

LAPORAN
SISTEM PARKIR OTOMATIS INDIKATOR SLOT PARKIR
MENGUNAKAN METODE ANN BERBASIS IoT



Oleh:

Devan Yusfa Sukmadya

210491100027

PROGRAM STUDI TEKNIK MEKATRONIKA
FAKULTAS TEKNIK
UNIVERSITAS TRJOYO MADURA
BANGKALAN

2023

KATA PENGANTAR

Puji syukur kami panjatkan ke hadirat Allah SWT atas segala rahmat dan karunia-Nya yang telah diberikan kepada kita semua, sehingga penyusun bisa menyelesaikan laporan proyek IoT ini. Adapun tujuan disusunnya laporan praktikum ini adalah sebagai syarat untuk memenuhi tugas mata kuliah IoT (*Internet of Things*).

Tersusunnya laporan ini tentu bukan karena buah kerja keras kami semata, melainkan juga atas bantuan dari berbagai pihak. Untuk itu, kami ucapkan terima kasih kepada semua pihak yang turut membantu dalam menyelesaikan laporan praktikum ini diantaranya dosen pengampu, serta teman-teman yang turut berpartisipasi dalam penyusunan laporan praktikum ini.

Kami selaku penyusun menyadari bahwa laporan ini belum bisa dikatakan sempurna. Untuk itu, kami menerima kritik dan saran yang membangun dari pembaca sekalian, agar kedepannya laporan proyek ini dapat disempurnakan. Semoga laporan proyek ini bermanfaat untuk kita semua.

Bangkalan, 28 November 2023

Kelompok III

DAFTAR ISI

COVER	i
KATA PENGANTAR.....	i
DAFTAR ISI.....	ii
BAB I PENDAHULUAN.....	1
1.1. Latar Belakang	1
1.2. Rumusan Masalah	2
1.3. Tujuan Pembuatan.....	2
BAB II LANDASAN TEORI	3
2.1. Tinjauan Pustaka.....	3
2.1.1. Arduino R4 WIFI	3
2.1.2 LED (<i>Light Emitting Diode</i>)	4
2.1.3 Sensor Infrared.....	5
2.1.4. Arduino IoT Cloud.....	6
2.1.5. Arduino Create agent	7
BAB III METODE PENELITIAN	10
3.1. Metode <i>Artificial Neural Network</i> (ANN)	10
3.1.1. Struktur Pada <i>Artificial Neural Network</i>	11
3.1.2. <i>Feed-forward</i> Pada <i>Artificial Neural Network</i>	12
3.1.3. <i>Backpropagation</i> Pada <i>Artificial Neural Networks</i>	13
3.2. Cara Kerja Metode <i>Artificial Neural Network</i> (ANN).....	14
3.3. Gambar Rangkaian Projek	16
3.5. Cara Kerja Rangkaian	18
BAB IV HASIL PENELITIAN DAN PEMBAHASAN.....	20
4.1. Hasil Pengujian Sensor Infrared pada Setiap Slot Parkir.....	20
4.2. Simulasi Percobaan Rangkaian	20
4.3. Implementasi Metode ANN pada Arduino R4 WiFi	22
4.4. Struktur Pada <i>Artificial Neural Network</i>	22
4.5. <i>Feed-forward</i> Pada <i>Artificial Neural Network</i>	23
4.6. <i>Backpropagation</i> Pada <i>Artificial Neural Networks</i>	24
4.7. Cara Kerja Metode <i>Artificial Neural Network</i> (ANN).....	26

4.8.	Integrasi Sistem dan Pengujian Keseluruhan	27
4.9.	Efisiensi dan Keandalan Sistem	28
4.10.	Implikasi dan Rekomendasi	28
4.11.	Evaluasi Pengaruh Jaringan WiFi pada <i>Respons</i> Sistem	29
4.12.	Analisis Data Hasil Monitoring dan Pengendalian Sistem	30
4.13.	Perbandingan dengan Sistem Konvensional	31
4.14.	Diskusi Hasil dan Kesimpulan	32
BAB V KESIMPULAN		34
DAFTAR PUSTAKA		35
LAMPIRAN.....		36

BAB I

PENDAHULUAN

1.1. Latar Belakang

Semarak penggunaan mobil di daerah perkotaan cukuplah sangat pesat dan tidak adanya penyediaan area parkir yang memadai untuk parkir dari mobil tersebut. Oleh karena itu, banyaknya mengakibatkan para pengendara yang parkir di sembarangan tempat. Sistem parkir yang ada di daerah perkotaan hanya memanfaatkan area parkir dan petugas parkir yang mengawasi atau mengatur. Pengendara harus bersusah untuk mencari tempat parkir yang kosong atau tidak ditempati, namun apabila petugas atau dari pengendara tidak memperhatikan tentang adanya daya tampung area parkir. Teknologi yang terus berkembang pada bidang sistem kontrol. Teknologi tersebut sering digunakan dalam pabrik-pabrik industri, teknologi sistem kontrol dapat mengontrol mesin atau sistem secara otomatis untuk meringankan pekerjaan manusia, dengan adanya permasalahan pada pengendara ketika ingin melihat tersedianya area parkir tersebut dapat dibuat dengan merancang sebuah sistem parkir yang sederhana dengan menggunakan sebuah mikrokontroler yaitu Arduino R4 WIFI sebagai mikrokontroler, IR sensor sebagai pendeteksi mobil pada area parkir, dan LED sebagai indikator untuk mengetahui bahwa area tersebut kosong.

Penggunaan Arduino sebagai *platform* pengembangan merupakan pilihan yang sangat populer karena kemudahan penggunaannya. Arduino dilengkapi dengan berbagai *pin input/output* yang memungkinkan pengembang untuk menghubungkan berbagai sensor dan aktuator dengan mudah. Selain itu, bahasa pemrograman Arduino yang *open source* membuatnya dapat diakses oleh berbagai kalangan pengembang, dari pemula hingga yang berpengalaman. Proyek ini adalah contoh implementasi jaringan saraf buatan (ANN) pada perangkat Arduino menggunakan platform Arduino IoT Cloud. Jaringan saraf buatan digunakan untuk mengambil data dari sensor inframerah dan memberikan kontrol kepada lampu LED berdasarkan hasil inferensi. Proyek ini memberikan gambaran tentang bagaimana menggabungkan jaringan saraf buatan ke dalam perangkat Arduino untuk aplikasi berbasis sensor.

1.2. Rumusan Masalah

Berdasarkan uraian latar belakang masalah diatas, maka permasalahan yang akan dikaji pada laporan tugas berikut ini dapat dirumuskan sebagai berikut:

1. Bagaimana cara untuk mengetahui area parkir dan nomor parkir yang kosong dengan menggunakan sistem kontrol yang dapat memudahkan pengendara?
2. Bagaimana agar metode ANN dapat mengubah nilai dari sensor sehingga dapat mengkontrol LED

1.3. Tujuan Pembuatan

Berikut di bawah ini merupakan tujuan yang hendak dicapai dalam merangkain sebuah project mata kuliah *internet of things* ini adalah:

1. Membuat sistem kontrol pada area parkir menggunakan mikrokontroler arduino
2. Membuat para pengendara mengetahui dengan adanya area parkir yang kosong
3. Mengadaptasi metode ANN atau *artificial intelegant neural networks*

BAB II|

LANDASAN TEORI

2.1. Tinjauan Pustaka

Berikut dibawah ini merupakan tinjauan pustaka untuk menunjang pembuatan pada tugas mata kuliah *internet of things* dengan judul sisitem parkir otomatis menggunakan metode *artificial intelegent neural networks* berbasis IoT sebagai berikut.

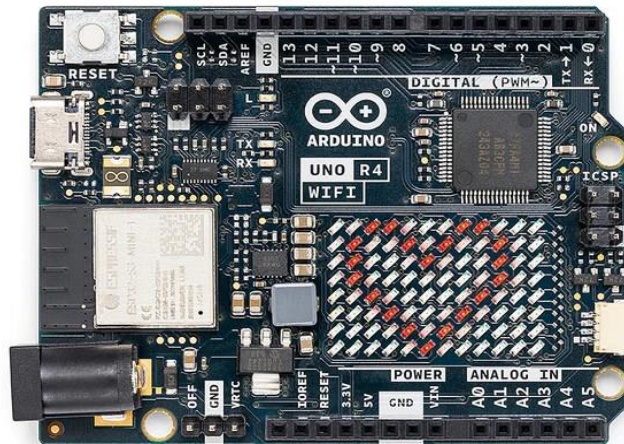
2.1.1. Arduino R4 WIFI

Arduino R4 WiFi merupakan sebuah papan pengembangan yang dirancang oleh Arduino dengan fokus pada kemampuan konektivitas nirkabel. Papan ini memanfaatkan teknologi WiFi yang terintegrasi untuk memungkinkan pengguna mengakses jaringan tanpa kabel, membuka pintu untuk berbagai aplikasi dalam lingkup *Internet of Things* (IoT) dan proyek elektronika yang memerlukan konektivitas online.

Arduino R4 WIFI dilengkapi dengan mikrokontroler yang dapat diprogram, memberikan fleksibilitas tinggi dalam merancang dan mengontrol perangkat elektronik. Arduino R4 WiFi mendukung berbagai sensor dan modul eksternal, memungkinkan integrasi yang lebih kompleks dengan perangkat keras tambahan. Dengan menggunakan bahasa pemrograman Arduino yang dapat gunakan secara *open source*, pengembang dapat dengan mudah membuat kode untuk mengendalikan papan ini dan memanfaatkan kemampuan WiFi-nya.

Fitur tambahan papan Arduino R4 WiFi mencakup berbagai pin *input/output* (I/O) untuk menghubungkan komponen tambahan, antarmuka USB untuk pemrograman dan transfer data, serta dukungan untuk berbagai protokol komunikasi. Semua ini membuat Arduino R4 WiFi menjadi pilihan populer di kalangan pengembang yang ingin merancang proyek-proyek inovatif yang terhubung ke internet, mulai dari aplikasi monitoring lingkungan hingga sistem otomatisasi rumah yang terintegrasi dengan *cloud*. Keunggulan konektivitas nirkabel membuka peluang baru untuk eksplorasi dan pengembangan solusi berbasis IoT yang dapat diakses dan dikendalikan dari mana saja melalui jaringan

WiFi. Berikut dibawah ini merupakan Arduino R4 WIFI yang ditunjukkan pada Gambar 2.1.



