

SISTEM PAKAR SEBAGAI PENGENDALI LAMPU LALU-LINTAS PADA PERSIMPANGAN JALAN MENGGUNAKAN FUZZY LOGIC BERBASIS ANDROID

Siswaya^{1*}, Sunardi² dan Anton Yudhana²

¹Program Studi Teknik Informatika, Universitas Ahmad Dahlan

²Program Studi Teknik Elektro, Universitas Ahmad Dahlan

Jl. Prof.Dr. Soepomo Janturan, Umbulharjo, Yogyakarta 55164.

*Email: siswaya.stmik@gmail.com

Abstrak

Perkembangan teknologi kecerdasan buatan memungkinkan sistem traffic light mengatur kendaraan berdasarkan level kepadatan kendaraan di perempatan jalan. Lampu hijau diatur agar fleksibel, ini bertujuan supaya setiap jalur mendapatkan hitungan detik yang tepat dengan tingkat kepadatan yang terjadi saat itu. Permasalahan yang muncul sehari-hari adalah didominasi bukan biner serta bersifat non linier, hal demikian sangat cocok jika menggunakan fuzzy logic sebagai solusinya, karena fuzzy logic mampu mengambil keputusan yang lebih manusiawi dan adil. Metodologi penelitian ini menggunakan Rational Unified Process (RUP) dimana metode ini unggul pada proses iteratif dan inkrementalnya. RUP juga menggunakan pendekatan pengembangan software yang fokus pada architecture serta mengarah berdasarkan pemakaian kasusnya. RUP termasuk well defined dan well structured dalam hal proses rekayasa softwaranya. RUP membagi tahapan pengembangan software mulai inception, elaboration, construction, dan transition. Riset ini menggunakan fuzzy logic terutama sebagai penghitung tingkat kepadatan kendaraan dan lama lampu hijau menyala, untuk itu digunakan fuzzy inference system serta melibatkan fuzzy logic toolbox. Realisasi sistem fuzzy logic control ini dimulai dengan pemetaan linguistic variable yaitu variable input dan variable output. Aplikasi android pada sistem smart traffic light diharapkan dapat memberikan informasi yang akurat secara real time kepada para petugas di lapangan dalam memantau kondisi kepadatan lalu-lintas..

Kata kunci : Android, Fuzzy Logic, Sistem Pakar

1. PENDAHULUAN

Dasar pengaturan atau pengendalian lampu lalu-lintas di perempatan atau persimpangan selama ini menggunakan basis waktu tetap, sehingga padat atau tidaknya kendaraan pengendali lalu-lintas akan tetap bekerja sesuai dengan program yang telah dimasukkan di dalam chip mikrokontrolernya. Hal demikian menjadi tidak efektif terutama di saat semua jalur di perempatan tersebut sepi, sehingga kendaraan harus menanti sampai lampu hijau menyala dalam waktu yang lama. Sama halnya juga pada saat keadaan dan waktu tertentu kendaraan dari satu atau dua jalur padat, sedang pada jalur yang lain tidak, tentunya hal ini akan mengakibatkan antrian kendaraan yang panjang jika tetap menggunakan pewaktuan lalu-lintas yang sesuai dengan *setting* awalnya. Jadi tujuan utama dari sistem pengendali lampu lalu-lintas dalam mengatasi penumpukan dan kemacetan lalu-lintas belum tercapai.

Konsep penyelesaian masalah dalam penelitian harus dirumuskan sejak awal secara jelas sebagai solusi akhir dari penelitian yang dilakukan. Pelaksanaan penelitian harus memiliki landasan teori dalam bentuk kajian pustaka, di samping itu kajian teori juga berfungsi untuk mengorganisasikan temuan-temuan peneliti yang telah dilakukan sebelumnya (Prawira, Sunardi and Fadlil, 2017).

Penelitian tentang pemanfaatan kecerdasan buatan untuk mengendalikan *traffic light* (lampu lalu-lintas) dapat dilakukan dengan cara mengimplementasikan *fuzzy logic control*. Informasi data dari sensor dapat digunakan sebagai indikator utama tentang *level* kepadatan kendaraan yang melintas dan program *fuzzy* diimplementasikan dengan arduino mega 2560 (Nugroho, 2017).

