2023 IEEE 3rd International Maghreb Meeting of the Conference on Sciences and Techniques of Automatic Control and Computer Engineering (MI-STA), Benghazi, Libya 21-23 May 2023.

# Design and Implementation of a Smart Traffic Light System with Libyan License Plate Recognition on FPGA

1st Lamia Atif Ali
*Department of Computer Engineering*
*University of Tripoli*
Tripoli, Libya
l.abd-elmotaleb@uot.edu.ly

2nd Mohamed Muftah Eljhani
*Department of Computer Engineering*
*University of Tripoli*
Tripoli, Libya
m.eljhani@uot.edu.ly

2023 IEEE 3rd International Maghreb Meeting of the Conference on Sciences and Techniques of Automatic Control and Computer Engineering (MI-STA) | 979-8-3503-1989-7/23/$31.00 ©2023 IEEE | DOI: 10.1109/MI-STA57575.2023.10169241

*Abstract*— **The primary purpose of conducting this research is to create, test, and apply a traffic light system based on traffic density that operates in real-time with red light violation detection and automatic Libyan license plate recognition. A group of infrared sensors positioned on each intersection roadway are used to measure traffic density. It uses radio frequency identification technology to prioritize emergency cars. The system detects red light violations using passive infrared sensors and then captures the violating vehicle image, then the image is handed to the automatic license plate recognition system which processes the image to localize the license plate region and recognize the digits and characters. The suggested traffic light system is designed and simulated using Verilog Hardware Description Language and tested on Cyclone IV GX field-programmable gate array. The automatic license plate recognition system is implemented using machine learning and image processing algorithms.**

***Keywords—Density Based Traffic Light System, Automatic License Plate Recognition, Digital Image Processing, Machine Learning, Verilog Hardwar Description Language, Field-Programmable Gate Array Design.***

## I. INTRODUCTION

Today, vehicle traffic is an important economic component in both urban and rural areas due to the ever-increasing number of vehicles, and it needs appropriate administration and supervision to guarantee this large number of vehicles does not conflict [1]. Due to the huge number of vehicles around the world, different levels of traffic breaking-rule arise, particularly red-light violations [1]. Effective traffic violation detection and number plate recognition systems are required to arrest offenders and address the shortcomings and failings of human traffic administrators who are unable to be everywhere at once [1]. Moreover, this huge increase in vehicles number can easily lead to traffic jams. A traffic jam happens when a car is stationary for a long period [2]. Conventional traffic lights have a fixed time duration regardless of traffic congestion and do not consider the presence of emergency vehicles. This prompted us to consider developing a smarter and more effective method of solving those problems. A field-programmable gate array (FPGA) based intelligent traffic light controllers (TLCs) are being explored to design more effective traffic systems. This is in contrast to conventional traffic light controllers, which use a 24-hour delay and apply specific traffic light periods to individual time zones[3]. A method of prioritizing routes has been created where the waiting time for each is manually adjusted[4]. Four routes have been established and the movement of traffic on them is managed by a 24-hour TLC [5]. In [6]. Real-time density-based TLC manages intersections with different delays for roads in the intersection and provides an emergency handling feature [7]. A dynamic TLC has been suggested, in which traffic density is calculated as a function of the number of active infrared (IR) sensors[8]. A red light violation detection system is implemented using IR sensors and image processing techniques to recognize the violated vehicle license plate number [1]. A system to recognize Egypt's license plate is implemented utilizing image processing and an artificial neural network [9]. Libyan license plate recognition system uses a support vector machine [10]. To manage and supervise traffic, the proposed system employs radio frequency identification (RFID) scanners and infrared sensors, while machine learning and digital image processing are used to determine the offending vehicle's license plate number. As a system on a chip (SOPC), FPGA manages and controls the traffic based on traffic density and gives priority to emergency vehicles automatically. A new function suggested in our system is that to prevent blockage at the junction, the traffic signal shifts based on data received from sensors that are installed on the roads. Another powerful feature is implemented in our system which is red light violation detection using IR sensors and License plate recognition. In our system license plate recognition is implemented using machine learning and image processing techniques, the main advantage of this license plate system over the other license plate recognition systems is that it takes into account different illumination conditions of the image and different angles and different distances from which the image was captured.

