IJARCCE

ISSN (Online) 2278-1021
ISSN (Print) 2319-5940

International Journal of Advanced Research in Computer and Communication Engineering
ISO 3297:2007 Certified
Vol. 7, Issue 7, July 2018

Smart Traffic controller using Fuzzy logic

Adnan Shaout[1], Pavan Kumar Dasari[2], Abdelwahed Motwakel[3]

Electrical and Computer Engineering Department, The University of Michigan – Dearborn - USA[1,2]

Computer Science Department, Faculty of Computer Science and Information Technology,

Omdurman Islamic University & National University - Sudan, Khartoum- Sudan[3]

Abstract: This project implements a Smart traffic controller using fuzzy logic based on Vehicle and pedestrians count. Smart traffic controllers are needed these days to maintain busy traffic junctions featuring both vehicles and pedestrians. Computer vision technologies are available to see and identify the vehicles and pedestrians for regulating traffic. Various Fuzzy controllers based on neural networks are available to handle the modern metro traffic junctions. Fuzzy inference system used in this project decides the signalling time based on the fuzzified count of pedestrians and vehicles. This input variable for the decision is the fuzzified count of the vehicles and pedestrians from all the streets. This project is built in MATLAB using Fuzzy systems toolbox. This Smart traffic controller based on Fuzzy logic is capable of delivering unbiased traffic signalling and manage the traffic effectively.

Keywords: Fuzzy, Traffic, Vehicles, Pedestrians

I. INTRODUCTION

Modern metro traffic junctions are getting more difficult to manage in terms of traffic management. Day by day, the ever-increasing traffic is making it more difficult. Even the increase in real estate costs leading towards central parking facilities for thousands, which increase the pedestrians on the roads walking to nearby offices. The traffic systems should accommodate the pedestrians and vehicles equally.

Various traffic controllers based on fuzzy and computer vision just consider the vehicles in traffic decision [2], [3]. They are also biased by starving the street with fewer vehicles.

In [4] authors proposed Unified Modelling Language (UML) model for an Adaptive Road Traffic Control System. They provide a technique for controlling the traffic in highway network using signals that are automatically controlled by detectors whilst coordinating all the signals, as per the changes in traffic flow.

In [5]authors proposed an adaptive traffic control system based on a new traffic infrastructure using Wireless Sensor Network (WSN) and presented techniques for controlling the traffic flow sequences. These techniques are dynamically adaptive to traffic conditions on both single and multiple intersections.

In [6]authors proposed Field Programmable Gate Arrays (FPGA) design of A 24-hour traffic light controller system of a four roads structure with six traffic lights has been simulated, implemented and tested. The system has been designed using VHDL, and implemented on hardware using Xilinx Spartan 3 FPGA Starter kit.

In [7] authors introduced project called intelligent of traffic signal controller .This project has two major phases. The first stage is to design a program, which consists of reading, research, planning and designing a program. The second phase is to continue with the hardware implementation using the gate logic and the interface light is using LED. The blinking is depending on the state machine transition. GSM Interface is also provided for sending traffic alerts signals for drivers on road and precautions are taken not to indulge in traffic congestion.

In [8] authors proposed traffic light control system for emergency vehicle using Radio Frequency (RF) facilitate emergency vehicle to cross at the intersection of traffic light. The system-using Radio Frequency (RF) as the medium for emergency vehicle communicates with traffic light system. The system solve problem for emergency vehicle when approaching traffic light with ease.

In [9] authors proposed intelligent Traffic Lights Based on Radio Frequency Identification (RFID) with this system considered the priority of different type of vehicles and considers the density of traffic on the roads by installing RF reader on the road intersections. Radio frequency identification is a technique that uses the radio waves to identify the object uniquely. Under the proposed work, each intersection contains 8 RFID readers. The road was divided into two lanes. Each lane has its RFID reader to track the vehicles passing through it. Each intersection point has its own database to store the information regarding the vehicles that passed from it with timestamp and traffic light. Every

ISSN (Online) 2278-1021
ISSN (Print) 2319-5940

IJARCCE

International Journal of Advanced Research in Computer and Communication Engineering
ISO 3297:2007 Certified
Vol. 7, Issue 7, July 2018

vehicle has a RFID enabled device that stores a vehicle identification number (VIN). The current project introduces an intelligent decision making fuzzy system for controlling the traffic based on pedestrians and vehicles, which is similar to human decision making. This project also uses a round robin signalling system to avoid the unbiased signalling where the signal time is based on the final fuzzified count of vehicles and pedestrians. Modern technological developments are allowing us to detect the vehicles count and pedestrians and update the count live to feed to the controller for handling the traffic effectively.

