

Kendali Lampu Lalu Lintas Dengan Menggunakan Metode *Fuzzy Logic Mamdani*

Paula Juniana¹, Lukman Hakim²

¹*Teknik Informatika, Universitas Bunda Mulia
JL.Lodan Raya No. 2 Jakarta Utara, DKI Jakarta
paulajuniana@gmail.com*

²*Teknik Informatika, Universitas Bunda Mulia
JL.Lodan Raya No. 2 Jakarta Utara, DKI Jakarta
lhakim@bundamulia.ac.id*

Abstract— The cause of the delay in the distribution of goods, reduced productivity of employees and economic wheels that are not good enough, one of which is congestion. The congestion itself is due to the arrangement of traffic lights that are not dynamic so that the lane settings do not match the existing conditions. Jakarta is one of the largest and second largest cities in Asia, it needs serious handling such as good traffic regulation so as to reduce congestion and other economic losses. The traffic system by applying artificial intelligence capabilities is more likely to be used in big cities like Jakarta. Based on these problems, a study on the three-lane traffic system or one third by adding infrared sensors at the traffic lights and increasing the time each detected a traffic jam and a smooth lane. Traffic light control system uses the Mamdani Fuzzy Logic method. The Mamdani method compares based on the conditions of each infrared sensor as a variable input of heat from a vehicle on each pathway, the denser the path the more hot the path is. Detection of this jam will help the system to control the time of the green light, the lights are based on congestion conditions. Based on the experiment, the prototype responded to the vehicle-filled path, the prototype added the green light time to the conditions of 0 seconds, 5 seconds, 10 seconds and 15 seconds. However, when the road is not detected by traffic jams, the green light will return to normal at 15 seconds without extra time.

Keywords— Mamdani Fuzzy Logic, Green Light Time, Prototype, Traffic

Abstrak –Penyebab terlambatnya distribusi barang, berkurangnya produktivitas karyawan serta roda ekonomi yang tidak cukup baik salah satunya adalah kemacetan. Kemacetan sendiri akibat pengaturan lampu lalu lintas yang tidak dinamis sehingga pengaturan jalur tidak sesuai kondisi yang ada. Jakarta adalah salah satu kota terbesar dan termacet ke-2 di Asia, perlu penanganan serius seperti pengaturan lalu lintas yang baik sehingga mengurangi kemacetan dan kerugian ekonomi lainnya. Sistem lalu lintas dengan menerapkan kemampuan kecerdasan buatan lebih memungkinkan digunakan pada kota besar seperti Jakarta. Berdasarkan masalah tersebut, dilakukan penelitian pada sistem lalu lintas tiga jalur atau sepertiga dengan penambahan sensor inframerah pada lampu lalu lintas dan penambahan waktu setiap terdeteksi kemacetan dan jalur lancar. Sistem kontrol lampu lalu lintas menggunakan metode Fuzzy Logic Mamdani. Metode Mamdani membandingkan berdasarkan kondisi dari setiap sensor infrared sebagai variabel masukan panas dari kendaraan pada setiap jalur, semakin padat jalur semakin panas jalur tersebut. Deteksi

kemacetan ini akan membantu sistem dalam mengendalikan waktu lampu hijau, lampu berdasarkan kondisi kemacetan. Berdasarkan percobaan, prototipe merespon jalur yang dipenuhi kendaraan, prototipe menambahkan waktu lampu hijau sesuai dengan kondisi yaitu 0 detik, 5 detik, 10 detik, dan 15 detik. Namun, ketika jalanan tidak terdeteksi oleh kemacetan lalu lintas, lampu hijau akan kembali normal pada 15 detik tanpa waktu tambahan.

Kata Kunci— Logika Fuzzy Mamdani, Waktu Lampu Hijau, Prototipe, Lalu Lintas

I. PENDAHULUAN

Jakarta merupakan kota termacet dengan peringkat 22 dan minimnya penggunaan sistem transportasi cerdas seperti optimasi sistem lalu lintas atau jalur yang dinamis [1]. Kemacetan telah menimbulkan banyak hal yang akan terjadi, seperti hilangnya produktivitas, hilangnya pendapatan dari masyarakat yang bekerja, pemborosan penggunaan BBM, meningkatnya tingkat stress [2].

Kemacetan juga disebabkan karena sistem kendali lampu lalu lintas yang dibuat dengan waktu yang tetap sehingga tidak dapat mendeteksi terjadi kepadatan di setiap jalur tertentu. Lampu lalu lintas di Jakarta, sering terjadi kerusakan yang membuat kepadatan serta arus jalan nya kendaraan tidak dapat dikendalikan.

Kemacetan lalu lintas terus meningkat dari tahun ke tahun. Jakarta menduduki peringkat ke-2 di Asia dengan waktu yang dihabiskan oleh pengendara selama 55 jam di jalan saat kemacetan [1]. Kemacetan lalu lintas di Jakarta terus meningkat disebabkan karena meningkatnya jumlah kendaraan. Pada tahun 2013, jumlah kendaraan di Jakarta ada 9.257.801 kendaraan, di tahun 2014 jumlah kendaraan ada 9.831.142 kendaraan, di tahun 2015 ada 10.413.974 kendaraan, di tahun 2016 ada 10.905.533 kendaraan, dan di awal tahun 2017 tercatat jumlah kendaraan ada 10.940.102 kendaraan.

Sistem otomatis *traffic light* sudah dipasang dimana-mana terutama di negara-negara maju [3]. Sistem yang saat ini digunakan di Indonesia kurang efektif untuk menanggulangi setiap titik kemacetan pada setiap kota besar serta pengaturan sistem lampu lalu lintas tidak dinamis yang secara otomatisasi mengatur kondisi setiap jalur kemacetan yang ada. Berdasarkan hal tersebut perlu adanya

pengembangan sistem pengaturan otomatis lampu lalu lintas dengan teknologi kecerdasan buatan untuk mengatasi kemacetan pada setiap jalur kendaraan, untuk itu penulis membuat *prototype Traffic Light* dengan menggunakan algoritma Mandani.

II. LANDASAN TEORI

A. Traffic Light

Traffic light adalah sebuah alat pengendali arus lalu lintas menggunakan lampu yang dipasang pada persimpangan jalan. Tujuannya untuk mengatur arus lalu lintas pada setiap persimpangan agar dapat berjalan secara teratur dan tidak saling bertabrakan. Lampu lalu lintas atau *traffic light* dikenal pertama kali pada tahun 1868. Sistem menggunakan gas yang dipasang di *Westminster* Inggris. Kemudian pada tahun 1918 di *New York*, dengan format merah, kuning, hijau yang dioperasikan secara manual. Pada tahun 1926 dilakukan operasi lampu secara semi otomatis di *Wolverhampton* Inggris. Secara garis besar *traffic light* digunakan sebagai pengatur arus lalu lintas, mencegah kemacetan di persimpangan, memberikan kesempatan kepada kendaraan lain/ pejalan kaki dan meminimalisasi konflik kendaraan [4].