## II. METHODOLOGY

### A. Traffic Light System

*1) System overview:* This system is responsible for managing the traffic flow at a three-way intersection. To monitor the number of vehicles on each road, five IR sensors are placed on each of them. three RFID readers are also installed on each road, which use radio waves to detect and

979-8-3503-1989-7/23/$31.00 ©2023 IEEE

identify emergency vehicles by communicating with the RFID tags placed on them. Proper setup and assessment of the frequency spectrums used by RFIDs coexisting in the same environment are crucial to avoid interference that could disrupt the system's functionality.

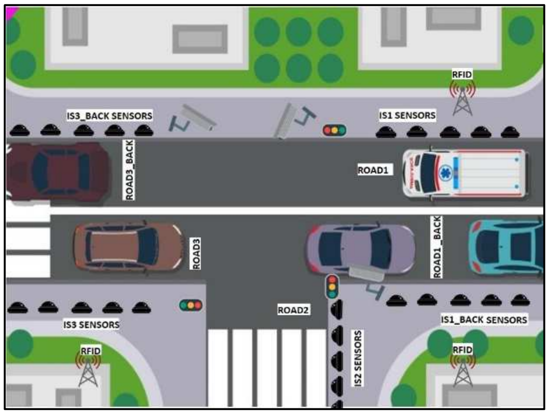

Fig. 1. The system's structure.

*2) System algorithm:* the algorithm operates by allocating two states, green and yellow, to each road, as well as a shared state in which all roads are red. Each state is illustrated in TABLE. I. Also, the system has three different states' transition flow based on the traffic volume of each road and the emergency signals. The traffic system manages a three-way intersection. It always starts by turning on the yellow light for ROAD1 for 3 seconds, and while counting up to 3 seconds it checks for any emergency or road blockage signals. If not found it determines the green light duration for ROAD2 by continuously monitoring the active sensors count on ROAD2, which indicate the traffic density level. After the 3 seconds are done, the green light is turned on for ROAD2 and the system flow continues. Fig. 2 shows the normal states transition without emergency or road blockage signal. Fig. 3 shows the state's transition in case of an emergency signal on ROAD3. The road blockage signal is determined based on the count of active IR sensors on the two main roads from which the vehicles leave the intersection if the active sensors count on them is five this indicates that the road is blocked and to prevent the intersection blockage the system goes to ALLRED state (where the red light is on for all roads) to make time for the blocked road to get less crowded. Fig. 4 shows the state's transition in case of a road blockage signal. Fig. 5 shows ROAD2's green state flow chart. Fig. 6 shows the flow chart of ROAD1's yellow state. Fig. 7 views the ALLRED state flow chart. We illustrated ROAD2's green state and ROAD1's yellow state flow charts only as the remaining roads' green and yellow states are similar to them.

TABLE. I. THE SYSTEM STATES

| States | Meaning |
| --- | --- |
| R1G | ROAD1's green light is on. |
| R1Y | ROAD1's yellow light is on. |
| R2G | ROAD2's green light is on. |
| R2Y | ROAD2's yellow light is on. |
| R3G | ROAD3's green light is on. |
| R3Y | ROAD3's yellow light is on. |
| ALLRED | The traffic light is red for All roads. |

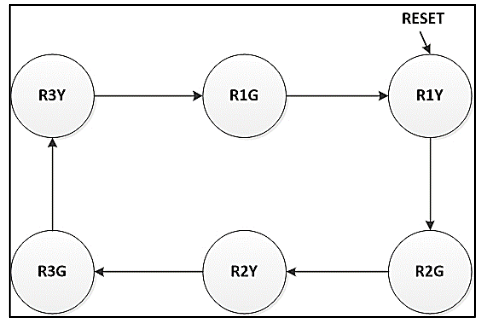

Fig. 2. The normal flow of the system's states.

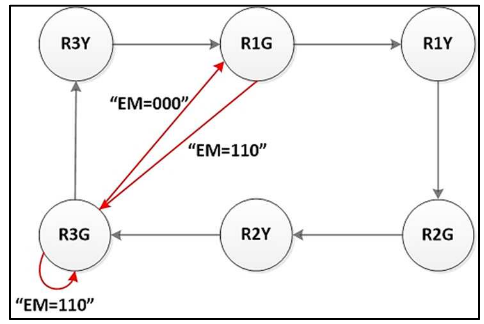

Fig. 3. The system's states flow with emergency vehicle on ROAD3.