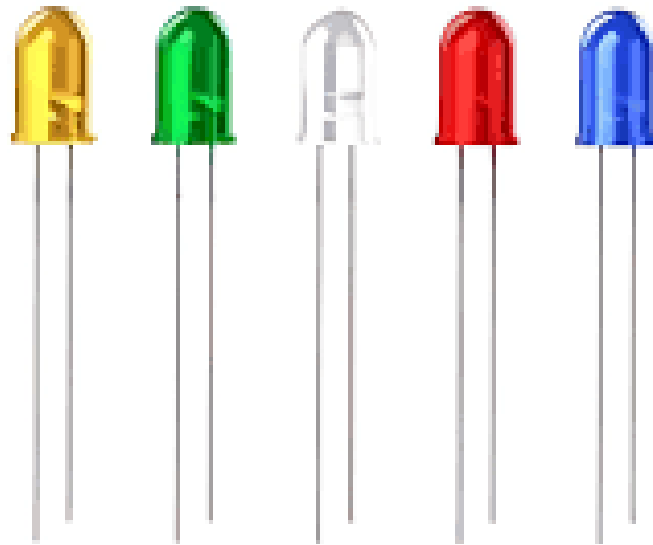
Gambar 2.1. Arduino R4 WIFI

2.1.2 LED (*Light Emitting Diode*)

Light Emitting Diodes (LEDs) adalah semikonduktor yang menghasilkan cahaya ketika arus listrik melewati mereka. Mereka bekerja berdasarkan fenomena *elektroluminescence*, di mana ketika elektron dan lubang bergabung dalam material semikonduktor LED, mereka melepaskan energi dalam bentuk foton, menghasilkan cahaya yang terlihat. Terdapat berbagai jenis dan warna LED, termasuk merah, hijau, biru, dan putih, yang dapat dihasilkan berdasarkan bahan dan struktur yang digunakan dalam pembuatannya.

Salah satu keunggulan utama LED dibandingkan dengan lampu tradisional seperti bola lampu *incandescent* adalah efisiensinya. LED jauh lebih efisien dalam menghasilkan cahaya per *watt* daya listrik yang digunakan dibandingkan dengan lampu *incandescent* yang menghasilkan cahaya dengan memanaskan filamen kawat. Selain itu, umur hidup LED jauh lebih panjang, seringkali bertahan puluhan ribu jam, yang mengurangi kebutuhan penggantian yang sering. Mereka juga lebih tahan lama dan tahan terhadap guncangan serta getaran, menjadikannya pilihan yang ideal untuk berbagai aplikasi, mulai dari penanda lampu pada perangkat elektronik hingga penerangan dalam lampu jalan dan televisi layar datar. Karena efisiensinya dan fleksibilitasnya, LED telah menjadi pilihan populer dalam

berbagai kebutuhan penerangan dan pencahayaan sehari-hari. Berikut ini merupakan LED (*Light Emitting Diode*) yang ditunjukkan pada Gambar 2.2.



Gambar 2.2. LED (*Light Emitting Diode*)

2.1.3 Sensor Infrared

Sensor inframerah adalah perangkat elektronik yang digunakan untuk mendeteksi cahaya inframerah (IR) dalam spektrum elektromagnetik. Sensor ini bekerja dengan mendeteksi perubahan dalam intensitas cahaya inframerah yang diterimanya, dan kemudian mengonversinya menjadi sinyal listrik yang dapat diinterpretasikan oleh perangkat elektronik. Sensor inframerah memiliki banyak aplikasi yang beragam, termasuk dalam sistem pengendalian jarak jauh, kendaraan otonom, pengukuran jarak, deteksi gerakan, dan pengendalian peralatan elektronik. Mereka juga sering digunakan dalam perangkat seperti *remote control*, kamera keamanan, dan sensor otomatisasi industri.

Sensor inframerah menggunakan fotodetektor khusus yang peka terhadap cahaya inframerah. Ketika cahaya inframerah mengenai permukaan sensor, fotodetektor akan menghasilkan arus listrik yang sebanding dengan intensitas cahaya yang diterima. Kemudian, sinyal ini dapat diolah oleh sirkuit elektronik untuk mengidentifikasi perubahan dalam lingkungan, seperti pergerakan objek atau perbedaan suhu. Sensor inframerah dapat bekerja dalam berbagai jarak, mulai dari jarak dekat hingga jarak jauh, tergantung pada jenis sensor dan aplikasinya. Mereka

sangat berguna dalam pengembangan perangkat yang memerlukan deteksi atau pengukuran berdasarkan cahaya inframerah, membuatnya menjadi komponen penting dalam berbagai teknologi modern. Berikut ini merupakan sensor infrared yang akan ditunjukkan pada Gambar 2.3.



Gambar 2.3. Sensor Infrared

2.1.4. Arduino IoT Cloud

Arduino IoT Cloud adalah platform yang dirancang untuk mempermudah pengembangan dan pengelolaan proyek Internet of Things (IoT) menggunakan perangkat keras Arduino. Platform ini menyediakan lingkungan yang terintegrasi dan ramah pengguna untuk menghubungkan perangkat Arduino ke cloud, memungkinkan pengguna untuk mengakses dan mengontrol perangkat mereka dari mana saja melalui internet.

Salah satu fitur utama dari Arduino IoT Cloud adalah kemampuannya untuk mengelola perangkat secara otomatis melalui antarmuka web yang sederhana. Pengguna dapat dengan mudah mengonfigurasi dan memantau perangkat Arduino mereka, serta membuat aturan dan logika kendali untuk mengotomatiskan *respons* perangkat terhadap berbagai kondisi. Ini memungkinkan implementasi proyek IoT yang lebih canggih dan terstruktur.

Arduino IoT Cloud juga menyediakan layanan terhubung ke perangkat keras Arduino, memungkinkan data yang dihasilkan oleh sensor atau perangkat diintegrasikan ke dalam aplikasi atau layanan lainnya. Dengan demikian, pengguna dapat membuat proyek-proyek yang lebih kompleks, seperti sistem pemantauan atau kendali rumah pintar. Dengan menyederhanakan proses pengembangan dan menghadirkan kemudahan pengelolaan, Arduino IoT Cloud menjadi pilihan yang

menarik bagi pengembang dan hobiis yang ingin eksplorasi lebih lanjut dalam dunia *Internet of Things*.



Gambar 2.4. Arduino IoT Cloud

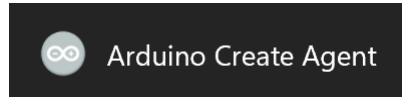
2.1.5. Arduino Create agent

Arduino Create Agent adalah perangkat lunak yang memungkinkan komunikasi yang lancar antara perangkat keras Arduino dan Arduino Create, platform pengembangan berbasis cloud dari Arduino. Fungsinya sangat penting dalam memfasilitasi proses pengunggahan kode (sketch) dari Arduino Create ke perangkat Arduino yang terhubung ke komputer. Dengan menggunakan Arduino Create Agent, pengguna dapat mengelola dan mengendalikan perangkat Arduino mereka dari antarmuka web Arduino Create tanpa kesulitan.

Perangkat lunak ini memastikan bahwa pengguna tidak perlu lagi mengunduh atau mengelola driver khusus secara manual untuk berbagai jenis perangkat Arduino. Arduino Create Agent berfungsi sebagai perantara yang menghubungkan perangkat keras Arduino dengan *platform cloud*, membuat pengaturan dan konfigurasi perangkat menjadi lebih mudah dan cepat. Hal ini juga memungkinkan pembaruan firmware dan pustaka perangkat keras Arduino secara otomatis, mengurangi kompleksitas dan waktu yang dibutuhkan untuk memulai proyek.

Selain itu, Arduino Create Agent mendukung pengembangan lintas platform, memungkinkan pengguna untuk bekerja dengan perangkat keras Arduino dari berbagai sistem operasi, termasuk Windows, macOS, dan Linux. Dengan demikian, perangkat ini menciptakan pengalaman pengembangan yang seragam

dan dapat diakses dari berbagai jenis komputer. Kesederhanaan dan kemudahan penggunaan Arduino Create Agent menjadikannya komponen kunci dalam ekosistem Arduino, mendukung visi Arduino dalam memberdayakan komunitas untuk menciptakan proyek elektronika yang inovatif dan terhubung.



Gambar 2.5. Arduino Create Agent

BAB III

METODE PENELITIAN

3.1. Metode *Artificial Neural Network* (ANN)

Artificial Neural Network (ANN), atau Jaringan Saraf Buatan, adalah paradigma pengolahan informasi yang terinspirasi oleh struktur dan fungsi jaringan saraf biologis. ANN terdiri dari neuron buatan yang saling terhubung dan diorganisir dalam lapisan-lapisan. Setiap neuron menerima sejumlah *input*, mengalami transformasi matematis menggunakan bobot koneksi yang dapat diubah, dan menghasilkan *output* yang menjadi *input* bagi neuron-neuron pada lapisan berikutnya. Konfigurasi dan hubungan antar neuron ini memungkinkan ANN untuk memodelkan dan memahami pola-pola kompleks dalam data.

Proses pelatihan ANN melibatkan penggunaan algoritma seperti *backpropagation*, yang melakukan penyesuaian bobot berdasarkan perbandingan antara *output* yang dihasilkan dan nilai yang seharusnya. Proses ini berulang hingga model mencapai tingkat akurasi yang memadai. ANN memiliki kemampuan untuk belajar dari data pelatihan dan kemudian menerapkan pengetahuannya untuk memproses data baru, menjadikannya alat yang kuat untuk tugas-tugas seperti pengenalan pola, klasifikasi, dan prediksi.