B. Sistem Kontrol dan Penambahan Waktu

Traffic light diatur oleh suatu sistem kontrol yang dipasang didalam suatu panel yang diletakkan tidak jauh dari *traffic light*. Beberapa tempat (seperti *New York*) masih menggunakan elektro mekanik kontroler. Panel biasanya berisi suatu panel listrik, untuk mendistribusikan listrik, detektor antar muka panel, *relay* cepat, dan komponen lain. Jenis-jenis pengontrolan pada *traffic light* adalah [4].

1. Kontrol waktu tetap :

sistem kontrol ini tidak berubah-ubah. Waktu sirkulasi setiap jalur mendapatkan lamanya lampu merah menyala adalah tetap.

2. Kontrol dinamis :

Sistem kontrol ini akan memprioritaskan jalur yang padat, dan bahkan jika tidak ada kendaraan yang mengantri, maka jalur tersebut tidak mendapatkan lampu hijau sama sekali.

3. Kontrol terkoordinasi :

Sistem kontrol ini mengkoordinasikan antara *traffic light* yang berdekatan, sehingga tidak terjadi antrian yang panjang diantara kedua *traffic light* tersebut. Misalnya ketika terdeteksi kepadatan akan menambah waktu pada lampu hijau yaitu 5, 10 detik, dan 15 detik.

4. Kontrol interupsi :

Beberapa tempat yang padat arus kendaraan dan tidak ada *zebra cross* (tempat menyebarang jalan melalui tangga) pejalan kaki boleh meminta untuk menyebarang. bernilai $0^\circ, 45^\circ, 90^\circ, \text{ dan } 135^\circ$.

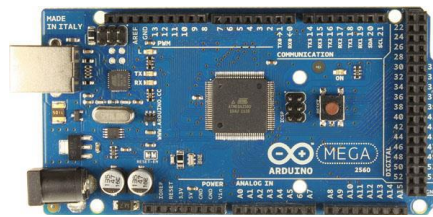
C. Arduino

Arduino adalah kit elektronik atau papan rangkaian elektronik *open source* yang di dalamnya terdapat komponen utama yaitu sebuah *chip* dengan jenis AVR dari perusahaan Atmel. *Arduino* dikatakan sebagai sebuah *platform* dari *physical computing* yang bersifat *open source*.

Pertama-tama perlu dipahami bahwa kata “*platform*” disini adalah sebuah pilihan kata yang tepat. *Arduino* tidak hanya sekedar sebuah alat pengembangan, tetapi kombinasi dari *hardware*, bahasa pemrograman dan *Integrated Development Environment* (IDE) yang canggih. IDE adalah sebuah *software* yang sangat berperan untuk menulis program, meng-*compile* menjadi kode *biner* dan meng-*upload* ke dalam *memory* mikrokontroler [5].

1) Arduino Mega 2560

Pada percobaan prototipe menggunakan *Arduino Mega 2560*, mikrokontroler yang sesuai untuk kebutuhan lampu lalu lintas 3 jalur, untuk mikrokontroler *arduino Mega 2560* dapat dilihat pada gambar 1.



Gambar. 1. *Arduino Mega 2560*

Arduino merupakan sebuah board mikrokontroler yang didasarkan pada ATmega 2560. *Arduino Mega 2560* adalah suatu mikrokontroler pada ATmega 2560 yang mempunyai 54 input/ output digital yang mana 16 pin digunakan sebagai keluaran, 16 masukan analog, dan di dalamnya terdapat 16 MHZ *osilator* kristal, USB koneksi, *power*, ICSP, dan tombol *reset*. Kinerja *arduino* ini memerlukan dukungan mikrokontroler dengan menghubungkan *arduino* pada suatu komputer dengan USB kabel untuk menghidupkannya menggunakan arus AC atau DC dan bisa juga dengan menggunakan baterai [7].

D. Fuzzy Logic

Logika *fuzzy* merupakan salah satu komponen pembentuk *soft computing*. Dasar logika *fuzzy* adalah teori himpunan *fuzzy*. Logika *fuzzy* merupakan suatu cara yang tepat untuk memetakan suatu ruang input ke dalam suatu ruang output [8].

Himpunan *fuzzy* merupakan suatu pengembangan lebih lanjut tentang konsep himpunan dalam matematika. Pada himpunan tegas (*crisp*), nilai keanggotaan dalam suatu himpunan yang memiliki dua kemungkinan, yaitu :

- a) Satu (1) yang berarti bahwa suatu item menjadi anggota dalam suatu himpunan, atau
- b) Nol (0) yang berarti bahwa suatu item tidak menjadi anggota dalam suatu himpunan.

E. Metode Mamdani

Metode *Mamdani* dikenal juga sebagai metode *min-max*, diperkenalkan oleh Ebrahim Mamdani pada tahun 1975 [8]. Tahapan untuk memperoleh *output* adalah sebagai berikut [8]:

1. Pembentukan Himpunan Fuzzy :

Pembentukan anggota himpunan dan variabel *fuzzy* ini didasarkan pada standar perusahaan.

2. *Penentuan Fungsi Keanggotaan :*

Fungsi keanggotaan pada penelitian ini ditunjukkan dengan pemetaan titik – titik *input* himpunan tegas yang digambarkan pada bentuk kurva segitiga, kurva trapesium, dan kurva bentuk bahu.

3. *Penentuan Aturan Fungsi Fuzzy :*

Metode yang digunakan dalam komposisi aturan dan aplikasi fungsi implikasi adalah metode *max – min* dengan operator *AND*. Secara umum aturan tersebut dapat dituliskan: *IF* (x1 is A1) • (x2 is A2) • ... (xn is An) *THEN* y is B dengan • adalah operator *AND*, xn adalah skalar yang berupa variabel *fuzzy* dan An adalah variabel linguistik berupa himpunan *fuzzy*.

4. *Penegasan :*

Output himpunan *fuzzy* yang diperoleh dari komposisi aturan *fuzzy* akan mengalami proses defuzifikasi. Terdapat 5 metode yang digunakan dalam *defuzifikasi*.

Metode terbaik dipilih berdasarkan nilai Mean Square Error (MSE) terkecil. Metode yang digunakan dalam *defuzifikasi* antara lain:

1. *Centroid*
2. *Bisector*
3. *Mean of Maximum (MOM)*
4. *Largest of Maximum (LOM)*
5. *Smallest of Maximum (SOM)*

F. *Infrared Sensor*

Penggunaan sensor infrared sangat membantu untuk deteksi keberadaan setiap kendaraan, serta mengetahui banyaknya kendaraan setiap jalur. Untuk sensor yang digunakan dapat dilihat pada gambar 2.