Penelitian dan pengembangan ilmu pengetahuan lebih lanjut yang ingin menyamai kemampuan otak manusia dinamakan kecerdasan buatan. Perkembangan teknologi kecerdasan buatan memungkinkan sistem pengendali lampu lalu-lintas mampu mengatur kendaraan berdasarkan *level* kepadatan kendaraan di perempatan jalan, dengan tetap memperhatikan kebutuhan bagi jalur-jalur yang lainnya (Nugroho, 2017).

Lampu hijau perlu diatur agar lebih fleksibel, ini bertujuan supaya masing-masing jalur mendapatkan hitungan detik yang tepat dengan tingkat kepadatan yang terjadi saat itu, dan efek positifnya adalah simpang jalan yang lain tidak harus menunggu antrian lampu hijau terlalu lama, selain itu kepadatan lalu-lintas pada masing-masing jalur bisa terurai dan lancar (Yudanto *et al.*, 2013).

Konsep pemecahan masalah di atas kemudian dikenal dengan istilah *smart traffic light*, dengan menambahkan sebuah *chip* mikrokontroler sebagai kendali utamanya. Cara ini dapat mengatur berapa lama lampu hijau menyala berdasar tingkat kepadatan kendaraan yang semua itu memanfaatkan data keluaran yang dihasilkan oleh sensor-sensornya. Penentuan lama lampu hijau menyala telah ditetapkan berdasar persentase kepadatan kendaraan pada setiap jalurnya, jika persentasenya <30% maka lamanya 10 detik, persentasenya >30% & <60% maka lamanya 20 detik dan persentasenya >60% maka lamanya 30 detik (Alfith, 2014).

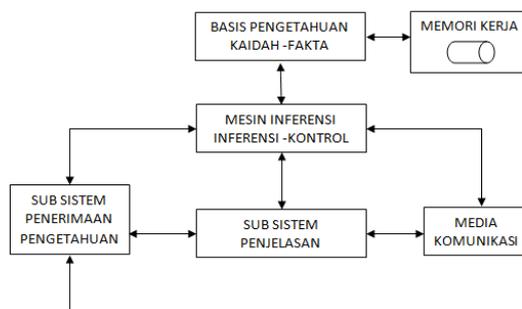
Komputer yang pada awalnya hanya sekedar digunakan untuk alat menghitung saja, maka sesuai dengan perkembangan teknologi, komputer sekarang telah mampu mengambil peran dan tugas-tugas rumit yang sebelumnya harus dilakukan oleh manusia, bahkan saat ini sudah mampu meniru perilaku manusia dalam mengambil keputusan yang kemudian dinamakan kecerdasan buatan (Listiyono, 2008). Sistem kecerdasan buatan berbeda dengan program komputer biasa, sehingga dapat dipakai untuk mengatasi masalah yang tidak terstruktur dan belum ada prosedur yang untuk memecahkan masalah itu. (Listiyono, 2008). Perkembangan kecerdasan buatan mengalami kemajuan yang cepat, dimana sistem didesain mampu menirukan keahlian seorang ahli atau pakar tertentu untuk menjawab pertanyaan dan menyelesaikan masalah baik di bidang bisnis, ekonomi, kesehatan, politik, keuangan dan lain sebagainya (Listiyono, 2008).

Kecerdasan buatan merupakan salah satu bagian daripada ilmu komputer yang sudah terbukti mampu membantu manusia, dan hal ini sesuai dengan tujuan praktisnya. Tujuan utama kecerdasan buatan adalah memprogram sebuah komputer yang mampu berfikir dan berlogika seperti otak manusia, sehingga dapat membantu mengambil keputusan, pencarian informasi secara akurat atau menjadikan komputer lebih mudah dipahami dengan memanfaatkan bahasa yang lebih natural. Sistem pakar adalah salah satu komponen dari kecerdasan buatan yang dirancang secara spesifik untuk mentransfer atau menduplikat kepakaran seorang ahli bidang tertentu ke dalam bentuk program komputer (Triyanto and Fadlil, 2014).

