

**PURWARUPA SISTEM DETEKSI DAN PENGATURAN
LAMPU LALU LINTAS BERBASIS TINYML UNTUK
PRIORITAS LAYANAN GAWAT DARURAT
AMBULANS**

TUGAS AKHIR

Diajukan sebagai syarat menyelesaikan jenjang strata Satu (S-1) di
Program Studi Teknik Informatika, Fakultas Teknologi Industri,
Institut Teknologi Sumatera

Oleh:

MUHAMMAD A.M HIKAR SYAHRIAL

119140031



**PROGRAM STUDI TEKNIK INFORMATIKA
FAKULTAS TEKNOLOGI INDUSTRI
INSTITUT TEKNOLOGI SUMATERA
LAMPUNG SELATAN**

2024

DAFTAR ISI

DAFTAR ISI.....	ii
DAFTAR TABEL.....	iv
DAFTAR GAMBAR	v
BAB I PENDAHULUAN.....	1
1.1 Latar Belakang.....	1
1.2 Rumusan Masalah.....	3
1.3 Tujuan Penelitian	3
1.4 Batasan Masalah	3
1.5 Manfaat Penelitian	4
1.6 Sistematika Penulisan	4
1.6.1 BAB I PENDAHULUAN.....	4
1.6.2 BAB II TINJAUAN PUSTAKA	4
1.6.3 BAB III METODE PENELITIAN	5
1.6.4 BAB IV HASIL DAN PEMBAHASAN	5
1.6.5 BAB V KESIMPULAN DAN SARAN	5
BAB II TINJAUAN PUSTAKA	6
2.1 Tinjauan Pustaka.....	6
2.2 Dasar Teori	9
2.2.1 Lampu Lalu Lintas.....	9
2.2.2 Layanan Darurat Ambulans	10
2.2.3 Mikrokontroler ESP32.....	11
2.2.4 Modul Mikrofon INMP441	12
2.2.5 Modul Sensor Infra Merah TCRT5000	13
2.2.6 Modul LED Lampu Lalu Lintas	13
2.2.7 <i>Push-Button</i>	14

2.2.8 Buzzer.....	15
2.2.9 TinyML.....	15
2.2.10 Convolutional Neural Network (CNN).....	16
2.2.11 Confusion Matrix	18
2.2.12 Mel Frequency Cepstrum Coefficients (MFCC).....	19
BAB III METODE PENELITIAN	20
3.1 Alur Penelitian	20
3.2 Kebutuhan Perangkat Keras.....	22
3.3 Rancangan Sistem.....	22
3.3.1 Rancangan Umum.....	22
3.3.2 Rancangan Perangkat Keras	24
3.4 Rancangan Model TinyML.....	26
3.4.1 Akuisisi Data.....	26
3.4.2 Data Preprocessing	27
3.4.3 Modelling	28
3.4.4 Deployment	29
3.5 Rancangan Pengujian.....	30
3.5.1 Pengujian Perangkat Keras	30
3.5.2 Pengujian Model TinyML	31
3.5.3 Evaluasi Model TinyML.....	31
DAFTAR PUSTAKA	32

DAFTAR TABEL

Tabel 2.1 Penelitian Terkait.....	6
Tabel 2.2 <i>Literature Review</i>	6
Tabel 3.1 Kebutuhan Perangkat Keras	22
Tabel 3.2 Pengujian Perangkat Keras	30

DAFTAR GAMBAR

Gambar 2.1 Lampu Lalu Lintas	10
Gambar 2.2 Mobil Ambulans	11
Gambar 2.3 Mikrokontroler ESP32	12
Gambar 2.4 Modul Mikrofon INMP441	13
Gambar 2.5 Modul IR TCRT5000.....	13
Gambar 2.6 Modul LED Lampu Lalu Lintas	14
Gambar 2.7 <i>Push-Button</i>	14
Gambar 2.8 <i>Buzzer</i>	15
Gambar 2.9 Arsitektur TinyML.....	16
Gambar 2.10 Arsitektur CNN.....	17
Gambar 2.11 <i>Confusion Matrix</i>	18
Gambar 3.1 Diagram Alur Penelitian	20
Gambar 3.2 Blok Diagram Rancangan Sistem	23
Gambar 3.3 Desain Perangkat Keras Dari Samping.....	25
Gambar 3.4 Desain Perangkat Keras Dari Depan.....	25

BAB I

PENDAHULUAN

1.1 Latar Belakang

Setiap tahun, jumlah kendaraan di Indonesia terus meningkat. Menurut data yang dikutip dari GAIKINDO, penjualan mobil secara keseluruhan selama tahun 2022 mencapai 1,013 juta unit, mengalami pertumbuhan sebesar 17,4 persen dibandingkan dengan periode tahun 2021 sebelumnya. Data yang dirilis oleh GAIKINDO juga menunjukkan peningkatan signifikan dalam penjualan ritel, dengan lonjakan sebanyak 150.224-unit dari 863.358-unit pada periode 2021 [1]. Peningkatan jumlah kendaraan yang tidak diiringi dengan pelebaran atau perluasan infrastruktur jalan mengakibatkan kemacetan, khususnya di kota-kota besar.

Kemacetan menimbulkan dampak yang beragam, baik dari segi waktu, ekonomi, dan lingkungan [2]. Dampak lain yang ditimbulkan kemacetan adalah mengganggu kelancaran kendaraan darurat dalam menjalankan tugasnya [3]. Kurangnya pemahaman dan kesadaran pengemudi kendaraan bermotor terhadap aturan dan etika berlalu lintas menyebabkan rendahnya toleransi di antara pengguna jalan khususnya dalam memberi prioritas untuk ambulans melewati jalan saat terjebak dalam kemacetan, seperti dipengaruhi oleh pemberhentian lampu merah. Kendaraan darurat memiliki tugas untuk bergerak cepat ke lokasi tujuan. Tetapi pada kenyataannya, kejadian ambulans terjebak macet sehingga membuat pasien meninggal dalam perjalanan tidak sedikit [4].

Banyak kasus yang melibatkan pasien yang meninggal sebelum tiba di rumah sakit, akibat keadaan lalu lintas yang tidak kondusif. Lebih dari 20% pasien yang membutuhkan perawatan darurat telah meninggal dalam perjalanan ke rumah sakit karena penundaan karena kemacetan lalu lintas dan pengendara yang tidak kooperatif, kata Sekretaris Jenderal National Institute of Emergency Medicine (NIEM) Anucha Setthasathian [5]. Padahal terdapat peraturan yang dimuat dalam Undang-Undang No 22 Tahun 2009 mengenai Lalu Lintas dan Angkutan Jalan Pasal 134 yang menetapkan hak prioritas bagi beberapa pengguna jalan, seperti kendaraan pemadam kebakaran, ambulans yang membawa orang sakit, kendaraan yang

memberikan bantuan pada kecelakaan lalu lintas, dan kendaraan pimpinan lembaga negara, yang semuanya telah diatur dengan tepat dalam undang-undang [6].

Untuk saat ini alat pengatur lalu lintas yang biasa dikenal dengan lampu lalu lintas atau *traffic light* telah dipasang di berbagai macam persimpangan jalan dan area penyebrangan guna untuk mengatur pergerakan pengguna jalan, baik pengendara kendaraan bermotor maupun pejalan kaki [4] [7]. Namun, lampu lalu lintas saat ini masih memiliki kekurangan dalam mengatur kendaraan di saat kemacetan terjadi. Lampu lalu lintas belum mampu mendeteksi kepadatan lalu lintas secara *real-time*, sehingga pengaturan waktu lampu seringkali tidak efektif dan justru memperparah kemacetan. Selain itu, lampu lalu lintas saat ini belum mampu mendeteksi kendaraan darurat seperti ambulans yang membutuhkan prioritas untuk melewati jalan dengan cepat. Akibatnya, kendaraan darurat sering terhambat oleh kemacetan, yang dapat berakibat fatal dalam situasi kritis. Untuk melihat urgensi dibutuhkannya solusi di permasalahan ini, penulis melakukan wawancara kepada pihak kepolisian. Dalam wawancara, mereka menyoroti tantangan yang sering dihadapi dalam memberikan prioritas kepada kendaraan darurat, salah satunya ambulans, di tengah kemacetan lalu lintas.

Salah satu upaya dalam mengatasi permasalahan tersebut adalah membangun sebuah sistem yang dapat mengatur lampu lalu lintas di saat kendaraan darurat harus melewati persimpangan. Untuk membangun sistem tersebut diperlukan *embedded system* dan *tiny machine learning*. *Embedded system* adalah sistem yang biasanya dibangun dalam satu kesatuan sistem dan digunakan untuk menjalankan tugas tertentu [8]. Dalam konteks ini *embedded system* merupakan rangkaian lampu lalu lintas. *Tiny machine learning* atau TinyML adalah cabang studi dalam *machine learning* yang meneliti model yang bisa dijalankan pada perangkat kecil berdaya rendah, seperti mikrokontroler [9].

Ambulans menyampaikan informasi melalui variasi bunyi sirine yang khas. Setiap variasi bunyi sirine digunakan oleh supir ambulans dalam situasi yang berbeda-beda. Dalam permasalahan ini, TinyML digunakan untuk mendeteksi suara sirine ambulans dan mengatur lampu lalu lintas. Berdasarkan permasalahan tersebut, penulis membangun “Purwarupa Sistem Deteksi Dan Pengaturan Lampu Lalu Lintas Berbasis TinyML untuk Prioritas Layanan Gawat Darurat Ambulans”

Evaluasi model TinyML yang ditanamkan dalam sistem pada penelitian ini adalah pengukuran evaluasi utama untuk tugas klasifikasi dengan menggunakan *confusion matrix*. *Confusion matrix* adalah matriks yang berisi informasi tentang hasil prediksi klasifikasi dan data aktual yang dilakukan oleh sistem klasifikasi [10]. Selain itu akan dibahas juga mengenai evaluasi rangkaian alat yang digunakan seperti mikrofon yang digunakan untuk menangkap suara ambulans dan modul IR yang digunakan untuk mendeteksi kepadatan kendaraan.

1.2 Rumusan Masalah

Berdasarkan latar belakang yang sudah dijelaskan maka rumusan masalah yang diangkat dalam penelitian ini adalah sebagai berikut:

1. Bagaimana merancang dan mengimplementasikan sistem yang dapat mendeteksi suara sirine ambulans?
2. Bagaimana hasil evaluasi kinerja sistem dalam pengaturan lampu lalu lintas?