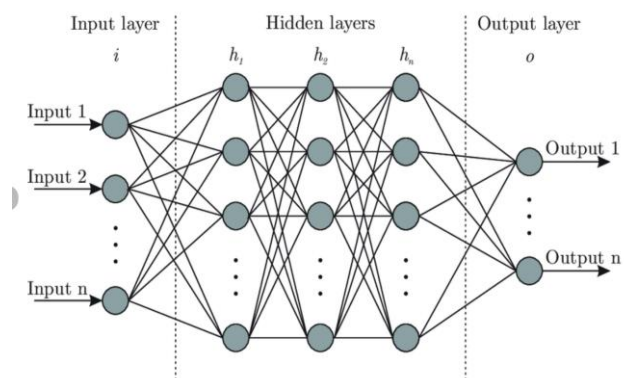
Keunggulan ANN meliputi kemampuannya untuk menangani masalah non-linier dan kompleks, serta fleksibilitasnya dalam berbagai domain aplikasi seperti pengolahan citra, pengenalan suara, dan analisis data besar. Meskipun demikian, terdapat beberapa tantangan, termasuk interpretabilitas rendah di mana hasil keputusan model sulit dijelaskan, dan kebutuhan akan jumlah data pelatihan yang signifikan untuk menghasilkan model yang akurat.

$$net_j^{(h)} = \sum_{i=1}^I w_{ij}^{(h)} x_i^{(h-1)} + b_j^{(h)},$$

Gambar 3.1. *Equations* Pada Metode ANN

3.1.1. Struktur Pada *Artificial Neural Network*

Struktur pada *Artificial Neural Network* (ANN) mencerminkan organisasi dan hubungan antar elemen-elemen utama yang membentuk jaringan tersebut. ANN terdiri dari tiga jenis lapisan utama, yaitu lapisan *input*, lapisan tersembunyi, dan lapisan *output*. Lapisan *input* menerima data atau fitur dari *input* eksternal dan meneruskannya ke lapisan tersembunyi. Lapisan tersembunyi, yang dapat terdiri dari satu atau beberapa lapisan, melakukan transformasi matematis terhadap *input* menggunakan bobot koneksi yang dapat diubah. Setiap neuron dalam lapisan tersembunyi memiliki fungsi aktivasi yang menentukan sejauh mana informasi diumpungkan ke lapisan selanjutnya. Lapisan *output* menghasilkan hasil akhir atau prediksi berdasarkan hasil transformasi yang telah dilakukan oleh lapisan tersembunyi.



Gambar 3.2. *Input, Hidden Dan Output Layers Pada ANN*

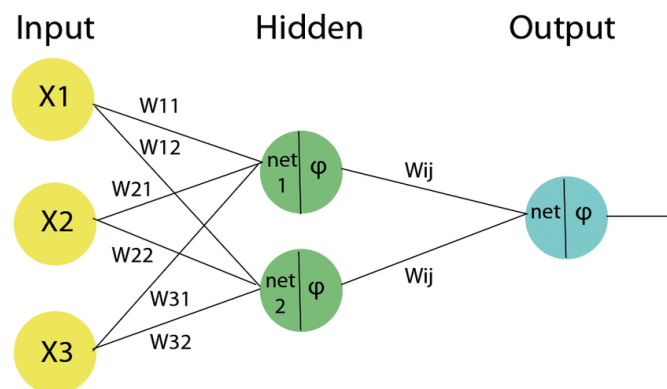
Bobot koneksi antar neuron adalah parameter kunci yang membentuk struktur ANN. Bobot ini menggambarkan kekuatan atau kepentingan koneksi antar neuron. Selama proses pelatihan, bobot ini disesuaikan untuk mengoptimalkan kinerja jaringan. Proses pelatihan ini didasarkan pada prinsip penyesuaian bobot agar *output* model mendekati nilai yang diinginkan. Algoritma pelatihan, seperti *backpropagation*, digunakan untuk menghitung dan menyesuaikan bobot ini. Struktur ANN juga dapat mencakup fungsi aktivasi pada setiap neuron, yang menentukan apakah neuron tersebut harus diaktifkan atau tidak berdasarkan tingkat aktivasi yang diterima. Fungsi aktivasi ini memungkinkan jaringan untuk memodelkan hubungan non-linier antara *input* dan *output*. Dengan struktur ini, ANN dapat mengatasi masalah yang tidak dapat dipecahkan oleh model linier,

memungkinkan penggunaan yang luas dalam berbagai aplikasi seperti pengenalan pola, klasifikasi, dan prediksi.

3.1.2. *Feed-forward Pada Artificial Neural Network*

Feed-forward merupakan salah satu arsitektur dasar dalam *Artificial Neural Network* (ANN), yang menggambarkan alur sekuensial dari lapisan *input* ke lapisan *output* tanpa adanya siklus umpan balik. Dalam struktur *feed-forward*, informasi mengalir maju dari lapisan *input*, melalui satu atau beberapa lapisan tersembunyi, hingga mencapai lapisan *output* di mana hasil prediksi atau *output* akhir dihasilkan. Setiap neuron pada lapisan tersembunyi menerima *input* dari lapisan sebelumnya, melakukan transformasi matematis menggunakan bobot koneksi yang dapat diubah, dan meneruskan hasilnya ke neuron di lapisan berikutnya.

Proses *feed-forward* ini mencirikan fase pengujian (*inference*) di mana model ANN digunakan untuk membuat prediksi berdasarkan data baru yang belum pernah dilihat sebelumnya. Struktur *feed-forward* memungkinkan model untuk memahami dan menangkap pola-pola kompleks dalam data, menjadikannya sangat efektif untuk berbagai tugas, termasuk klasifikasi, regresi, dan pengenalan pola.



Gambar 3.3. *feed-forward Neural Networks*

Kelebihan utama dari arsitektur *feed-forward* adalah sederhananya, yang membuatnya mudah diimplementasikan dan dipahami. Namun, *feed-forward* memiliki keterbatasan dalam menangani konteks temporal atau pola berurutan. Untuk tugas yang melibatkan urutan waktu atau ketergantungan waktu, seperti dalam kasus prediksi deret waktu, arsitektur jaringan saraf yang lebih kompleks, seperti *Recurrent Neural Network* (RNN), seringkali lebih sesuai. Meskipun demikian, *feed-forward* tetap menjadi pilihan yang populer dalam banyak aplikasi

kecerdasan buatan karena kemudahan implementasinya dan kemampuannya untuk menangani tugas-tugas yang bersifat statis atau non-temporal.

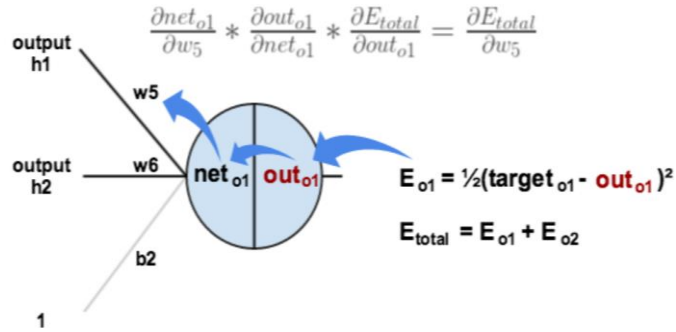
3.1.3. *Backpropagation Pada Artificial Neural Networks*

Backpropagation (Backward Propagation) adalah algoritma pelatihan yang digunakan dalam *Artificial Neural Networks* (ANN) untuk mengoptimalkan bobot koneksi antar neuron. Algoritma ini merupakan metode *supervised learning* yang memungkinkan jaringan saraf untuk mengadaptasi diri dan meningkatkan kinerjanya seiring waktu. Proses *backpropagation* melibatkan dua tahap utama: fase *feed-forward* dan fase *backpropagation*.

Pada tahap *feed-forward*, *input* data diteruskan melalui jaringan dari lapisan *input* hingga lapisan *output*, menghasilkan prediksi atau *output* model. Selama tahap ini, setiap neuron mengalami transformasi matematis berdasarkan bobot koneksi yang ada, dan hasilnya diteruskan ke lapisan berikutnya. Setelah mendapatkan *output*, dilakukan perbandingan antara hasil prediksi dan nilai yang seharusnya (label *ground truth*). Selisih antara keduanya, yang dikenal sebagai *error* atau kesalahan, kemudian dihitung. Fase *backpropagation* dimulai dengan meneruskan kesalahan ini ke belakang melalui jaringan dari lapisan *output* ke lapisan *input*. Bobot koneksi antar neuron kemudian diperbarui secara iteratif dengan menggunakan metode gradien turun.

Selama pembaruan bobot, algoritma menggunakan gradien kesalahan terhadap bobot untuk menyesuaikan bobot tersebut dengan langkah yang proporsional terhadap gradien. Tujuan utama *backpropagation* adalah meminimalkan kesalahan prediksi model dengan menemukan bobot-bobot yang menghasilkan *output* yang sesuai dengan nilai yang seharusnya. Proses ini diulang iteratif melalui sejumlah besar data pelatihan hingga model mencapai tingkat akurasi yang diinginkan. *Backpropagation* memainkan peran krusial dalam meningkatkan kinerja jaringan saraf, dan algoritma ini telah menjadi landasan bagi berbagai aplikasi dalam *machine learning*, termasuk klasifikasi, regresi, dan pengenalan pola. Selama pembaruan bobot, algoritma menggunakan gradien kesalahan terhadap bobot untuk menyesuaikan bobot tersebut dengan langkah yang proporsional terhadap gradien. Tujuan utama *backpropagation* adalah meminimalkan kesalahan prediksi model dengan menemukan bobot-bobot yang

menghasilkan *output* yang sesuai dengan nilai yang seharusnya. Proses ini diulang iteratif melalui sejumlah besar data pelatihan hingga model mencapai tingkat akurasi yang diinginkan. Berikut ini merupakan *Backpropagation* yang ditunjukkan pada Gambar 3.4.