Pengetahuan dan kepakaran seseorang disimpan secara khusus dalam sebuah komputer dalam bentuk program, sehingga seseorang yang tidak memiliki pengetahuan khusus atau bukan pakarnya mampu memberi solusi atas pertanyaan yang ada, serta mampu memberi keputusan yang seharusnya dilakukan oleh pakarnya, meskipun tingkat kepakaran seseorang yang diadopsi ke sistem komputer akan menentukan kualitas sistem pakarnya (Listiyono, 2008).

Pemecahan masalah dalam sistem pakar menggunakan atau memanfaatkan *knowledge* (keahlian khusus yang tidak setiap orang menguasainya), fakta serta teknik logika yang biasanya dipakai oleh seorang pakar (Rackman, 2014).

Komponen-komponen dalam sistem pakar antara lain adalah *knowledge base*, *data base*, *inference engine* dan *user interface*. Keterkaitan komponen-komponen tersebut dapat dilihat pada Gambar 1.



Gambar 1. Keterkaitan komponen-komponen utama sistem pakar

Fuzzy logic dikenalkan pertama kali oleh Prof. Lotfi A.Zadeh dari Universitas California di Barkeley. Jauh sebelum itu telah dikenal sebuah logika *crisplogic* yang mengenal dua kondisi

secara tegas yaitu nilai benar dan salah. Berbeda dengan *fuzzy logic* yang samar atau tidak tegas antara nilai benar atau salah. Namun demikian, *fuzzy logic* merupakan suatu metode yang tepat untuk memetakan suatu ruang masukan ke dalam suatu ruang keluaran (Rackman, 2014).

Sistem kendali yang menggunakan prinsip *fuzzy logic* dinamakan *fuzzy logic control*. Pemahaman *fuzzy logic* sendiri adalah aturan pengambilan keputusan dengan tujuan pemecahan suatu masalah untuk sistem yang tidak bisa dibuat model atau muncul adanya ambiguitas serta kesamaran. *Fuzzy logic* sering disebut sebagai logika yang tidak tegas atau samar, hal ini disebabkan *fuzzy logic* mendapat banyak informasi yang bersifat tidak pasti untuk kemudian menjadi logika bernilai yang diperhitungkan. Persamaan logika menentukan *fuzzy logic* jadi bukan dari persamaan diferensial rumit. *Fuzzy logic system* meliputi himpunan *fuzzy* dan aturan *fuzzy* (Nugroho, 2017).

Permasalahan yang muncul di kehidupan sehari-hari adalah lebih dominan bukan biner serta bersifat non linier, hal demikian cocok sekali kiranya jika menggunakan *fuzzy logic* sebagai solusinya, karena *fuzzy logic* menerapkan nilai linguistik non linier. *Fuzzy logic* juga mampu mengambil keputusan yang lebih manusiawi dan adil (Yudanto *et al.*, 2013).

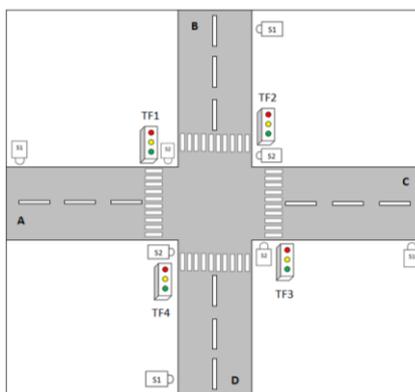
Mobile device menggunakan komponen *android* yang terdiri dari aplikasi inti, *middleware* dan sistem operasi. *Eclipse* digunakan untuk pengembangan software dan *Integrated Development Eclipse* (IDE) adalah *environment* untuk mengembangkannya. Kelebihan *eclipse* adalah bisa berfungsi di mayoritas *platform* dan bersifat *multi-languade*, *multi-platform* dan *multirole*, di samping itu juga *open source* serta gratis. *Open source* berarti dapat melihat kode program dari *software* ini. Pengembangan *open source* dapat menggunakan komponen *plug-in* (Umar and Hari Prabowo, 2016).

Dukungan kemajuan di bidang elektronika terutama di bidang mikrokontroler dan *android* sangatlah pesat. Salah satu *board* yang berisi mikrokontroler adalah *arduino*, yang merupakan *chip* serbaguna yang mudah dalam pemrogramannya. *Android* adalah sebuah sistem operasi yang *open source*. *Android* bisa dikoneksikan dengan piranti elektronika untuk pengendalian atau menerima data informasi dari piranti elektronika (Yudhana, Dwi and Putra, 2018).