II. DESIGN

The Smart traffic controlling system has three sub stages, Robotic eyes, Fuzzy Logic controller and signalling system. This projects implements only the fuzzy logic controller with expectations set for the other two systems in terms of variables based on the fuzzy controller.

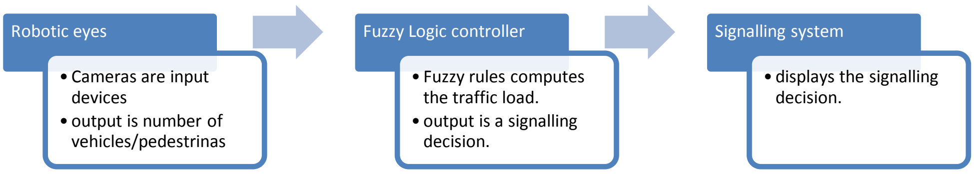

Fig: 1 Block diagram of system design

A. Robotic Eyes

Robotic eyes are the sensors used for extracting the count necessary for fuzzy controller. These robotic eyes are equipped with advanced computer vision algorithms to detect the vehicles and pedestrians and estimate count. This is not implemented in the project.

B. Fuzzy logic controller

Fuzzy logic controller is fed with the input from the robotic eyes for decision-making. It combines the vehicle and pedestrians count from each street to form a fuzzified count, which is then used to form a final fuzzified count used in calculating the signalling time. This is the system implemented in this project.

C. Signalling system

This is the displaying system of the signal. It processes the decision of Fuzzy controller and implements the signal based on the decision.

III. IMPLEMENTATION

The system is implemented using MATLAB and Fuzzy systems toolbox. The Fuzzy inference system for decision-making is built using Fuzzy systems toolbox in MATLAB. The fuzzy inference system works in three layers, street level, opposite streets and final decision.

A. Street Level Fuzzification

At street level the Fuzzy Inference System (FIS) combines the input variables vehicle count and pedestrians count to come up with a fuzzy value for each street. This FIS is used for calculating fuzzy value for all the four streets.

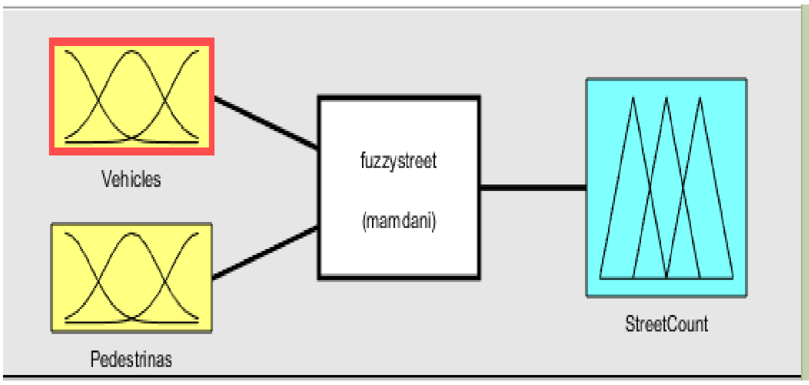

Fig: 2 Structure of Street level FIS

IJARCCE

ISSN (Online) 2278-1021
ISSN (Print) 2319-5940

International Journal of Advanced Research in Computer and Communication Engineering
ISO 3297:2007 Certified
Vol. 7, Issue 7, July 2018

Triangular membership functions are used for the two inputs vehicles and pedestrians. The range of vehicles and pedestrians are assumed to be from 0-30.