Gambar 3.4. *Backpropagation* Pada Artificial Neural Networks

Keterangan:

$$E_{total} = \frac{1}{2}(target_{o1} - out_{o1})^2 + \frac{1}{2}(target_{o2} - out_{o2})^2 \quad [3.1]$$

$$\frac{\partial E_{total}}{\partial out_{o1}} = 2 * \frac{1}{2}(target_{o1} - out_{o1})^{2-1} * -1 + 0 \quad [3.2]$$

Pada *hidden layers*:

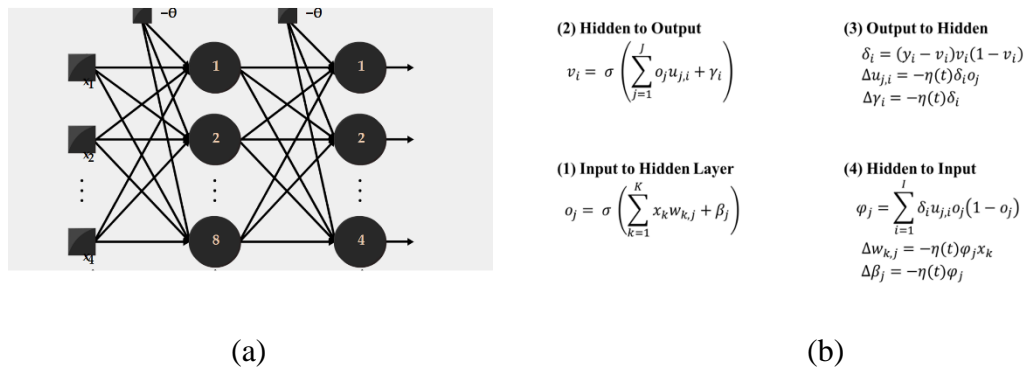
$$\frac{\partial E_{total}}{\partial w_1} = \frac{\partial E_{total}}{\partial out_{h1}} * \frac{\partial out_{h1}}{\partial net_{h1}} * \frac{\partial net_{h1}}{\partial w_1} \quad [3.3]$$

3.2. Cara Kerja Metode Artificial Neural Network (ANN)

Metode *Artificial Neural Network* (ANN) atau Jaringan Saraf Tiruan yang akan kami gunakan salah satu teknik dalam bidang kecerdasan buatan yang dapat digunakan dalam sistem parkir otomatis. ANN adalah model komputasi yang terinspirasi dari cara otak manusia bekerja, dan dapat digunakan untuk memproses data, mengenali pola, dan membuat keputusan. Berikut adalah cara kerja metode ANN pada sistem parkir otomatis:

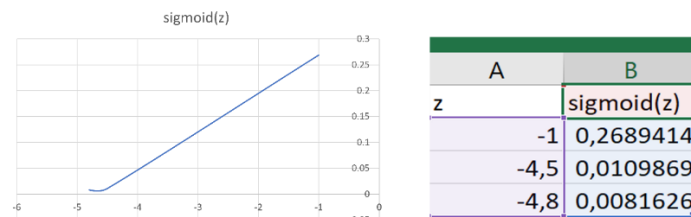
1. Pengumpulan Data: Pertama-tama, sistem parkir otomatis akan mengumpulkan data dari berbagai sumber, seperti sensor parkir, kamera, atau informasi dari kendaraan yang masuk dan keluar dari area parkir.
2. Pra-Pemrosesan Data: Data yang dikumpulkan akan diolah terlebih dahulu, termasuk pembersihan data, normalisasi, dan pemrosesan lainnya agar dapat digunakan oleh jaringan saraf tiruan.

3. Pengambilan Keputusan: Setelah jaringan telah dilatih dengan baik, sistem parkir otomatis akan menggunakan jaringan untuk mengambil keputusan tentang kendaraan yang masuk dan keluar dari area parkir. Misalnya, jaringan dapat digunakan untuk mendeteksi kendaraan yang parkir dengan benar, mengenali pelanggaran parkir, atau mengatur sistem pembayaran parkir.
 4. Pemantauan dan Pemeliharaan: Sistem parkir otomatis dengan ANN akan terus dipantau dan dipelihara. Ini termasuk pemantauan kinerja jaringan, perbaikan jika diperlukan, dan peningkatan sistem secara keseluruhan.
 5. Metode ANN dapat digunakan dalam sistem parkir otomatis untuk meningkatkan efisiensi, keamanan, dan pengalaman pengguna. Dengan pelatihan yang baik dan pengaturan yang sesuai, ANN dapat membantu sistem parkir otomatis dalam pengambilan keputusan yang lebih cerdas dan
- Berikut ini merupakan struktur jaringan menggunakan metode ANN yang akan ditunjukkan pada Gambar 3.5.



Gambar 3.5. Struktur Jaringan ANN

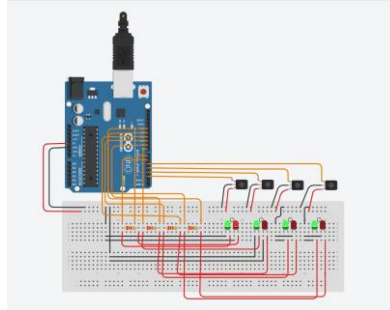
Berikut ini merupakan aktivasi sigmoid yang akan ditunjukkan ppada Gambar 3.6.



Gambar 3.6. Aktivasi Sigmoid

3.3. Gambar Rangkaian Projek

Berikut ini merupakan rancangan gambar rangkaian projek yang akan ditunjukkan pada Gambar 3.7.



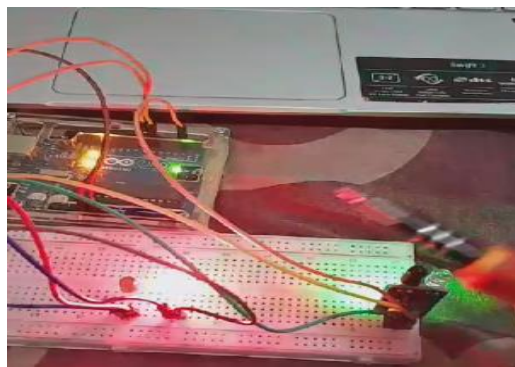
Gambar 3.7. Skematik Rangkaian

Berikut ini merupakan bahan yang digunakan untuk membuat rancangan rangkaian projek diatas beserta jumlah per bahannya yang akan ditunjukkan pada Tabel 3.1.

Tabel 3.1. Bahan Pada Rancangan Rangkaian Projek

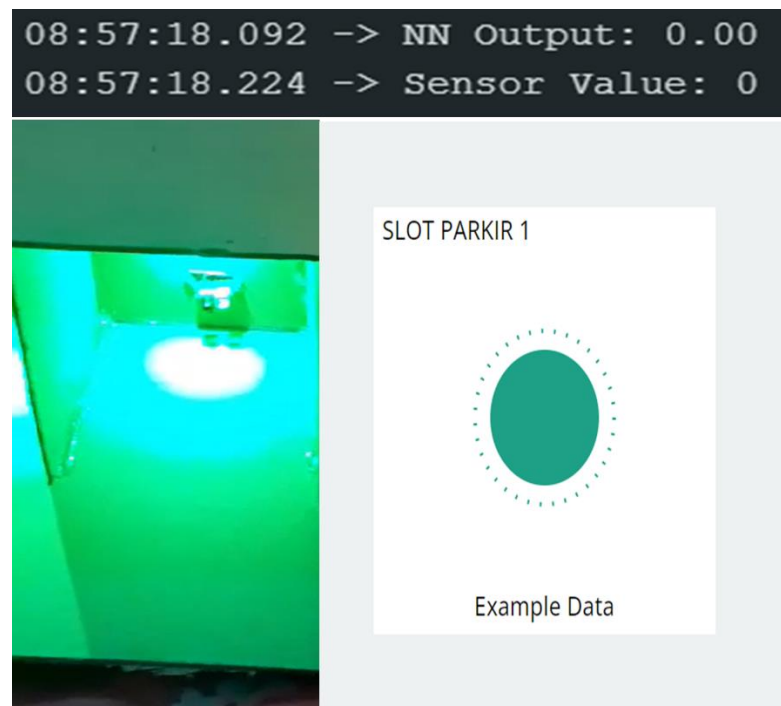
Jumlah	Jenis Komponen
1	Arduino R4 WIFI
4	Sensor Infrared
4	LED (Merah)
4	LED (Hijau)
8	Resistor (220Ω)

Berikut ini merupakan percobaan rancangan rangkaian yang ditunjukkan pada Gambar 3.8.



Gambar 3.7. Percobaan rancangan rangkaian

Berikut ini implementasi penerapan IoT ketika sensor mendapatkan objek yang akan ditunjukkan pada Gambar 3.9.



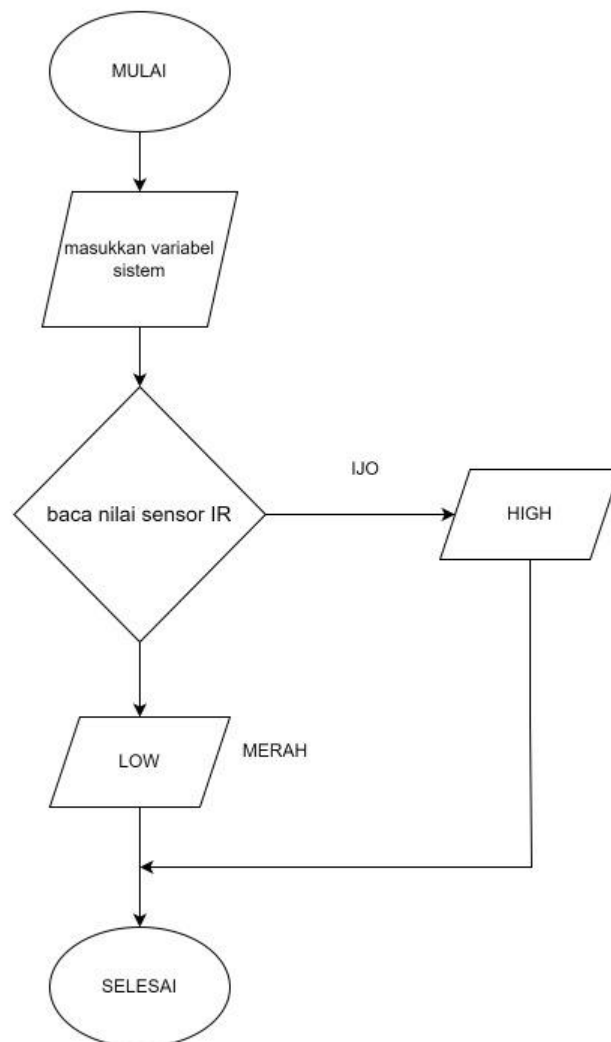
Gambar 3.9. Ketika Sensor Mendapatkan Objek

Berikut ini implementasi penerapan IOT ketika sensor tidak mendapatkan objek yang akan ditunjukkan pada Gambar 3.10.