1.3 Tujuan Penelitian

Berdasarkan latar belakang yang sudah dijelaskan maka tujuan dari penelitian ini adalah memberikan jawaban dan solusi dari permasalahan diatas, yaitu sebagai berikut:

1. Merancang dan mengimplementasikan sistem yang dapat mendeteksi suara sirine ambulans dan mengatur lampu lalu lintas.
2. Mengetahui hasil evaluasi kinerja sistem dalam mengurangi penundaan dan meningkatkan prioritas layanan gawat darurat ambulans, terutama dalam perjalanan pasien ke rumah sakit.

1.4 Batasan Masalah

Berdasarkan latar belakang yang sudah dijelaskan maka batasan masalah pada penelitian ini adalah sebagai berikut:

1. Sistem akan difokuskan pada mendeteksi dan memberikan prioritas untuk suara sirine ambulans sebagai representasi layanan gawat darurat dalam konteks lalu lintas perkotaan.

2. Suara sirine ambulans yang akan dideteksi sistem adalah khusus yang digunakan saat membawa pasien dalam kondisi gawat darurat.
3. Pendeteksian suara dalam sistem akan difokuskan pada mikrofon, tanpa mempertimbangkan metode lain seperti kamera atau sensor visual.
4. Implementasi sistem akan diujicobakan dalam skala prototipe dengan ukuran 1-meter persegi.

1.5 Manfaat Penelitian

Manfaat yang diperoleh dari penelitian ini adalah sebagai berikut:

1. Penulis dapat mengimplementasikan ilmu yang didapat selama berkuliah di Institut Teknologi Sumatera.
2. Sebagai suatu inovasi terbaru dalam membantu kelancaran layanan darurat khususnya ambulans dalam menjemput dan mengantarkan pasien ke rumah sakit atau pusat layanan kesehatan darurat.
3. Dapat memberikan acuan untuk penelitian atau pengembangan selanjutnya dalam bidang pengaturan lalu lintas untuk layanan darurat.

1.6 Sistematika Penulisan

Untuk mempermudah pemahaman terhadap keseluruhan penulisan ini, penulis akan menguraikan sistematika penulisan yang terdiri dari lima bab, dengan susunan sebagai berikut:

1.6.1 BAB I PENDAHULUAN

Bab ini berisi penjelasan mengenai latar belakang, rumusan masalah, tujuan penelitian, batasan masalah, manfaat penelitian dan sistematika penulisan.

1.6.2 BAB II TINJAUAN PUSTAKA

Bab ini berisi tentang pembahasan penelitian terdahulu. Serta berisi teori penunjang yang berhubungan dengan penelitian yang dilakukan, seperti definisi lampu lalu lintas dan layanan darurat, TinyML, penjelasan tentang modul sensor yang digunakan serta konsep-konsep lain yang mendasari sistem.

1.6.3 BAB III METODE PENELITIAN

Bab ini berisi penjabaran mengenai alur penelitian, menjelaskan tentang langkah-langkah yang dilakukan dalam penelitian, alat dan bahan yang digunakan dalam penelitian, metode yang digunakan serta rancangan pengujian.

1.6.4 BAB IV HASIL DAN PEMBAHASAN

Bab ini menyajikan hasil penelitian, hasil pengujian, analisis hasil dan pembahasan hasil pengujian terhadap sistem.

1.6.5 BAB V KESIMPULAN DAN SARAN

Bab ini berisi kesimpulan dan saran dari hasil penelitian yang dilakukan oleh penulis. Kesimpulan didasarkan pada hasil uji sistem yang dilakukan secara bertahap dan saran diberikan untuk perkembangan penelitian selanjutnya. Penulis berharap agar penelitian ini dapat dikembangkan untuk mencapai hasil yang lebih baik dari sebelumnya.

BAB II

TINJAUAN PUSTAKA

2.1 Tinjauan Pustaka

Adapun beberapa penelitian terdahulu yang menjadi pendukung penelitian ini adalah sebagai berikut:

Tabel 2.1 Penelitian Terkait

Kode	Judul	Tahun
J1	Sistem Kontrol Lampu Lalu Lintas Untuk Layanan Darurat Berbasis Internet of Things (IoT) [3]	2019
J2	Rancang Bangun Traffic Light System Tanggap Darurat Berbasis IoT [11]	2020
J3	Intelligent Traffic Light for Emergency Vehicles Clearance [12]	2020
J4	An Intelligent IoT Based Traffic Light Management System: Deep Reinforcement Learning [13]	2022
J5	Recent development of smart traffic lights [14]	2021
J6	A smart traffic light using a microcontroller based on the fuzzy Logic [15]	2022

Tabel 2.2 Literature Review

Kode	Permasalahan	Metode	Hasil
J1	Mobil ambulans menghadapi kendala saat melintas di persimpangan dengan lampu lalu lintas merah dan antrian kendaraan Panjang.	Sistem kontrol lampu lalu lintas menggunakan Arduino Mega 2560 dan <i>web server</i> sebagai inputan untuk mengontrol 4 lampu lalu lintas di persimpangan dan 1 LCD sebagai penginformasian status lampu lalu lintas.	<i>Web server</i> mampu mengontrol lampu lalu lintas dengan tingkat keberhasilan sebesar 90,6%. LCD berhasil menampilkan informasi status lampu lalu lintas khususnya dalam keadaan prioritas untuk ambulans.

Kode	Permasalahan	Metode	Hasil
J2	Kepadatan kendaraan di persimpangan jalan menghambat kinerja kendaraan prioritas.	Sistem mikrokontroler lampu lalu lintas berbasis sensor suara.	Sistem mengatur kelancaran lalu lintas secara efektif, memberikan prioritas kepada kendaraan darurat.
J3	Sistem lampu lalu lintas saat ini dapat membuat ambulans terjebak dalam kemacetan atau menyebabkan kecelakaan saat melintasi lampu merah.	Sistem memungkinkan ambulans untuk mengubah status lampu lalu lintas. <i>Hardware</i> yang digunakan mencakup Arduino UNO dan Mega dengan <i>shield</i> jaringan (ZigBee).	Sistem mampu mengurangi waktu ambulans bergerak dari titik A ke B, dari 10 menit menjadi 8 menit.
J4	Trafik menjadi permasalahan yang membawa konsekuensi negatif seperti pemborosan waktu dan peningkatan kemungkinan kecelakaan.	Sistem memanfaatkan Distributed Multi-Agent Reinforcement Learning dan menggunakan sensor kamera pengawas.	Hasil simulasi menunjukkan sistem lebih efisien daripada kontrol lampu lalu lintas yang berlaku di Kota Shiraz, dilihat dari panjang antrian kendaraan rata-rata dan waktu tunggu di persimpangan.
J5	Diperlukan mekanisme agar lampu lalu lintas dapat mengelola alokasi waktu sinyal secara cerdas dan adaptif sesuai dengan kondisi aliran lalu lintas.	Pengaturan siklus lampu lalu lintas cerdas dapat dikelompokkan berdasarkan kepadatan lalu lintas, skenario untuk kendaraan darurat, dan kepentingan pejalan kaki.	Penelitian ini lebih bersifat analisis dan eksploratif terhadap metode dan teknologi yang digunakan dalam pengembangan lampu lalu lintas pintar.
J6	Kemacetan lalu lintas yang disebabkan oleh penumpukan kendaraan di jalan telah menjadi masalah penting, mengakibatkan gangguan bagi para pengemudi.	Sistem berbasis logika <i>fuzzy</i> menggunakan mikrokontroler ATmega2560. Sensor ultrasonik, inframerah, dan cahaya digunakan dalam sistem ini	Sistem ini menggunakan logika <i>fuzzy</i> untuk mempertimbangkan kepentingan dari bagian lalu lintas lain dengan memberikan waktu minimum 9 detik dan maksimum 27 detik dalam 1 siklus pengaturan lampu lalu lintas.

Penelitian pertama berjudul “Sistem Kontrol Lampu Lalu Lintas Untuk Layanan Darurat Berbasis Internet of Things (IoT)”. Dalam penelitian ini dibuat sebuah alat sistem kontrol lampu lalu lintas menggunakan web server sebagai inputan untuk mengontrol 4 lampu lalu lintas di persimpangan dan 1 LCD sebagai penginformasian status lampu lalu lintas. Mikrokontroler Arduino Mega 2560 digunakan untuk mengendalikan seluruh rangkaian. Hasil yang didapatkan adalah *web server* mampu mengontrol lampu lalu lintas dengan tingkat keberhasilan sebesar 90,6%. LCD dapat menampilkan informasi mengenai keadaan lampu lalu lintas dengan menjelaskan lampu lalu lintas dalam keadaan prioritas. Pengujian dilakukan, menunjukkan bahwa web server mampu mengontrol lampu lalu lintas dengan tingkat keberhasilan sebesar 90,6%. LCD berfungsi sebagai penampil informasi status lampu lalu lintas, khususnya dalam keadaan prioritas untuk ambulans [3].

Penelitian kedua berjudul “Rancang Bangun Traffic Light System Tanggap Darurat Berbasis IoT”. Penelitian ini menggunakan pendekatan dengan merancang sistem mikrokontroler lampu lalu lintas berbasis sensor suara. Sensor suara digunakan sebagai teknologi identifikasi otomatis dengan memanfaatkan gelombang suara diatas 118 desibel. Hasil yang didapatkan adalah sistem dapat memberikan prioritas kepada kendaraan darurat seperti ambulans, mobil pemadam kebakaran, atau pejabat [11].

Penelitian ketiga berjudul “Intelligent Traffic Light for Emergency Vehicles Clearance”. Penelitian ini memberikan solusi dengan membuat sebuah sistem yang memungkinkan ambulans mengendalikan lampu lalu lintas untuk mengatasi masalah keterlambatan. Sistem ini memungkinkan ambulans untuk mengubah status lampu lalu lintas, memungkinkan pergerakan darurat tanpa hambatan. *Hardware* yang digunakan mencakup Arduino UNO dan Mega dengan shield jaringan (ZigBee). Hasil evaluasi menunjukkan bahwa dengan sistem ini waktu yang diperlukan untuk melewati titik A ke titik berhasil diperkecil dari 10 menit menjadi 8 menit [12].