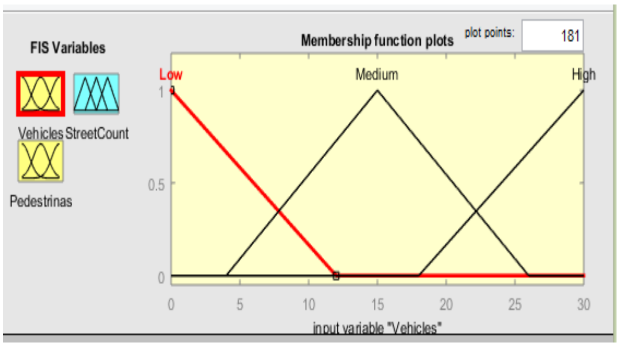

Fig: 3 Vehicles membership function

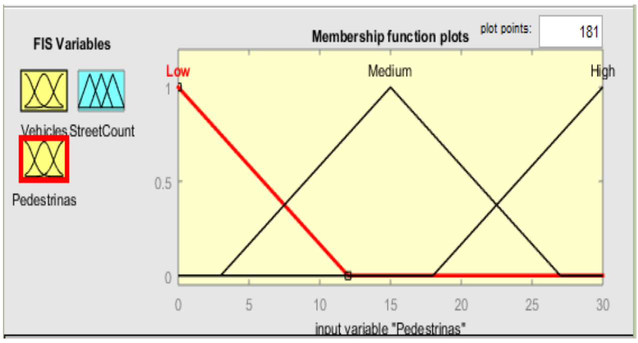

Fig: 4 Pedestrians membership function

The output of fuzzy inference system is the fuzzified count for the street on a scale range of 0-30.

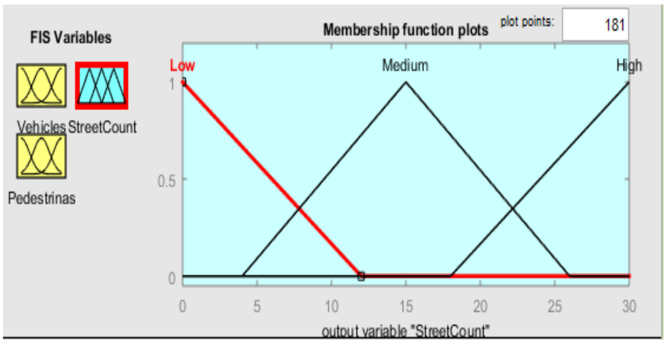

Fig: 5 Street output membership function

Rule base for street level fuzzification is as below,

Pedestrians

Vehicles	Low	Medium	High
Low	L	M	H
Medium	L	M	H
High	M	M	H

Fig: 6 Rule base for street FIS

B. Opposite Streets Fuzzification

At this level, the fuzzy output of the opposite streets (East-West/ North-South) at street level is used as inputs for calculating the combined fuzzy count for the opposite streets. This FIS is used for calculating fuzzy value for two sets of opposite streets.

ISSN (Online) 2278-1021
ISSN (Print) 2319-5940

International Journal of Advanced Research in Computer and Communication Engineering
ISO 3297:2007 Certified
Vol. 7, Issue 7, July 2018

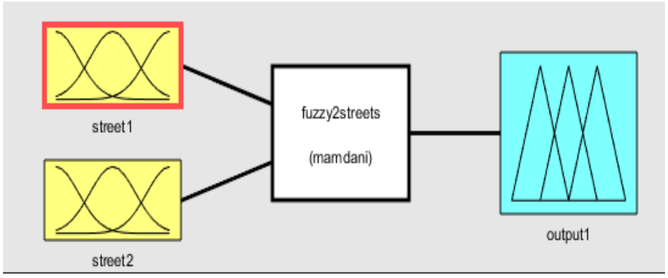

Fig: 7 Structure of opposite streets FIS

Triangular membership functions are used for the two inputs Street 1 and Street 2. The range of Street 1 and Street 2 are calculated to be from 0-30.