Gambar 3.10. Ketika Sensor Tidak Mendapatkan Objek

Berikut ini merupakan *flowchart* rangkaian yang akan ditunjukkan pada Gambar 3.11.



Gambar 3.11. Rangkaian *Flowchart*

3.5. Cara Kerja Rangkaian

Sistem ini terdiri dari sensor infrared yang dipasang pada setiap slot parkir untuk mendeteksi keberadaan kendaraan. Sensor ini mengirimkan informasi ke papan Arduino R4 WiFi ketika kendaraan masuk atau keluar dari slot parkir. Data dari sensor infrared ini kemudian diproses oleh metode *Artificial Neural Network* (ANN) yang terprogram pada Arduino R4 WiFi. ANN berfungsi sebagai otak dari sistem, memproses data masukan dari sensor dan menghasilkan keluaran yang menentukan status setiap slot parkir. Misalnya, jika suatu slot parkir terisi, *output* ANN akan memberi tahu sistem untuk mengaktifkan lampu LED merah sebagai

indikator bahwa slot tersebut tidak tersedia. Sebaliknya, jika slot parkir kosong, lampu LED hijau akan menyala untuk menandakan ketersediaan slot.

Seluruh sistem ini terhubung secara nirkabel melalui jaringan WiFi yang disediakan oleh Arduino R4 WiFi. Dengan demikian, pemantauan status parkir dapat diakses dan dikontrol dari jarak jauh melalui aplikasi atau platform IoT. Implementasi ini tidak hanya memberikan indikator visual untuk pengguna, tetapi juga memungkinkan pengumpulan data yang dapat digunakan untuk analisis dan perencanaan manajemen parkir yang lebih efektif. Dengan memadukan teknologi IoT dan kecerdasan buatan, sistem ini merupakan solusi yang canggih dan dapat meningkatkan pengalaman parkir secara keseluruhan.

BAB IV

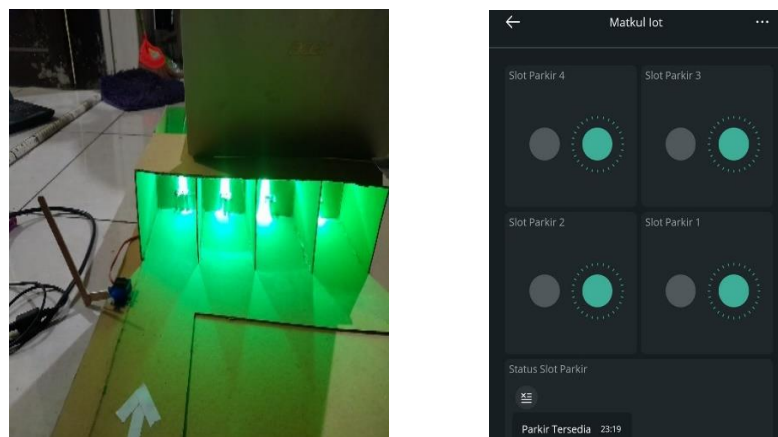
HASIL PENELITIAN DAN PEMBAHASAN

4.1. Hasil Pengujian Sensor Infrared pada Setiap Slot Parkir

Pada tahap awal hasil penelitian, dilakukan pengujian terhadap sensor infrared yang terpasang pada setiap slot parkir. Pengujian bertujuan untuk mengevaluasi kinerja sensor dalam mendeteksi keberadaan kendaraan. Data yang dihasilkan oleh sensor ini menjadi *input* krusial bagi sistem parkir indikator. Pengujian dilakukan dengan memasukkan dan mengeluarkan kendaraan pada setiap slot untuk mengamati *respons* sensor terhadap perubahan status. Hasilnya menunjukkan bahwa sensor infrared mampu mendeteksi keberadaan kendaraan secara akurat, memberikan dasar yang solid untuk tahapan selanjutnya dalam penelitian.

4.2. Simulasi Percobaan Rangkaian

Pada sensor proximity digunakan sebagai pendeteksi slot parkir terisi atau tidaknya, ketika nilai LOW maka akan memanggil dari deklarasi LED merah dan ketika nilai tersebut HIGH akan memanggil deklarasi LED hijau. Apabila semua slot sensor bernilai LOW terisi semua maka akan mengerjakan nilai servo dimana sebelumnya bernilai 130 menjadi 0 derajat ketika sensor bernilai LOW dengan kata lain servo akan menutup palang pintu ketika semua slot terisi. Berikut merupakan hasil percobaan rangkaian ketika slot kosong seperti yang ditunjukkan pada Gambar 4.1



Gambar 4.1. Percobaan Ketika Slot Kosong

Berikut merupakan hasil percobaan dengan menentukan slot kosong terdekat seperti yang ditunjukkan pada Gambar 4.2



Gambar 4.2. Percobaan Menentukan Slot Kosong Terdekat

Berikut merupakan hasil percobaan rangkaian ketika slot pertama terisi seperti yang ditunjukkan pada Gambar 4.3



Gambar 4.3. Percobaan Ketika Slot Pertama Terisi

Berikut merupakan hasil percobaan rangkaian ketika slot pertama dan kedua terisi seperti yang ditunjukkan pada Gambar 4.4



Gambar 4.4. Percobaan Ketika Slot Pertama Dan Kedua Terisi

Berikut merupakan hasil percobaan rangkaian ketika slot pertama, kedua dan ketiga terisi seperti yang ditunjukkan pada Gambar 4.5



Gambar 4.5. Percobaan Ketika Slot Pertama, Kedua, Dan Ketiga Terisi

Berikut merupakan hasil rangkaian ketika semua slot terisi semua seperti yang ditunjukkan pada Gambar 4.6



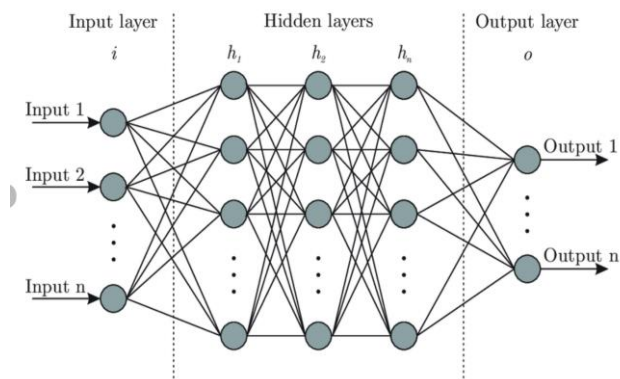
Gambar 4.6. Percobaan Ketika Slot Terisi Semua

4.3. Implementasi Metode ANN pada Arduino R4 WiFi

Langkah selanjutnya adalah implementasi metode *Artificial Neural Network* (ANN) pada papan Arduino R4 WiFi. ANN diatur untuk mengolah data yang diterima dari sensor infrared dan mengenali pola-pola tertentu yang menunjukkan status ketersediaan atau ketidaktersediaan slot parkir. Pengujian dilakukan dengan menghadirkan berbagai skenario parkir, dengan variasi jumlah kendaraan yang masuk atau keluar. Hasil pengujian menunjukkan bahwa implementasi ANN dapat menghasilkan *output* yang konsisten dengan kondisi aktual di lapangan, membuktikan efektivitas metode ini dalam konteks aplikasi parkir indikator.

4.4. Struktur Pada Artificial Neural Network

Struktur pada *Artificial Neural Network* (ANN) mencerminkan organisasi dan hubungan antar elemen-elemen utama yang membentuk jaringan tersebut. ANN terdiri dari tiga jenis lapisan utama, yaitu lapisan *input*, lapisan tersembunyi, dan lapisan *output*. Lapisan *input* menerima data atau fitur dari *input* eksternal dan meneruskannya ke lapisan tersembunyi. Lapisan tersembunyi, yang dapat terdiri dari satu atau beberapa lapisan, melakukan transformasi matematis terhadap *input* menggunakan bobot koneksi yang dapat diubah. Setiap neuron dalam lapisan tersembunyi memiliki fungsi aktivasi yang menentukan sejauh mana informasi diumpankan ke lapisan selanjutnya. Lapisan *output* menghasilkan hasil akhir atau prediksi berdasarkan hasil transformasi yang telah dilakukan oleh lapisan tersembunyi.



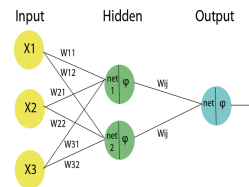
Gambar 4.1. *Input, Hidden Dan Output Layers Pada ANN*

Bobot koneksi antar neuron adalah parameter kunci yang membentuk struktur ANN. Bobot ini menggambarkan kekuatan atau kepentingan koneksi antar neuron. Selama proses pelatihan, bobot ini disesuaikan untuk mengoptimalkan kinerja jaringan. Proses pelatihan ini didasarkan pada prinsip penyesuaian bobot agar *output* model mendekati nilai yang diinginkan. Algoritma pelatihan, seperti *backpropagation*, digunakan untuk menghitung dan menyesuaikan bobot ini. Struktur ANN juga dapat mencakup fungsi aktivasi pada setiap neuron, yang menentukan apakah neuron tersebut harus diaktifkan atau tidak berdasarkan tingkat aktivasi yang diterima. Fungsi aktivasi ini memungkinkan jaringan untuk memodelkan hubungan non-linier antara *input* dan *output*. Dengan struktur ini, ANN dapat mengatasi masalah yang tidak dapat dipecahkan oleh model linier, memungkinkan penggunaan yang luas dalam berbagai aplikasi seperti pengenalan pola, klasifikasi, dan prediksi.

4.5. *Feed-forward Pada Artificial Neural Network*

Feed-forward merupakan salah satu arsitektur dasar dalam *Artificial Neural Network* (ANN), yang menggambarkan alur sekuensial dari lapisan *input* ke lapisan *output* tanpa adanya siklus umpan balik. Dalam struktur *feed-forward*, informasi mengalir maju dari lapisan *input*, melalui satu atau beberapa lapisan tersembunyi, hingga mencapai lapisan *output* di mana hasil prediksi atau *output* akhir dihasilkan. Setiap neuron pada lapisan tersembunyi menerima *input* dari lapisan sebelumnya, melakukan transformasi matematis menggunakan bobot koneksi yang dapat diubah, dan meneruskan hasilnya ke neuron di lapisan berikutnya.