Gambar. 2. *Infrared Sensor*

Infrared Sensor adalah rangkaian yang memberi keluaran rendah tanpa adanya sinyal *Infrared*. Sensor ini hanya dapat mendeteksi apakah ada benda yang menghalangi atau tidak ada, sensor tidak dapat mengetahui jarak ke benda tersebut. *Transmitter* dan *receiver* yang ada di dalam sensor tersebut menghadap ke arah yang sama, dimana *receiver* akan menerima pantulan sinar *infrared* dari *transmitter* [10].

G. *Model Prototype*

Model *Prototype* merupakan suatu teknik untuk mengumpulkan informasi tertentu mengenai kebutuhan-kebutuhan informasi pengguna secara cepat. Berfokus pada penyajian dari aspek-aspek perangkat lunak tersebut yang akan nampak bagi pelanggan atau pemakai. *Prototype* tersebut akan dievaluasi oleh pelanggan/pemakai dan dipakai untuk menyaring kebutuhan pengembangan perangkat lunak [10].

Langkah – langkah dalam metode *Prototype* meliputi:

1. Mengidentifikasi kebutuhan pemakai. Pada tahap ini, analisis sistem akan melakukan studi kelayakan dan studi

terhadap kebutuhan pemakai, baik yang meliputi model interface, teknik prosedural maupun dalam teknologi yang akan digunakan

2. Pengembangan *prototype*. Pada tahap kedua ini, analisis sistem bekerja sama dengan pemrogram mengembangkan *prototype* sistem untuk memperlihatkan kepada pemesan pemodelan sistem yang akan dibangunnya.
3. Menentukan *prototype*, apakah dapat diterima oleh pemesan atau pemakai. Analisis sistem pada tahap ini akan mendeteksi dan mengidentifikasi sejauh mana pemodelan yang dibuatnya dapat diterima oleh pengguna.
4. Penggunaan *prototype*. Pada tahap ini, peneliti akan melakukan pemrogram untuk mengimplementasikan pemodelan yang dibuatnya menjadi satu sistem.

III. METODOLOGI PENELITIAN

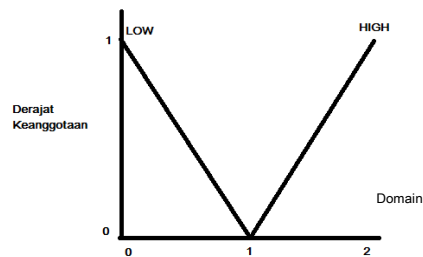
Kebutuhan fungsional dari penelitian ini adalah sebagai berikut:

1. *Prototype* ini menggunakan 2 sensor *Infrared* untuk mendeteksi kepadatan kendaraan.
2. *Prototype* ini akan menambah waktu pada lampu hijau ketika terdeteksi kepadatan kendaraan.
3. *Prototype* ini menggunakan metode *Fuzzy Logic Mamdani* yang menghasilkan *output* yaitu lampu warna hijau dan waktu nyala hijau.

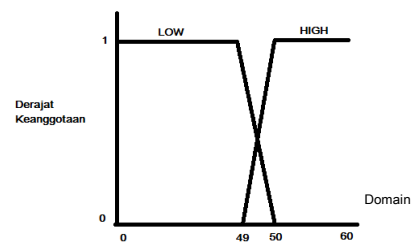
Pada artikel ini, digunakan metode *Fuzzy Logic Mamdani* untuk pengaturan lalu lintas berdasarkan kemacetan yang terjadi:

Fuzzy Logic Mamdani

Pada artikel ini, digunakan metode *Fuzzy Logic Mamdani* untuk mengatur sistem kendali lampu lalu lintas, untuk nilai variabel berdasarkan masukan infrared dan waktu tunggu dapat dilihat pada gambar 3 dan 4. Berikut langkah-langkah pengerjaannya sebagai berikut:



Gambar. 3. Set Himpunan *Infrared 1 & Infrared 2*



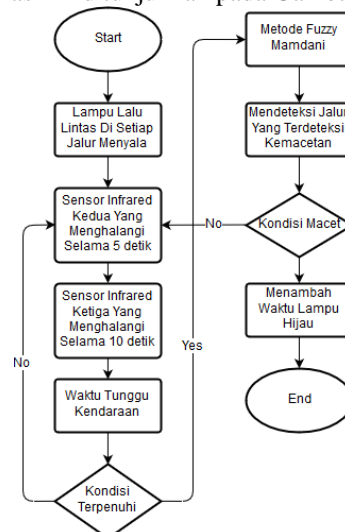
Gambar. 4. Set Himpunan Waktu Tunggu

1. Menentukan himpunan universal pada *variable input*.
 Pada tahap awal, penulis menggunakan 2 variabel yang akan dipakai dalam penerapan pada *prototype*.
 1. *Variabel waktu tunggu* : waktu tunggu adalah waktu yang digunakan pada jalur lalu lintas untuk melihat berapa waktu lampu merah yang telah berjalan.
 2. *Variabel infrared1* : (jika ada kendaraan yang berhenti di *infrared1* selama 5 detik maka *infrared 1* akan menjadi *input/masukan* dalam pendeteksian *prototype*).
 3. *Variabel infrared2* : (jika ada kendaraan yang berhenti di *infrared2* selama 10 detik maka *infrared2* akan menjadi *input/masukan* dalam pendeteksian *prototype*).
2. Menentukan himpunan keanggotaan.
 - a) Waktu Tunggu : terdiri atas 2 himpunan *fuzzy*, yaitu: *LOW* dan *HIGH*.
 - b) *Infrared1* :terdiri atas 2 himpunan *fuzzy*, yaitu: 1 (*ADA/HIGH*) dan 0 (*TIDAK_ADA/LOW*).
 - c) *Infrared2* :terdiri atas 2 himpunan *fuzzy*, yaitu: 1 (*ADA/HIGH*) dan 0 (*TIDAK_ADA/LOW*).
3. Membentuk aturan *fuzzy*.
 - a. *Rule 1*
 If waktu_tunggu *LOW* dan *infrared1* *TIDAK_ADA* dan *infrared2* *TIDAK_ADA* then waktu_hijau *TIDAK_TAMBAH*.
 - b. *Rule 2*
 If waktu_tunggu *LOW* dan *infrared1* *TIDAK_ADA* dan *infrared2* *ADA* then waktu_hijau *TAMBAH_5_DETIK*.
 - c. *Rule 3*
 If waktu_tunggu *LOW* dan *infrared1* *ADA* dan *infrared2* *TIDAK_ADA* then waktu_hijau *TAMBAH_5_DETIK*.
 - d. *Rule 4*
 If waktu_tunggu *LOW* dan *infrared1* *ADA* dan *infrared2* *ADA* then waktu_hijau *TAMBAH_10_DETIK*.
 - e. *Rule 5*
 If waktu_tunggu *HIGH* dan *infrared1* *TIDAK_ADA* dan *infrared2* *TIDAK_ADA* then waktu_hijau *TAMBAH_5_DETIK*.
 - f. *Rule 6*
 If waktu_tunggu *HIGH* dan *infrared1* *TIDAK_ADA* dan *infrared2* *ADA* then waktu_hijau *TAMBAH_10_DETIK*.
 - g. *Rule 7*
 If waktu_tunggu *HIGH* dan *infrared1* *ADA* dan *infrared2* *TIDAK_ADA* then waktu_hijau *TAMBAH_10_DETIK*.
 - h. *Rule 8*
 If waktu_tunggu *HIGH* dan *infrared1* *ADA* dan *infrared2* *ADA* then waktu_hijau *TAMBAH_15_DETIK*.
4. Melakukan *defuzzifikasi (Mean Of Maximum)*.
 Dalam proses *defuzzifikasi* dilakukan proses perbandingan antar nilai variabel yang ada menjadi *output* hasil keluaran dari *prototype* sebagai berikut:

- a. *Output*: Jalur yang terdeteksi ($WT=LOW \& Inf1=0 \& Inf2=0$) maka LED berwarna hijau tidak terjadi penambahan waktu. $Def=(0+0+0)/3=0$
- b. *Output*: Jalur yang terdeteksi ($WT=LOW \& Inf1=0 \& Inf2=1$) maka LED berwarna hijau terjadi penambahan waktu 5 detik. $Def=(0+0+1)/3=0.33$
- c. *Output*: Jalur yang terdeteksi ($WT=LOW \& Inf1=1 \& Inf2=0$) maka LED berwarna hijau terjadi penambahan waktu 5 detik. $Def=(0+1+0)/3=0.33$
- d. *Output*: Jalur yang terdeteksi ($WT=LOW \& Inf1=1 \& Inf2=1$) maka LED berwarna hijau terjadi penambahan waktu 10 detik. $Def=(0+1+1)/3=0.67$
- e. *Output*: Jalur yang terdeteksi ($WT=HIGH \& Inf1=0 \& Inf2=0$) maka LED berwarna hijau terjadi penambahan waktu 5 detik. $Def=(1+0+0)/3=0.33$
- f. *Output*: Jalur yang terdeteksi ($WT=HIGH \& Inf1=0 \& Inf2=1$) maka LED berwarna hijau terjadi penambahan waktu 10 detik. $Def=(1+0+1)/3=0.67$
- g. *Output*: Jalur yang terdeteksi ($WT=HIGH \& Inf1=1 \& Inf2=0$) maka LED berwarna hijau terjadi penambahan waktu 10 detik. $Def=(1+1+0)/3=0.67$
- h. *Output*: Jalur yang terdeteksi ($WT=HIGH \& Inf1=1 \& Inf2=1$) maka LED berwarna hijau terjadi penambahan waktu 15 detik. $Def=(1+1+1)/3=1$

5. Melakukan pengujian model *fuzzy*.

Perancangan *flowchart* bertujuan untuk menggambarkan aliran proses dalam sistem. *Flowchart* dari aplikasi ini ditunjukkan pada Gambar 5.



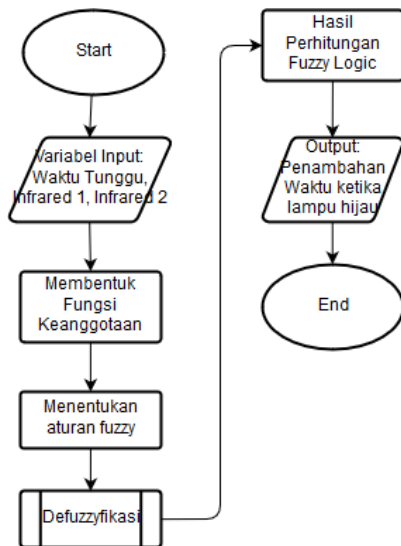
Gambar. 5. *Flowchart* Aplikasi

Penjelasan tahapan pengujian sebagai berikut :

1. Pada setiap lampu lalu lintas 3 jalur, sebagai indikator kondisi jalur lalu merah, hijau, kuning.
2. Sensor *infrared* sebagai variabel masukan yang membaca banyaknya kendaraan yang berada setiap jalur, apabila jalur dipenuhi kendaraan maka *infrared* memberikan indikator *high* atau nyala.
3. Sensor *infrared* yang memberikan nilai maka algoritma *fuzzy* akan mengatur nilai setiap jalur

kendaraan berdasarkan kepadatan setiap jalur, maka itu berulang terus berdasarkan kondisi dari sensor *infrared*.

Flowchart dari metode Mamdani ditunjukkan pada Gambar 6.

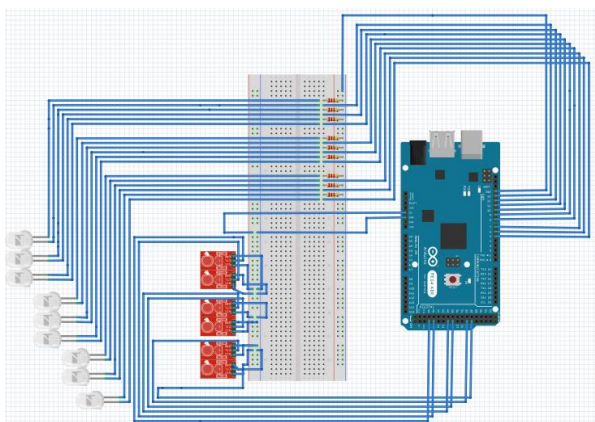


Gambar. 6. Flowchart Metode Mamdani

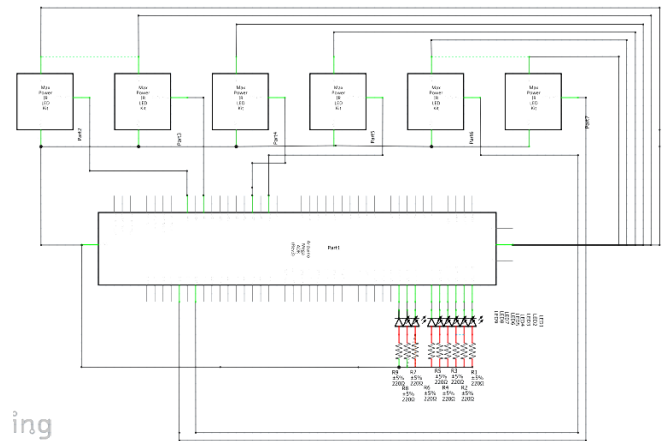
Penjelasan tahapan pada metode mandani sebagai berikut :

1. Pada metode mandani harus memiliki variabel masukan diantaranya waktu tunggu, infrared.
2. Berdasarkan nilai yang sudah ditentukan pada setiap anggota pada variabel sebagai penentu kondisi.
3. Berdasarkan nilai yang sudah ditentukan *fuzzy* melakukan seleksi dan *defuzzifikasi* pada nilai yang didapatkan berdasarkan masukan variabel seperti sensor infrared.
4. Berdasarkan perhitungan *fuzzy* mandani maka nilai setiap kondisi lampu lalu lintas dapat ditentukan.