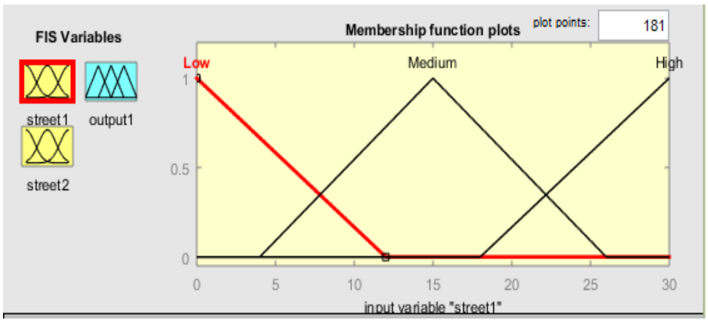

Fig: 8 Street 1 membership function

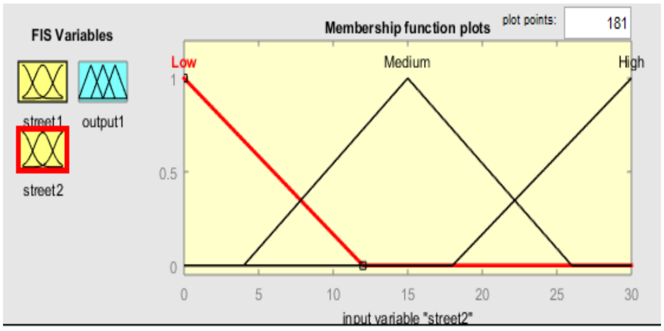

Fig: 9 Street 2 membership function

The output of fuzzy inference system is the fuzzified count for the set of opposite streets on a scale range of 0-30.

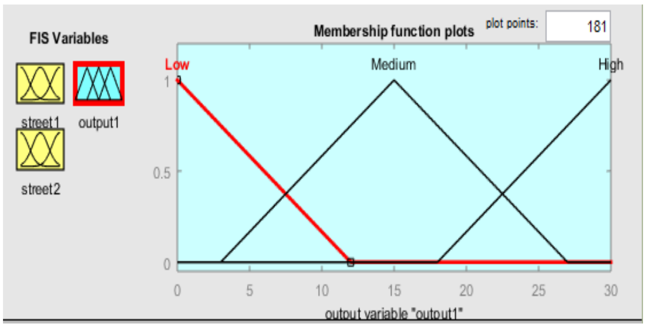

Fig: 10 Opposite Street output membership function

Rule base for street level fuzzification is as below,

Street2 \ Street1	Low	Medium	High
Low	L	M	H
Medium	M	M	H
High	H	H	H

Fig: 11 Rule base for opposite streets

ISSN (Online) 2278-1021
ISSN (Print) 2319-5940

International Journal of Advanced Research in Computer and Communication Engineering
ISO 3297:2007 Certified
Vol. 7, Issue 7, July 2018

C. Final Decision

Final decision contains two FIS, one for East- West and the second one for North-South. Both are similar except for rule base. The final decision is based on the fuzzy count of opposite streets calculated using the fuzzy opposite streets FIS.

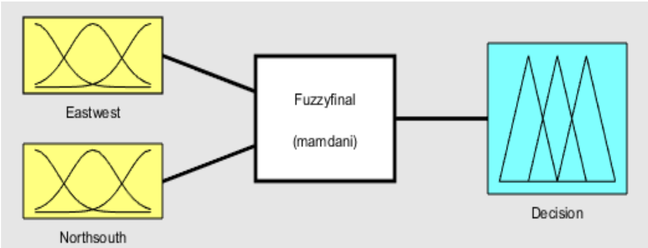

Fig: 12 Structure of final FIS

Triangular membership functions are used for the two inputs East-West and North-South. The range of Street 1 and Street 2 are calculated to be from 0-30.

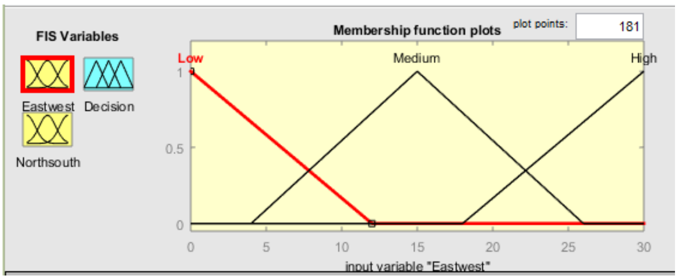

Fig: 13 East West membership function

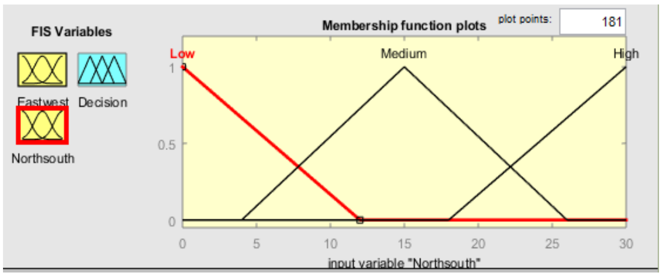

Fig: 14 North South membership function

The output of fuzzy inference system is the fuzzified count for the final decision for deciding the green time on a scale range of 0-30.