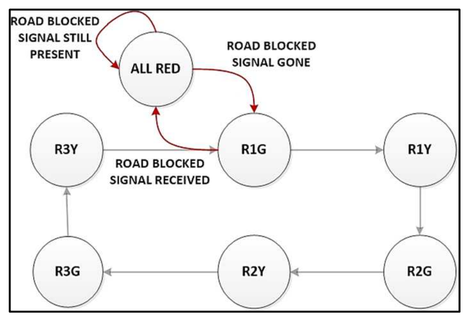

Fig. 4. The state transition with road blockage signal.

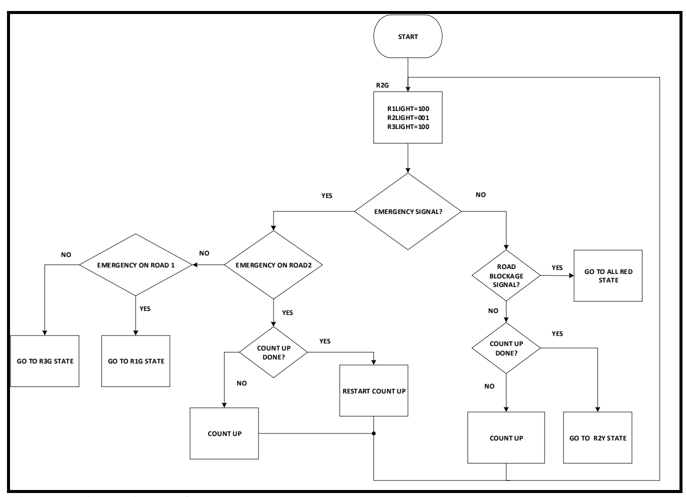

Fig. 5. Flow chart of ROAD2 green state.

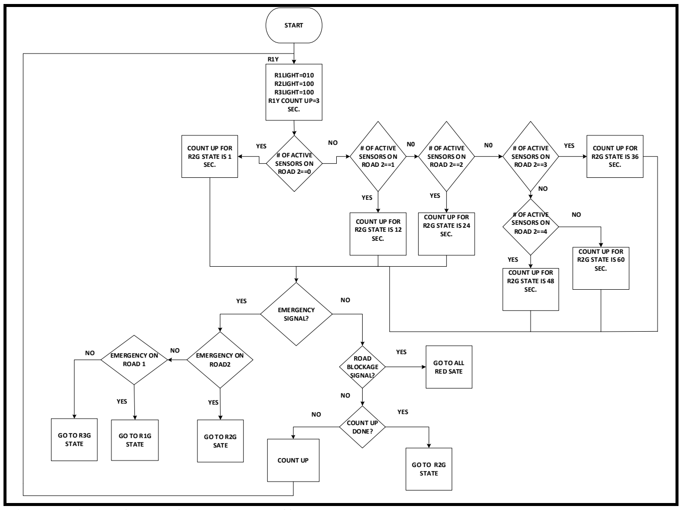

Fig. 6. Flow chart of ROAD1 yellow state.

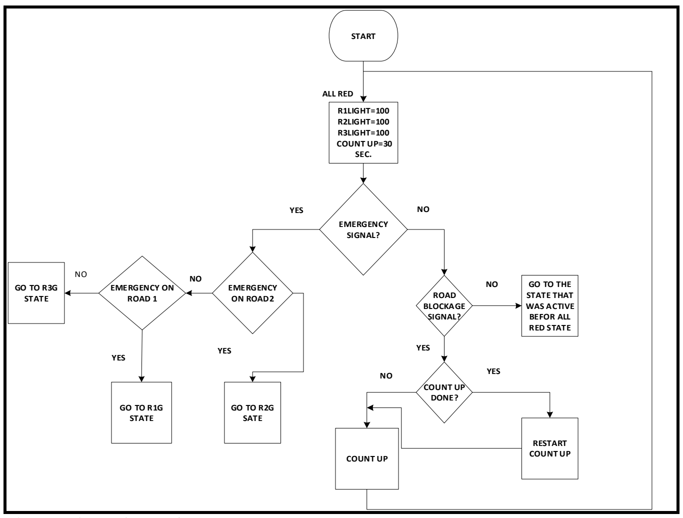

Fig. 7. Flow chart of ALLRED state.

