

NASKAH PUBLIKASI

**SIMULASI PENGATURAN LAMPU LALU LINTAS BERDASARKAN
KEPADATAN KENDARAAN MENGGUNAKAN METODE PELABELAN
OBJEK DAN FUZZY INFERENCE SYSTEM TSUKAMOTO**

PROYEK TUGAS AKHIR

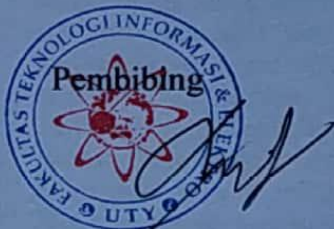
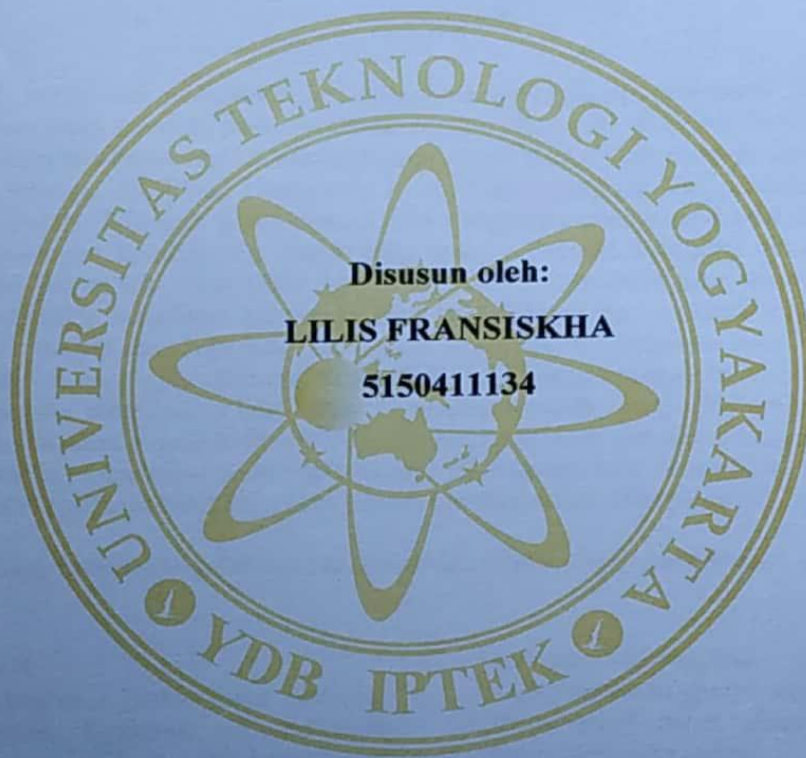


Disusun oleh:
Lilis Fransiskha
5150411134

**PROGRAM STUDI INFORMATIKA
FAKULTAS TEKNOLOGI INFORMASI DAN ELEKTRO
UNIVERSITAS TEKNOLOGI YOGYAKARTA
2020**

NASKAH PUBLIKASI

**SIMULASI PENGATURAN LAMPU LALU LINTAS BERDASARKAN KEPADATAN
KENDARAAN MENGGUNAKAN METODE PELABELAN OBJEK DAN FUZZY
INFERENCE SYSTEM TSUKAMOTO**



Dr. Enny Itje Sela, S.Si., M.Kom.

Tanggal: 29-02-2020

SIMULASI PENGATURAN LAMPU LALU LINTAS BERDASARKAN KEPADATAN KENDARAAN MENGGUNAKAN METODE PELABELAN OBJEK DAN FUZZY INFERENCE SYSTEM TSUKAMOTO

Liliz Fransiskha

*Program Studi Informatika, Fakultas Teknologi Informasi dan Elektro
Universitas Teknologi Yogyakarta
Jl. Siliwangi (Ringroad Utara) Jombor, Sleman, Yogyakarta
E-mail: LilizFransiskha@gmail.com*

ABSTRAK

Pertumbuhan jumlah kendaraan mengakibatkan terjadinya peningkatan kepadatan lalu lintas. Saat ini dengan pertumbuhan kendaraan yang semakin pesat membuat kondisi lalu lintas di Indonesia semakin padat setiap harinya. Sistem pengaturan lampu lalu lintas saat ini masih menggunakan sistem yang bekerja dengan waktu penetapan durasi nyala lampu merah, kuning, dan hijau. Pengaturan ini berlaku sepanjang hari, sementara jumlah arus lalu lintas tidaklah sama setiap waktu. Sehingga perlu suatu sistem yang dapat mengatur durasi nyala lampu lalu lintas yang dapat menyesuaikan dengan kondisi lalu lintas yang terjadi. Dengan teknologi pengolahan citra digital dapat diterapkan pada sistem lampu lalu lintas yaitu dengan mendeteksi kepadatan kendaraan atau jumlah kendaraan dari setiap ruas jalan. Pendeteksi kepadatan kendaraan atau jumlah kendaraan menggunakan metode Pelabelan Objek merupakan cara untuk mendeteksi kepadatan kendaraan atau jumlah kendaraan dengan cara memberi label pada setiap objek yang ada dan kemudian dihitung. Sedangkan untuk menghasilkan lama durasi lampu hijau yang akan diberikan pada setiap ruas jalur lampu lalu lintas, digunakan metode *Fuzzy Tsukamoto* sebagai parameter yaitu berdasarkan kepadatan kendaraan atau jumlah kendaraan dan lebar jalur dari setiap ruas jalan. Hasil yang didapatkan yaitu jumlah durasi nyala lampu hijau pada tiap-tiap ruas jalan lampu lalu lintas, lama durasi lampu hijau menyala ditentukan berdasarkan kepadatan kendaraan atau jumlah kendaraan dan lebar jalurnya.

Kata Kunci: Lampu Lalu Lintas, Pengolahan Citra, Pelabelan Objek, *Fuzzy Tsukamoto*.

1. PENDAHULUAN

[1] Berdasarkan data Badan Pusat Statistik, setiap tahunnya jumlah kendaraan di Indonesia semakin meningkat, dimulai dari mobil penumpang (mobil pribadi), mobil bus, mobil barang dan sepeda motor. Pada tahun 2017 jumlah kendaraan mencapai 138,556 juta unit dan pada tahun 2018 jumlah kendaraan mencapai 145,858 juta unit, dengan jumlah tersebut menunjukkan peningkatan sebesar 5,27% dari tahun sebelumnya. Semakin meningkatnya jumlah kendaraan mengakibatkan kepadatan lalu lintas pun terus meningkat dan menjadi salah satu faktor penyebab kemacetan. Kemacetan lalu lintas ini menimbulkan banyak kerugian diantaranya kerugian waktu yang terbuang di jalan, menguras energi, dan membuat bahan bakar kendaraan menjadi lebih boros, oleh karena itu kemacetan ini harus diatasi.

Saat ini pengaturan lalu lintas yang diterapkan adalah dengan penetapan durasi nyala lampu merah, kuning, dan hijau yang baku. Pengaturan ini berlaku sepanjang hari, sementara jumlah arus lalu lintas tidaklah sama setiap waktu,

sehingga diperlukan suatu sistem yang dapat mengendalikan durasi nyala lampu hijau pada lalu lintas yang dapat disesuaikan dengan kepadatan kendaraan yang terjadi.

Pada penelitian ini alternatif solusi yang mungkin dilakukan adalah dengan menggunakan penggunaan aturan logika *fuzzy* pada sistem pengaturan lampu lalu lintas. Perancangan sistem pengaturan lampu lalu lintas baru yang berbasis logika *fuzzy* ini digunakan dengan menggunakan metode *Fuzzy Inference System Tsukamoto* untuk menentukan jumlah durasi lampu hijau, sedangkan untuk mendeteksi objek berupa kendaraan (mobil), menggunakan kamera *Smartphone* dengan mengambil gambar dari replika jalan yang sudah dirancang. Gambar yang telah diambil akan diproses dengan metode Pelabelan Objek. Dengan pengaturan lampu lalu lintas berbasis logika *fuzzy* dapat diketahui kondisi dari tiap-tiap jalan yang memungkinkan pemberian durasi lampu yang berubah sesuai dengan keadaan dan kebutuhan dari tiap jalan berdasarkan konsep sistem inferensi *fuzzy*

2. LANDASAN TEORI

2.1 Simulasi

[10] Simulasi adalah suatu metodologi untuk melaksanakan percobaan dengan menggunakan model dari satu sistem nyata. Menurut [3] simulasi merupakan suatu model pengambilan keputusan dengan mencontoh atau mempergunakan gambaran sebenarnya dari suatu sistem kehidupan dunia nyata tanpa harus mengalaminya pada keadaan yang sesungguhnya.