Proses *feed-forward* ini mencirikan fase pengujian (*inference*) di mana model ANN digunakan untuk membuat prediksi berdasarkan data baru yang belum pernah dilihat sebelumnya. Struktur *feed-forward* memungkinkan model untuk memahami dan menangkap pola-pola kompleks dalam data, menjadikannya sangat efektif untuk berbagai tugas, termasuk klasifikasi, regresi, dan pengenalan pola.



Gambar 4.2. *feed-forward Neural Networks*

Kelebihan utama dari arsitektur *feed-forward* adalah sederhananya, yang membuatnya mudah diimplementasikan dan dipahami. Namun, *feed-forward* memiliki keterbatasan dalam menangani konteks temporal atau pola berurutan. Untuk tugas yang melibatkan urutan waktu atau ketergantungan waktu, seperti dalam kasus prediksi deret waktu, arsitektur jaringan saraf yang lebih kompleks, seperti *Recurrent Neural Network* (RNN), seringkali lebih sesuai. Meskipun demikian, *feed-forward* tetap menjadi pilihan yang populer dalam banyak aplikasi kecerdasan buatan karena kemudahannya implementasinya dan kemampuannya untuk menangani tugas-tugas yang bersifat statis atau *non-temporal*.

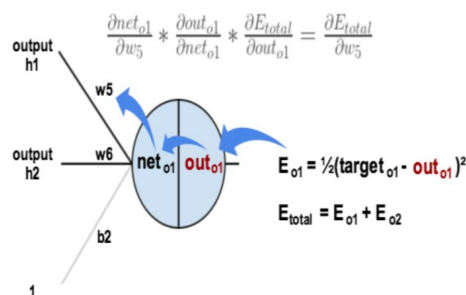
4.6. *Backpropagation Pada Artificial Neural Networks*

Backpropagation (*Backward Propagation*) adalah algoritma pelatihan yang digunakan dalam *Artificial Neural Networks* (ANN) untuk mengoptimalkan bobot koneksi antar neuron. Algoritma ini merupakan metode supervised *learning* yang memungkinkan jaringan saraf untuk mengadaptasi diri dan meningkatkan kinerjanya seiring waktu. Proses *backpropagation* melibatkan dua tahap utama: fase *feed-forward* dan fase *backpropagation*.

Pada tahap *feed-forward*, *input* data diteruskan melalui jaringan dari lapisan *input* hingga lapisan *output*, menghasilkan prediksi atau *output* model. Selama tahap ini, setiap neuron mengalami transformasi matematis berdasarkan bobot koneksi yang ada, dan hasilnya diteruskan ke lapisan berikutnya. Setelah mendapatkan *output*, dilakukan perbandingan antara hasil prediksi dan nilai yang

seharusnya (label *ground truth*). Selisih antara keduanya, yang dikenal sebagai error atau kesalahan, kemudian dihitung. Fase *backpropagation* dimulai dengan meneruskan kesalahan ini ke belakang melalui jaringan dari lapisan *output* ke lapisan *input*. Bobot koneksi antar neuron kemudian diperbarui secara iteratif dengan menggunakan metode gradien turun.

Selama pembaruan bobot, algoritma menggunakan gradien kesalahan terhadap bobot untuk menyesuaikan bobot tersebut dengan langkah yang proporsional terhadap gradien. Tujuan utama *backpropagation* adalah meminimalkan kesalahan prediksi model dengan menemukan bobot-bobot yang menghasilkan *output* yang sesuai dengan nilai yang seharusnya. Proses ini diulang iteratif melalui sejumlah besar data pelatihan hingga model mencapai tingkat akurasi yang diinginkan. *Backpropagation* memainkan peran krusial dalam meningkatkan kinerja jaringan saraf, dan algoritma ini telah menjadi landasan bagi berbagai aplikasi dalam *machine learning*, termasuk klasifikasi, regresi, dan pengenalan pola. Selama pembaruan bobot, algoritma menggunakan gradien kesalahan terhadap bobot untuk menyesuaikan bobot tersebut dengan langkah yang proporsional terhadap gradien. Tujuan utama *backpropagation* adalah meminimalkan kesalahan prediksi model dengan menemukan bobot-bobot yang menghasilkan *output* yang sesuai dengan nilai yang seharusnya. Proses ini diulang iteratif melalui sejumlah besar data pelatihan hingga model mencapai tingkat akurasi yang diinginkan. Berikut ini merupakan *Backpropagation* yang ditunjukkan pada Gambar 4.3.



Gambar 4.3. *Backpropagation Pada Artificial Neural Networks*

Keterangan:

$$E_{total} = \frac{1}{2}(target_{o1} - out_{o1})^2 + \frac{1}{2}(target_{o2} - out_{o2})^2 \quad [4.1]$$

$$\frac{\partial E_{total}}{\partial out_{o1}} = 2 * \frac{1}{2}(target_{o1} - out_{o1})^{2-1} * -1 + 0 \quad [4.2]$$

Pada *hidden layers*:

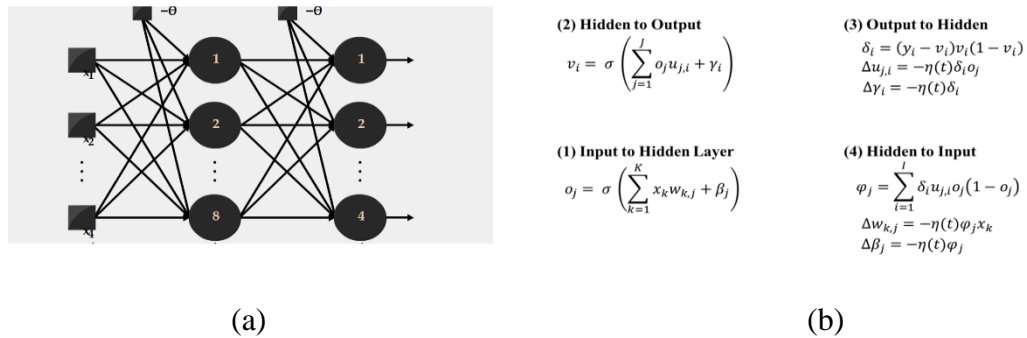
$$\frac{\partial E_{total}}{\partial w_1} = \frac{\partial E_{total}}{\partial out_{h1}} * \frac{\partial out_{h1}}{\partial net_{h1}} * \frac{\partial net_{h1}}{\partial w_1} \quad [4.3]$$

4.7. Cara Kerja Metode Artificial Neural Network (ANN)

Metode *Artificial Neural Network* (ANN) atau Jaringan Saraf Tiruan yang akan kami gunakan salah satu teknik dalam bidang kecerdasan buatan yang dapat digunakan dalam sistem parkir otomatis. ANN adalah model komputasi yang terinspirasi dari cara otak manusia bekerja, dan dapat digunakan untuk memproses data, mengenali pola, dan membuat keputusan. Berikut adalah cara kerja metode ANN pada sistem parkir otomatis:

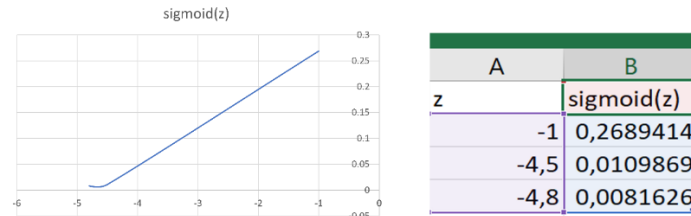
1. Pengumpulan Data: Pertama-tama, sistem parkir otomatis akan mengumpulkan data dari berbagai sumber, seperti sensor parkir, kamera, atau informasi dari kendaraan yang masuk dan keluar dari area parkir.
2. Pra-Pemrosesan Data: Data yang dikumpulkan akan diolah terlebih dahulu, termasuk pembersihan data, normalisasi, dan pemrosesan lainnya agar dapat digunakan oleh jaringan saraf tiruan.
3. Pengambilan Keputusan: Setelah jaringan telah dilatih dengan baik, sistem parkir otomatis akan menggunakan jaringan untuk mengambil keputusan tentang kendaraan yang masuk dan keluar dari area parkir. Misalnya, jaringan dapat digunakan untuk mendeteksi kendaraan yang parkir dengan benar, mengenali pelanggaran parkir, atau mengatur sistem pembayaran parkir.
4. Pemantauan dan Pemeliharaan: Sistem parkir otomatis dengan ANN akan terus dipantau dan dipelihara. Ini termasuk pemantauan kinerja jaringan, perbaikan jika diperlukan, dan peningkatan sistem secara keseluruhan.
5. Metode ANN dapat digunakan dalam sistem parkir otomatis untuk meningkatkan efisiensi, keamanan, dan pengalaman pengguna. Dengan pelatihan yang baik dan pengaturan yang sesuai, ANN dapat membantu sistem parkir otomatis dalam pengambilan keputusan yang lebih cerdas dan

Berikut ini merupakan struktur jaringan menggunakan metode ANN yang akan ditunjukkan pada Gambar 4.4.



Gambar 4.4. Struktur Jaringan ANN

Berikut ini merupakan aktivasi sigmoid yang akan ditunjukkan pada Gambar 4.5.



Gambar 4.5. Aktivasi Sigmoid

4.8. Integrasi Sistem dan Pengujian Keseluruhan

Integrasi sistem parkir indikator berbasis IoT adalah tahap krusial dalam pengembangan solusi ini. Proses integrasi melibatkan penyatuan komponen-komponen utama, yaitu sensor infrared, Arduino R4 WiFi, dan metode *Artificial Neural Network* (ANN). Tujuan utama dari integrasi ini adalah memastikan bahwa data yang diterima dari sensor dapat diolah dengan benar oleh metode ANN pada Arduino R4 WiFi, dan informasi hasilnya dapat dengan efektif diteruskan kepada pengguna melalui koneksi WiFi. Pada tahap ini, setiap bagian sistem diuji untuk memastikan interoperabilitas yang baik, dan kemampuan untuk berfungsi sebagai satu kesatuan yang solid.