Penelitian keempat berjudul “An Intelligent IoT Based Traffic Light Management System: Deep Reinforcement Learning”. Penelitian ini memanfaatkan Internet of Things (IoT) dan kecerdasan buatan (AI) untuk mengontrol lampu lalu lintas secara efisien. Sistem ini menggunakan sensor seperti kamera pengawas untuk

mendapatkan informasi trafik real-time. Sistem kontrol lampu lalu lintas cerdas diimplementasikan dengan memanfaatkan Distributed Multi-Agent Reinforcement Learning (MARL) dan memanfaatkan data trafik dari persimpangan yang berdekatan bersama dengan informasi lokal. Evaluasi dilakukan melalui simulasi numerik untuk dua persimpangan sintetis dengan data simulasi dan peta dunia nyata di Kota Shiraz, Iran. Data trafik dunia nyata diperoleh dari organisasi transportasi dan lalu lintas pemerintah kota. Hasil simulasi menunjukkan bahwa pendekatan yang diusulkan lebih efisien daripada penjadwalan kontrol lampu lalu lintas berdasarkan waktu tetap yang berlaku di Shiraz, dilihat dari panjang antrian kendaraan rata-rata dan waktu tunggu di persimpangan [13].

Penelitian kelima berjudul “Recent development of smart traffic lights”. Penelitian ini lebih bersifat analisis dan eksploratif terhadap metode dan teknologi yang digunakan dalam pengembangan lampu lalu lintas pintar. Evaluasi lebih pada bagaimana teknologi tersebut dapat diterapkan secara cerdas dan adaptif untuk mengatasi masalah kemacetan lalu lintas dalam situasi yang berbeda, seperti kepadatan lalu lintas, keadaan darurat, dan kebutuhan pejalan kaki [14].

Penelitian keenam berjudul “A smart traffic light using a microcontroller based on the fuzzy Logic”. Penelitian ini memberikan solusi dengan menciptakan lampu lalu lintas pintar menggunakan mikrokontroler ATmega2560 berbasis logika fuzzy. Empat set sensor ultrasonik, inframerah, dan cahaya ditempatkan di empat bagian menentukan proses pengaturan waktu lampu lalu lintas. Ketika antrian kendaraan mencapai sensor, sinyal dikirim sebagai input digital mikrokontroler. Sistem ini menggunakan logika *fuzzy* untuk mempertimbangkan kepentingan dari bagian lalu lintas lain dengan memberikan waktu minimum 9 detik dan maksimum 27 detik dalam 1 siklus pengaturan lampu lalu lintas.

2.2 Dasar Teori

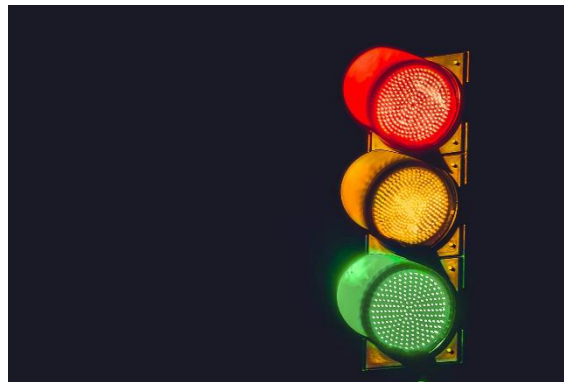
2.2.1 Lampu Lalu Lintas

Lampu lalu lintas, juga dikenal sebagai sinyal lalu lintas, lampu berhenti, lampu stop-and-go, robot atau semaphore, adalah perangkat pemberi sinyal yang ditempatkan di persimpangan jalan, tempat penyeberangan pejalan kaki, atau lokasi

lainnya. Perangkat ini biasanya digunakan untuk menunjukkan kapan pengemudi harus berhenti dan kemudian melanjutkan dengan hati-hati [16].

Lampu lalu lintas modern menggunakan berbagai warna, termasuk merah, kuning, dan hijau, untuk menandakan kapan pengemudi harus berhenti, memperlambat, atau melanjutkan perjalanan. Susunan yang paling umum adalah merah di atas, kuning di tengah, dan hijau di bawah. Lampu merah menandakan bahwa pengemudi harus berhenti di perempatan dan tidak melanjutkan perjalanan hingga lampu berubah menjadi hijau. Lampu kuning berfungsi sebagai peringatan bahwa lampu akan berubah menjadi merah dan pengemudi harus memperlambat dan bersiap untuk berhenti. Lampu hijau menunjukkan bahwa pengemudi aman untuk melewati persimpangan [17].

Menurut Institute of Transportation Engineers, lampu lalu lintas adalah salah satu alat paling efektif untuk mengatur lalu lintas dan meningkatkan keselamatan di persimpangan. Mereka dapat digunakan untuk mengontrol arus kendaraan, sepeda dan pejalan kaki, mengurangi potensi konflik dan tabrakan. Mereka juga membantu meningkatkan mobilitas dan mengurangi keterlambatan bagi semua pengguna jalan [18].



Gambar 2.1 Lampu Lalu Lintas

2.2.2 Layanan Darurat Ambulans

Layanan ambulans adalah komponen penting dari Emergency Medical Services (EMS) yang menyediakan perawatan medis darurat dan transportasi untuk individu yang sakit atau terluka [19]. Ambulans dikelola oleh profesional medis terlatih, seperti paramedis dan Emergency Medical Team (EMT), yang mampu memberikan perawatan penyelamat jiwa, seperti Cardiopulmonary Resuscitation (CPR) dan defibrilasi, dalam perjalanan ke rumah sakit. Penggunaan ambulans dapat

sangat meningkatkan peluang bertahan hidup bagi individu yang menderita cedera atau penyakit serius [20]. Menurut sebuah penelitian yang diterbitkan dalam *Journal of Emergency Medical Services (JEMS)*, pasien dengan luka parah yang menerima perawatan dari paramedis di lapangan sebelum tiba di rumah sakit memiliki peluang bertahan hidup 40% lebih tinggi daripada mereka yang tidak menerima perawatan tersebut [21].

Ambulans menyampaikan informasi melalui variasi bunyi sirine yang khas, dan setiap variasi bunyi digunakan dalam situasi yang berbeda-beda. Pertama, suara yang menyerupai palang pintu kereta dengan tempo lambat menandakan bahwa ambulans sedang dalam perjalanan menuju lokasi pasien. Kedua, perubahan sedikit lebih cepat dalam tempo sirene mengindikasikan bahwa ambulans membawa pasien yang tidak dalam kondisi darurat. Ketiga, ketika frekuensi suara sirene meningkat, ini menunjukkan bahwa ambulans membawa pasien dalam kondisi gawat darurat. Terakhir, suara sirene dengan tempo lebih lambat dan panjang memberi sinyal bahwa ambulans membawa jenazah. Sistem ini memberikan pengguna jalan informasi yang jelas tentang keadaan ambulans dan kebutuhan mendesaknya [22] [23].



Gambar 2.2 Mobil Ambulans

2.2.3 Mikrokontroler ESP32

Diciptakan oleh Espressif Systems, ESP32 adalah seri sistem-on-a-chip (SoC) yang murah dan hemat daya dengan kemampuan Wi-Fi dan Bluetooth dual-mode! Keluarga ESP32 mencakup chip ESP32-D0WDQ6 (dan ESP32-D0WD), ESP32-D2WD, ESP32-S0WD, dan sistem dalam paket (SiP) ESP32-PICO-D4. Di pusatnya, terdapat mikroprosesor Tensilica Xtensa LX6 dual-core atau single-core dengan

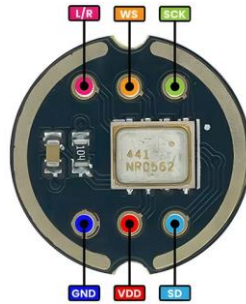
kecepatan clock hingga 240 MHz. ESP32 sangat terintegrasi dengan sakelar antenna bawaan, balun RF, penguat daya, penerima amplifier rendah-noise, filter, dan modul manajemen daya. Dirancang untuk perangkat mobile, elektronik yang dapat dipakai, dan aplikasi IoT, ESP32 mencapai konsumsi daya yang sangat rendah melalui fitur penghematan daya, termasuk pemangkasan clock dengan resolusi tinggi, beberapa mode daya, dan penskalaan daya dinamis [24].



Gambar 2.3 Mikrokontroler ESP32

2.2.4 Modul Mikrofon INMP441

INMP441 adalah mikrofon MEMS digital output, omnidireksional yang memiliki kinerja tinggi, konsumsi daya rendah, dan port bawah. Solusi lengkap INMP441 terdiri dari sensor MEMS, kondisi sinyal, konverter analog-digital, filter anti-aliasing, manajemen daya, dan antarmuka I²S standar industri 24-bit. Antarmuka I²S memungkinkan INMP441 terhubung langsung ke prosesor digital, seperti DSP dan mikrokontroler, tanpa memerlukan codec audio dalam sistem. INMP441 memiliki Signal-to-Noise Ratio (SNR) yang tinggi, menjadikannya pilihan yang sangat baik untuk aplikasi di daerah dekat. INMP441 memiliki respons frekuensi wideband yang datar, menghasilkan suara alami dengan daya paham yang tinggi. INMP441 tersedia dalam kemasan permukaan berukuran tipis $4,72 \times 3,76 \times 1$ mm. Kompatibel dengan solder reflow tanpa degradasi sensitivitas [25].



Gambar 2.4 Modul Mikrofon INMP441

2.2.5 Modul Sensor Infra Merah TCRT5000

Unit sensor IR TCRT5000 mencakup Photodiode yang dikopelkan dengan Phototransistor dalam satu paket. Photodiode (Tx) dalam sensor ini memiliki dua pin, yaitu anoda dan katoda, yang digunakan untuk menghasilkan sinyal inframerah. Demikian juga, Phototransistor dalam sensor ini memiliki dua pin, yaitu emitter dan kolektor. Kedua pin ini memainkan peran kunci dalam membaca sinyal inframerah yang dipantulkan Kembali [26]. Sensor inframerah ini digunakan untuk mendeteksi keberadaan suatu objek dengan menggunakan pemrograman tertentu; juga dapat mengukur jarak objek di depannya. Namun, jarak dapat diukur hanya untuk rentang kecil objek karena gangguan lingkungan. Jadi, sensor ini berguna jika ingin mengukur jarak objek di depannya. Sensor ini dapat digunakan sebagai sensor analog/digital. Jika hanya ingin mendeteksi keberadaan target atau objek, maka sensor ini dapat digunakan sebagai sensor digital. Jika ingin mengukur jarak target, maka sensor ini digunakan sebagai sensor analog [27].