2.2 Lampu Lalu Lintas

Menurut penjelasan UU Lalu Lintas No. 14 tahun 1992 pasal 8 ayat 1 huruf C menyebutkan bahwa "Pengertian alat pemberi isyarat lalu lintas adalah peralatan teknis berupa isyarat lampu yang dapat di lengkapi dengan bunyi untuk memberi peringatan atau mengatur lalu lintas orang atau kendaraan di persimpangan, persilangan sebidang ataupun pada arus jalur". Jadi lampu lalu lintas dapat diartikan sebagai lampu yang digunakan untuk mengatur kelancaran lalu lintas di suatu persimpangan jalan dengan cara memberi kesempatan pengguna jalan dari masing-masing arah untuk berjalan secara pergantian. Pada setiap lampu lalu lintas terdapat 3 buah lampu yang berwarna merah, kuning, hijau. Merah untuk berhenti, kuning berarti hati-hati, sedangkan hijau berarti jalan.

2.3 Logika Fuzzy

Logika fuzzy pertama kali diperkenalkan oleh Prof. Lotfi A. Zadeh dari Universitas California, pada tahun 1965. Logika fuzzy merupakan suatu metode pengambilan keputusan berbasis aturan yang digunakan untuk memecahkan keabu-abuan masalah pada sistem yang sulit dimodelkan atau memiliki ambiguitas. Dasar logika fuzzy adalah teori himpunan fuzzy. Dalam logika fuzzy nilai kebenaran suatu pernyataan berkisar sepenuhnya benar sampai dengan sepenuhnya salah.[7]



Gambar 2.1 Diagram Blok 'Logika Fuzzy Sebagai Blank Blok'

Logika fuzzy dapat dianggap sebagai kotak hitam yang berhubungan antara ruang input menuju ruang output. Kotak hitam yang dimaksudkan adalah metode yang dapat digunakan untuk mengolah data input menjadi output dalam bentuk informasi yang baik.

Ada beberapa hal yang perlu di ketahui dalam sistem fuzzy, yaitu:

1. Variabel Fuzzy

Variabel fuzzy merupakan variabel yang hendak dibahas dalam suatu sistem fuzzy. Contoh : umur, temperatur, permintaan, dan lain-lain.

2. Himpunan fuzzy

Himpunan fuzzy merupakan suatu grup yang mewakili suatu kondisi atau keadaan tertentu dalam suatu variabel fuzzy. Contoh variabel temperatur terbagi menjadi 5 himpunan fuzzy, yaitu : Dingin, Sejuk, Normal, Hangat, dan Panas. Himpunan fuzzy merupakan suatu group yang mewakili suatu kondisi atau keadaan tertentu dalam suatu variabel fuzzy. Himpunan fuzzy disebut himpunan tegas (crisp), nilai keanggotaan suatu item x dalam suatu himpunan A yang dituliskan dengan $\mu_A[x]$, dimana memiliki dua buah kemungkinan nilai yaitu:

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

Himpunan fuzzy memiliki dua atribut yaitu:

- Linguistik, merupakan penamaan grub yang mewakili suatu keadaan atau kondisi tertentu dengan menggunakan bahasa alami / sehari-hari. Contohnya : Muda, Parobaya, Tua
- Numeris, merupakan suatu nilai angka yang menunjukkan ukuran dari suatu variabel. Contohnya : 20, 40, 80

3. Semesta pembicaraan

Semesta pembicaraan adalah keseluruhan nilai yang diperbolehkan untuk dioperasikan dalam satu variabel fuzzy. Semesta pembicaraan merupakan himpunan bilangan real yang senantiasa bertambah secara monoton dari kiri ke kanan. Nilai semesta pembicaraan dapat berupa bilangan positif maupun negatif. Contoh semesta pembicaraan untuk variabel IPK: $[0, 4]$:

4. Domain

Domain himpunan fuzzy adalah keseluruhan nilai yang diizinkan dalam semesta pembicaraan dan boleh dioperasikan dalam suatu himpunan fuzzy. Seperti halnya semesta pembicaraan, domain merupakan bilangan real yang senantiasa naik (bertambah) secara monoton dari kiri ke kanan. Nilai domain dapat berupa bilangan positif maupun negatif. Contoh domain bilangan fuzzy:

- Kurang = $[0, 3]$
- Cukup = $[2.5, 3.5]$
- Baik = $[3, 4]$

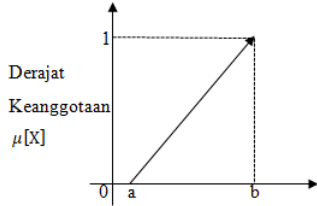
2.4 Fungsi Keanggotaan

Fungsi keanggotaan fuzzy adalah suatu kurva yang menentukan pemetaan titik-titik input data ke dalam derajat keanggotaanya yang nilainya berkisar antara 0 hingga 1. Beberapa fungsi keanggotaan fuzzy, yaitu:

a. Representasi Linear

Representasi linear adalah pemetaan input ke derajat keanggotaannya digambarkan sebagai suatu garis lurus. Pada representasi linear terdapat 2 kemungkinan, yaitu:

Pertama, kenaikan himpunan dimulai pada nilai domain yang memiliki derajat keanggotaan nol (0) bergerak ke kanan menuju ke nilai domain yang memiliki derajat keanggotaan lebih tinggi. Representasi linear naik dapat dilihat pada gambar 2.2.

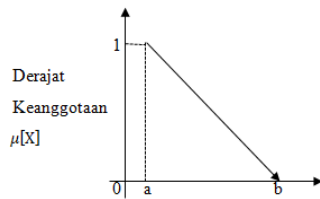


Gambar 2.2 Representasi Linear Naik

Fungsi Keanggotaan Naik:

$$\mu[x] = \begin{cases} 0; & , x \leq a \\ \frac{x - a}{b - a}; & , a \leq x \leq b \\ 1; & , x \geq b \end{cases} \quad (2.1)$$

Kedua, merupakan kebalikan yang pertama. himpunan dimulai dari nilai domain dengan derajat keanggotaan tertinggi pada sisi kiri, kemudian bergerak menurun ke nilai domain yang memiliki derajat keanggotaan lebih rendah. Representasi kurva linear turun dapat dilihat pada gambar 2.3.



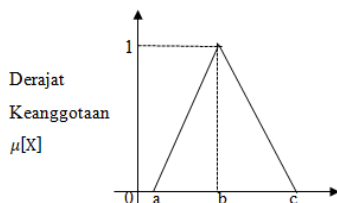
Gambar 2.3 Representasi Linear Turun

Fungsi keanggotaan Linear Turun:

$$\mu[x] = \begin{cases} \frac{b - x}{b - a}; & , a \leq x \leq b \\ 0; & , x \geq b \end{cases} \quad (2.2)$$

b. Representasi Kurva Segitiga

Kurva segitiga pada dasarnya terbentuk dari gabungan antara 2 garis (linear). Representasi kurva segitiga dapat dilihat pada gambar 2.4.



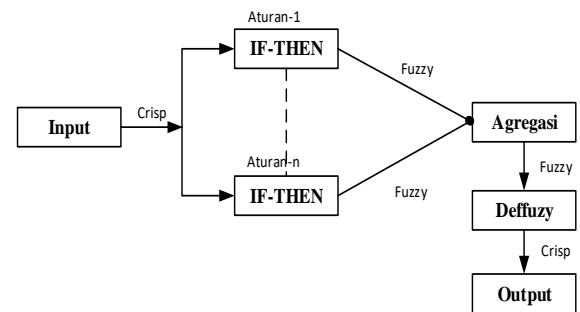
Gambar 2.4 Representasi Kurva Segitiga

Fungsi Keanggotaan Kurva Segitiga:

$$\mu[x] = \begin{cases} 0; & , x \leq a \text{ atau } x \geq c \\ \frac{x - a}{b - a}; & , a \leq x \leq b \\ \frac{b - x}{c - b}; & , b \leq x \leq c \end{cases} \quad (2.3)$$

2.5 Sistem Inferensi Fuzzy

Sistem Inferensi Fuzzy merupakan suatu kerangka komputasi yang didasarkan pada teori himpunan fuzzy, aturan fuzzy berbentuk IF-THEN dan penalaran fuzzy. Secara garis besar, diagram blok proses inferensi fuzzy.[10]



Gambar 2.5 Diagram Blok Sistem Inferensi Fuzzy

Sistem inferensi fuzzy menerima input crisp. Input ini kemudian dikirim ke basis pengetahuan yang berisi n aturan fuzzy dalam bentuk IF-THEN. Fire strength akan dicari pada setiap aturan. Apabila jumlah aturan lebih dari satu, maka akan dilakukan agregasi dari semua aturan. Selanjutnya, pada hasil agregasi akan dilakukan defuzzy untuk mendapatkan nilai crisp sebagai output sistem.