*3) Features proposed in the system:*

*a) Dynamic traffic signal duration:*

This smart traffic system operates in real-time and makes traffic signal scheduling choices based on traffic density. Assume that it is ROAD1's turn to have the green light but ROAD1 is not crowded by vehicles at all at the same time ROAD2 is very crowded by vehicles, it is not a good idea to give a fixed duration for various traffic densities. This traffic system is capable of realizing this issue and gives time duration for ROAD1's green signal based on the traffic density on that specific road, guaranteeing adequate administration of traffic, relying on the traffic density of each road. Fig. 6 views ROAD1's yellow state flow chart which contains an illustration for various time durations depending on traffic densities.

*b) Prioritizing emergency cars:* Whenever an emergency car is spotted by the system it takes permission to pass immediately. Fig. 3 views the case when an emergency car is spotted on ROAD3 and its state transition. while the green light is turned on for ROAD1. The traffic system turns on the red light for ROAD1 and the green light for ROAD3 allowing the emergency vehicle to pass after that the traffic system returns to ROAD1.

*c) Preventing three-ways intersection blockage:* The system prevents intersection blockage that happens when one or both of the main roads from which vehicles exit the intersection are crowded with vehicles. If the system gives one of the roads the green light the intersection blockage will happen even if the other roads are not crowded. Fig. 4 shows the state transition when a road blockage signal is received. The system goes to an ALLRED state where red lights are turned on for all roads to provide extra time for the crowded roads to get less crowded. Fig.8 views the case when ROAD1_BACK and ROAD3_BACK are crowded with vehicles and the green light is given to ROAD2.

Fig. 8. Three-way intersection blockage view.

## B. License Plate System

The proposed automatic license plate system consists of five parts:

1. Detection of red-light violations and image acquisition.
2. License plate localization.
3. Image pre-processing.
4. Extract the license number.
5. License plate number recognition.

*1) Detection of red-light violation and image acquisition.*

Passive infrared sensors are installed at a distance from each of the intersecting roads' entrance points at the intersection of a three-way road. These sensors are coordinated with the red-light traffic signal so that each road's sensors are enabled and disabled simultaneously. Any car that enters the junction while the light is red is picked up by the infrared sensor. The camera designated to that segment of the road is triggered upon detection of any violating vehicles and takes a photo of the offending car. The FPGA receives the vehicle image at that point. The camera was positioned so that it could take a picture of the front of a car that was breaking the law, including the license plate. Each camera is placed at a distance after the entrance point of the intersection facing the road assigned to it.

*2) License plate localization*

A machine learning algorithm is implemented to determine the license plate region in the image. The aggregate channel features (ACF) machine learning method is used to train the license plate detector. The detector was trained on a custom-made dataset of Libyan license plates. The detector detects license plates from various angles and various distances and also can detect skewed license plates. The license plate detector is trained on 3069 images and tested on 746 images of cars with license plates from different angles and distances. The data set was collected from streets with different illumination conditions and was labeled manually. The machine learning approach is used for localizing the license plate because it gives us the best results that image processing techniques cannot give us in many conditions such as differences in, illumination, distances, and angles of the captured image. The evaluation of the license plate detector on the test data produces average precision of 0.9.

Fig. 9. Detected license plate

*3) Image pre-processing*

After the license plate region is determined the license region is cropped and goes through image processing steps to prepare it for digit extraction and recognition.

1. Extract the license plate region.

Fig. 10. Cropped license plate.

2. Convert the license plate to a binary image.

Fig. 11. Binary image of license plate.

3. Skew Detection and Correction.

Skewed license plates cause errors in recognizing license numbers, therefore, correcting skewed license plates is extremely important. Hough transformation is used in this process. The following steps describe the skew detection and correction process:

a) Convert the image to grayscale.
b) Apple Canny or Sobel filter.
c) Find Hough lines between 0.1 to 180-degree angle.
d) Round the angles from line peaks to 2 decimal places.
e) Find the angle with the highest occurrence.
f) Rotate the image with that angle.