Gambar 2.5 Modul IR TCRT5000

2.2.6 Modul LED Lampu Lalu Lintas

Modul lampu lalu lintas LED umumnya digunakan dalam simulasi lalu lintas. Biasanya, modul ini terdiri dari tiga unit LED, masing-masing mengeluarkan cahaya dengan warna yang berbeda: merah, kuning dan hijau [28].

1. LED Merah: Menunjukkan bahwa lalu lintas harus berhenti.
2. LED Kuning: Memperingatkan bahwa isyarat akan berubah, baik dari merah ke hijau atau dari hijau ke merah. Ini merupakan isyarat bagi pengemudi untuk bersiap-siap berhenti.
3. LED Hijau: Menunjukkan bahwa lalu lintas dapat melanjutkan [17].



Gambar 2.6 Modul LED Lampu Lalu Lintas

2.2.7 Push-Button

Button atau tombol berfungsi sebagai *input* sederhana untuk mikrokontoler. *Button* digunakan untuk menyambungkan sinyal HIGH atau LOW pada pin mikrokontroler sehingga dapat dikenali sebagai input saat tombol ditekan. *Button* sering digunakan untuk tugas seperti menghidupkan atau mematikan perangkat, memilih opsi, memicu tindakan, atau memasukkan perintah. Terdapat berbagai bentuk, ukuran, dan desain, dan dapat ditemukan dalam berbagai perangkat elektronik, mulai dari alat rumah tangga sederhana hingga mesin industri yang kompleks [29] [30].



Gambar 2.7 Push-Button

2.2.8 Buzzer

Buzzer merupakan perangkat untuk menghasilkan suara sebagai *output*, dimana suara yang dihasilkan oleh *buzzer* bergantung pada tegangan yang diberikan kepadanya. Jika *buzzer* diberi tegangan penuh sebesar 5V, maka suara yang dihasilkan akan terdengar keras, sedangkan jika tegangan yang diberikan berupa frekuensi atau osilasi, maka suara yang dihasilkan akan berupa nada [31].



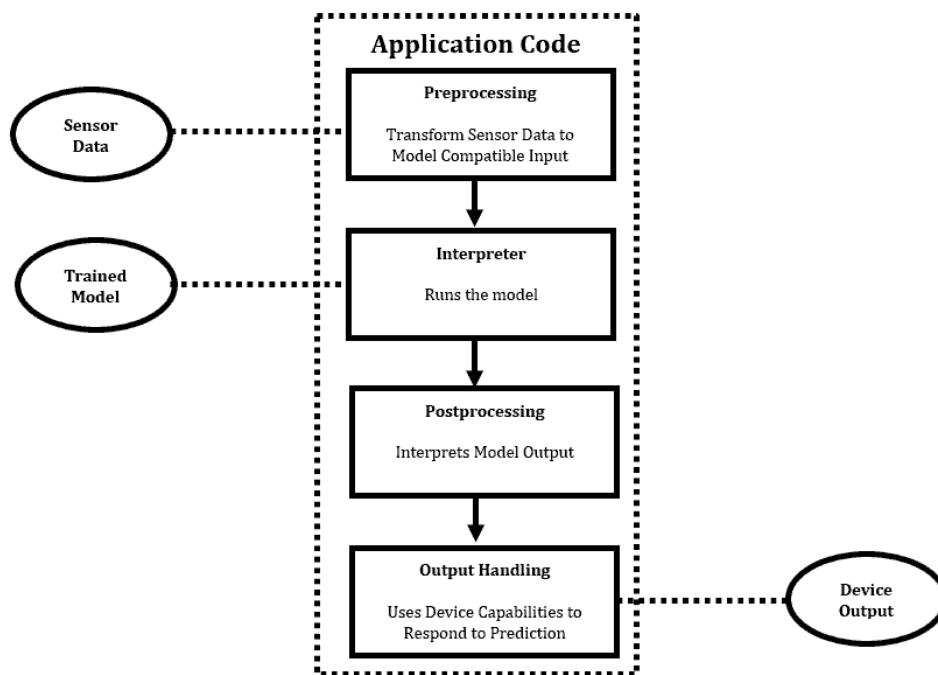
Gambar 2.8 Buzzer

2.2.9 TinyML

TinyML merupakan bidang studi dalam Pembelajaran Mesin dan Sistem Tertanam yang mengeksplorasi jenis model yang dapat dijalankan pada perangkat kecil dengan daya rendah, seperti mikrokontroler. Ini memungkinkan inferensi model dengan latensi rendah, konsumsi daya rendah, dan *bandwidth* rendah pada *edge devices* [9]. Sementara CPU konsumen standar mengonsumsi antara 65-watt dan 85-watt, dan GPU konsumen standar mengonsumsi antara 200-watt hingga 500-watt, mikrokontroler tipikal mengonsumsi daya dalam urutan miliwatt atau mikrowatt. Ini sekitar seribu kali lebih rendah dari konsumsi daya. Konsumsi daya rendah ini memungkinkan perangkat TinyML berjalan tanpa terhubung ke baterai selama minggu, bulan, dan dalam beberapa kasus, bahkan tahun, sambil menjalankan aplikasi ML pada perangkat tepi atau *edge devices* [32].

Keuntungan dari TinyML melibatkan beberapa aspek penting. Dalam hal latensi, data tidak perlu ditransfer ke server untuk inferensi karena model beroperasi pada perangkat tepi. Transfer data biasanya memerlukan waktu, yang menyebabkan keterlambatan kecil. Dengan menghilangkan kebutuhan ini, latensi berkurang. Selain itu, penghematan energi juga menjadi manfaat utama karena mikrokontroler

membutuhkan jumlah daya yang sangat kecil, memungkinkan mereka beroperasi untuk periode yang lama tanpa perlu diisi daya. Tanpa infrastruktur server yang luas juga diperlukan karena tidak ada transfer informasi, yang berarti penghematan energi, sumber daya, dan biaya. Pengurangan bandwidth juga terjadi karena inferensi membutuhkan sedikit atau tanpa koneksi internet. Sensor di perangkat mengambil dan memproses data, sehingga tidak ada data sensor mentah yang terus-menerus dikirim ke server. Keamanan data juga terjamin karena data Anda tidak disimpan di server, karena model berjalan di tepi. Tidak ada transfer informasi ke server meningkatkan jaminan privasi data [9] [32] [33].



Gambar 2.9 Arsitektur TinyML

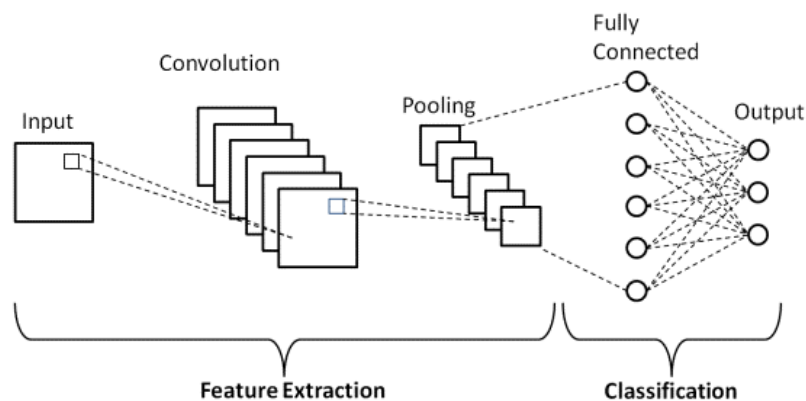
2.2.10 Convolutional Neural Network (CNN)

Convolutional Neural Network (CNN) merupakan model pembelajaran mesin yang efektif untuk menganalisis data visual, menggunakan prinsip konvolusi untuk mengekstrak fitur dalam gambar. Terinspirasi oleh pola koneksi otak manusia, CNN mampu mengidentifikasi objek dalam gambar dan sering digunakan dalam berbagai aplikasi visi komputer seperti pengenalan gambar dan deteksi objek. Berbeda dengan jaringan saraf tradisional, CNN memproses gambar secara komprehensif dan dapat

mengungguli kinerja model-model tradisional pada tugas-tugas yang berkaitan dengan gambar, serta dalam tingkat yang lebih kecil, pemrosesan suara dan audio [34].

Pada dasarnya, CNN bekerja dengan menerima gambar sebagai input dan mengolahnya melalui serangkaian lapisan (*layers*) untuk menghasilkan output yang diinginkan, seperti klasifikasi gambar. Lapisan-lapisan utama dalam CNN meliputi lapisan konvolusi (*convolutional layer*), lapisan aktivasi (*activation layer*), lapisan pooling (*pooling layer*), dan lapisan sepenuhnya terhubung (*fully connected layer*). Lapisan konvolusi bertugas menerapkan filter atau kernel pada gambar input untuk menghasilkan peta fitur (*feature map*). Filter ini bergerak melintasi gambar dengan langkah tertentu (*stride*), mengidentifikasi pola dan fitur dasar seperti tepi, sudut, dan tekstur [34] [35].

Setiap lapisan dalam CNN memiliki bobot (*weights*) dan bias yang dioptimalkan selama proses pelatihan. Bobot ini menentukan seberapa kuat suatu fitur dikenali oleh jaringan, sementara bias membantu menggeser fungsi aktivasi. Setelah melalui lapisan konvolusi, peta fitur tersebut seringkali melewati lapisan aktivasi seperti ReLU (*Rectified Linear Unit*) yang memperkenalkan non-linearitas ke dalam model. Selanjutnya, lapisan pooling, seperti max-pooling, mengurangi dimensi peta fitur sambil mempertahankan informasi penting, sehingga mengurangi beban komputasi dan risiko *overfitting*. Akhirnya, lapisan sepenuhnya terhubung menggabungkan semua fitur yang diekstrak dan menghasilkan output akhir, misalnya probabilitas untuk setiap kelas dalam tugas klasifikasi [34] [35].