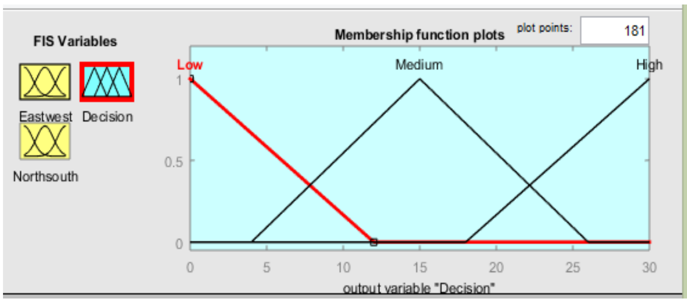

Fig: 15 Final output membership functions.

Rule base for East-West signal decision is as below,

East & West \ North & South	Low	Medium	High
Low	L	L	L
Medium	M	M	L
High	H	H	M

Fig: 16 Rule base for East-West signal

Rule base for North-South signal decision is as below,

ISSN (Online) 2278-1021
ISSN (Print) 2319-5940

IJARCCE

International Journal of Advanced Research in Computer and Communication Engineering
ISO 3297:2007 Certified
Vol. 7, Issue 7, July 2018

East & West

North & South	Low	Medium	High
Low	L	L	L
Medium	M	M	L
High	H	H	M

Fig: 17 Rule base for North-South signal

D. Traffic Signal

The traffic signal part of the project is implemented in the MATLAB. It is implemented by using the below two functions,

1. Createfigure1.m (See Appendix 1)
2. Createfigure2.m (See Appendix 2)

The final traffic controller is implemented using final .m (See Appendix 3). Random number generator function is used to generate vehicles and pedestrians count for simulation.

IV. RESULTS

Simulation of the smart traffic controller using fuzzy logic is made in MATLAB editor. Results are satisfactory in calculating the green time. The sample results are shown below,

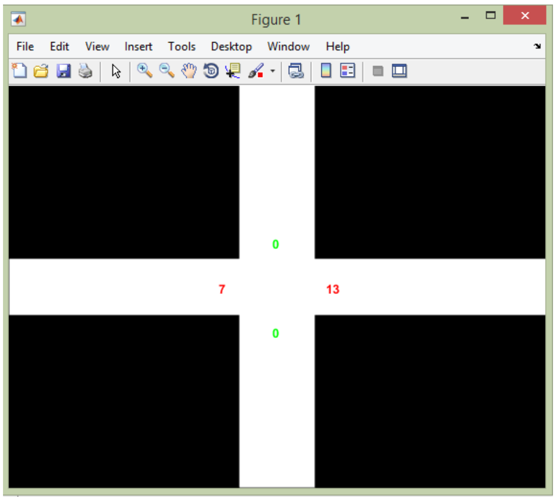

Fig: 18 Sample output for East-West signal

In the fig 18, the displayed numbers in the simulation are the effective counts of pedestrians and vehicles for each street.

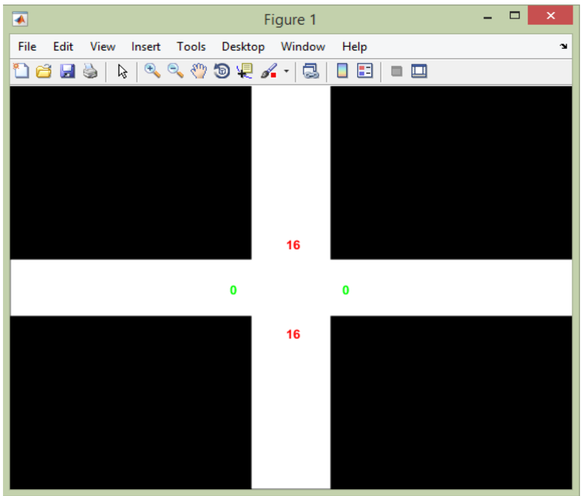

Fig: 19 Sample output for North-South signal

IJARCCE

ISSN (Online) 2278-1021
ISSN (Print) 2319-5940

International Journal of Advanced Research in Computer and Communication Engineering
ISO 3297:2007 Certified
Vol. 7, Issue 7, July 2018

The efficiency of the smart traffic controller can be studied by using below data from the simulated controller.