Fig. 12. Skew corrected license plate.

4. Remove the black border.

The black border could cause errors in extracting the license's number; therefore, the black border is removed as follows:

a) Perform two-dimensional median filtering on license plate image.
b) Perform hole Filling.
c) NOT (XOR(result of median filtering, result of hole filling)).

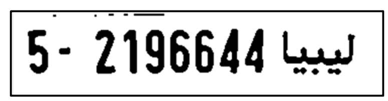

Fig. 13. License image after removing black border.

5. Invert the image color.

Inverting license plate color is important because the numbers and characters need to be white to get extracted using the connected-component labeling.

Fig. 14. Inverted license plate image.

6. Remove small objects.

The license plate is resized to a fixed size, which in our case [136 × 513], and then the objects with a size less than 400 pixels, that cannot be digits or the word "ليبيا", are removed.


Fig. 15. License plate image after removing small objects.

*4) Extract the license number.*

The extraction of a license plate number is done via connected-component labeling. From top to bottom and left to right, the connected-component labeling begins extracting objects from an image. The connected-component extracting method is appropriate given that Libyan license plates are rectangular in shape and include all of their numbers in a single line.

Fig. 16. Extracted license number

*5) License plate recognition*

After extracting the license plate number, the digits, and words are classified using Image category classification by generating a histogram of visual word occurrences that represent the images. This histogram called a bag of visual words or bag of features, is used to train an image category classifier. The classifier is trained with a custom-made dataset of digits and the word "ليبيا" Libya. Five hundred and fifty images are used in total to train and evaluate the classifier. The dataset is divided into 323 images for training and 137 images for evaluating the classifier. The evaluation set produces average accuracy of 0.966.

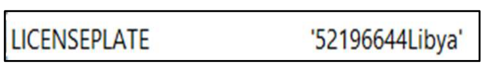


Fig. 17. Result of license plate recognition.

## III. Results and Discussions

We obtained a timing diagram for the system using ModelSim-Intel-ALTER as a means of verifying its functionality. The traffic signals for the three roadways are represented by R1light, R2light, and R3light in the timing diagram, where 100, 010, and 001 indicate green, yellow, and red, respectively. The IR sensors for ROAD1, ROAD2, ROAD3, ROAD1_BACK, and ROAD3_BACK are represented by the inputs IS1, IS2, IS3, IS1_BACK, and IS3_BACK, with numbers ranging from 00000 to 11111 representing the various traffic volumes on every road. EM input represents the passing emergency vehicle on a particular road, for simulation purposes The EM input values are 0xx,100,101, and 110, which sequentially indicate no emergency cars, emergency car on ROAD1, emergency car on ROAD2, and emergency car on ROAD3. State register represents the current state of the traffic system; the delayR1G, delayR2G, and delayR3G, represent the time assigned to ROAD1, ROAD2, and ROAD3 for the green signal, delayRY represents the time for all roads for the yellow traffic signal. The different green light period values are 60, 48, 36, 24, 12, and 1 sec. these values depend on the active IR sensors. Fig. 18 views various traffic light times based on the various traffic densities on each road. Fig. 19 views a case when the ROAD1_BACK is crowded if we grant green light to any road that will cause the problem referred to in Fig. 8 This issue is solved by turning the red lights for all roads to enable the road to become less crowded. Fig. 20 views an emergency car on ROAD3 while the yellow light on ROAD1 is on the traffic system gives the priority to ROAD3 to enable the emergency car to pass, after that, the system returns to the previous state when ROAD1's yellow light was on, and continues on the normal state transitions. Fig. 21 shows the system implementation on FPGA.

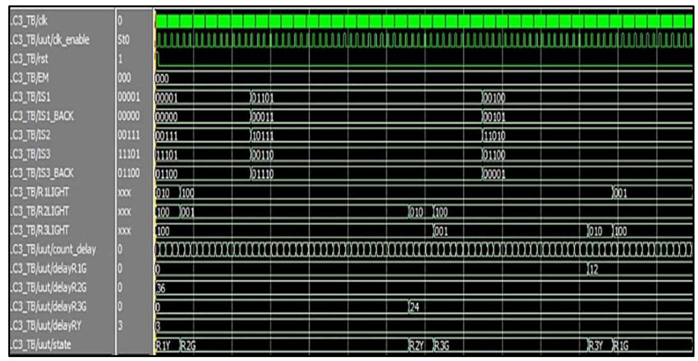

Fig. 18. The timing diagram for the real-time density-based traffic system.