Schematic Diagram prototype kendali lalu lintas ini menggambarkan apa saja komponen yang digunakan didalam *arduino* dapat dilihat pada gambar 7.



Gambar. 7. Breadboard Prototype



Gambar. 8. Schematic Diagram Prototype

Dilihat pada gambar 8 adalah rangkaian keseluruhan *prototype* yang berguna sebagai kendali lampu lalu lintas untuk mengatasi kemacetan. Setiap komponen yang membutuhkan energi listrik harus mempunyai *pin Vcc* dan *GND*, karena dua *pin* ini berfungsi mengalirkan arus ke komponen yang bersangkutan. *Anoda* dan *katoda* juga memiliki arti listrik yang berlawanan arus dilambangkan dengan (+) dan (-), *Anoda* adalah nama lain dari *VCC* yang artinya harus mendapat kan arus positif dan *Katoda* adalah nama lain dari listrik negatif yang memiliki arti *Ground (gnd)*. *Data* yang disalurkan semuanya terhubung di *mainboard* untuk menjalankan instruksi yang dibuat, sehingga masing-masing komponen bisa saling bekerja dan menjalankan fungsi masing-masing dari setiap perintah. Berikut ini adalah penjelasan secara menyeluruh dari komponen yang digunakan untuk merancang sebuah *prototype*.

1. Light Emitting Diode (LED)

Lampu LED digunakan untuk menandakan kondisi kendaraan yang harus berhenti, berjalan, atau bersiap sebelum waktu jalannya. Lampu LED yang digunakan ada 3 warna yaitu merah, kuning, dan hijau. Lampu lalu lintas akan digunakan pada 3 jalur pada *prototype*.

a. Lampu Merah (Jalur 1,2,3)

Lampu merah pada jalur 1,2,3 memiliki 2 pin yang bisa dilihat pada tabel I.

TABEL I
LED MERAH

PIN	Sambungan
GND	GND (Extender GND and Vcc)
D0	13, 10, 7 (Arduino Mega)

b. Lampu Kuning (Jalur 1,2,3)

Lampu Kuning pada jalur 1,2,3 memiliki 2 pin yang bisa dilihat pada tabel II.

TABEL II
LED KUNING

PIN	Sambungan
GND	GND (Extender GND and Vcc)
D0	12, 9, 6 (Arduino Mega)

c. Lampu Hijau (Jalur 1,2,3)

Lampu Hijau pada jalur 1,2,3 memiliki 2 pin yang bisa dilihat pada tabel III.

TABEL III
LED MERAH

PIN	Sambungan
GND	GND (Extender GND and Vcc)
D0	11, 8, 5 (Arduino Mega)

2. Sensor Infrared

Sensor *Infrared* adalah sensor yang mampu mendeteksi kendaraan yang berhenti tepat disensor selama jangka waktu yang ditentukan. Di setiap jalur, terdapat 2 sensor infrared. Sensor *Infrared* memiliki 3 pin yang bisa dilihat pada tabel IV dan tabel V.

TABEL IV
INFRARED SENSOR 1

PIN	Sambungan
GND	GND (Extender GND and Vcc)
VCC	VCC (5V: Extender GND and Vcc)
OUTPUT	31, 39, 47 (Arduino Mega)

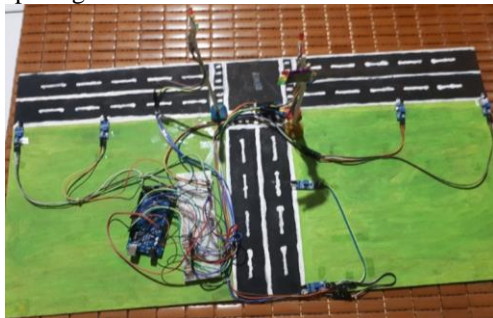
TABEL V
INFRARED SENSOR 2

PIN	Sambungan
GND	GND (Extender GND and Vcc)
VCC	VCC (5V: Extender GND and Vcc)
OUTPUT	33, 41, 49 (Arduino Mega)

IV. HASIL DAN ANALISIS

A. Hasil Prototype Traffic Light 3 Jalur

Rancangan 3 jalur lalu lintas yang diimplementasikan menggunakan infrared pada setiap lampu lalu lintas sebagai masukan untuk mengetahui kepadatan setiap jalur, dapat dilihat pada gambar 9.

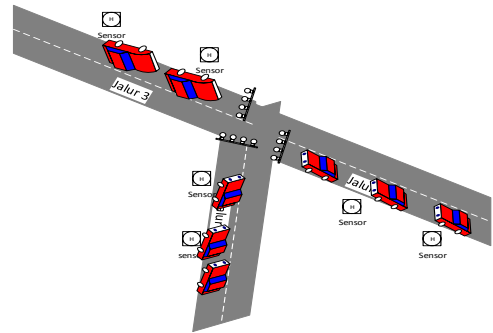


Gambar. 9. Rancangan 3 jalur lampu lalu lintas prototype

Ketika *user* menjalankan *prototype*, lampu lalu lintas pada 3 jalur akan menyala. Lampu lalu lintas yang akan menyala pada 3 jalur secara bergantian, misalnya jika jalur 1 lampu hijau menyala maka jalur 2 dan 3 lampu menyala warna merah. Pada kendali lampu lalu lintas ini digunakan 3 *input* yang akan dimasukkan kedalam proses *fuzzy* yaitu, waktu tunggu, sensor *infrared* 1 (jika kendaraan berhenti 5 detik), dan sensor *infrared* 2 (jika kendaraan berhenti 10 detik). Ketika proses *fuzzy* dijalankan, waktu lampu hijau sebagai *output* akan ditambahkan waktunya jika memenuhi *rule* pada *fuzzy* seperti terjadi penambahan waktu 0, 5, 10, dan 15 detik pada lampu hijau.