	Pedestrianas			
S.no	P_E	P_N	P_W	P_S
1	4	10	8	15
2	3	26	27	24
3	5	22	4	20
4	26	11	21	7
5	27	19	19	26
6	26	2	15	6
7	3	21	3	4
8	22	6	20	16
9	14	25	4	5
10	3	13	16	13
11	2	29	6	4
12	28	27	3	22
13	13	29	10	21
14	6	5	29	6
15	7	12	14	29
16	7	13	15	5
17	9	10	19	9
18	18	5	27	26

Fig: 20 Pedestrians data from Matlab simulation

	Vehicles			
S.no	V_E	V_N	V_W	V_S
1	0	20	5	20
2	21	13	12	29
3	4	9	11	21
4	15	23	22	27
5	2	22	16	15
6	24	18	7	8
7	29	21	16	15
8	16	4	24	24
9	29	20	24	14
10	6	12	25	24
11	20	19	10	14
12	12	7	15	11
13	9	13	17	28
14	20	17	21	20
15	2	17	26	20
16	6	25	20	12
17	18	8	12	18
18	25	29	22	11

Fig: 21 Vehicles data from Matlab simulation

	Fuzzified count with signal time			
S.no	F_E	F_N	F_W	F_S
1	7	14	13	15
2	11	24	26	20
3	9	18	7	17
4	24	15	18	15
5	26	16	16	24
6	23	5	15	12
7	16	17	5	7
8	18	11	17	16
9	16	22	14	9
10	5	16	16	16
11	9	26	11	7
12	26	26	5	18
13	16	26	14	17
14	11	9	26	11
15	12	16	16	26
16	12	16	15	9
17	14	14	16	14
18	16	16	26	24

Fig: 22 Fuzzified count from Matlab simulation

ISSN (Online) 2278-1021
ISSN (Print) 2319-5940
IJARCCE
International Journal of Advanced Research in Computer and Communication Engineering
ISO 3297:2007 Certified
Vol. 7, Issue 7, July 2018

S.no	F_E&W	F_N&S	Greentime	Signal
1	15	15	23	E-W
2	26	21	16	N-S
3	14	16	23	E-W
4	21	15	20	N-S
5	26	21	30	E-W
6	19	16	22	N-S
7	16	15	23	E-W
8	16	15	23	N-S
9	15	18	23	E-W
10	16	15	23	N-S
11	15	26	7	E-W
12	26	26	23	N-S
13	15	26	7	E-W
14	26	15	7	N-S
15	16	26	7	E-W
16	16	16	23	N-S
17	15	15	23	E-W
18	26	21	16	N-S

Fig: 23 Final decision data from Matlab simulation.

By evaluating the data from fig 20, 21, 22 and 23 the signal performance is evaluated and it looks satisfactory. The decision system works fine without any biasing. The simulation results show that, the controller is able to decide the green time based on pedestrians and vehicles count effectively. The results satisfied the requirements of the project.

V. CONCLUSION

The Smart traffic controller based on fuzzy logic is developed which calculated green time based on pedestrians and vehicles count. The solution is helpful in implementing the unbiased traffic control by considering both vehicles and pedestrians. The membership functions might need to be tuned to achieve higher efficiency in decision process. In future more inputs can be added to the FIS system to achieve human like decision in traffic control. The improvements in the technology seems to be an important factor in achieving human like decision in traffic control, which needs more sophisticated sensors to get the parameters for decision.