2. METODOLOGI

Rancangan metode *fuzzy logic control* untuk pengendali lalu-lintas atau *traffic light* ini akan diimplementasikan untuk perempatan jalan atau jalan simpang empat. Setiap jalur atau ruas jalan akan dipasang masing-masing dua buah sensor yaitu sensor 1 (S1) dan sensor 2 (S2) pada jarak tertentu, yang berfungsi untuk membaca kendaraan yang datang yang masuk dalam wilayah antrian lampu lalu-lintas. Masukan dari 2 buah sensor ini kemudian akan dikalkulasi untuk menghitung nilai linguistik pada *fuzzy logic control*-nya. *Traffic rule* pada persimpangan jalan dapat diuraikan secara urut berikut ini :

- Pengendalian utama dari proses fuzzifikasi ini adalah lampu hijau, sedangkan lampu merah akan menyesuaikan lampu hijau yang aktif saja.
- Waktu penyalaan lampu hijau dibatasi oleh nilai maksimal dan minimal tertentu, sehingga walaupun tingkat kepadatan kendaraan padat sekali tetap akan mendapat antrian berhenti oleh lampu merah, kemudian ruas jalan lain diberi kesempatan untuk berjalan.

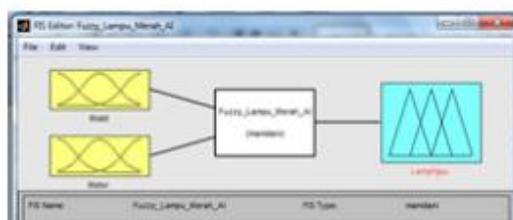


Gambar 2. Posisi sensor dan ruas perempatan jalan

Penghitung jumlah kendaraan masuk adalah S1 (sensor 1), sedang penghitung jumlah kendaraan keluar adalah S2 (sensor 2). Mempertimbangkan jumlah I/O-nya dimana *input* yang akan dipakai ada 8 *pin* dan *output*-nya ada 12 *pin* maka dipilih *arduino mega 2560* (Nugroho, 2017).

Metodologi penelitian ini menggunakan *Rational Unified Process* (RUP) dimana metode ini unggul pada proses iteratif dan inkrementalnya yaitu hasil pengembangan *software* yang dilakukan secara berulang serta berjenjang dengan cara menaik. RUP juga menggunakan pendekatan pengembangan *software* yang fokus pada *architecture* serta mengarah berdasarkan pemakaian kasusnya. RUP termasuk *well defined* dan *well structured* dalam hal proses rekayasa *software*-nya. RUP membagi tahapan pengembangan *software* sebagai berikut : a. permulaan (*inception*), b. perluasan atau perencanaan (*elaboration*), c. konstruksi (*construction*), dan transisi (*transition*).

Riset ini menggunakan *fuzzy logic* terutama sekali sebagai penghitung tingkat kepadatan kendaraan dan lama lampu hijau menyala supaya lebih efisien, untuk itu digunakan *fuzzy inference system* (FIS) serta melibatkan *fuzzy logic toolbox* yang terdapat dalam Matlab dengan tipe mamdani (Yudanto *et al.*, 2013). Gambar 3 – 6 adalah permodelan beberapa variabel.

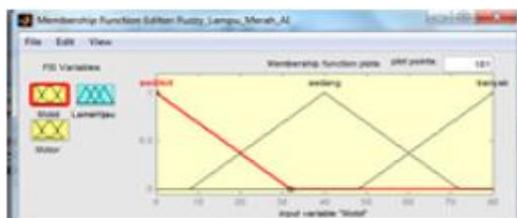


Gambar 3. Model *fuzzy logic*

Variabel linguistik yang akan digunakan.

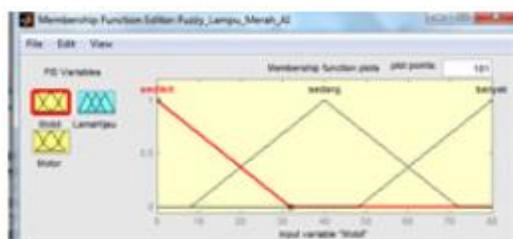
Variabel masukan :

- Jumlah mobil : sedikit, sedang dan banyak.