B. Analisis Proses dan Algoritma

Tampilan rancangan *prototype* lampu lalu lintas 3 jalur. Dapat dilihat pada gambar 10 dibawah ini :



Gambar. 10. Tampilan map lalu lintas 3 jalur

Pada rancangan *prototype* pengaturan lampu lalu lintas untuk 3 jalur atau pertigaan dimana setiap jalan dilengkapi sensor panas untuk mendeteksi jumlah kendaraan, apabila terdeteksi kemacetan dengan suhu panas maka jalur tersebut akan ditambahkan waktu lampu hijaunya dan sebaliknya jalur yang lancar, lampu merahnya diberikan waktu lebih lama dengan penambahan 5, 10, 15 detik berdasarkan kondisi pada lalu lintas. Pada map diatas terdapat beberapa infrared pada setiap jalur, semakin banyak sensor yang digunakan untuk mendeteksi kepadatan jumlah kendaraan semakin baik nilai akurasi pada penerapan algoritma *fuzzy* dan metode mandani.

Untuk implementasi algoritma terdapat beberapa tahapan yaitu :

1) Implementasi Metode Mamdani

Penggunaan metode mandani lebih memudahkan selain pada penerapan kondisi tertentu seperti digunakan pada pengaturan lalu lintas dinamis. Terlihat pada gambar 11.



Gambar. 11. Flowchar implementasi metode Mandani

Pada Gambar 11. dapat dilihat bahwa metode *Fuzzy Logic Mamdani* telah mendeklarasikan *variable input* ke dalam program. *Variable input* itu terdiri dari waktu tunggu, sensor *infrared* 1, dan sensor *infrared* 2. Setelah dideklarasikan *input* tersebut, *fuzzy* akan menentukan *rule* (aturan) himpunan *fuzzy*. Aturan-aturan himpunan *fuzzy* nantinya akan melakukan *defuzifikasi* dan melakukan pengujian terhadap jalur-jalur yang akan terdeteksi kemacetan. Aturan *fuzzy* yang dibuat diatas terdiri dari 12 aturan dengan hasil *output* yaitu penambahan waktu lampu hijau selama 0, 5, 10, dan 15 detik sesuai dengan kondisi jalur lalu lintas nya.

2) PROSES ATUR KENDALI LAMPU MERAH, KUNING, HIJAU

Pseudocode pengaturan lampu merah, kuning dan hijau dapat dilihat pada gambar 12. Sebagai berikut :

```
Lamp(){
  Read→lampu hijau,
  Read→lampu merah,
  Read→ lampu kuning
  Lampsett→1
  Millis→timesetted
}
Mulai kuning(){
  Write→lampuhijau, low
  Write→lampumerah, low
  Write→lampumerah,low
  Write→kuning, high
  Write→kuning, low
  Write →hijau, low
  Write→hijau,high
  Write→merah, low
  Write→merah,high
}
Reset(){
  Waktutunggu→0
  Waktutunggu→waktutunggu+waktutunggu/1000
}
```

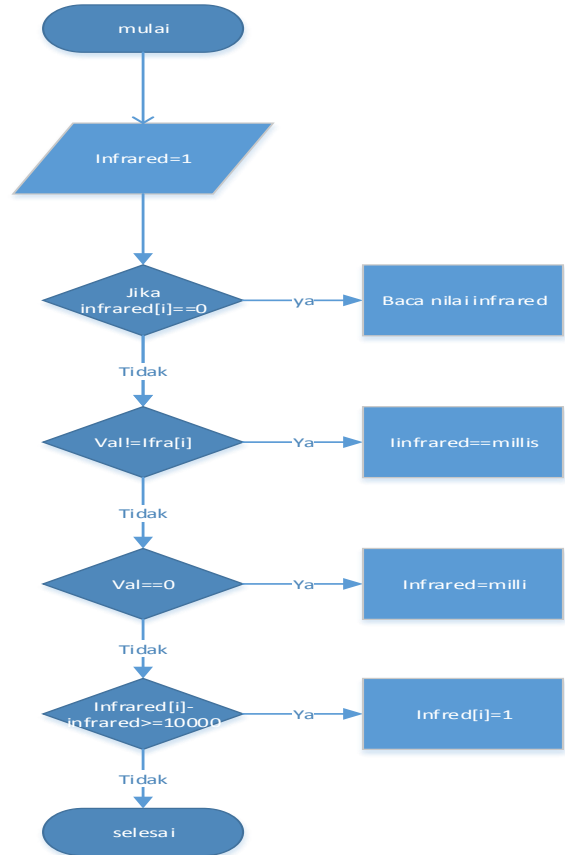
Gambar. 12. Pseudocode proses Kendali Lampu Lalu Lintas

Pada gambar 12, dapat dilihat bahwa telah terjadi proses atur kendali lampu lalu lintas pada prosedur *setLamp()*, *beginYellow()*, dan *resetLane()*. Pada proses *setLamp()*, dijelaskan bahwa ketika proses 1 jalur lampunya berwarna hijau, maka 2 jalur lainnya akan berwarna merah dan lampu kuning pada 3 jalur tidak diaktifkan. Tetapi, pada proses *beginYellow()*, proses tersebut mengatur lampu kuning yang akan menyala ketika terjadi pergantian dari lampu hijau ke merah atau sebaliknya. Proses *resetLane()* dilakukan ketika jalur tersebut lampunya menyala warna hijau. Ketika lampu akan menyala lampu hijau, waktu tunggu ketika hijau di jalur lain akan ditambahkan untuk mendapatkan waktu tunggu di jalur yang akan menyala lampu hijau. Pada proses ini juga, ketika lampu menyala hijau, seluruh *infrared* di jalur yang hijau akan diatur ulang (*reset*) dan akan melakukan proses *fuzzy* kepada jalur yang sedang merah. Lampu hijau yang akan dinyalakan di setiap jalur secara berurutan yaitu jalur 1 – 2 – 3.

Jalur pada setiap sensor yang ditempatkan semakin banyak semakin baik untuk memastikan nilai kepadatan kendaraan setiap jalur dan memastikan jalur tersebut lancar atau padat, sehingga pengaturan dengan algoritma *fuzzy mamdani* memberikan kemudahan kondisi setiap jalur secara otomatis.

3) IMPLEMENTASI DETEKSI SENSOR INFRARED 1 DAN 2

Untuk flowchar proses sensor infrared 1 dan infrared 2 dapat dilihat pada gambar 13.