Gambar 2.10 Arsitektur CNN

2.2.11 Confusion Matrix

Dalam mengevaluasi model TinyML yang ditanamkan dalam sistem, metrik yang digunakan melibatkan penggunaan *confusion matrix*. *Confusion matrix* menyajikan informasi detail tentang performa model dengan membandingkan hasil klasifikasi terhadap kelas sebenarnya, termasuk *True Positive*, *True Negative*, *False Positive*, dan *False Negative* [36]. Dengan mengamati elemen-elemen ini, bisa dilakukan pengukuran tingkat *accuracy*, *precision*, dan *recall* yang krusial untuk menilai sejauh mana model dapat membedakan suara sirine ambulans dengan akurasi tinggi [37].

		Nilai Sebenarnya	
		TRUE	FALSE
Nilai Prediksi	TRUE	TP (True Positive) <i>Correct result</i>	FP (False Positive) <i>Unexpected result</i>
	FALSE	FN (False Negative) <i>Missing result</i>	TN (True Negative) <i>Correct absence of result</i>

Gambar 2.11 Confusion Matrix

$$accuracy = \frac{TP + TN}{TP + TN + FP + FN} \quad (II.1)$$

$$precision = \frac{TP}{TP + FP} \quad (II.2)$$

$$recall = \frac{TP}{TP + FN} \quad (II.3)$$

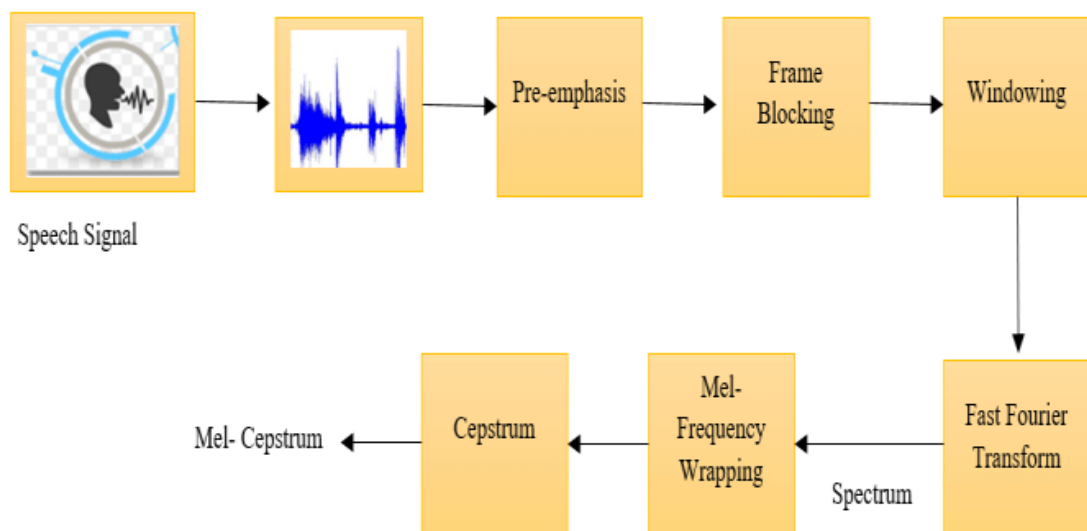
Accuracy didefinisikan sebagai tingkat kedekatan antara nilai prediksi dengan nilai aktual. *Precision* adalah tingkat ketepatan antara informasi yang diminta oleh pengguna dengan jawaban yang diberikan oleh sistem. *Recall* adalah tingkat keberhasilan sistem dengan menemukan kembali sebuah informasi [10]. Istilah *true positive*, *true negative*, *false positive* dan *false negative* membandingkan hasil dari klasifikasi yang diuji dengan nilai aktual yang terpercaya. Istilah *positive* dan *negative* merujuk prediksi klasifikasi, dan istilah *true* dan *false* mengacu pada apakah prediksi tersebut sesuai dengan penilaian aktual [36].

2.2.12 Mel Frequency Cepstrum Coefficients (MFCC)

Mel Frequency Cepstrum Coefficients atau MFCC adalah metode yang umum digunakan dalam teknologi pengenalan suara, baik untuk pengenalan penutur (*speaker recognition*) maupun pengenalan ucapan (*speech recognition*). Metode ini berfungsi untuk melakukan ekstraksi fitur, yakni proses mengonversi sinyal suara menjadi sejumlah parameter penting. Beberapa keunggulan dari MFCC adalah:

1. Mampu menangkap karakteristik suara yang penting untuk pengenalan, sehingga dapat mengekstrak informasi esensial dari sinyal suara.
2. Menghasilkan data dalam jumlah minimal tanpa menghilangkan informasi penting.
3. Mengadaptasi cara kerja organ pendengaran manusia dalam memproses sinyal suara.

Perhitungan pada MFCC menggunakan analisis jangka pendek (*short-term analysis*), karena sinyal suara bersifat *quasi-stationary*. Analisis dalam waktu pendek (sekitar 10-30 milidetik) menunjukkan sinyal suara yang stasioner, sementara dalam periode yang lebih lama, karakteristik sinyal akan berubah seiring dengan ucapan. MFCC juga meniru cara kerja sistem pendengaran manusia, di mana frekuensi rendah difilter secara linier (di bawah 1000Hz), dan frekuensi tinggi secara logaritmik (di atas 1000Hz). Berikut ini adalah blok diagram untuk MFCC [38].



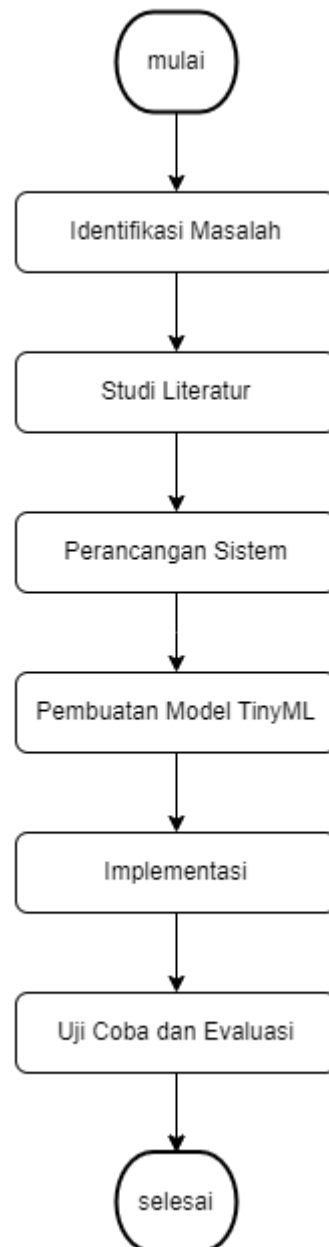
Gambar 2.12 Blok Diagram MFCC

BAB III

METODE PENELITIAN

3.1 Alur Penelitian

Penelitian “Purwarupa Sistem Deteksi Dan Pengaturan Lampu Lalu Lintas Otomatis Berbasis TinyML Untuk Prioritas Layanan Gawat Darurat Ambulans” memiliki alur yang ditunjukkan Gambar 3.1 berikut:



Gambar 3.1 Diagram Alur Penelitian

Berikut adalah penjelasan langkah-langkah alur penelitian dari diagram alur penelitian diatas:

1. Identifikasi Masalah

Langkah pertama adalah mengidentifikasi masalah yang ingin dipecahkan. Dalam konteks ini, terdapat masalah terhadap kondisi lalu lintas diakibatkan peningkatan jumlah kendaraan dan dampaknya terhadap layanan gawat darurat, khususnya ambulans.

2. Studi Literatur

Setelah mengidentifikasi masalah, langkah berikutnya adalah melakukan studi literatur untuk memahami solusi-solusi terdahulu, penelitian sejenis, dan produk yang telah ada. Analisis teknologi TinyML dan penerapannya dalam pengolahan suara menjadi bagian kritis dalam merancang solusi baru.

3. Perancangan Sistem

Langkah selanjutnya adalah merancang sistem. Pada tahap ini, dilakukan perencanaan desain pada sistem, bagaimana sistem akan beroperasi, serta bagaimana interaksi antara sistem dan pengguna akan terjadi.

4. Pembuatan Model TinyML

Pada tahap ini dilakukan pembuatan model TinyML yang akan digunakan sistem. Langkah-langkah yang dilakukan dalam tahap ini adalah pengumpulan data, melatih model dan evaluasi model menggunakan *confusion matrix*.

5. Implementasi

Setelah pembuatan model TinyML, langkah berikutnya adalah melakukan implementasi terhadap sistem. Tahap ini berupa pengkodean pada sistem dan *deploy* model TinyML pada sistem.

6. Uji Coba dan Evaluasi

Langkah berikutnya adalah melakukan uji coba terhadap sistem yang telah diimplementasikan dalam skenario simulasi kondisi lalu lintas. Data akan dikumpulkan dan hasilnya akan dievaluasi untuk melihat kinerja sistem dalam pengaturan jalan lalu lintas, memberikan prioritas dan mengoptimalkan pergerakan kendaraan darurat.

3.2 Kebutuhan Perangkat Keras

Otak penggerak sistem yang menjadi kebutuhan pada perangkat keras adalah mikrokontroler ESP32. Aktuator yang digunakan dalam perangkat keras terdapat 3 jenis yaitu modul LED lampu lalu lintas, *push-button*, dan *buzzer*. Adapun sensor yang digunakan di dalam perangkat keras terdapat 2 jenis yaitu modul mikrofon INMP441 dan modul sensor infra merah TCRT5000. Kebutuhan perangkat keras dapat dilihat pada Tabel 3.1.