Gambar 4. Model masukan jumlah mobil

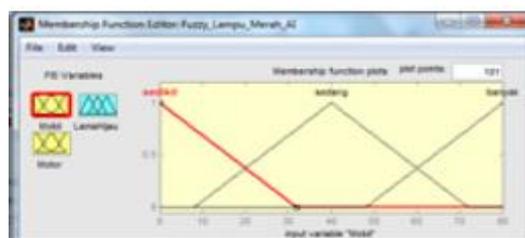
- Jumlah motor : sedikit, sedang dan banyak.



Gambar 5. Model masukan jumlah motor

Variabel keluaran :

- Lama lampu hijau (detik) : sebentar, sedang, lama.



Gambar 6. Model keluaran lama lampu hijau

Keterangan :

- Sedikit : jumlah kendaraan (mobil / motor) memenuhi <25% suatu simpang jalan.
- Sedang : jumlah kendaraan (mobil / motor) memenuhi <75% suatu simpang jalan.
- Banyak : jumlah kendaraan (mobil / motor) memenuhi >75% suatu simpang jalan.
- Tidak ada : tidak ada kendaraan (mobil / motor) pada suatu simpang jalan.
- Sebentar : 0 – 40 detik.
- Sedang : 40 – 80 detik.
- Lama : 80 – 120 detik.

3. HASIL DAN PEMBAHASAN

3.1. Deskripsi Masalah

Distribusi lama lampu hijau nyala pada setiap ruas jalan dipandang belum efektif, indikatornya adalah saat persimpangan jalan dalam kondisi tidak ramai maupun kondisi ramai, lama lampu hijau menyala masih sama. Kondisi ini mengakibatkan selain tidak efektif, juga banyak waktu yang terbuang percuma saat lampu hijau nyala serta adanya penumpukkan kendaraan di ruas jalan yang lainnya. Penerapan *fuzzy logic* diharapkan dapat memperbaiki kondisi ini.

Perhitungan jumlah kendaraan yang keluar atau masuk simpang jalan digunakan untuk menentukan tingkat kepadatan jalan, sehingga keberadaan sensor baik sensor *input* atau sensor *output* yang dipasang pada masing-masing ruas jalan sangat dibutuhkan. Sensor juga difungsi sebagai pendeteksi kendaraan yang sedang berada pada area kuning.

Penentuan waktu maksimal lampu hijau pada tiap-tiap ruas jalan adalah berbeda-beda tergantung pada panjang jalan (P) dan kecepatan rata-rata kendaraan yang melintas pada tiap ruasnya. Rumusnya adalah : Waktu (det) = $(P/100) * 60$ det.

Realisasi sistem dengan *fuzzy logic control* ini dimulai dengan membuat pemetaan *linguistic variable* yang meliputi *variable input* dan *variable output*. *Variable input* untuk jumlah kendaraan yang akan dideteksi oleh *detector pair* S1 dan S2 di tiap ruas persimpangan. Jadi rancangan sistem ini akan dipasang pada perempatan jalan sehingga terdapat empat *variable input*-nya. *Variable output* diperuntukkan bagi lama waktu lampu hijau menyala pada tiap ruas persimpangan. *Variable output* dibuat di dalam *fuzzy inference system* terutama dipakai untuk menterjemahkan pengendalian lama waktu lampu hijau menyala di setiap ruas jalan pada persimpangan. *Membership function* dari sistem diperlihatkan pada Tabel 1.

Tabel 1. Membership function pada fuzzy inference system

SimpangA	SimpangB	SimpangC	SimpangD	OUTPUT
sedikit	sedikit	sedikit	sedikit	cepat
sedang	sedang	sedang	sedang	sedang
banyak	banyak	banyak	banyak	lama
sangat banyak	sangat banyak	sangat banyak	sangat banyak	sangat lama

Memanfaatkan kombinasi empat data *variable input* pada Tabel 1 maka dapat dipakai untuk membuat IF – THEN *rule*-nya. Lama waktu lampu hijau menyala sangat dipengaruhi oleh kondisi ke empat *variable input* tersebut pada setiap simpang jalan. Langkah selanjutnya adalah membuat derajat keanggotaan jumlah kendaraan secara ril dengan menetapkan masing-masing kategori jumlah sedikit, sedang, banyak dan sangat banyak. Tabel 2 memperlihatkan derajat keanggotaan jumlah kendaraan.