Gambar. 13. Flowchar deteksi Sensor Infrared 1 dan Infrared 2

Pada Gambar 13, dapat dilihat bahwa, telah dilakukannya proses deteksi sensor *Infrared* pertama dan kedua pada jalur yang sedang menyala lampu merah. Sensor *Infrared* pertama akan aktif setelah lampu hijau berubah menjadi merah dan mendeteksi kendaraan yang berhenti tepat di sensor selama 5 detik dan dimasukkan sebagai salah satu *input fuzzy*. Sensor *Infrared* 2 juga aktif ketika lampu sedang merah dan mendeteksi kendaraan tepat di sensor selama 10 detik dan dihitung ke dalam aturan *fuzzy*. Setiap sensor sangat mempengaruhi kondisi lampu lalu lintas secara dinamis berdasarkan setiap jalur yang memberikan nilai kepadatan pada sensor infrared.

4) IMPLEMENTASI DEKLARASI, SETUP, LOOP PADA PROTOTYPE

Pseudocode deklarasi pengaturan kondisi infrared 1 dan infrared 2 yang digunakan sebagai masukan serta

penambahan waktu dapat dilihat pada gambar 14, sebagai berikut :

```

DEKLARASI:
int red[] → {0, 13, 10, 7};
int yellow[] → {0, 12, 9, 6};
int green[] → {0, 11, 8, 5};
int sisawaktu[] → {0, 0, 0, 0};
int waktutunggu[] → {0, 0, 0, 0};
int inputIR1[] → {0, 31, 39, 47}; //input IR 2 untuk 3 jalur
int inputIR2[] → {0, 33, 41, 49}; //input IR 3 untuk 3 jalur
double statIR1[] → {0.0, 0.0, 0.0, 0.0};
double statIR2[] → {0.0, 0.0, 0.0, 0.0};
double selisihIR1[][2] → {{0.0, 0.0}, {0.0, 0.0}, {0.0, 0.0}};
double selisihIR2[][2] → {{0.0, 0.0}, {0.0, 0.0}, {0.0, 0.0}};
double selisihwaktu[][3] → {{0.0, 0.0, 0.0}, {0.0, 0.0, 0.0}, {0.0, 0.0, 0.0}};
int plusgreen[] → {0, 0, 0, 0};
int timegreen [] → 15000;
int timeyellow → 2000;
boolean IR1checked[] → {0, 0, 0, 0};
boolean IR2checked[] → {0, 0, 0, 0};
boolean prevIR1[] → {1, 1, 1, 1};
boolean prevIR2[] → {1, 1, 1, 1};
int curGreen → 1;
int curRed[] → {2, 3};
int IR1StartTime[] → {0, 0, 0, 0};
int IR1CurTime[] → {0, 0, 0, 0};
int IR2StartTime[] → {0, 0, 0, 0};
int IR2CurTime[] → {0, 0, 0, 0};

```

Gambar. 14. Deklarasi, Setup, Loop pada Prototype

Pada Gambar 14, pertama harus mendeklarasikan apa saja yang akan digunakan didalam program seperti input lampu hijau, merah, kuning, sensor *infrared* 1, sensor *infrared* 2, waktu lampu hijau, waktu lampu kuning, dan waktu tunggu. *Void Setup()* digunakan untuk menjelaskan bahwa lampu hijau, kuning, dan merah akan menjadi *output* dalam program, serta sensor *infrared* 1 dan 2 sebagai *input* dalam program. *Void Loop()* akan menjelaskan proses keseluruhan *prototype* dari pengaturan lampu, deteksi sensor, terjadinya penambahan waktu lampu hijau serta pengaturan ulang lampu hijau pada *resetLane()* ketika lampu kuning sudah lebih dari 2 detik.

C. Pengujian Aplikasi dan Algoritma

Berikut adalah hasil penelitian dengan menggunakan metode Mamdani yang tertera pada tabel VI.

Dari hasil yang didapat pada tabel VI, pengujian tersebut untuk mengetahui keberhasilan metode Mamdani dalam pendeteksian yang diterapkan didalam *prototype* tersebut. Pengujian ini dilakukan sebanyak 50 kali pengujian dan penulis melakukan pengujian metode secara berurutan dimana di dalamnya terdapat beberapa fungsi keanggotaan untuk menguji kebenaran *output* yang akan dihasilkan dari hasil pendeteksian *prototype*. Dalam pengujian ini, secara rinci dapat dilihat pada tabel VII. Sebagai berikut:

TABEL VII
KESIMPULAN HASIL PENGUJIAN PROTOTYPE

Pengujian	Hasil	%
Pengujian gagal	3	6
Pengujian berhasil	47	94
Total	50	100

Berikut ini adalah himpunan *fuzzy* dari *prototype*, yaitu:

1. Waktu Tunggu terdiri atas 2 himpunan *fuzzy*, yaitu: *LOW* dan *HIGH*.
2. *Infrared* 1 terdiri atas 2 himpunan *fuzzy*, yaitu: 1 (ADA) dan 0 (TIDAK_ADA).
3. *Infrared* 2 terdiri atas 2 himpunan *fuzzy*, yaitu: 1 (ADA) dan 0 (TIDAK_ADA).

METODE FUZZY LOGIC MAMDANI YANG DIGUNAKAN ADALAH METODE FUZZY LOGIC MAMDANI YANG MENGHASILKAN DEFUZZYFIKASI BERUPA OUTPUT YAITU PENAMBAHAN WAKTU LAMPU HIJAU SELAMA 0 DETIK, 5 DETIK, 10 DETIK, DAN 15 DETIK