REFERENCES

[1] http://therantyhighwayman.blogspot.com/2014/07/traffic-signal-pie-third-slice-floating.html - Accessed on 7/28/2015
[2] Khan, Bilal Ahmed, and Nai Shyan Lai. "An intelligent traffic controller based on fuzzy logic." The First International Conference on Green Computing, Technology and Innovation (ICGCTI2013). The Society of Digital Information and Wireless Communication, 2013.
[3] A.R. Zade, D.R. Dandekar, “Simulation of Adaptive Traffic Signal Controller in Matlab Simulink Based on Fuzzy Inference system”, International Journal of Computer Applications
[4] Ranjini, K., A. Kanthimath, and Y. Yasmine. "Design of adaptive road traffic control system through unified modeling Language." International Journal of Computer Applications 14.7 (2011): 36-41.
[5] Yousef, Khalil M., Mamal N. Al-Karaki, and Ali M. Shatnawi. "Intelligent traffic light flow control system using wireless sensors networks." J. Inf. Sci. Eng. 26.3 (2010): 753-768.
[6] El-Medany, W. M., and M. R. Hussain. "FPGA-based advanced real traffic light controller system design." Intelligent Data Acquisition and Advanced Computing Systems: Technology and Applications, 2007. IDAACS 2007. 4th IEEE Workshop on. IEEE, 2007.
[7] Chavan, Shilpa S., R. S. Deshpande, and J. G. Rana. "Design of intelligent traffic light controller using embedded system." Emerging Trends in Engineering and Technology (ICETET), 2009 2nd International Conference on. IEEE, 2009.
[8] Ilmudin, M. H. A., et al. "Traffic Light Control System using 434 MHz Radio Frequency." International Journal of Research in Advent Technology 2.8 (2014): 26-31.
[9] Singh, Harpal, Krishan Kumar, and Harbans Kaur. "Intelligent traffic lights based on RFID." International Journal of Computing & Business Research, ISSN (2012): 2229-6166.

APPENDIX

1. Createfigure1.m

```
function createfigure1(X1, Y1, figure1)
% Create axes
axes1 = axes('Parent',figure1,...
'Position',[0 0 1 1]);
xlim([0 3.5])
ylim([0 3.5])
%%
hold(axes1,'all');
% Create rectangle
```

IJARCCE

ISSN (Online) 2278-1021
ISSN (Print) 2319-5940

International Journal of Advanced Research in Computer and Communication Engineering
ISO 3297:2007 Certified
Vol. 7, Issue 7, July 2018

```
annotation(figure1,'rectangle',...
    [4/7 0 3/7 3/7],...
'LineWidth',1,...
'FaceColor',[0 0 0]);
% Create rectangle
annotation(figure1,'rectangle',...
    [4/7 4/7 3/7 3/7],...
'LineWidth',1,...
'FaceColor',[0 0 0]);
% Create rectangle
annotation(figure1,'rectangle',...
    [0 4/7 3/7 3/7],...
'LineWidth',1,...
'FaceColor',[0 0 0]);
% Create rectangle
annotation(figure1,'rectangle',...
    [0 0 3/7 3/7],...
'LineWidth',1,...
'FaceColor',[0 0 0]);
Holdoff
```

2. Createfigure2.m

```
function createfigure1(x,y,figure1)
red=[1 0 0];
green=[0 1 0];
yellow=[1 .8 0];
% drawnow
if y(1)==1
    light1=green;
elseif y(1)==0.5
    light1=yellow;
else
    light1=red;
end
if y(2)==1
    light2=green;
elseif y(2)==0.5
    light2=yellow;
else
    light2=red;
end
if y(3)==1
    light3=green;
elseif y(3)==0.5
    light3=yellow;
else
    light3=red;
end
if y(4)==1
    light4=green;
elseif y(4)==0.5
    light4=yellow;
else
    light4=red;
end
annotation(figure1,'textbox',...
    [0.38 0.46 0.03 0.067],...
'String',{x(1)},...
'FitBoxToText','on','FontWeight','bold',...
'EdgeColor','none','BackgroundColor',[1 1 1],'Color',light1);
annotation(figure1,'textbox',...
    [0.48 0.35 0.03 0.067],...
'String',{x(2)},...
'FitBoxToText','on','FontWeight','bold',...
'EdgeColor','none','BackgroundColor',[1 1 1],'Color',light2);
```

```
annotation(figure1,'textbox',...
   [0.58 0.46 0.03 0.067],...
'String',{x(3)},...
'FitBoxToText','on','FontWeight','bold',...
'EdgeColor','none','BackgroundColor',[1 1 1],'Color',light3);
annotation(figure1,'textbox',...
   [0.48 0.57 0.03 0.067],...
'String',{x(4)},...
'FitBoxToText','on','FontWeight','bold',...
'EdgeColor','none','BackgroundColor',[1 1 1],'Color',light4);
```