2.6 Metode Fuzzy Tsukamoto

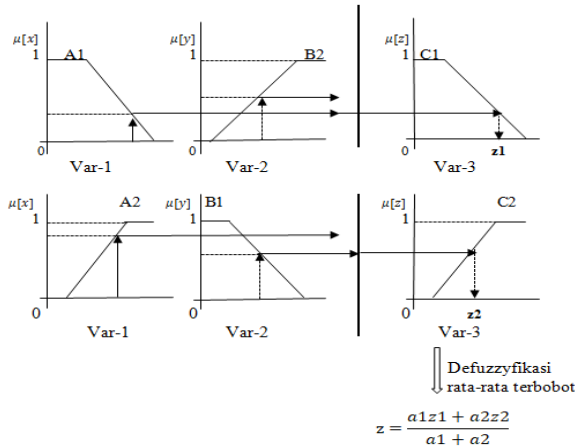
Menurut [9] pada metode fuzzy inference system Tsukamoto implikasi setiap aturan berbentuk implikasi “Sebab = Akibat” / implikasi “Input = Output” dimana antara anteseden dan konsekuen harus ada hubungannya. Setiap aturan dipresentasikan menggunakan himpunan-himpunan fuzzy, dengan fungsi keanggotaan yang monoton dan seleksi ciri. Kemudian untuk menentukan hasil tegas (Crisp Solution) digunakan rumusan penegasan (defuzzyfikasi) yang disebut center average defuzzyfie (CAD).

Misalkan ada 2 variabel input, var-1 (x) dan var-2(y), serta variabel output, var-3(z), dimana var-1 terbagi atas 2 himpunan yaitu A1 dan A2, var-2 terbagi atas 2 himpunan B1 dan B2, var-3 juga terbagi atas 2 himpunan yaitu C1 dan C2 (C1 dan C2 harus monoton). Ada 2 aturan yang digunakan, yaitu :

[R1] IF (x is A1) and (y is B2) THEN (z is C1)

[R2] IF (x is A2) and (y is B1) THEN (z is C2)

Pertama-tama dicari fungsi keanggotaan dari masing-masing himpunan *fuzzy* dari setiap aturan, yaitu himpunan A1, B2 dan C1 dari aturan *fuzzy* [R1], dan himpunan A2, B1 dan C2 dari aturan *fuzzy* [R2]. Aturan *fuzzy* R1 dan R2 dapat direpresentasikan dalam gambar 2.6 untuk mendapatkan suatu nilai *crisp* Z.



Gambar 2. 6 Inferenai menggunakan Metode Tsukamoto

Karena pada metode Tsukamoto operasi himpunan yang digunakan adalah konjungsi (AND), maka nilai keanggotaan anteseden dari aturan *fuzzy* [R1] adalah irisan dari nilai keanggotaan A1 dan var-1 dengan nilai keanggotaan B1 dan var-2. Menurut teori operasi himpunan pada persamaan 2.4.

$$\mu A \cap B = \mu A(X) \cap \mu B(Y) = \min(\mu A(X), \mu B(Y)) \quad (2.4)$$

Maka nilai keanggotaan anteseden dari operasi konjungsi (AND) dari aturan *fuzzy* [R1] adalah nilai minimum antara lain keanggotaan A1 dari var-1 dan nilai keanggotaan B2 dari var-2 dan anteseden dari aturan *fuzzy* [R2] adalah nilai minimum antara nilai keanggotaan A2 dari var-1 dengan nilai keanggotaan B1 dari var-2. Selanjutnya, nilai keanggotaan dari aturan *fuzzy* [R1] dan [R2] masing-masing disebut dengan α_1 dan α_2 . Nilai α_1 dan α_2 kemudian disubsitusikan pada fungsi keanggotaan himpunan C1 dan C2 sesuai aturan *fuzzy* [R1] dan [R2] untuk memperoleh nilai z_1 dan z_2 , yaitu nilai z (nilai perkiraan produksi) untuk aturan *fuzzy* [R1] dan [R2]. Untuk memperoleh nilai *output crisp* / nilai tegas Z, dicari dengan cara mengubah *input* (berupa himpunan *fuzzy* yang diperoleh dari komposisi aturan-aturan *fuzzy*) menjadi suatu bilangan pada domain himpunan *fuzzy* tersebut. Cara ini disebut dengan metode *defuzzyfikasi* (penegasan). Metode *defuzzyfikasi* yang digunakan dalam metode Tsukamoto adalah metode *defuzzyfikasi* rata-rata

terpusat (*Center Average Defuzzyfie*) yang dirumuskan sebagai:

$$Z = \frac{\sum_{i=1}^n \alpha_i z_i}{\sum_{i=1}^n \alpha_i} \text{ (Defuzzyfikasi rata-rata terpusat) [7]}$$

2.7 Citra

Dalam bukunya Dasar Pengolahan Citra dengan Delphi [5] menjelaskan bahwa citra adalah gambaran visual mengenai suatu objek atau beberapa objek. Citra terbagi atas 2 jenis, yaitu citra analog dan citra digital. Citra analog ialah citra yang bersifat kontinu, biasa dijumpai pada kertas atau media lain seperti film rontgen, televisi dan hasil CT Scan. Citra analog tidak dapat di proses oleh komputer secara langsung, sehingga diperlukan proses konversi analog ke digital. Citra digital adalah citra yang dinyatakan dalam kumpulan data digital dan dapat diproses oleh komputer. Citra digital direpresentasikan dengan matriks berukuran N baris dan M kolom dan masing-masing elemen pada citra digital disebut piksel/pixel. Pada penelitian ini penyebutan citra pada pembahasan selanjutnya merujuk kepada citra digital.

a. Citra Berwarna(RGB)

Citra berwarna atau RGB tersusun atas tiga komponen, yaitu komponen merah (R atau *Red*), hijau (G atau *Green*) dan biru (B atau *Blue*). Setiap piksel akan diwakili tiga komponen tersebut. Setiap komponen warna menggunakan delapan bit (nilainya berkisaran antara 0 sampai dengan 255), kemungkinan warna yang dapat disajikan mencapai $255 \times 255 \times 255$ atau 16.581.475 warna.

b. Citra Berskala Keabuan (Grayscale)

Citra yang menggunakan gradasi warna abu-abu yang merupakan kombinasi antara hitam dan putih. Setiap warna dalam citra *grayscale* dinyatakan dengan sebuah nilai bulat antara 0-255 (untuk yang aras keabuan sama dengan 256) dan nilai tersebut disebut sebagai intensitas. Dalam pengolahan citra, citra RGB seringkali dikonversi terlebih dahulu ke citra *grayscale*. Kemudian melalui citra *grayscale* inilah dilakukan pemrosesan. Misalnya untuk memperoleh tekstur objek.

c. Citra Biner (Hitam-Putih)

Citra yang nilai pikselnya berupa angka nol atau satu saja atau dua keadaan seperti 0 dan 255. Kata *biner* yang berarti dua menyatakan dua kemungkinan nilai tersebut. Citra seperti ini biasanya dipakai untuk kepentingan segmentasi, yang memisahkan objek dari latar belakangnya.

2.8 Preprocessing

Preprocessing adalah proses awal dilakukannya perbaikan citra sebelumnya dilakukan proses lebih lanjut terhadap citra masukan. *Image processing* bertujuan supaya meningkatkan

kemungkinan keberhasilan pada tahap pengolahan lebih lanjut pada sebuah citra. *Preprocessing* merupakan suatu proses untuk menghilangkan bagian-bagian yang tidak diperlukan pada gambar *input* untuk proses selanjutnya [2]. Dalam pengolahan citra, citra berwarna seringkali harus dikonversi ke dalam bentuk citra berskala keabuan (*Grayscale*) atau citra biner (hitam-putih). Mengingat kedua citra ini yang mudah diinterpretasikan [6]. Berikut beberapa *preprocessing* yang dilakukan:

2.8.1 Resize

Resize adalah proses mengubah resolusi atau mengubah ukuran horizontal dan vertikal citra

2.8.2 Grayscale

Untuk mengubah citra berwarna (RGB) menjadi citra skala abu-abu (*Grayscale*) menggunakan rumus sebagai berikut[9]:

$$CG = (R * 0,2989) + (G * 0,5870) + (B * 0,1141) \quad (2.5)$$

Keterangan:

CG = Citra *Grayscale*

R = nilai piksel *red* (merah)

G = nilai piksel *green* (hijau)

B = nilai piksel *blue* (biru)

2.8.3 Thresholding

Threshold merupakan suatu cara bagaimana mempertegas citra dengan mengubah citra menjadi hitam dan putih (nilainya hanya tinggal menjadi antara 0 dan 1) [11]. Pada operasi ini nilai piksel yang memenuhi syarat ambang batas dipetakan ke suatu nilai yang dikehendaki. Dalam hal ini syarat ambang batas dan nilai yang dikehendaki disesuaikan dengan kebutuhan.

