahasa pemrograman *Python* merupakan bahasa pemrograman yang cukup populer (Angelina M. T. I. Sambu dkk 2023, h. 90). Keunggulan yang dimiliki bahasa pemrograman ini dibandingkan dengan bahasa perograman yang lain yaitu mudah dipahami, mempunyai banyak library serta modul yang memudahkan berbagai keperluan, mendukung pemrograman yang berorientasi pada objek, serta sifatnya portable dan dapat dijalankan di berbagai sistem operasi (Rahman dkk 2023, h. 18).

2.21 Django

Django merupakan sebuah *framework* berbasis *Python*. *Django* secara resmi mendukung berbagai basis data seperti *PostgreSql*, *MySQL*, *Oracle* dan *SQLite*. Berdasarkan penjelasan tersebut, penulis menggunakan *MySQL* dalam

pembuatan sistem ini. Dalam penggunaannya, *Django* menerapkan arsitektur *Model View Controller* (MVC) yang terlihat dari berkas-berkas yang dibuat dalam setiap aplikasi yang dibuat.

Di sisi lain, *Django* juga dikenal sebagai *batterly included framework* yang mengklaim sebagian besar ekstensi dan juga pustaka yang diperlukan untuk melakukan bootstrap aplikasi umum yang memberikan pengembang lebih banyak waktu dalam mengimplementasikan logika bisnis sebagaimana yang dinyatakan oleh.

2.22 Script pada website

2.22.1 SweetAlert

SweetAlert adalah pengganti *Alert JavaScript* yang membuat tampilan lebih menarik Ramadhan & Putro (2020, h. 4). Dalam *jurnal* lain yang dilakukan oleh (Simanihuruk dkk 2021, h. 1250) dijelaskan bahwa *SweetAlert* ini adalah salah satu *frameork* dari *JavaScript* yang berlaku sebagai pihak ketiga untuk membuat notifikasi pada suatu aplikasi. Jenis notifikasi yang dapat dihasilkan dari penggunaan *SweetAlert* ini seperti *alert* biasa, *alert* konfirmasi, *alert error* dan lainnya.

2.22.2 WebcamJS

WebcamJS merupakan pustaka JavaScript yang digunakan untuk mengambil gambar melalui kamera komputer pengguna lalu mengirimkan kepada pengguna dalam format JPEG atau PNG. Gambar yang telah diambil tersebut kemudian dapat ditampilkan pada halaman web pengguna, dirender ke dalam kanvas ataupun dikirimkan ke server

pengguna. Pada *WebcamJS* menggunakan *getUserMedia* HTML5, namun menyediakan penggantian *Adobe Flash* otomatis dan tidak terlihat (Jhuckaby, 2017).

2.23 Perangkat Lunak yang dibutuhkan

2.23.1 Visual Studio Code

Menurut Del Sole (2023, h. 2) *Visual Studio Code* merupakan sebuah kerangka kerja untuk membuat aplikasi lintas platform dengan teknologi asli. Ini menggabungkan kesederhanaan editor kode yang kuat dengan alat yang dibutuhkan pengembang untuk mendukung pengembangan siklus hidup aplikasi, termasuk *debugger* dan integrasi kontrol versi berdasarkan Git. Oleh karena itu, *Visual Studio Code* merupakan alat pengembangan yang lengkap, bukan hanya editor kode sederhana.

2.23.2 Command Prompt

Command Prompt atau sering disingkat CMD pada dasarnya merupakan aplikasi command line interpreter atau CLI yang berfungsi untuk menjalankan perintah yang dimasukkan oleh penggunanya. Dengan memasukkan perintah ke CMD, user bisa mendapatkan kendali lebih terkait dengan pengoperasian komputer (Rosalina dkk 2022, h. 30).

2.23.3 Laragon

Laragon merupakan suatu perangkat lunak bersifat open source atau terbuka yang dapat mendukung berbagai sistem operasi, dimana laragon ini bertugas sebagai server virtual atau yang biasa dikenal dengan *localhost* (Andarsyah dkk 2022, h. 113). Dalam literatur lain, disebutkan bahwa laragon ialah *universaldevelopment environment* bagi bahasa pemrograman *Python*, PHP, Java, Ruby, Node.js dan Go (Fatmawati dkk 2021, h. 101).

2.24 Pemodelan dengan UML

Pemodelan *Unified Modeling Language* (UML) membantu proses perancangan sistem untuk mengurangi risiko kegagalan pengembangan program aplikasi. UML menjelaskan bagaimana aktor menggunakan aplikasi, aktivitas setiap aktor, proses dan mekanisme sistem parkir. Perancangan UML membantu perancangan dan pengkodean aplikasi web untuk sistem parkir (Friadi dkk 2023, h. 126).

2.24.1 Use Case Diagram

Use Case Diagram yakni contoh dari perilaku yang ditunjukkan oleh sistem informasi yang akan dikembangkan. *Use case* ini digunakan untuk mengidentifikasi fungsi manakah tersedia di dalam sistem informasi dan siapa yang memiliki otoritas untuk menggunakan fungsi ini (Haris Andri dkk 2023, h. 81).

2.24.2 Activity Diagram

Activity Diagram adalah diagram yang digunakan untuk menunjukkan aktivitas atau aliran kerja dari suatu sistem atau menggambarkan aktivitas sistem bukan apa yang dilakukan aktor (Amelia Sari Lubis dkk 2023, h. 52).

2.24.3 Sequence Diagram

Sequence Diagram adalah suatu diagram yang menggambarkan urutan dari proses yang dilakukan *user* dalam sistem informasi. Secara mudahnya *sequence diagram* yaitu gambaran dari tahap demi tahap yang dilakukan untuk menghasilkan suatu sistem yang sesuai (Amelia Sari Lubis dkk 2023, h. 52).