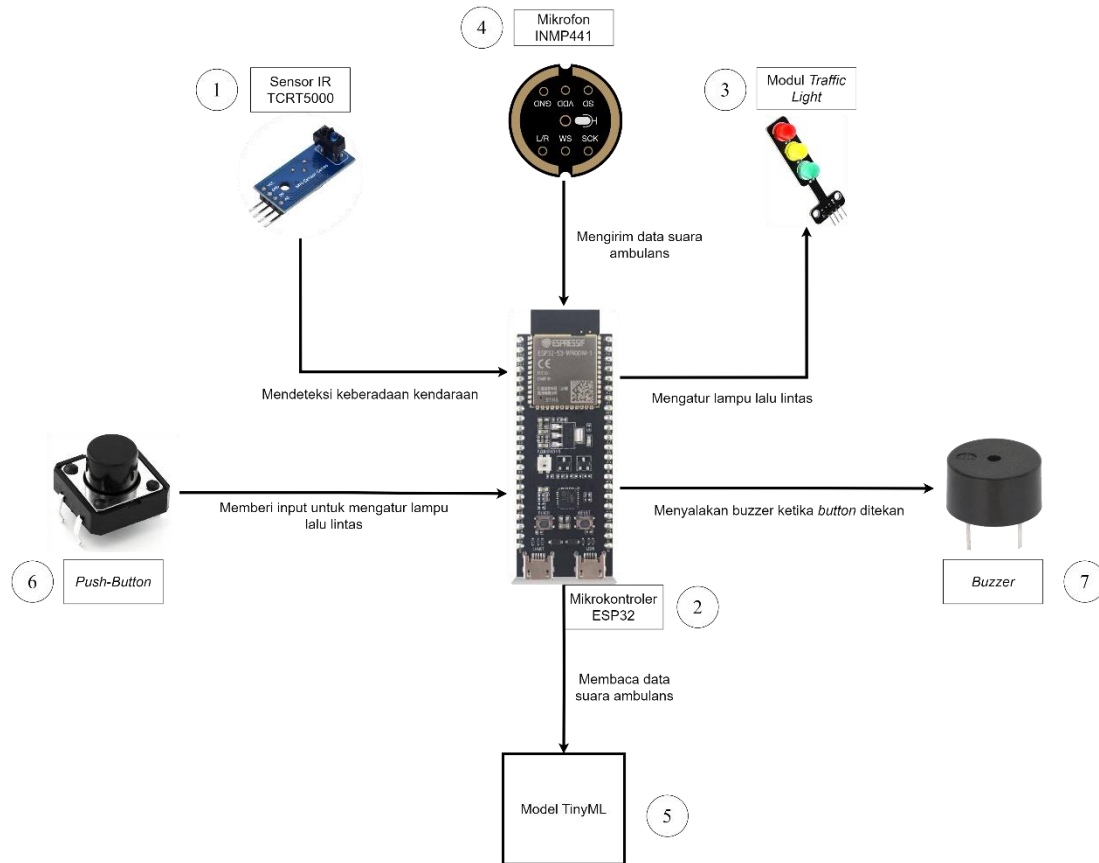
Tabel 3.1 Kebutuhan Perangkat Keras

No	Nama Perangkat	Fungsi
1	Mikrokontroler ESP32	Mikrokontroler yang berfungsi sebagai pengontrol aktuator dan sensor.
2	Modul LED Lampu Lalu Lintas	Modul ini digunakan untuk mengatur jalan lalu lintas.
3	<i>Push-Button</i>	Berfungsi sebagai tombol untuk pejalan kaki ketika ingin menyeberangi jalan.
4	<i>Buzzer</i>	Berfungsi sebagai pemberi tahu pejalan kaki kapan harus menyeberangi jalan.
5	Modul Mikrofon INMP441	Berfungsi sebagai penangkap suara ambulans yang akan melewati lampu lalu lintas.
6	Modul Sensor Infra Merah TCRT5000	Digunakan untuk mendeteksi penumpukan kendaraan di persimpangan jalan.

3.3 Rancangan Sistem

3.3.1 Rancangan Umum

Sistem dirancang dari berbagai komponen dan diperlukan sebuah diagram yang menunjukkan hubungan antara tiap komponen yang digunakan dalam penelitian ini. Rancangan sistem disajikan dalam bentuk diagram blok sistem seperti yang ditampilkan pada Gambar 3.2 berikut:



Gambar 3.2 Blok Diagram Rancangan Sistem

Gambar 3.2 merupakan blok diagram untuk menggambarkan sistem secara umum di dalam penelitian ini. Sistem akan bekerja dalam tiga situasi yaitu situasi normal, situasi ketika ambulans menyalakan sirine, dan ketika pejalan kaki ingin menyeberangi jalan. Berikut penjelasan proses yang terjadi pada tabel 3.2 sampai tabel 3.4.

Tabel 3.2 Situasi Normal

No	Nama Perangkat	Proses
1	Modul Sensor Infra Merah TCRT5000	Mendeteksi keberadaan kendaraan dan mengirimkan data ke mikrokontroler.
2	Mikrokontroler ESP32	Mengubah status lampu lalu lintas
3	Modul LED Lampu Lalu Lintas	Menyala sesuai perintah dari mikrokontroler.

Tabel 3.3 Situasi Ambulans Menyalakan Sirine

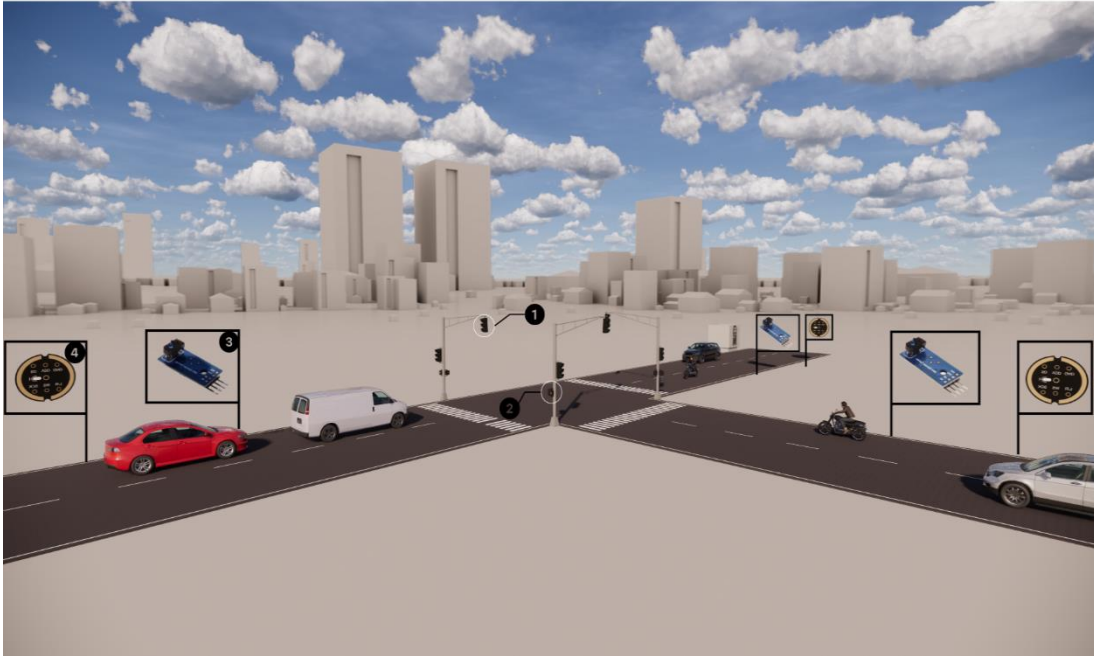
No	Nama Perangkat	Proses
4	Modul Mikrofon INMP441	Menangkap suara sirine ambulans dan mengirimkan data ke mikrokontroler.
5	Model TinyML	Mengenali suara ambulans.
2	Mikrokontroler ESP32	Mengubah status lampu lalu lintas
3	Modul LED Lampu Lalu Lintas	Menyala sesuai perintah dari mikrokontroler.

Tabel 3.4 Situasi Pejalan Kaki Menyeberangi Jalan

No	Nama Perangkat	Proses
6	<i>Push-Button</i>	Memberikan input ke mikrokontroler untuk mengubah status lampu lalu lintas.
2	Mikrokontroler ESP32	Mengubah status lampu lalu lintas
3	Modul LED Lampu Lalu Lintas	Menyala sesuai perintah dari mikrokontroler.
7	<i>Buzzer</i>	Mengeluarkan suara sebagai tanda untuk menyeberangi jalan.

3.3.2 Rancangan Perangkat Keras

Rancangan perangkat pada penelitian ini merupakan rancangan perangkat keras berupa bentuk fisik miniatur persimpangan jalan lalu lintas. Desain rancangan miniatur persimpangan jalan lalu lintas dapat dilihat pada Gambar 3.3 dan Gambar 3.4.



Gambar 3.3 Desain Perangkat Keras Dari Samping






Gambar 3.4 Desain Perangkat Keras Dari Depan

Keterangan dari penomoran pada gambar-gambar di atas tertera pada tabel 3.5.

Tabel 3.5 Penomoran Alat Pengikat Keras

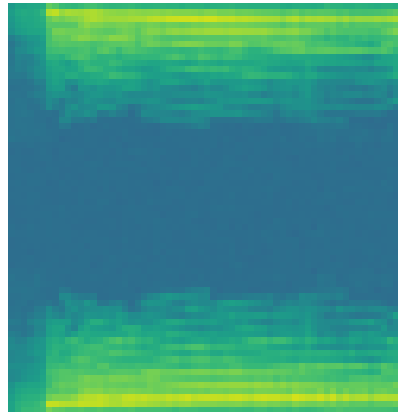
No	Nama Perangkat	Gambar
----	----------------	--------

No	Nama Perangkat	Gambar
1	Lampu lalu lintas	
2	Button dan buzzer	
3	Mikrofon	
4	Sensor IR	

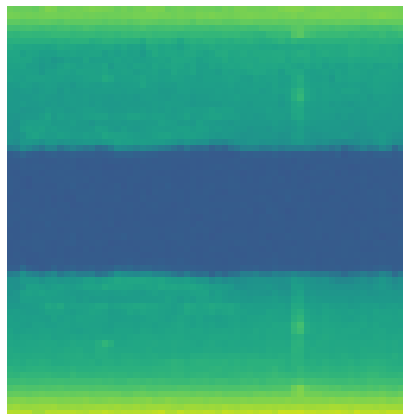
3.4 Rancangan Model TinyML

3.4.1 Akuisisi Data

Untuk akuisisi, data diakuisisi dari situs Kaggle. Dataset terdiri dari file audio berformat WAV yang masing-masing berdurasi 3 detik. File-file tersebut berisi suara sirine dari ambulans dan suara jalan lalu lintas. Setiap kategori memiliki 200 file suara, serta 200 gambar spektrogram untuk setiap file audio. Spektrogram adalah representasi visual yang menunjukkan bagaimana konten frekuensi dari sinyal suara berubah seiring waktu. Spektrogram mengonversi data audio menjadi format gambar 2D, di mana satu sumbu mewakili waktu, sumbu lainnya mewakili frekuensi, dan intensitas warna menggambarkan amplitudo.