Contoh perhitungan digital:

Misal diketahui citra grayscale 256 warna dengan ukuran 6 x 6 piksel pada Gambar 2.7.

40	60	80	84	75	100
55	79	90	89	34	87
37	69	97	80	56	90
150	70	75	50	70	60
105	160	59	45	32	20
40	60	245	205	79	77

Gambar 2.7 Contoh Nilai Citra *Grayscale*

Akan dilakukan operasi ambang tunggal dengan fungsi berikut

$$f_0(x, y) = \begin{cases} 0, & f_i(x, y) < 80 \\ 255, & f_i(x, y) \geq 80 \end{cases} \quad (2.6)$$

Maka perhitungan digital yang dilakukan adalah sebagai berikut: setiap nilai intensitas citra

grayscale yang nilainya < 80, diubah menjadi 0, sedangkan untuk setiap nilai intensitas citra *grayscale* yang nilainya ≥ 80 diubah menjadi 255.

40	60	80	84	75	100	0	0	255	255	0	255
55	79	90	89	34	87	0	0	255	255	0	255
37	69	97	80	56	90	0	0	255	255	0	255
150	70	75	50	70	60	255	0	0	0	0	0
105	160	59	45	32	20	255	255	0	0	0	0
40	60	245	205	79	77	0	0	255	255	0	0

Gambar 2.8 Contoh Nilai Citra *Grayscale* Setelah diubah

2.8.4 Pelabelan Objek

Menurut [6] pelabelan terhadap objek sesungguhnya berupa tindakan untuk memberikan label berbeda (berupa nomor) pada setiap objek sehingga dapat menghitung jumlah objek. Pemrosesannya dapat dilaksanakan pada citra *biner*. Adapun ketentuan yang digunakan dalam proses pelabelan objek sebagai berikut:

$$B(y, x) = \begin{cases} 0 \\ 1 \\ 2, 3, \dots \end{cases} \quad (2.7)$$

Keterangan :

B : Banjiri atau piksel-piksel yang berdekatan

(y-x) : Baris dan Kolom

0 : Piksel latar belakang sebagai *background*

1 : Piksel latar depan sebagai objek

2,3...n : Label pada Objek

Pada penelitian ini menggunakan model 4-ketetanggaan (4-konektivitas) secara *rekursif*, adapun rumus yang digunakan dalam model 4-ketetanggaan (4-konektivitas) secara *rekursif* sebagai berikut:

	B(x, y-1, label)	
B(x-1, y, label)	B(x, y, label)	B(x, y+1, label)
	B(x+1, y, label)	

Gambar 2.9 Rumus 4-Ketetanggaan Secara *Rekursif*

Keterangan:

B : Banjiri

B(y, x, label) : Banjiri (baris, kolom dan nilai label pada objek)

B(x-1, y, label) : Banjiri (baris-1, kolom, label)

B(x+1, y, label) : Banjiri (baris+1, kolom, label)

B(x, y-1, label) : Banjiri (baris-1, kolom-1, label)

B(x, y+1, label) : Banjiri (baris, kolom+1, label)

Piksel-piksel yang berdekatan dikatakan memiliki hubungan 4-konektivitas atau 4-ketetanggaan jika piksel-piksel tersebut terletak berdampingan secara horisontal dan vertikal. Dengan

menggunakan pendekatan 4-ketetanggan atau 4-konektivitas, pembanjiran secara *rekursif* dapat dilakukan melalui langkah-langkah berikut:

1. Cek nilai piksel pertama pada citra masukan yaitu baris 1 kolom 1.
2. Apabila nilai tersebut 0 (*background*), lewati dan pindah ke kolom selanjutnya pada baris yang sama.
3. Apabila piksel bernilai 1, cek nilai tetangganya.
 - Apabila tetangga terdekatnya (*neighbour*) belum terlabel, beri label pada piksel yang ditinjau dengan label 2.
 - Apabila memiliki satu tetangga terdekat yang sudah terlabel, beri label yang sama dengan label tetangga.
 - Apabila memiliki lebih dari satu tetangga terdekat dengan label berbeda, beri label menggunakan label terkecil dari tetangga-tetangga tersebut.
 - Setiap selesai melakukan pelabelan dan bertemu kembali dengan *background*, perbarui indeks label.
4. Lakukan proses ini sampai keseluruhan baris dan kolom dicek.

2.9 Database

Menurut [12] *database* Atau memiliki istilah basis data merupakan suatu kumpulan data yang saling berhubungan dan berkaitan dengan subjek tertentu pada tujuan tertentu pula, hubungan antardata ini dapat dilihat oleh adanya field ataupun kolom.

Sedangkan menurut [8] *database* itu didefinisikan sebagai kumpulan data yang terintegrasi dan diatur sedemikian rupa sehingga data tersebut dapat dimanipulasi, diambil, dan dicari secara cepat.

2.10 Data Flow Diagram (DFD)

Menurut [4] (DFD) merupakan diagram yang menggunakan notasi-notasi untuk menggambarkan arus data sistem secara logika. DFD merupakan dokumentasi grafik yang menggunakan simbol penomoran di dalam mengilustrasikan arus data yang saling berhubungan diantara pemrosesan data untuk diubah menjadi informasi.

2.11 Flowchart

Menurut [13] *flowchart* merupakan langkah awal pembuatan sebuah program. Dengan adanya *flowchart* aturan setiap kegiatan akan menjadi lebih jelas. Setelah *flowchart* selesai disusun, selanjutnya programmer menerjemahkannya ke bentuk program dengan bahasa pemrograman.

3. METODOLOGI PENELITIAN

3.1 Data dan Bahan

3.1.1 Replika Jalan

Pada penelitian ini bahan yang digunakan untuk tiruan jalan pada saat pengambilan gambar yang akan digunakan sebagai data masukan pada aplikasi simulasi pengaturan lampu lalu lintas yaitu dengan membangun replika jalan dan mobil mainan yang mana penyusunannya dibuat mirip dengan kondisi aslinya. Sedangkan untuk mensimulasikan atau menjalankan aplikasi lalu lintas dengan membuat animasi yang diterapkan pada *platform* desktop komputer.



Gambar 3.1 Hasil Replika Ruas Jalan

3.1.2 Data Penelitian

Proses pengumpulan data merupakan sebuah masukan nilai lebar jalur atau replika gambar dari kendaraan yang berhenti di jalan raya. Berikut proses data-data yang dikumpulkan untuk penelitian simulasi pengaturan lampu lalu lintas.

a. Data Gambar Kendaraan

Data penelitian yang digunakan yaitu hasil pengambilan gambar atau foto dengan kamera *smartphone* dengan format yang digunakan *joint Photographic Expert Group Image* atau jpeg. Pada tahap ini peneliti menggunakan replika atau tiruan jalan raya dan kendaraan mainan dengan ukuran panjang 60 cm dan lebar 20 cm foto. Proses pengambilan data gambar menggunakan sudut kemiringan 14° dengan pencahayaan yang cukup. Sampel data gambar kendaraan yang digunakan pada penelitian dapat dilihat pada Gambar 3.2.

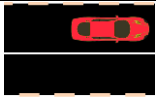
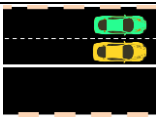



Gambar 3.2 Data Gambar Kendaraan

b. Lebar Jalur

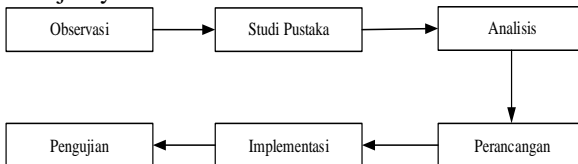
Lebar jalur yang dimasukkan nilainya ditentukan berdasarkan pengamatan banyaknya banjar yang ada di jalan raya. Nilai jalur yang dimasukkan yaitu 1 sampai 3 jalur, ilustrasi lebar jalur ditunjukkan pada Tabel 3.1.

Tabel 3.1 Lebar Jalur yang digunakan

Lebar Jalur	Gambar Ilustrasi
1	
2	
3	

3.2 Metode Penelitian

Metode penelitian merupakan teknik atau cara-cara yang digunakan untuk mengumpulkan informasi serta menganalisis informasi yang telah didapatkan. Metode penelitian memberikan gambaran tentang rancangan penelitian yang meliputi langkah-langkah yang harus ditempuh, sumber data, langkah-langkah dalam memproses data yang sudah didapatkan, dan langkah-langkah memproses data selanjutnya.