TABEL VI
METODE MAMDANI

Tabel Pengujian Prototype Kendali Lampu Lalu Lintas										
Pengujian ke-	Jalur 1	Jalur 2	Jalur 3	Infrared 1	Infrared 2	Waktu Tunggu	Lampu Hijau (detik)	Tambah Waktu (detik)	Total Waktu (detik)	Hasil
1	v			0	0	LOW	15	0	15	Sesuai
2		v		0	0	LOW	15	0	15	Sesuai
3			v	0	0	LOW	15	0	15	Sesuai
4	v			1	0	LOW	15	5	20	Sesuai
5		v		1	0	LOW	15	5	20	Sesuai
6			v	0	1	LOW	15	5	20	Sesuai
7	v			1	1	LOW	15	10	25	Sesuai
8		v		1	1	LOW	15	10	25	Sesuai
9			v	1	1	HIGH	15	15	30	Tidak Sesuai (Waktu= 25s)
10	v			1	1	HIGH	15	15	30	Sesuai
11		v		1	1	HIGH	15	15	30	Sesuai
12			v	1	1	HIGH	15	15	30	Sesuai
13	v			0	0	HIGH	15	5	20	Sesuai
14		v		0	0	HIGH	15	5	20	Sesuai
15			v	0	0	LOW	15	0	15	Sesuai
16	v			1	1	LOW	15	10	25	Sesuai
17		v		0	1	LOW	15	5	20	Sesuai
18			v	1	1	LOW	15	10	25	Sesuai
19	v			1	1	LOW	15	10	25	Sesuai
20		v		1	1	HIGH	15	15	30	Tidak Sesuai (Waktu= 25s)
21			v	1	1	HIGH	15	15	30	Sesuai
22	v			1	1	HIGH	15	15	30	Sesuai
23		v		1	1	HIGH	15	15	30	Sesuai
24			v	1	1	HIGH	15	15	30	Sesuai
25	v			0	1	HIGH	15	10	25	Sesuai
26		v		0	1	HIGH	15	10	25	Sesuai
27			v	0	0	HIGH	15	5	20	Sesuai
28	v			0	0	LOW	15	0	15	Sesuai
29		v		1	0	LOW	15	5	20	Sesuai
30			v	1	1	LOW	15	10	25	Sesuai
31	v			1	1	LOW	15	10	25	Sesuai
32		v		1	0	HIGH	15	10	25	Sesuai
33			v	1	0	HIGH	15	10	25	Sesuai
34	v			1	1	HIGH	15	15	30	Tidak Sesuai (Waktu= 25s)
35		v		1	1	HIGH	15	15	30	Sesuai
36			v	1	1	HIGH	15	15	30	Sesuai
37	v			1	0	HIGH	15	5	20	Sesuai
38		v		0	1	HIGH	15	10	25	Sesuai
39			v	0	1	LOW	15	5	20	Sesuai
40	v			0	0	LOW	15	0	15	Sesuai
41		v		1	0	LOW	15	5	20	Sesuai
42			v	1	1	LOW	15	10	25	Sesuai
43	v			1	0	LOW	15	5	20	Sesuai
44		v		1	0	LOW	15	5	20	Sesuai
45			v	1	1	LOW	15	10	25	Sesuai
46	v			1	1	LOW	15	10	25	Sesuai
47		v		1	1	HIGH	15	15	30	Sesuai
48			v	1	1	HIGH	15	15	30	Sesuai
49	v			1	1	HIGH	15	15	30	Sesuai
50		v		0	0	HIGH	15	5	20	Sesuai

V. KESIMPULAN

A. Kesimpulan

Dari penelitian dan pengujian yang dilakukan maka dapat disimpulkan sebagai berikut:

1. *Prototype* kendali lampu lalu lintas ini menggunakan metode *Fuzzy Logic Mamdani* dan 50 pengujian yang dilakukan 47 percobaan sesuai dengan kondisi dengan presentase 94% akurat yang telah diuji, 3 pengujian tidak sesuai kondisi atau 3% belum akurat.
2. *Prototype* kendali lampu lalu lintas diterapkan pada pengujian di 3 jalur dengan *output* yang dihasilkan pada defuzzifikasi *fuzzy logic mamdani* yaitu *Mean Of Maximum*.
3. penerapan lampu lalu lintas dinamis dengan penambahan sensor infrared memberikan nilai yang akurat untuk mengetahui nilai pada setiap jalur.

B. Saran

Prototype kendali lampu lalu lintas ini diharapkan akan menjadi lebih baik dengan beberapa saran yang dapat digunakan:

1. Untuk pengembangan lebih lanjut, disarankan menambahkan dan membuat fitur pemantau pelanggaran lalu lintas dengan menggunakan kamera.
2. Untuk pengembangan lebih lanjut, disarankan implementasi sensor menggunakan sensor lain untuk diuji seperti sensor kecepatan, sensor panas, dan lainnya karena sensor *infrared* tidak dapat membedakan jenis benda yang akan dideteksi oleh sensor.
3. Untuk penerapan lampu lalu lintas dinamis diharapkan ditambahkan kamera untuk membaca deteksi kejadian kecelakaan sehingga dapat memberikan kemudahan bagi kepolisian dalam menangani kemacetan yang berakibat pada kemacetan.
4. Untuk penggunaan metode lain seperti Tsukamoto, atau lain bisa dibandingkan untuk akurat pemberian nilai pada titik kepadatan jalur.

DAFTAR PUSTAKA

- [1] Mangatur, Edison, & Suandi. (2018). Analisis Dampak Kemacetan Lalu Lintas Terhadap Pendapatan Masyarakat dan aksesibilitas di kota Jambi . *Jurnal Pembangunan berkelanjutan, vol 1 no 1*, 1-10.
- [2] Zulfikar, Tarmizi, & Adria, A. (2011). Perancangan Pengontrolan Traffic Light Otomatis. *Jurnal Rekayasa Elekrika Vol 9 No. 3*, 126-131.
- [3] Ramadhani, A. (2018, 02 25). *Properti*. Dipetik 03 12, 2019, dari Kompas.com: <https://properti.kompas.com/read/2018/02/25/143845721/jakarta-kota-termacet-ke-12-dunia>
- [4] E. E. Prasetyo, "Pembuatan Sistem Otomasi Dispenser Menggunakan Mikrokontroler Arduino Mega 2560," Seminar Nasional Teknologi Informasi dan Multimedia, 6-7 Februari 2016. ISSN: 2302-3805.
- [5] Zulfikar, Tarmizi, dan Oktavina, "Desain Sistem Kontrol Traffic Light Adaptif Pada Persimpangan Empat Berbasis PLC Siemens," Jurnal Nasional Teknik Elektro, Vol. 4, No. 1, Maret 2015. ISSN: 2302-2949.
- [6] A. Giyartono, dan P. E. Kresnha, "Aplikasi Android Pengendali Lampu Rumah Berbasis Mikrokontroler ATmega 328," Seminar Nasional Sains dan Teknologi, Jakarta, 17 November 2015. P-ISSN: 2407-1846/ E-ISSN: 2460-8416.

- [7] B. Sutomo, "Pemodelan Sistem Kontrol Traffic Light Berdasarkan Kepadatan Kendaraan Dengan Teknik Edge Detection dan Logika Fuzzy," Jurnal Informatika, Vol. 15, No. 2, Desember 2015.
- [8] A. Saleh, "Implementasi Metode Fuzzy Mamdani Dalam Memprediksi Tingkat Kebisingan Lalu Lintas," Seminar Nasional Teknologi Informasi dan Multimedia, 6-8 Februari 2015. ISSN: 2302-3805.
- [9] R. C. Kumar, dkk, "Obstacle Avoiding Robot - A Promising One," International Journal of Advanced Research in Electrical, Electronics and Instrumentation Engineering, Vol. 2, Issue 4, April 2013. P-ISSN: 2320-3765/ E-ISSN: 2278-8875.
- [10] W. P. Sari, dan R. Wijaya, "Sistem Keamanan Rumah Menggunakan Sensor Getaran Dengan Output Suara Berbasis PC Menggunakan Bahasa Pemrograman Visual Basic," Jurnal KomTekInfo Vol. 4, No. 1, Juni 2017. ISSN: 2356-0010