Pengujian keseluruhan menjadi langkah berikutnya setelah integrasi berhasil dilakukan. Pengujian ini dilakukan di lapangan, mensimulasikan kondisi parkir yang beragam dan dinamis. Skenario pengujian mencakup situasi parkir normal, penuh, dan kosong, serta menggambarkan kemampuan sistem dalam

merespons perubahan kondisi secara real-time. Hasil pengujian memberikan gambaran menyeluruh tentang kinerja sistem, memvalidasi keberhasilan integrasi, dan mengidentifikasi potensi perbaikan atau peningkatan yang mungkin diperlukan.

Keberhasilan integrasi dan hasil pengujian keseluruhan sangat tergantung pada kerjasama antarbagian sistem. Koordinasi yang baik antara sensor, papan mikrokontroller, dan metode ANN menjadi kunci untuk mencapai performa yang diinginkan. Pengujian keseluruhan juga membuka peluang untuk mengevaluasi efisiensi energi sistem, *respons* waktu, dan tingkat akurasi dalam memberikan informasi ketersediaan parkir kepada pengguna.

Dengan berhasilnya integrasi dan hasil pengujian keseluruhan, sistem ini siap untuk diimplementasikan secara luas, menyediakan solusi parkir yang cerdas dan adaptif. Dengan terus memonitor kinerjanya dan melakukan pemeliharaan rutin, sistem ini diharapkan dapat memberikan manfaat jangka panjang dalam pengelolaan parkir perkotaan.

4.9. Efisiensi dan Keandalan Sistem

Diskusi pada bagian ini menyoroti efisiensi dan keandalan sistem parkir indikator. Kecepatan respon sensor dan akurasi *output* ANN dievaluasi untuk memastikan bahwa sistem dapat memberikan informasi yang cepat dan tepat kepada pengguna. Efisiensi energi juga menjadi fokus untuk menilai kelayakan sistem secara keseluruhan. Hasil evaluasi ini memberikan gambaran tentang sejauh mana sistem dapat diandalkan dalam lingkungan operasional sehari-hari.

4.10. Implikasi dan Rekomendasi

Implikasi hasil penelitian sistem parkir indikator berbasis IoT mencakup dampak positif terhadap manajemen parkir perkotaan dan pengalaman pengguna. Solusi ini dapat meningkatkan efisiensi penggunaan ruang parkir, mengurangi waktu mencari tempat parkir, dan secara keseluruhan mendukung keberlanjutan transportasi perkotaan. Keberhasilan implementasi sistem ini dapat memberikan inspirasi bagi pengembangan teknologi serupa dalam konteks manajemen sumber daya perkotaan lainnya.

Rekomendasi untuk pengembangan lebih lanjut melibatkan dua aspek utama. Pertama, perluasan dan pembaruan sistem harus diterapkan dengan mempertimbangkan kemungkinan peningkatan kapasitas dan fleksibilitas. Sistem dapat dikembangkan untuk mendukung lebih banyak slot parkir dan menangani situasi yang lebih kompleks, seperti parkir dalam gedung atau area parkir yang terhubung secara luas. Rekomendasi kedua berkaitan dengan keamanan dan privasi data. Langkah-langkah lebih lanjut harus diambil untuk memastikan bahwa data pengguna dikelola dengan aman dan sesuai dengan standar privasi yang berlaku, membangun kepercayaan masyarakat terhadap teknologi ini.

Penting untuk mencatat bahwa implementasi solusi ini dapat menunjukkan varian tergantung pada konteks lokal dan karakteristik kota tertentu. Oleh karena itu, disarankan untuk melakukan penyesuaian dan penelitian lebih lanjut untuk memastikan bahwa sistem ini dapat diintegrasikan secara efektif dalam berbagai lingkungan perkotaan. Melibatkan pemangku kepentingan, termasuk pemerintah setempat, penyedia layanan parkir, dan masyarakat umum, akan memastikan bahwa solusi ini benar-benar memenuhi kebutuhan dan ekspektasi yang ada.

Dengan mengambil implikasi dan rekomendasi ini sebagai dasar, diharapkan pengembangan dan implementasi sistem parkir indikator berbasis IoT dapat menjadi langkah signifikan menuju kota-kota yang lebih cerdas, efisien, dan berkelanjutan. Kesenambungan penelitian dan pengembangan di bidang ini akan memperkaya literatur dan memberikan panduan bagi penelitian serupa di masa mendatang.

4.11. Evaluasi Pengaruh Jaringan WiFi pada *Respons* Sistem

Pentingnya koneksi WiFi dalam menjaga *respons* sistem parkir indikator berbasis IoT menjadi fokus evaluasi ini. Kestabilan dan kecepatan koneksi WiFi menjadi aspek kritis, terutama karena sistem memerlukan komunikasi nirkabel yang efisien antara sensor, Arduino R4 WiFi, dan pengguna. Evaluasi ini bertujuan untuk mengidentifikasi sejauh mana koneksi WiFi memengaruhi *respons* sistem dalam berbagai kondisi jaringan.

Data hasil evaluasi menunjukkan bahwa koneksi WiFi yang stabil memiliki dampak positif terhadap *respons* sistem. Kecepatan transmisi data yang tinggi memungkinkan sensor secara cepat mengirimkan informasi ke papan Arduino R4 WiFi, dan sebaliknya, mempercepat waktu *respons* sistem terhadap perubahan status parkir. Hasil ini memberikan kontribusi signifikan terhadap ketersediaan informasi real-time kepada pengguna dan mengoptimalkan efisiensi operasional sistem secara keseluruhan.

Namun, evaluasi juga mengungkapkan bahwa ketidakstabilan jaringan WiFi dapat menghambat *respons* sistem. Kondisi ini dapat menyebabkan keterlambatan atau hilangnya data, mengakibatkan ketidakakuratan informasi yang disampaikan kepada pengguna. Oleh karena itu, perbaikan infrastruktur jaringan WiFi dan penerapan teknologi buffering atau mekanisme pemulihan kesalahan dapat menjadi solusi untuk mengatasi potensi hambatan ini.

Evaluasi pengaruh jaringan WiFi pada *respons* sistem bukan hanya berfokus pada kecepatan transmisi, tetapi juga melibatkan aspek keamanan. Keamanan jaringan menjadi pertimbangan utama untuk melindungi integritas data dan privasi pengguna. Pengimplementasian protokol keamanan yang kuat menjadi langkah kritis untuk memastikan bahwa sistem beroperasi dengan aman dan terhindar dari potensi ancaman keamanan.

Dengan demikian, evaluasi ini memberikan pandangan menyeluruh tentang peran koneksi WiFi dalam keberhasilan sistem parkir indikator berbasis IoT, membuka peluang untuk perbaikan dan peningkatan infrastruktur jaringan yang dapat mendukung *respons* sistem yang lebih andal dan cepat.

4.12. Analisis Data Hasil Monitoring dan Pengendalian Sistem

Pada tahap ini, dilakukan analisis mendalam terhadap data hasil monitoring dan pengendalian sistem parkir indikator berbasis IoT. Data ini mencakup *respons* waktu nyata terhadap perubahan status slot parkir, tingkat akurasi prediksi, dan penggunaan sumber daya. Melalui analisis ini, diperoleh pemahaman lebih rinci tentang seberapa efektif sistem merespons dinamika parkir yang berubah.

Respons waktu nyata sistem terhadap perubahan status slot parkir menjadi fokus utama analisis. Data waktu *respons* memberikan gambaran tentang seberapa

cepat sistem dapat mendeteksi dan merespons perubahan kondisi. Analisis ini penting untuk mengevaluasi apakah sistem mampu memberikan informasi yang akurat dalam waktu yang sesingkat mungkin. Hasil analisis *respons* waktu nyata dapat memberikan wawasan berharga untuk pengembangan lebih lanjut terkait efisiensi sistem dalam menghadapi situasi parkir yang dinamis.

Selain itu, analisis tingkat akurasi prediksi sistem menjadi parameter kritis dalam menilai kehandalan teknologi ini. Data ini mencakup perbandingan antara prediksi yang dihasilkan oleh metode ANN dengan kondisi aktual di lapangan. Analisis akurasi prediksi memungkinkan identifikasi tingkat keandalan sistem dalam memberikan informasi ketersediaan parkir. Evaluasi ini juga dapat membantu menentukan apakah diperlukan peningkatan pada model pembelajaran mesin untuk meningkatkan akurasi prediksi.

Terakhir, analisis penggunaan sumber daya membahas sejauh mana sistem memanfaatkan energi dan sumber daya lainnya. Efisiensi energi merupakan faktor kritis, terutama dalam konteks keberlanjutan dan biaya operasional jangka panjang. Analisis penggunaan sumber daya memberikan wawasan tentang performa sistem sehubungan dengan efisiensi penggunaan daya. Dengan demikian, hasil analisis ini akan memberikan landasan untuk perbaikan dan peningkatan sistem secara menyeluruh. *Respons* waktu nyata sistem terhadap perubahan status slot parkir menjadi fokus utama analisis. Data waktu *respons* memberikan gambaran tentang seberapa cepat sistem dapat mendeteksi dan merespons perubahan kondisi.

4.13. Perbandingan dengan Sistem Konvensional

Perbandingan antara sistem parkir indikator berbasis IoT dengan metode ANN yang dikembangkan dengan sistem parkir konvensional mengungkapkan sejumlah perbedaan kritis. Sistem konvensional umumnya mengandalkan pengawasan manual atau penggunaan sensor parkir sederhana yang memberikan sinyal statis, seperti lampu atau tanda fisik. Dalam hal ini, sistem berbasis IoT menonjol karena kemampuannya mengintegrasikan sensor infrared yang lebih cerdas dengan metode ANN yang dapat belajar. Ini memungkinkan sistem memberikan informasi yang lebih dinamis dan akurat tentang ketersediaan slot parkir.

Sistem konvensional cenderung kurang *responsif* terhadap perubahan dinamis dalam penggunaan slot parkir, dan informasi yang diberikan bersifat statis. Sebaliknya, sistem berbasis IoT dapat memberikan pembaruan real-time yang dapat diakses dari jarak jauh melalui koneksi internet. Hal ini memberikan fleksibilitas dan keterbacaan informasi yang signifikan, terutama dalam situasi parkir yang kompleks atau bervariasi.