Gambar 3.3 Metode Penelitian

3.2.1 Observasi

Observasi yang dilakukan untuk penelitian pembuatan aplikasi simulasi pengaturan lampu lalu lintas berdasarkan kepadatan kendaraan menggunakan metode Pelabelan objek dan *Fuzzy Inference System Tsukamoto* adalah dengan melakukan observasi pengamatan cara kerja lampu lalu lintas di suatu perempatan. Observasi tersebut memberikan kesimpulan bahwa cara kerja lampu lalu lintas disuatu perempatan yaitu dengan lama waktu lampu hijau yang sudah ditetapkan sepanjang waktu.

3.2.2 Studi Pustaka

Studi pustakka dilakukan untuk mendapatkan teori acuan, informasi dan referensi dalam kegiatan penelitian. Sehingga dapat digunakan sebagai pengetahuan dasar dalam pengolahan citra, algoritma pelabelan objek dan perhitungan *Fuzzy Inference System Tsukamoto*.

3.2.3 Analisis

Kebutuhan pada penelitian ini mengikuti tujuan penelitian sehingga diharapkan data sebagai bahan mentah dapat dijadikan sumber yang valid. Untuk memperoleh data pada penelitian ini dibutuhkan program dan replica atau alat peraga jalan dan mobil mainan. Program digunakan untuk mengolah data simulasi, menghitung durasi lampu hijau dengan metode *Fuzzy Inference System Tsukamoto* dan menjalankan simulasi animasi lampu lalu lintas. Selanjutnya gambar yang sudah dikumpulkan dari hasil replica atau alat peraga berupa jalanan dan mobil mainan digunakan sebagai data masukan dan akan diproses untuk menghasilkan jumlah kendaraan atau banyak objek.

3.2.4 Perancangan

Perancangan pada penelitian ini meliputi beberapa tahapan proses, yaitu rancangan terhadap alur kerja program atau *flowchart*, DFD, rancangan terhadap *preprocessing* citra, pengolahan citra/segmentasi citra, ekstraksi fitur citra dan rancangan antarmuka simulai lampu lalu lintas berdasarkan kepadatan dengan metode FIS *Tsukamoto*.

3.2.5 Implementasi

Implementasi dilakukan dengan menggunakan bahasa pemrograman Java dan basis data MySQL. Bahasa pemrograman java dikenal sebagai *Object Oriented Programming (OOP)* bertujuan untuk mempermudah pengembangan program. Sedangkan MySQL digunakan untuk menyimpan data hasil simulasi atau perhitungan dengan metode yang digunakan

3.2.6 Pengujian

Aplikasi simulasi lampu lalu lintas yang telah selesai dibangun kemudian dilakukan pengujian untuk mengetahui apakah hasil analisis telah sesuai dngan yang diharapkan atau belum. Pengujian dilakukan dengan menggunakan sistem simulasi pada animasi yang telah dibuat dan pengujian tingkat keakurasian sistem dalam memprediksi kepadatan kendaraan/jumlah kendaraan. Selain itu pengujian juga dilakukan menggunakan cara *desk checking* atau melakukan perhitungan manual dan dicocokkan dengan *output* yang diberikan oleh program.

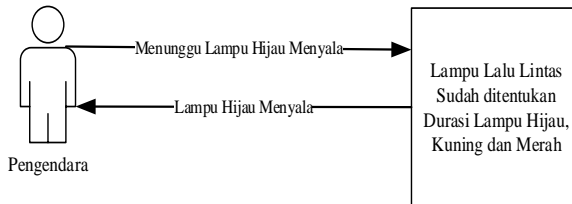
4. HASIL DAN PEMBAHASAN

4.1 Analisis Sistem

4.1.1 Analisis Sistem yang Sedang Berjalan

Selama ini pengaturan lalu lintas yang digunakan hanya berorientasi terhadap waktu dengan menentukan terlebih dahulu siklus waktunya dan

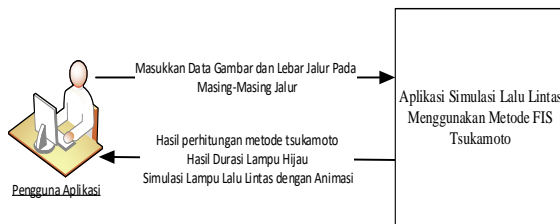
tidak menyesuaikan kondisi lalu lintas. Misalnya nyala lampu merah dan hijau akan berganti setiap 30 detik. Durasi dari sistem ini dirasa punya kelemahan, yaitu mengabaikan keadaan jalan yang diaturnya. Dengan begitu bisa saja selama durasi nyala lampu hijau jarang kendaraan yang melintas. Sementara lalu lintas yang sedang mendapat lampu merah kendaraan amat padat dan harus menunggu giliran lampu hijau yang masih mengatur (jalan kosong). Seperti yang terlihat pada gambar 4.1.



Gambar 4.1 Sistem yang Sedang Berjalan

4.1.2 Analisis Sistem yang Diusulkan

Sistem yang diusulkan untuk simulasi lampu lalu lintas ini memiliki konsep bahwa durasi lama lampu hijau tidaklah statis atau tetap, melainkan dinamis, yaitu nyala lampu hijau pada sistem lalu lintas ini berdasarkan jumlah kendaraan pada masing-masing jalur dan lebar jalur. Simulasi lampu lalu lintas ini menggunakan dua parameter *input* yaitu banyaknya jumlah kendaraan dan lebar masing-masing jalur dengan memanfaatkan metode FIS Tsukamoto untuk menentukan durasi lampu hijau pada masing-masing jalur. Dengan pengaturan lampu lalu lintas berbasis logika *fuzzy* dapat diketahui kondisi dari tiap-tiap jalan yang memungkinkan pemberian layanan lampu yang berubah sesuai dengan keadaan dan kebutuhan dari tiap jalan. Seperti yang terlihat pada gambar 4.2.



Gambar 4.2 Sistem yang Diusulkan

4.2 Penyelesaian dengan Fuzzy Inference System Tsukamoto

Di dalam model base ini, terdapat tiga langkah untuk menentukan durasi lama lampu hijau yaitu pembentukan himpunan *fuzzy* (*fuzzyfikasi*), inferensi aturan dan penegasan (*defuzzyfikasi*).

4.2.1 Pembentukan Himpunan Fuzzy

Tahap pertama untuk melakukan perhitungan membuat himpunan *fuzzy* (*fuzzyfikasi*) pada masing-masing variabel. Terdapat dua variabel *input* yang dibuat fungsi keanggotaannya, yaitu banyak kendaraan pada jalur yang diatur dan lebar jalur yang diatur. Sedangkan untuk durasi lampu hijau sebagai keluarannya atau *output*. Himpunan *fuzzyfikasi* dapat dilihat pada tabel 4.1.

Tabel 4. 1 Himpunan Fuzzy Masing-Masing Variabel

Fungsi	Variabel	Himpunan Fuzzy	Domain
Input	Banyak Kendaraan	Sepi (S)	[0 ; 0 ; 6 ; 13]
		Normal (N)	[6 ; 13 ; 20]
		Padat (P)	[13 ; 20 ; 20 ; 20]
	Lebar Jalur	Sempit (Se)	[0 ; 0 ; 2 ; 4]
Lebar (L)		[2 ; 4 ; 4 ; 4]	
Output	Durasi Lampu Hijau	Cepat (C)	[0 ; 0 ; 20 ; 40]
		Sedang (Sd)	[20 ; 40 ; 60]
		Lama (Lm)	[40 ; 60 ; 60 ; 60]

4.2.2 Fungsi Keanggotaan Nilai

Fungsi derajat linear turun digunakan untuk himpunan SEPI variabel banyak kendaraan, dan himpunan SEMPIT variabel lebar jalur. Fungsi kurva segitiga digunakan untuk himpunan NORMAL variabel banyak kendaraan, dan fungsi linear naik digunakan untuk himpunan PADAT variabel banyak kendaraan, dan himpunan LEBAR variabel lebar jalur. Masing-masing linear digunakan untuk mencari nilai *fuzzyfikasi*.

a. Fungsi keanggotaan himpunan *fuzzy* SEPI dari variabel Banyak Kendaraan dapat dicari dengan fungsi berikut:

$$\mu_{MB \text{ Sepi}} [X] = \begin{cases} 1; & , X \leq 6 \\ \frac{13-X}{13-6}; & , 6 \leq X \leq 13 \\ 0; & , X \geq 13 \end{cases} \quad (4.1)$$