Gambar 3.5 Spektrogram Sirine Ambulan



Gambar 3.6 Spektrogram Suara Jalan Lalu Lintas

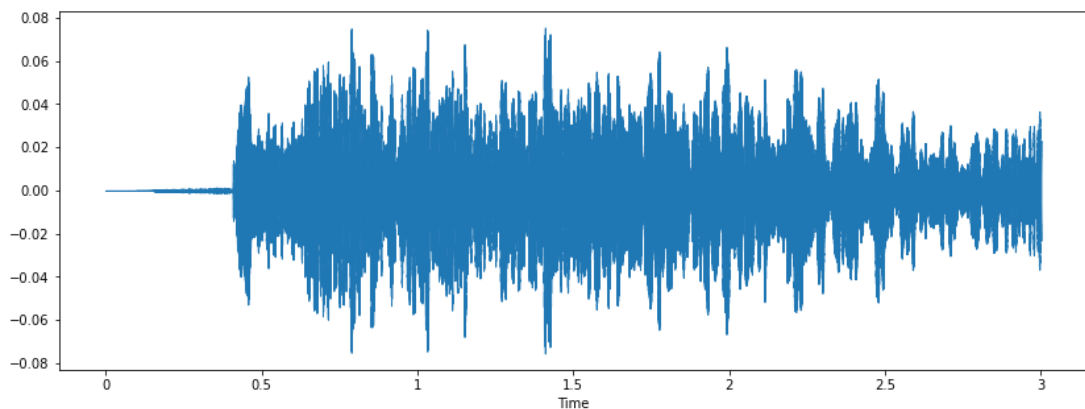
3.4.2 Data Preprocessing

Proses preprocessing data dimulai dengan memuat file audio yang berisi suara sirine ambulans dan suara jalan lalu lintas. Setelah file audio dimuat, langkah pertama adalah mengekstrak karakteristik penting dari sinyal suara tersebut. Karakteristik ini membantu dalam merepresentasikan informasi tonal dan frekuensi yang ada dalam suara.

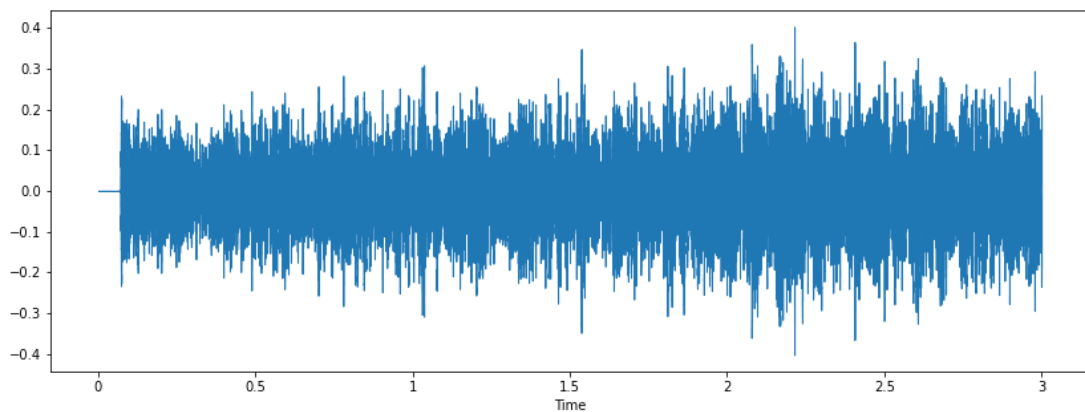
Selanjutnya, dari sinyal suara yang telah dimuat, dihasilkan representasi yang menangkap variasi frekuensi suara selama waktu tertentu. Proses ini melibatkan analisis spektral untuk mendapatkan gambaran lebih mendalam tentang bagaimana suara berubah seiring waktu.

Setelah karakteristik diekstraksi, rata-rata dari semua fitur yang diperoleh dihitung untuk menciptakan satu set fitur yang lebih sederhana namun tetap informatif. Fitur-fitur ini kemudian disimpan bersama dengan label kategori yang sesuai untuk setiap file audio.

Secara keseluruhan, proses ini bertujuan untuk mengubah data audio mentah menjadi representasi numerik yang dapat digunakan dalam tahap berikutnya, yaitu modeling, untuk membangun model yang mampu mendeteksi suara ambulans secara efektif.



Gambar 3.7 Gelombang Suara Sirine Ambulans



Gambar 3.8 Gelombang Suara Jalan Lalu Lintas

3.4.3 Modelling

Pada bagian ini, data latih dan data uji terlebih diubah ke dalam bentuk yang dapat diproses oleh model. Kemudian, model dibangun menggunakan *Convolutional Neural Network* (CNN). Model CNN yang dirancang terdiri dari beberapa lapisan konvolusi yang bertugas untuk mendeteksi pola-pola dari fitur suara yang telah diekstraksi pada tahap preprocessing. Setiap lapisan konvolusi akan memproses data

secara bertahap, menangkap pola frekuensi atau fitur lain yang ada dalam sinyal suara. Setelah data diproses oleh beberapa lapisan, hasilnya kemudian disederhanakan melalui teknik pooling untuk mereduksi dimensi dan menjaga informasi penting.

Selanjutnya, hasil akhir dari pemrosesan lapisan konvolusi ini dihubungkan dengan lapisan neuron padat, yang menggabungkan semua informasi yang telah ditangkap dari data sebelumnya. Lapisan terakhir dari model digunakan untuk memprediksi kategori suara (misalnya, suara ambulans atau suara jalan raya) berdasarkan informasi yang telah dipelajari selama proses pelatihan.

Agar model dapat berfungsi dengan baik, proses pelatihan dilakukan. Pada proses ini, model diberikan data pelatihan dan dioptimalkan agar dapat mengidentifikasi pola secara lebih akurat. Proses pelatihan dilengkapi dengan pengaturan yang memungkinkan model untuk berhenti ketika performa sudah optimal, sehingga tidak terjadi overfitting (model menjadi terlalu spesifik pada data pelatihan).

3.4.4 Deployment

Setelah model *Convolutional Neural Network* (CNN) dilatih dan dievaluasi, langkah berikutnya adalah mendeploy model ke perangkat mikrokontroler ESP32. Proses *deployment* ini melibatkan beberapa langkah penting untuk memastikan bahwa model dapat berfungsi secara efektif dalam lingkungan terbatas yang ada pada perangkat tersebut.

1. Konversi Model

- Model yang telah dilatih perlu diekspor ke dalam format yang kompatibel dengan mikrokontroler. Dalam hal ini, model diubah menjadi format TensorFlow Lite, yang dirancang untuk dapat berjalan di perangkat dengan sumber daya terbatas.

2. Kompresi dan Optimasi

- Agar model dapat dioperasikan dengan efisien pada ESP32, teknik kompresi dan optimasi diterapkan. Salah satu cara yang umum digunakan adalah quantization, yang mengurangi ukuran model tanpa mengorbankan

akurasi secara signifikan. Ini membuat model lebih ringan dan mempercepat waktu inferensi.

3. Integrasi dengan ESP32

- Model yang telah dikompresi kemudian dimasukkan ke dalam kode program yang dijalankan pada ESP32. Kode dirancang untuk menangkap input audio secara *real-time* dari mikrofon yang terhubung ke ESP32, melakukan ekstraksi fitur sesuai dengan model yang telah dilatih, dan akhirnya menerapkan model untuk membuat prediksi.

3.5 Rancangan Pengujian

Pada tahap ini dilakukan pengujian fungsionalitas alat yang digunakan untuk mengetahui keakuratan sensor yang digunakan serta memeriksa kesalahan pada perangkat sistem. Selain pengujian terhadap fungsionalitas alat, dilakukan juga evaluasi dan pengujian terhadap model TinyML untuk melihat performa model.

3.5.1 Pengujian Perangkat Keras

Pengujian perangkat keras pada sistem ini bertujuan untuk melihat kemampuan perangkat keras yang digunakan. Skema pengujian perangkat keras dapat dilihat pada tabel 3.6 berikut ini:

Tabel 3.6 Pengujian Perangkat Keras

No	Komponen Pengujian	Target	Indikator Keberhasilan
1	Mikrofon INMP441	Mikrofon dapat menangkap suara sirine ambulans dengan berbagai volume.	Mikrofon dapat mengirim data audio ke mikrokontroler.
2	Sensor Infra Merah TCRT5000	Sensor IR dapat mendeteksi kendaraan.	Sensor IR dapat mengirimkan data secara kontinu ke mikrokontroler.
3	Modul LED Lampu Lalu Lintas	LED merah, kuning dan hijau dapat menyala.	Warna LED menyala sesuai dengan kondisi simulasi.
4	<i>Push-Button</i>	Memberikan input ke mikrokontroler.	<i>Button</i> dapat mengubah status warna lampu lalu lintas ketika ditekan.

5	<i>Buzzer</i>	Mengeluarkan suara	<i>Buzzer</i> dapat mengeluarkan suara ketika <i>button</i> ditekan.
6	Uji Responsivitas	Uji waktu respons perangkat saat menerima input dan memberikan output.	Rentang waktu saat menerima input dan memberikan output tidak lebih dari 5 detik.

3.5.2 Pengujian Model TinyML

Pengujian model TinyML pada sistem ini bertujuan untuk melihat kemampuan model yang telah diintegrasikan ke mikrokontroler. Skema pengujian model TinyML dapat dilihat pada tabel 3.7 berikut ini:

Tabel 3.7 Pengujian Model TinyML

No	Komponen Pengujian	Target	Indikator Keberhasilan
1	Pengujian <i>Real-time</i>	Uji model dalam kondisi nyata dengan memberikan suara sirine ambulans secara langsung.	Model dapat mendeteksi suara sirine ambulans dalam berbagai kondisi suara latar belakang.
2	Pengujian <i>Robustness</i>	Uji model dengan variasi suara sirine (misalnya, jarak, volume, dan frekuensi yang berbeda).	Model dapat mendeteksi suara sirine ambulans dalam berbagai variasi suara sirine.
3	Uji Latensi	Uji model dengan mengukur waktu yang diperlukan untuk inferensi dari input suara hingga prediksi selesai.	Rentang waktu saat menerima input suara hingga prediksi tidak lebih dari 5 detik.