3. Final.m

```
clc; clear; close all;  %clean screen and variables to initial preparation
fuzzystreet=readfis('fuzzystreet.fis'); % import fuzzy controller for determining count on street.
fuzzy2streets=readfis('fuzzy2streets.fis'); % import fuzzy controller for determining the combined count of opposite streets.
fuzzyfinal=readfis('Fuzzyfinal.fis'); % import fuzzy controller for final decision for east &west.
fuzzyfinalns=readfis('FuzzyfinalNS.fis');% import fuzzy controller for final decision for North & South.
figure1=figure;
x=[0 20 5 20];        % initial condition (#cars at first time)[e n w s]
z=[4 10 8 15];        % initial condition for pedestrians [e n w s]
y=[0 0];   % y is a matrix that contain light state (0=Red, 1=Green, 0.5=Yellow)
yellowtime=2;          % times (in second) for yellow light
exittime=3;           % times (in second) for each exit cars
exitcar=2;            % #cars that exit in each step from green light
fff=1;
while (fff==1)
for ii=1:2
clf('reset');
ytemp=[0 0 0 0];
% fuzzy controller determining signal on time
if max(x)<30    % domain of fuzzy controller (if x>[30 30 30 30] control is out of fuzzy controller and control will be by MAX
function)
if max(z)<30
i=1;
fori=1:4
temp=[x(i);z(i)];
ytemp(i)=evalfis(temp,fuzzystreet); %run fuzzy controller for each street
end
ytemp=ceil(ytemp);
zt(1)=evalfis([ytemp(1);ytemp(3)],fuzzy2streets); %run fuzzy controller for opposite streets
zt(2)=evalfis([ytemp(2);ytemp(4)],fuzzy2streets);
ztt=[zt(1);zt(2)];
end
end
createfigure1(ytemp,y,figure1);  % creatfigure1 is a function that builds 4 streets
   y=zeros(1,4);
% Yello Light
if ii==1
y(ii)=0.5;  % yellow light
y(ii+2)=0.5;
y(ii+1)=0;      % red
y(ii+3)=0;
   createfigure2(x,y,figure1);       % creatfigure2 is a function that shows total count and lights
pause(yellowtime);
y(ii)=0;  % red light
y(ii+2)=0;
y(ii+1)=1;      % green
y(ii+3)=1;
elseif ii==2
y(ii)=0.5;  % yellow light
y(ii+2)=0.5;
y(ii-1)=0;      % red
y(ii+1)=0;
   createfigure2(x,y,figure1);       % creatfigure2 is a function that shows total count and lights
pause(yellowtime);
```

IJARCCE
ISSN (Online) 2278-1021
ISSN (Print) 2319-5940
International Journal of Advanced Research in Computer and Communication Engineering
ISO 3297:2007 Certified
Vol. 7, Issue 7, July 2018

```
y(ii)=0;  % red light
y(ii+2)=0;
y(ii-1)=1;      % green
y(ii+1)=1;
end
if ii==1
ff=evalfis(ztt,fuzzyfinal);    % run fuzzy controller for final decision.
greentime(ii)=ceil(ff*exittime/exitcar);     % time that is sufficient for all cars
disp(greentime(ii));
for j=1:greentime(ii)+1
ytemp(ii+1)=ceil(ytemp(ii+1)-exitcar);                    % #Cars that exit
ytemp(ii+3)=ceil(ytemp(ii+3)-exitcar);                  % #Cars that exit
ifytemp(ii+1)<=0
ytemp(ii+1)=0;
ifytemp(ii+3)<=0
ytemp(ii+3)=0;
createfigure2(ytemp,y,figure1);
pause(exittime);
break
break
end
end
createfigure2(ytemp,y,figure1);
pause(exittime);
end
elseif ii==2
ff=evalfis(ztt,fuzzyfinalns);   % run fuzzy controller for final decision.
greentime(ii)=ceil(ff*exittime/exitcar);
disp(greentime(ii));
for j=1:greentime(ii)+1
ytemp(ii-1)=ceil(ytemp(ii-1)-exitcar);                    % #Cars that exit
ytemp(ii+1)=ceil(ytemp(ii+1)-exitcar);                  % #Cars that exit
ifytemp(ii-1)<=0
ytemp(ii-1)=0;
ifytemp(ii+1)<=0
ytemp(ii+1)=0;
createfigure2(ytemp,y,figure1);
pause(exittime);
break
break
end
end
createfigure2(ytemp,y,figure1);
pause(exittime);
end
end
x=ceil(28.*rand(4,1) + 1);
z=ceil(28.*rand(4,1) + 1);
end
end
```