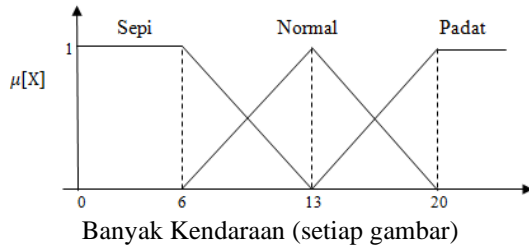
b. Fungsi keanggotaan himpunan *fuzzy* NORMAL dari variabel Banyak Kendaraan dapat dicari dengan fungsi berikut:

$$\mu_{MB \text{ Normal}} [X] = \begin{cases} 0; & , X \leq 6 \text{ atau } X \geq 20 \\ \frac{X-6}{13-6}; & , 6 \leq X \leq 13 \\ \frac{20-X}{20-13}; & , 13 \leq X \leq 20 \\ 1; & , X = 13 \end{cases}$$

- c. Fungsi keanggotaan himpunan *fuzzy* PADAT dari variabel Banyak Kendaraan dapat dicari dengan fungsi berikut:

$$\mu_{MB\ Padat} [X] = \begin{cases} 0; & , X \leq 13 \\ \frac{X-13}{20-13}; & , 13 \leq X \leq 20 \\ 1; & , X \geq 20 \end{cases} \quad (4.3)$$

Fungsi keanggotaan himpunan *fuzzy* SEPI, NORMAL, dan PADAT dari variabel Banyak Kendaraan dipresentasikan pada gambar 4.3.



Gambar 4.3 Fungsi Keanggotaan Himpunan *Fuzzy* SEPI, NORMAL dan PADAT

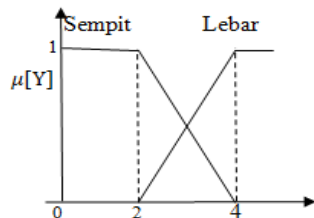
- d. Fungsi keanggotaan himpunan *fuzzy* SEMPIT dari variabel Lebar Jalur dapat dicari dengan fungsi berikut:

$$\mu_{LJ\ Sempit} [Y] = \begin{cases} 1; & , Y \leq 2 \\ \frac{4-Y}{4-2}; & , 2 \leq Y \leq 4 \\ 0; & , Y \geq 4 \end{cases} \quad (4.4)$$

- e. Fungsi keanggotaan himpunan *fuzzy* LEBAR dari variabel Lebar Jalur dapat dicari dengan fungsi berikut:

$$\mu_{LJ\ Lebar} [Y] = \begin{cases} 0; & , Y \leq 2 \\ \frac{Y-2}{4-2}; & , 2 \leq Y \leq 4 \\ 1; & , Y \geq 4 \end{cases} \quad (4.5)$$

Fungsi keanggotaan himpunan *fuzzy* SEMPI dan LEBAR dari variabel Lebar Jalur dipresentasikan pada gambar 4.4.



Gambar 4.4 Fungsi Keanggotaan Himpunan *Fuzzy* SEMPIT dan LEBAR

- f. Fungsi keanggotaan himpunan *fuzzy* CEPAT dari variabel Durasi Lampu Hijau dapat dicari dengan fungsi berikut:

$$\mu_{DL\ Cepat} [Z] = \begin{cases} 1; & , Z \leq 20 \\ \frac{40-Z}{40-20}; & , 20 \leq Z \leq 40 \\ 0; & , Z \geq 40 \end{cases} \quad (4.6)$$

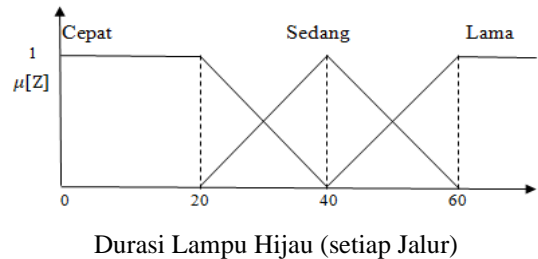
- g. Fungsi keanggotaan himpunan *fuzzy* SEDANG dari variabel Durasi Lampu Hijau dapat dicari dengan fungsi berikut:

$$\mu_{DL\ Sedang} [Z] = \begin{cases} 0; & , Z \leq 20 \text{ atau } Z \geq 60 \\ \frac{Z-20}{40-20}; & , 20 \leq Z \leq 40 \\ \frac{60-Z}{60-40}; & , 40 \leq Z \leq 60 \\ 1; & , Z = 40 \end{cases}$$

- h. Fungsi keanggotaan himpunan *fuzzy* LAMA dari variabel Durasi Lampu Hijau dapat di cari dengan fungsi berikut:

$$\mu_{DL\ Lama} [Z] = \begin{cases} 0; & , Z \leq 40 \\ \frac{Z-40}{60-40}; & , 40 \leq Z \leq 60 \\ 1; & , Z \geq 60 \end{cases} \quad (4.8)$$

Fungsi keanggotaan himpunan *fuzzy* CEPAT, SEDANG, dan LAMA dari variabel Durasi Lampu dipresentasikan pada gambar 4.5.



Gambar 4.5 Fungsi Keanggotaan Himpunan *Fuzzy* CEPAT, SEDANG dan LAMA

4.2.3 Rule

- [R1] IF Jumlah Mobil SEPI And Jalur SEMPIT THEN Durasi Lampu Hijau CEPAT
- [R2] IF Jumlah Mobil SEPI And Jalur LEBAR THEN Durasi Lampu Hijau CEPAT
- [R3] IF Jumlah Mobil NORMAL And Jalur SEMPIT THEN Durasi Lampu Hijau SEDANG
- [R4] Jumlah IF Mobil NORMAL And Jalur LEBAR THEN Durasi Lampu Hijau SEDANG
- [R5] IF Jumlah Mobil PADAT And Jalur SEMPIT THEN Durasi Lampu Hijau LAMA
- [R6] IF Jumlah Mobil PADAT And Jalur LEBAR THEN Durasi Lampu Hijau LAMA

4.3 Perancangan Sistem

Perancangan sistem aplikasi digunakan untuk menggambarkan aplikasi yang akan dibangun. Rancangan aplikasi simulasi lampu lalu lintas ada 3 yaitu data *flow diagram* (DFD), rancangan *database*,

dan rancangan antar muka:

4.3.1 Data Flow Diagram

Data flow diagram (DFD) adalah suatu model data atau proses yang dibuat untuk menggambarkan dari mana asal data dan kemana tujuan data yang keluar dari aplikasi, tempat data tersimpan dan proses apa yang dihasilkan serta keluarannya.

4.3.1.1 Context Diagram

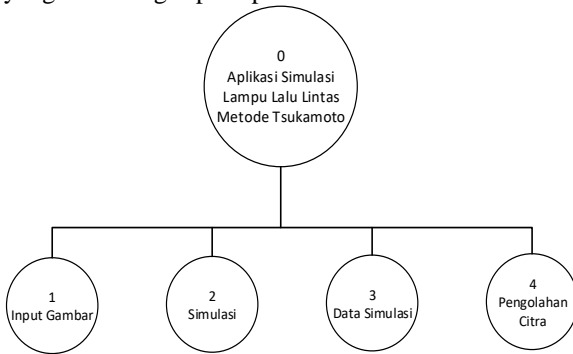
Context diagram yang dirancang hanya memiliki satu pengguna. Pada aplikasi terdapat input gambar, simulasi, data simulasi dan pengolahan citra. Context diagram yang dirancang seperti yang terlihat pada gambar 4.6.



Gambar 4.6 Context Diagram

4.3.1.2 Hierarchy Diagram

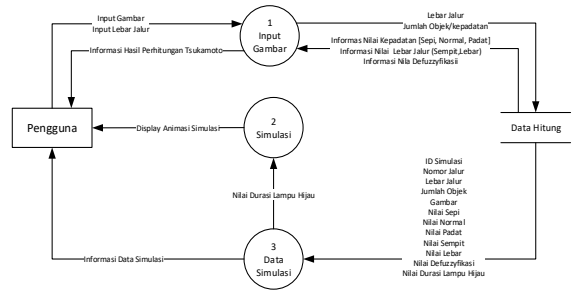
Hierarchy diagram merupakan diagram yang menggambarkan struktur dari aplikasi berupa suatu bagan berjenjang yang menggambarkan semua proses yang ada di aplikasi, dimana terdapat empat menu pada aplikasi yaitu input gambar, simulasi, data simulasi, dan pengolahan citra. Hierarchy diagram yang dirancang seperti pada Gambar 4.7.



Gambar 4.7 Hierarchy Diagram

4.3.1.3 Data Flow Diagram Level 1

Data flow diagram level 1 menggambarkan proses input gambar, simulasi dan data simulasi. Pada diagram ini sistem dijelaskan cara kerja keseluruhan aplikasi. Seperti dilihat pada gambar 4.8.