3.5.3 Evaluasi Model TinyML

Pengujian model TinyML akan menggunakan *confusion matrix*. Dengan menggunakan *confusion matrix* akan didapatkan hasil klasifikasi terhadap kelas sebenarnya, serta metrik seperti tingkat akurasi, presisi, recall, dan nilai F1, yang semuanya krusial untuk menilai sejauh mana model dapat membedakan suara sirine ambulans.

DAFTAR PUSTAKA

- [1] GAIKINDO, “Data BPS: Otomotif Sektor yang Mendatangkan Laba,” GAIKINDO, 2023. [Online]. Available: <https://www.gaikindo.or.id/data-bps-otomotif-sektor-yang-mendatangkan-laba/>. [Accessed 5 March 2023].
- [2] S. “Kemacetan Kendaraan Pengguna BBM Fosil dan Dampaknya Terhadap Kerugian Ekonomi dan Lingkungan,” *Rang Teknik Journal*, vol. 4, no. 1, pp. 143-163, 2021.
- [3] A. Machdani, B. Husodo and S. Attamimi, “Sistem Kontrol Lampu Lalu Lintas Untuk Layanan Darurat Berbasis Internet Of Things (IoT),” *Jurnal Teknologi Elektro, Universitas Mercu Buana*, vol. 10, no. 3, pp. 188-193, 2019.
- [4] M. Munir, M. I. Mahali, S. A. Dewanto, B. Wulandari and N. Hasanah, “Pengembangan Smart Traffic Light berbasis IoT (Internet Of Things) dengan Mobile Backend as a Service (MbaaS) sebagai wujud Smart City bidang transportasi,” Jurusan Pendidikan Teknik Elektronika dan Informatika UNY, Yogyakarta.
- [5] The Nation, “The Nation Thailand,” 16 January 2017. [Online]. Available: <https://www.nationthailand.com/in-focus/30304268>. [Accessed 8 November 2023].
- [6] Dewan Perwakilan Rakyat Republik Indonesia, “UU 22 Tahun 2009,” 2009. [Online]. Available: https://www.dpr.go.id/dokjdih/document/uu/UU_2009_22.pdf. [Accessed 12 December 2023].
- [7] N. Royan, “ANALISA PERENCANAAN TRAFFIC LIGHT DI PERSIMPANGAN BANDARA SMB II PALEMBANG,” *Berkala Teknik*, vol. 5, no. 12, pp. 837-855, 2015.
- [8] D. Setiawan, “Lebih Mengenal Embedded System Dan Contohnya,” Universitas STEKOM, 4 February 2022. [Online]. Available: <https://teknik-komputer-d3.stekom.ac.id/informasi/baca/Lebih-Mengenal-Embedded-System-dan-Contohnya/1e59224df65f667b8134085643c4d9c6e5a2e557>. [Accessed 21 May 2024].

- [9] V. Tsoukas, A. Gkogkidis, E. Boumpa, S. Papafotikas and A. Kakarountas, "A Gas Leakage Detection Device Based on the Technology of TinyML," *Technologies*, vol. 11, no. 45, pp. 1-16, 2023.
- [10] P. N. Andono, T. Sutojo and M. , *Pengolahan Citra Digital*, Semarang: Penerbit Andi, 2017.
- [11] Usman, P. Albert, I. V. Sari, I. Idris and R. Khair, "Rancang Bangun Traffic Light System Tanggap Darurat Berbasis IoT," *Jurnal Sistem Komputer dan Informatika (JSON)*, vol. 1, no. 3, pp. 195-199, 2020.
- [12] R. Nono, R. Alsudais, R. Alshmrani, S. Alamoudi and A. Aljahdali, "Intelligent Traffic Light for Emergency Vehicles Clearance," *ADCAIJ: Advances in Distributed Computing and Artificial Intelligence Journal*, vol. 9, no. 3, pp. 89-104, 2020.
- [13] S. Damadam, M. Zourbakhsh, R. Javidan and A. Faroughi, "An Intelligent IoT Based Traffic Light Management System: Deep Reinforcement Learning," *Smart Cities*, vol. 5, no. 4, pp. 1293-1311, 2022.
- [14] A. N. A. Yusuf, A. S. Arifin and F. Y. Zulkifli, "Recent development of smart traffic lights," *IAES International Journal of Artificial Intelligence*, vol. 10, no. 1, pp. 224-233, 2021.
- [15] D. M. A. Hamid, N. A. Bakar, M. Nurtanto and S. , "A smart traffic light using a microcontroller base on the fuzzy logic," *IAES International Journal of Artificial Intelligence*, vol. 11, no. 3, pp. 809-818, 2022.
- [16] V. R. Adittama, T. Firdaus and M. Yudi, "Lampu Lalu Lintas," Program Studi Teknik Komputer, Jurusan Teknologi Informasi, Politeknik Negeri Samarinda, Samarinda, 2021.
- [17] K. P. F. Apriyani, M. A. Asmara and R. Harini, "EDUKASI RAMBU-RAMBU LALU LINTAS TERHADAP ANAK-ANAK SDN 38 KOTA BENGKULU," *Jurnal Ilmiah Mahasiswa Kuliah Kerja Nyata*, vol. 2, no. 1, pp. 16-24, 2022.
- [18] DUNN ENGINEERING ASOCIATES, *TRAFFIC CONTROL SYSTEMS HANDBOOK*, New York: Office of Transportation Management Federal Highway Administration, 2005.

- [19] [ems.gov](https://www.ems.gov/what-is-ems/), “What Is EMS?,” NHTSA Office, 16 May 2023. [Online]. Available: <https://www.ems.gov/what-is-ems/>. [Accessed 21 December 2023].
- [20] Maine EMS, “What is EMS?,” Maine.gov, [Online]. Available: <https://www.maine.gov/ems/whatisems>. [Accessed 21 December 2023].
- [21] S. M. Sasser, R. C. Hunt, M. Faul and D. Sugerman, “Guidelines for Field Triage of Injured Patients,” *Recommendations and Reports*, vol. 61, no. 1, pp. 1-23, 2012.
- [22] A. Tanjung, Director, *Mengenal Jenis Suara Sirine ambulans, Ini Artinya / Explanatory*. [Film]. Yogyakarta: Brilio News, 2020.
- [23] CNN Indonesia, “Mengapa Suara Sirene Ambulans Terbagi 4 Jenis?,” 14 January 2023. [Online]. Available: <https://www.cnnindonesia.com/otomotif/20230113165624-584-900175/mengapa-suara-sirene-ambulans-terbagi-4-jenis#:~:text=Sementara%20itu%20mengutip%20situs%20Ambulance,ambulans%20ada%20di%20jalan%20lurus..> [Accessed 15 December 2023].
- [24] ESP32, “the internet of things with esp32,” Espressif Systems, [Online]. Available: <http://esp32.net/>. [Accessed 31 January 2024].
- [25] InvenSense Inc., “INMP441,” InvenSense Inc., 2014.
- [26] R. Aditya, “Perancangan Alat Monitoring Water Meter di PDAM Tirta Dharma Salatiga berbasis Arduino,” Fakultas Teknik Elektronika dan Komputer Universitas Kristen Satya Wacana, Salatiga, 2019.
- [27] EL-PRO-CUS, “TCRT5000 IR Sensor : Pin Configuration & Its Applications,” [Online]. Available: <https://www.elprocus.com/tcrt5000-ir-sensor/>. [Accessed 21 December 2023].
- [28] Retro Aksesoris, “Traffic Light Module Arduino,” September 2020. [Online]. Available: <https://www.aksesoriskomputerlampung.com/2020/09/traffic-light-module-arduino.html>. [Accessed 21 December 2023].
- [29] M. Roghib, “Program Tombol/Button,” Menara Ilmu Mikrokontroler Universitas Gadjah Mada, 2 October 2018. [Online]. Available: <https://mikrokontroler.mipa.ugm.ac.id/2018/10/02/program-tombol-button/>.

[Accessed 21 March 2024].

- [30] codebender_cc, “How to Use a Push Button - Arduino Tutorial,” Autodesk Instructables, 2015. [Online]. Available: <https://www.instructables.com/How-to-use-a-Push-Button-Arduino-Tutorial/>. [Accessed 21 March 2024].
- [31] M. Roghib, “Program Buzzer,” Menara Ilmu Mikrokontroler Universitas Gadjah Mada, 2 October 2018. [Online]. Available: <https://mikrokontroler.mipa.ugm.ac.id/2018/10/02/program-buzzer/>. [Accessed 21 March 2024].
- [32] Arun, “Medium,” Towards Data Science, 10 November 2020. [Online]. Available: <https://towardsdatascience.com/an-introduction-to-tinyml-4617f314aa79>. [Accessed 15 December 2023].
- [33] K. Pykes, “What is TinyML? An Introduction to Tiny Machine Learning,” datacamp, February 2023. [Online]. Available: <https://www.datacamp.com/blog/what-is-tinyml-tiny-machine-learning>. [Accessed 21 December 2023].
- [34] R. Awati, “convolutional neural network (CNN),” TechTarget, January 2024. [Online]. Available: <https://www.techtarget.com/searchenterpriseai/definition/convolutional-neural-network>. [Accessed 21 March 2024].
- [35] “Introduction to Convolution Neural Network,” GeeksforGeeks, 14 March 2024. [Online]. Available: <https://www.geeksforgeeks.org/introduction-convolution-neural-network/>. [Accessed 22 May 2024].
- [36] A. Bhandari, “Understanding & Interpreting Confusion Matrix in Machine Learning (Updated 2023),” Analytics Vidhya, 2 November 2023. [Online]. Available: https://www.analyticsvidhya.com/blog/2020/04/confusion-matrix-machine-learning/#What_Is_a_Confusion_Matrix?. [Accessed 18 December 2023].
- [37] G. Delnevo, . S. Mirri , C. Prandi and P. Manzoni, “An evaluation methodology to determine the actual limitations of a TinyML-based solution,” *Internet of Things*, vol. 22, pp. 1-17, 2023.
- [38] T. Nasution, “Metoda Mel Frequency Cepstrum Coefficients (MFCC) untuk

Mengenali Ucapan pada Bahasa Indonesia,” *Jurnal Sains dan Teknologi Informasi*, vol. 1, no. 1, pp. 22-31, 2012.

- [39] Global Traffic Technologies, “Solutions: Emergency Vehicle Preemption IR,” Opticom, January 2009. [Online]. Available: <https://www.gtt.com/wp-content/uploads/Model-700-Series.pdf>. [Accessed 5 March 2023].