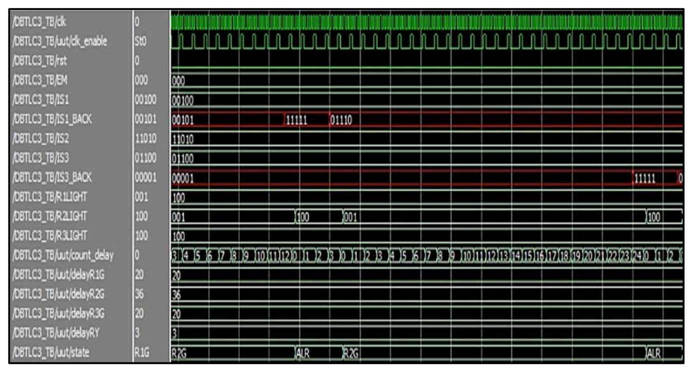

Fig. 19. The timing diagram for the real-time density-based traffic system with road blockage signal.

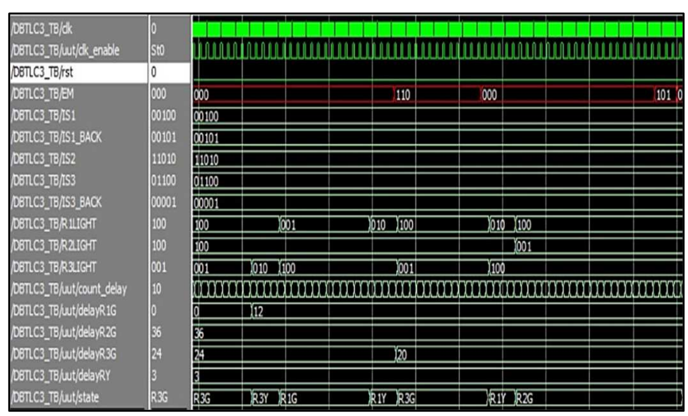

Fig. 20. The timing diagram for the real-time density-based traffic system with an emergency car on ROAD3.

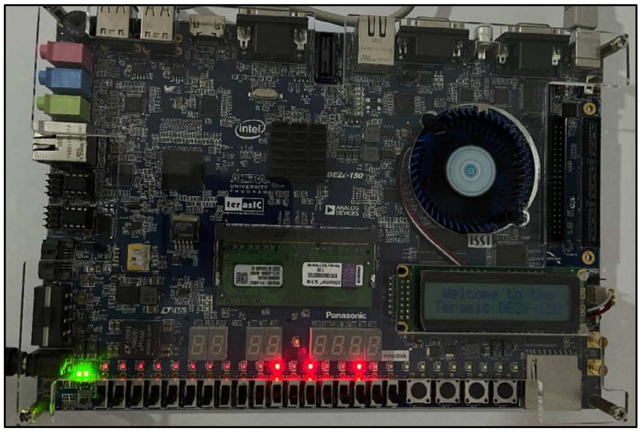


Fig. 21. The traffic system on FPGA.

The Libyan license plate recognition system is implemented using MATLAB. The license plate localization step was implemented using an ACF license plate detector, the detector was trained on 3069 images and tested on 746 images of license plates under different illuminations, distances, and tilted license plates. The evaluation of the license plate detector on the test data produces average precision of 0.9 as shown in Fig. 22. The license plate number Extraction step worked correctly on 70 test images due to the pre-processing step which helped a lot in improving the extracting step. In the license plate number recognition step, the classifier was trained on a custom dataset of digits and the word "ليبيا" Libya. 550 images were used in total to train and evaluate the classifier. The dataset was divided into 323 images for training and 137 images for evaluating the classifier. The evaluation produces average accuracy of 0.966.