Gambar 4.8 Data Flow Diagram Level 1

4.4 Implementasi Program

Program dibangun dengan menggunakan NetBeans Ide sebagai alat untuk membuat program dan MySQL untuk database.

1. Tampilan Halaman Beranda

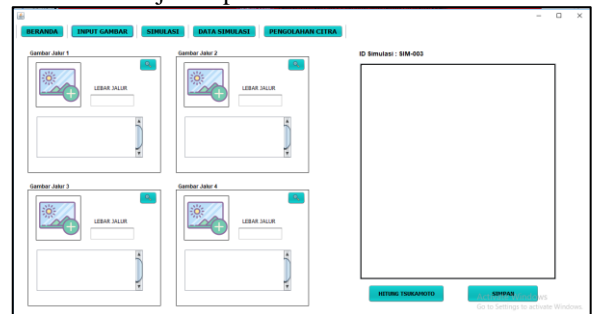
Halaman Beranda merupakan jendela pertama yang akan ditampilkan oleh aplikasi, saat menu Beranda aktif terdapat gambar dan judul aplikasi yang dibuat. Pada sisi atas jendela terdapat beberapa menu yaitu Input Gambar, Simulasi, Data Simulasi dan Pengolahan Citra. Halaman Beranda dapat dilihat pada Gambar 4.9.



Gambar 4.9 Halaman Menu Beranda

2. Tampilan Halaman Input Gambar

Halaman menu Input Gambar disini terdapat empat data jalur yaitu gambar jalur 1, gambar jalur 2, gambar jalur 3 dan gambar jalur 4 yang diartikan dalam empat persimpangan. Berikut menu Input Gambar ditunjukkan pada Gambar 4.10.



Gambar 4.10 Halaman Menu Input Gambar

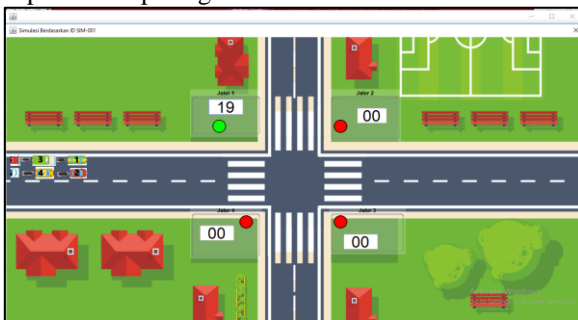
4. Tampilan Halaman Data Simulasi

Halaman Data Simulasi digunakan untuk menyimpan semua data yang telah diolah pada menu *Input Gambar*, kemudian disimpan yang nantinya bisa digunakan sebagai data informasi dari masing-masing jalur Simulasi. Dimana setiap ID Simulasi memiliki 4 jalur yang disimulasikan, setiap jalur terdapat beberapa data seperti nomor jalur, lebar jalur, banyak kendaraan, nilai sepi, nilai normal, nilai padat, nilai sempit, nilai lebar, nilai *defuzzyfikasi* dan nilai durasi. Masing-masing nilai diperoleh dari hasil perhitungan *fuzzy Tsukamoto*, nilai durasi di ambil dari nilai *defuzzyfikasi* yang telah dibulatkan ke atas untuk dijadikan detikan atau durasi lampu hijau. Berikut tabel penyimpanan pada menu data simulasi dapat dilihat pada Gambar 4.16.

No.	ID Simulasi	Nomor Jalur	Lebar Jalur	Banyak Kendaraan	Gambar	Sepi	Normal	Padat	Sempit	Lebar	DurasiKendaraan	Durasi
1	Sim-001	1	2	5	-	0.0	0.0	0.0	0.1	0.0	10.0	0.0
2	Sim-001	2	2	5	-	0.4	0.0	0.0	0.3	0.0	10.0	0.0
3	Sim-001	3	10	10	-	0.0	0.0	0.0	0.0	0.0	40.0	0.0
4	Sim-001	4	3	10	-	0.0	0.0	0.0	0.0	0.0	10.0	0.0
5	Sim-002	1	2	5	-	0.0	0.0	0.0	0.0	0.0	10.0	0.0
6	Sim-002	2	2	5	-	0.0	0.0	0.0	0.3	0.0	10.0	0.0
7	Sim-002	3	10	10	-	0.0	0.0	0.0	0.0	0.0	40.0	0.0
8	Sim-002	4	3	10	-	0.0	0.0	0.0	0.0	0.0	10.0	0.0

Gambar 4.16 Halaman Menu Data Simulasi

Apabila pengguna ingin menampilkan simulasi lampu lalu lintas berdasarkan ID Simulasi, pengguna tinggal menekan ID Simulasi yang terdapat pada tabel Data Simulasi. Berikut simulasi lampu lalu lintas yang akan ditampilkan berdasarkan ID Simulasi dapat dilihat pada gambar 5.9.

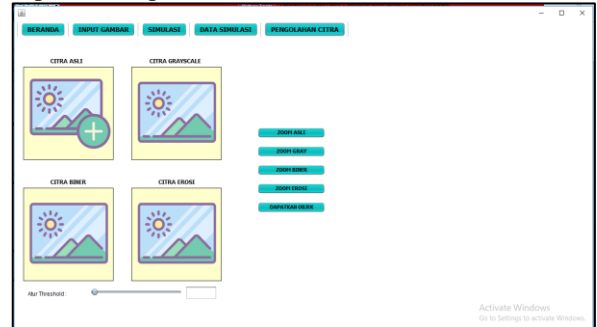


Gambar 4.17 Tampilan Simulasi Berdasarkan ID Simulasi

5. Tampilan Halaman Pengolahan Citra

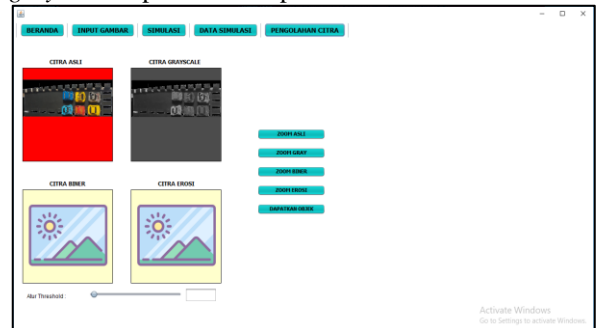
Halaman Pengolahan Citra digunakan untuk mengetahui proses pengolahan citra dari awal sampai akhir yaitu dimulai dari citra asli (RGB), citra grayscale (abu-abu), citra biner, citra erosi hingga pelabelan objek untuk mendapatkan nilai atau jumlah objek kendaraan pada suatu gambar. Selain itu data jumlah objek kendaraan akan digunakan sebagai

salah satu data masukan untuk perhitungan *fuzzy Tsukamoto* untuk menentukan durasi lampu hijau pada sebuah ruas jalur. Halaman pengolahan citra dapat dilihat pada Gambar 4.18.



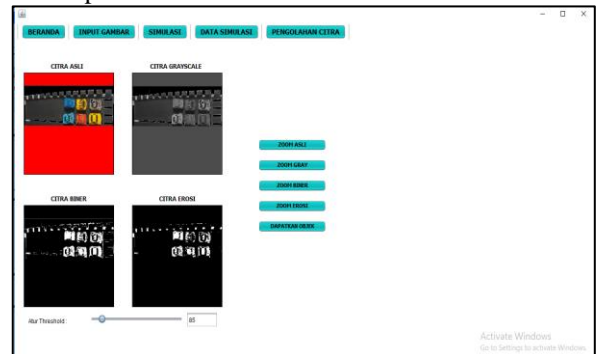
Gambar 4.18 Halaman Menu Pengolahan Citra

Pengguna memasukkan citra dengan cara klik icon pada kotak citra asli, kemudian memilih gambar mana yang ingin diolah. Setelah pengguna selesai memilih citra yang akan diolah, sistem secara langsung akan melakukan *resizing* dan *preprocessing* kemudian akan tampil gambar citra asli dan citra *grayscale* seperti terlihat pada Gambar 4.19.



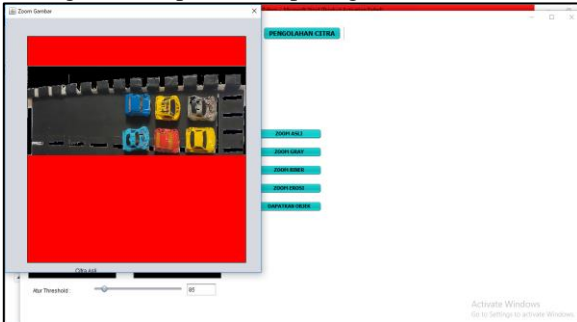
Gambar 4.19 Tampilan Setelah Memilih Gambar yang Akan diolah

Pengguna dapat mengatur *threshold* sesuai yang diinginkan, kemudian kotak dialog pada citra *biner* dan citra *erosi* akan menampilkan gambar sesuai citra yang telah dipilih pada citra asli. Dapat dilihat pada Gambar 4.20.