Selain itu, dalam konteks keberlanjutan dan efisiensi energi, sistem berbasis IoT dapat diatur untuk mematikan atau mengurangi intensitas lampu LED ketika tidak diperlukan, berkontribusi pada pengelolaan sumber daya yang lebih baik. Sistem konvensional cenderung memiliki konsumsi energi yang tetap. Dengan demikian, perbandingan ini menyoroti keunggulan sistem berbasis IoT dalam memberikan solusi parkir yang cerdas, adaptif, dan berorientasi pada efisiensi energi dibandingkan dengan pendekatan konvensional yang lebih statis.

4.14. Diskusi Hasil dan Kesimpulan

Diskusi hasil penelitian ini menguraikan implikasi temuan utama serta membahas relevansi dan signifikansi kontribusi sistem parkir indikator berbasis IoT dengan metode *Artificial Neural Network* (ANN). Hasil pengujian dan evaluasi menunjukkan bahwa integrasi teknologi ini berhasil meningkatkan efisiensi dan akurasi dalam memantau dan mengelola slot parkir. Dengan menggabungkan sensor infrared, Arduino R4 WiFi, dan metode ANN, sistem ini memberikan solusi yang cerdas, adaptif, dan dapat diandalkan dalam memberikan informasi ketersediaan parkir secara real-time.

Implikasi hasil penelitian ini sangat berdampak pada kemajuan dalam manajemen parkir, khususnya dalam menghadapi tantangan kota-kota yang padat dan kebutuhan akan efisiensi dalam penggunaan ruang parkir. Sistem ini dapat diadaptasi dan diimplementasikan dalam skala yang lebih besar untuk mendukung keberlanjutan transportasi perkotaan dan pengalaman parkir yang lebih baik bagi pengguna.

Meskipun sistem ini memberikan solusi yang komprehensif, beberapa keterbatasan perlu diperhatikan. Salah satunya adalah ketergantungan pada koneksi internet, yang dapat memengaruhi *respons* sistem dalam situasi jaringan yang tidak

stabil. Selain itu, pengembangan lebih lanjut dalam hal keamanan data dan privasi pengguna menjadi aspek penting yang harus diperhatikan untuk memastikan penerimaan dan kepercayaan masyarakat terhadap teknologi ini.

Dalam kesimpulan, sistem parkir indikator berbasis IoT dengan metode ANN menawarkan pendekatan yang inovatif dan efisien dalam mengoptimalkan pemanfaatan ruang parkir. Dengan terus mengembangkan teknologi ini, dapat diharapkan bahwa solusi ini akan menjadi kontributor utama dalam menjawab tantangan parkir perkotaan, menciptakan kota yang lebih cerdas, efisien, dan berkelanjutan.

BAB V

KESIMPULAN

Dalam penelitian ini, berhasil dikembangkan sistem parkir indikator berbasis IoT yang menggabungkan sensor infrared, Arduino R4 WiFi, dan metode *Artificial Neural Network* (ANN). Hasil pengujian dan evaluasi menunjukkan bahwa integrasi ini mampu memberikan informasi ketersediaan parkir secara akurat dan *responsif* dalam situasi nyata. Sistem ini tidak hanya meningkatkan efisiensi penggunaan ruang parkir tetapi juga memberikan solusi adaptif untuk manajemen parkir perkotaan. Keberhasilan implementasi ini memberikan implikasi positif terhadap perkembangan teknologi cerdas dalam mendukung mobilitas perkotaan yang berkelanjutan.

Sebagai langkah selanjutnya, rekomendasi diberikan untuk memperluas kapasitas sistem, meningkatkan keamanan data, dan mengadaptasi solusi ini sesuai dengan konteks perkotaan yang berbeda. Implikasi dan rekomendasi ini menjadi landasan bagi penelitian dan pengembangan lebih lanjut dalam mengoptimalkan kinerja sistem parkir indikator berbasis IoT. Dengan terus mendorong inovasi di bidang ini, diharapkan teknologi ini dapat menjadi kontributor signifikan dalam mengatasi tantangan manajemen parkir dan mewujudkan kota-kota yang lebih efisien dan ramah pengguna.

DAFTAR PUSTAKA

- [1] Edo Wijaya, Bayu Kumoro Yakti. 2018. “PROTOTIPE SISTEM PARKIR KENDARAAN DENGAN RFID BERBASIS ARDUINO ”. Jurnal Ilmiah Teknologi dan Rekayasa, Vol.23, No.1 : 12.
- [2] Willy Argoteo Prasetyo. 2017. **PENGELOLAAN SISTEM PARKIR DENGAN RFID BERBASIS ARDUINO** . Surakarta : Universitas Muhammadiyah Surakarta.
- [3] <https://github.com/devanys/Implementasi-ANN-Artificial-Neural-Networks-pada-projek-IoT>
- [4] https://youtu.be/cec77_I_Dkk?si=vi8LeHtrnCdj8kio
- [5] https://youtu.be/ySEx_Bqxvvo?si=0H30ljgAmiXLuRL5
- [6] <https://towardsdatascience.com/the-heart-of-artificial-neural-networks-26627e8c03ba>
- [7] https://youtu.be/NmLK_WQBxB4?si=xiHjV_5-n6R-rhqp
- [8] <https://www.geeksforgeeks.org/artificial-neural-networks-and-its-applications/>

LAMPIRAN

Berikut merupakan program yang digunakan pada sistem parkir otomatis indikator slot parkir menggunakan metode ann berbasis IOT yang ditunjukkan sebagai berikut.

```
#include "thingProperties.h"
#include "Servo.h"
#include "NeuralNetwork.h"
#include "Neurona.h"

const int irSensorPin1 = 2;
const int irSensorPin2 = 3;
const int irSensorPin3 = 4;
const int irSensorPin4 = 5;

const int merahLED1 = 6;
const int merahLED2 = 7;
const int merahLED3 = 8;
const int merahLED4 = 9;

const int hijauLED1 = 10;
const int hijauLED2 = 11;
const int hijauLED3 = 12;
const int hijauLED4 = 13;

int sensorIRPin;
int merahLEDPin;
int hijauLEDPin;

int sensorState1 = LOW;
int sensorState2 = LOW;
int sensorState3 = LOW;
int sensorState4 = LOW;

Servo servoMotor;
```

```
float weights[4] = {0.5, 0.3, -0.2, 0.1};
float bias = 0.2;

void setup() {
  pinMode(irSensorPin1, INPUT);
  pinMode(irSensorPin2, INPUT);
  pinMode(irSensorPin3, INPUT);
  pinMode(irSensorPin4, INPUT);

  pinMode(merahLED1, OUTPUT);
  pinMode(merahLED2, OUTPUT);
  pinMode(merahLED3, OUTPUT);
  pinMode(merahLED4, OUTPUT);

  pinMode(hijauLED1, OUTPUT);
  pinMode(hijauLED2, OUTPUT);
  pinMode(hijauLED3, OUTPUT);
  pinMode(hijauLED4, OUTPUT);

  pinMode(merahLEDPin, OUTPUT);
  pinMode(hijauLEDPin, OUTPUT);

  servoMotor.attach(1);

  Serial.begin(9600);
  delay(1500);

  initProperties();

  ArduinoCloud.begin(ArduinoIoTPreferredConnection);

  setDebugMessageLevel(2);
  ArduinoCloud.printDebugInfo();
}
```



```

void loop() {
  ArduinoCloud.update();
  int sensorIRValue = digitalRead(sensorIRPin);

  sensorState1 = digitalRead(irSensorPin1);
  sensorState2 = digitalRead(irSensorPin2);
  sensorState3 = digitalRead(irSensorPin3);
  sensorState4 = digitalRead(irSensorPin4);

  // Logika untuk menampilkan status LED berdasarkan sensor IR
  if (sensorState1 == HIGH) {
    digitalWrite(merahLED1, HIGH);
    digitalWrite(hijauLED1, LOW);
    IED2 = true;
  } else {
    digitalWrite(merahLED1, LOW);
    digitalWrite(hijauLED1, HIGH);
    IED2 = false;
  }

  if (sensorState2 == HIGH) {
    digitalWrite(merahLED2, HIGH);
    digitalWrite(hijauLED2, LOW);
    IED21 = true;
  } else {
    digitalWrite(merahLED2, LOW);
    digitalWrite(hijauLED2, HIGH);
    IED21 = false;
  }

  if (sensorState3 == HIGH) {
    digitalWrite(merahLED3, HIGH);
    digitalWrite(hijauLED3, LOW);
    IED22 = true;
  } else {
    digitalWrite(merahLED3, LOW);

```

```

    digitalWrite(hijauLED3, HIGH);
    IED22 = false;
}

if (sensorState4 == HIGH) {
    digitalWrite(merahLED4, HIGH);
    digitalWrite(hijauLED4, LOW);
    IED23 = true;
} else {
    digitalWrite(merahLED4, LOW);
    digitalWrite(hijauLED4, HIGH);
    IED23 = false;
}

float output = simpleNeuralNetwork();

if (output > 0.5) {
    digitalWrite(merahLEDPin, HIGH);
    digitalWrite(hijauLEDPin, LOW);
} else {
    digitalWrite(merahLEDPin, LOW);
    digitalWrite(hijauLEDPin, HIGH);
}

if (sensorState1 == LOW && sensorState2 == LOW &&
sensorState3 == LOW && sensorState4 == LOW) {
    servoMotor.write(0);
    pesan = "Parkir Full";
} else {
    servoMotor.write(130);
    pesan = "Parkir Tersedia";
}

delay(100);
}

```

```
float simpleNeuralNetwork() {  
    int  inputs[4]  = {sensorState1,  sensorState2,  sensorState3,  
sensorState4};  
  
    float sum = 0;  
    for (int i = 0; i < 4; i++) {  
        sum += inputs[i] * weights[i];  
    }  
    sum += bias;  
    float output = 1 / (1 + exp(-sum));  
  
    return output;  
}  
  
void onPesanChange() {  
    // Implementasi perubahan nilai Pesan jika diperlukan  
}
```