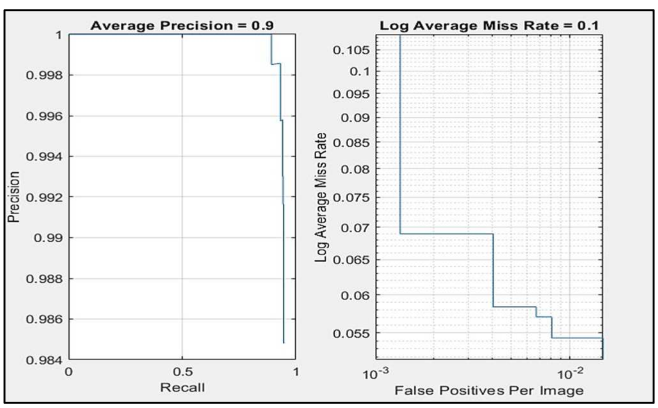


Fig. 22. Evaluation of license plate detector.

## IV. CONCLUSION

The primary contribution of this paper is the development, simulation, and deployment of a real-time traffic light system based on density, and a recognition system for Libyan license plates. The traffic system is designed to alleviate congestion and minimize traffic flow at intersections. Based on the volume of traffic, the system can make decisions in real time and select the best traffic signal time period for each road. The system's key benefits are its ability to prioritize emergency vehicles and make wise decisions to avoid blocking intersections. Following the testing of the traffic light system module, the design underwent simulation and validation with software utilities provided by Intel-Altera. The system was then developed and tested using the Cyclone IV GX: EP4CGX150DF31C8 FPGA board to ensure that it worked as intended in hardware. The system for recognizing license plates in Libya is used to track traffic violations using machine learning and image processing methods. The license plate software is programmed and tested using MATLAB software tool.

## REFERENCES

[1] B. U. Umar, O. M. Olaniyi, A. James, and O. R. Isah, "Traffic violation detection system using image processing," *Computer Engineering and Applications*, Vol. 10, No. 2, June 2021.

[2] J. R. Aworemi, I. A. Abdul-Azeez, A. J. Oyedokun, and J.O.Adewoye, "A study of the causes, effects and ameliorative measures of road traffic congestion in Lagos Metropolis," *European Journal of Social Sciences*, vol. 11, no. 1, pp. 119-128, 2009.

[3] P.K. Singh and P.Daniel, "Advanced real traffic light controller system design using cortex-m0 IP on FPGA," *IEEE International Conference on Advanced Communications, Control, and Computing Technologies*, pp. 1023–1026, 2014.

[4] B. Dilip, Y. Alekhya, and P. D. Bharathi, "FPGA implementation of an advanced traffic light controller using Verilog HDL," *International Journal of Advanced Research in Computer Engineering & Technology (IJARCET)*, vol. 1, no. 7, pp. 2278–1323, 2012.

[5] W. M. El-Medany and M. R. Hussain, "FPGA-based advanced real traffic light controller system design," Proceeding of 4th *IEEE International Workshop on Intelligent Data Acquisition and Advanced Computing Systems: Technology and Applications*, ISBN: 978-1-4244-1347-8, IEEE, pp. 100- 105, 2007.

[6] Y.Liu and X.Chen, "Design of traffic lights controlling system based on PLC and configuration technology," Proceeding of *International Conference on Multimedia Information Networking and Security*, ISBN: 978-0-7695-3843-3, pp.561-563, 2009.

[7] L.A.Ali, M.A.Albakkosh, and M.M.Eljhani," University of Tripoli's Real-Time Density-Based Traffic Light Controller on FPGA, *International Science and Technology Journal*, September 2022.

[8] S.Singh and S. C. Badwaik, "Design and implementation of FPGA-based adaptive dynamic traffic light controller," *International Conference on Emerging Trends in Networks and Computer Communications(ETNCC)*, pp. 324–330, 2011.

[9] S. Bayoumi, E. Korany, and S. Fathy,“ License Plate Recognition system for Egyptian Car Plates,” *International Conference on Image and Video Processing and Computer Vision,* July 2010.

[10] A. M. Hassan, S. A. Ghoul, and A. A.Alkabir. "Libyan Vehicle License Plate Recognition with Support Vector Machine," *Al-Mukhtar Journal of Sciences*, vol. 37, pp. 1--13, 2022.