Gambar 4.20 Proses Pengolahan Citra

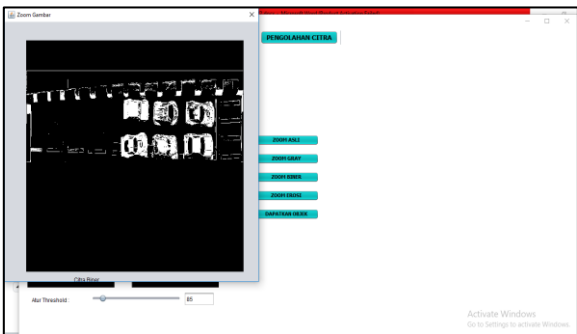
Apabila pengguna ingin melihat atau *zoom* masing-masing dari gambar pengolahan citra, pengguna dapat mengklik tombol *zoom* masing-masing citra. Dapat dilihat pada gambar dibawah ini.



Gambar 4.21 Zoom Citra Asli



Gambar 4.22 Zoom Citra Grayscale



Gambar 4.23 Zoom Citra Biner



Gambar 4.24 Zoom Citra Erosi

4.5 Sistem Simulasi

Pengujian sistem simulasi merupakan pengujian dengan menggabungkan seluruh algoritma dari sistem, baik itu pengolahan citra maupun *fuzzy inference system Tsukamoto*. Pada sebuah perempatan lampu lalu lintas terdapat 4 ruas jalur yang masing-masing jalur terdapat beberapa kendaraan disetiap arah dan memiliki lebar jalur, dari data pengujian simulasi 1 diketahui data kepadatan kendaraan atau banyak kendaraan dan lebar jalur seperti disajikan dalam Tabel 4.2.

Tabel 4.2 Data pengujian pada simulasi 1

No	Kepadatan kendaraan / Banyak Kendaraan	Lebar Jalur	Keterangan
1	9 Kendaraan	2 Jalur	Jalur 1
2	5 Kendaraan	2 Jalur	Jalur 2
3	16 Kendaraan	3 Jalur	Jalur 3
4	11 Kendaraan	3 Jalur	Jalur 4

Berdasarkan data pengujian diatas setelah dilakukan perhitungan nilai keanggotaan dan inferensi untuk setiap aturan maka didapatkan hasil seperti tabel berikut ini:

Tabel 4.3 Nilai Fungsi Keanggotaan Setiap Jalur

No Sim	Jalur	Sepi	Normal	Padat	Sempit	Lebar
Sim1	1	0,57~0,6	0,42~0,4	0,0	1,0	0,0
	2	1,0	0,0	0,0	1,0	0,0
	3	0,0	0,57~0,6	0,42~0,4	0,5	0,5
	4	0,28~0,3	0,71~0,7	0,0	0,5	0,5

Tabel 4.4 Nilai α – predikat dari Setiap Aturan

No Sim	Jalur 1		Jalur 2		Jalur 3		Jalur 4	
	α_1		α_1		α_1		α_1	
1	α_1	0,57	α_1	1,0	α_1	0	α_1	0,28
	α_2	0	α_2	0	α_2	0	α_2	0,28
	α_3	0,42	α_3	0	α_3	0,5	α_3	0,5
	α_4	0	α_4	0	α_4	0,5	α_4	0,5
	α_5	0	α_5	0	α_5	0,42	α_5	0
	α_6	0	α_6	0	α_6	0,42	α_6	0

Tabel 4.5 Nilai z dari Setiap Aturan

No Sim	Jalur 1		Jalur 2		Jalur 3		Jalur 4	
Sim 1	z1	28,57	z1	20	z1	40	z1	34,28
	z2	40	z2	40	z2	40	z2	34,28
	z3	28,57	z3	20	z3	30	z3	30
	z4	20	z4	20	z4	30	z4	30
	z5	40	z5	40	z5	48,57	z5	40
	z6	40	z6	40	z6	48,57	z6	40

Setelah nilai z dihitung maka diperoleh lama durasi lampu hijau menyala pada data simulasi 1 yaitu:

- Jalur 1 : 28,6 ~ 29 detik
- Jalur 2 : 20 ~ 20 detik
- Jalur 3 : 38,6 ~ 39 detik
- Jalur 4 : 31,6 ~ 32 detik

5. PENUTUP

5.1. Kesimpulan

Setelah melakukan penelitian ini dapat didapatkan beberapa kesimpulan antara lain:

1. Aplikasi simulasi pengaturan lampu lalu lintas ini dapat memberikan lama durasi lampu hijau pada setiap jalur lampu lalu lintas tergantung dari banyaknya kepadatan atau jumlah kendaraan dan lebar jalur pada satu jalur searah dari hasil perhitungan metode.
2. Metode pelabelan objek yang diterapkan pada pengolahan citra merupakan pemberian nilai 0 dan 1, kemudian dari proses tersebut menghasilkan piksel-piksel yang bertetangga yang dapat diberikan label untuk menentukan banyaknya kendaraan yang berada di atas jalur.
3. Metode *Fuzzy Inference System Tsukamoto* dapat digunakan untuk menentukan lama durasi lampu hijau menyala berdasarkan banyaknya kepadatan atau jumlah kendaraan dan lebar jalur pada suatu jalur.
4. Akurasi dari metode Pelabelan Objek yang digunakan untuk mendeteksi kepadatan kendaraan pada suatu citra sebesar 85%.
5. Faktor yang dapat mempengaruhi deteksi jumlah kendaraan yaitu intensitas cahaya, komposisi warna, kontras, bayangan dari kendaraan lain dan posisi saat replika jalan di foto. Sedangkan faktor yang berpengaruh pada pelabelan objek yaitu *threshold* yang digunakan untuk konversi citra biner.

5.2. Saran

Berdasarkan hasil dari melakukan penelitian ini, penulis menyadari banyak kekurangan dari sistem yang sudah dibuat. Adapun saran untuk mengembangkan penelitian ini yaitu, karena sistem yang dibuat hanya simulasi dan data sudah dikumpulkan terlebih dahulu, maka perlu ditambahkan webcam sehingga webcam langsung mengambil citra pada suatu jalur saat lampu jalur tersebut berwarna merah menuju hijau dan langsung dikirim kepada sistem tersebut memungkinkan program untuk memberikan durasi nyala dengan sendirinya, sehingga operator tidak perlu memasukan data citra pada setiap jalur.

DAFTAR PUSTAKA

- [1] Badan Pusat Statistik (2020), *Perkembangan Jumlah Kendaraan Bermotor Menurut Jenis, 1949-2018*, (<https://www.bps.go.id/linkTableDinamis/view/id/1133>).
- [2] Bahri, R. dan Maliki, I. (2012), *Perbandingan Algoritma Template Matching dan Feature Extraction Optical Character Recognition*, *Jurnal Komputer dan Informatika*, (1), 29–35.
- [3] Iqbal, M.H. (2002), *Pokok – Pokok Materi Teori Pengambilan Keputusan*, Ghalia Indonesia, Ed. Jakarta.
- [4] Jogiyanto, H.M. (2006), *Analisis dan Desain Sistem Informasi: Pendekatan Terstruktur Teori dan Praktek Aplikasi Bisnis*, Yogyakarta: ANDI ublisher.
- [5] Kadir, A. (2013), *Dasar Pengolahan Citra dengan Delphi*, Andi Offset.
- [6] Kadir, A. dan Susanto, A. (2013), *Teori dan Aplikasi Pengolahan Citra*, Andi Offset.
- [7] Kusumadewi, S. dan Purnomo, H. (2010), *Aplikasi Logika Fuzzy Untuk Mendukung Keputusan*, ed. 2 Yogyakarta: Graha Ilmu.
- [8] Prahasta, E. (2002), *Konsep-Konsep Dasar Sistem Informasi Geografis*, Bandung: Informatika.
- [9] Setiadi (2009), *Himpunan dan Logika Samar serta Aplikasinya*, Yogyakarta: Graha Ilmu.
- [10] Siagian, P. (1987), *Penelitian Operasional Teori dan Praktek*, Jakarta: Universitas Indonesia Press.

- [11] Usman Ahmad (2005), *Pengolahan Citra Digital & Teknik Pemrogramannya*, Yogyakarta: Graha Ilmu.
- [12] Waljiyanto (2003), *Sistem Basis Data : Analisis dan Pemodelan Data*, Yogyakarta: Graha Ilmu.
- [13] Whitten (2010), *System Analysis & Design Methods*, California: McGraw-Hil/Irwin.