Tabel 2. Variabel lingustik dan nilai lingustik

Variabel Lingustik	Nilai Lingustik	Nilai Puncak
sedikit	< 10	5
sedang	5 - 15	10
banyak	10 - 20	15
sangat banyak	> 15	20

Penggunaan metode *fuzzy logic control* setidaknya melibatkan empat tahapan yang harus dilalui dalam proses *fuzzy inference system* (FIS), antara lain *FIS Editor* untuk menyatakan *variable input* dan *variable output*, *Membership Function Editor* untuk menyatakan derajat keanggotaan variabel linguistik, *Rule Editor* untuk menyatakan aturan anteseden dan konsekuen, serta *Rule Viewer* untuk menyatakan defuzzyfikasinya (Nugroho, 2017).

Alasan penting penggunaan *rule viewer* pada proses defuzzyfikasi adalah dapat diaturnya nilai tegas dari setiap *variable input* dengan mencantumkan nilai lingustik *input* pada GUI *input*-nya, atau cukup menggeser garis merah pada setiap *variable input* maka *indicator lamp* akan menyatakan nilai tegasnya berupa waktu dalam detik.

4. KESIMPULAN

Aplikasi android pada sistem *smart traffic light* diharapkan dapat memberikan informasi yang akurat secara *real time* kepada para petugas di lapangan dalam memantau kondisi kepadatan lalu-lintas. Kombinasi sistem pakar dan *fuzzy logic control* yang keduanya merupakan cabang dari kecerdasan buatan diharapkan mampu menghasilkan pemahaman baru tentang kendali sistem khususnya pada penerapan di *traffic light system*. Sistem mampu secara responsif melayani kondisi-kondisi darurat saat kendaraan *ambulance* lewat atau pejabat sedang melintas di persimpangan jalan.

DAFTAR PUSTAKA

- Alfith (2014) 'Jurnal Momentum ISSN : 1693-752X Perancangan Smart Traffic Light Jurnal Momentum ISSN : 1693-752X', 16(2), pp. 95–100.
- Listiyono, H. (2008) 'Merancang dan Membuat Sistem Pakar', *Jurnal Teknologi Informasi DINAMIK*, XIII(2), pp. 115–124.
- Nugroho, E. A. (2017) 'Sistem pengendali lampu lalu lintas berbasis logika fuzzy', *Simetris*, 8(1), pp. 75–84.
- Prawira, T. Y., Sunardi dan Fadlil, A. (2017) 'Sistem Pendukung Keputusan / Decision Support System untuk Memprediksi Stok Barang pada Retail Mini Market Berbasis Mobile dengan Pendekatan Analisis Menggunakan Metode Algoritma Apriori', pp. 267–271.
- Rackman, M. B. dan Pakar, S. (2014) 'Sistem Pakar untuk Diagnosis Penyakit pada Tanaman Teh dengan Metode Fuzzy Logic Berbasis Android 1,2,3', pp. 628–636.
- Triyanto, S. dan Fadlil, A. (2014) 'Sistem Pakar Untuk Mendiagnosa Penyakit Kelinci Berbasis Web', *Jurnal Sarjana Teknik Informatika*, 1, pp. 1–13.
- Umar, R. dan Hari Prabowo, P. (2016) 'Pencarian Dan Pemesanan Travel Berbasis Mobile dengan Google Maps API', *Annual Research Seminar 2016*, 2(ISBN : 979-587-626-0), pp. 369–372.
- Yudanto, A. Y. *et al.* (2013) 'Optimalisasi Lampu Lalu Lintas dengan Fuzzy Logic', V(2), pp. 58–62.
- Yudhana, A., Dwi, M. dan Putra, D. (2018) 'Rancang Bangun Sistem Pemantauan Infus Berbasis Android', *Transmisi*, 20, (2), April 2018.