applied sciences

MDPI

Article

An Intelligent Traffic Light System for Improving Urban Quality of Life: Design and Simulation Scenarios

Mihaela Hnatiuc [1], Mirel Paun [1,*] and Diana-Ioana Urum [2]

[1] Electronics and Telecommunications Department, Constanta Maritime University, 900663 Constanta, Romania; mihaela.hnatiuc@cmu-edu.eu
[2] USONIA S.R.L., 900590 Constanta, Romania; dianahnatiuc9@gmail.com
* Correspondence: mirel.paun@cmu-edu.eu

Abstract

Growing traffic demand, congestion and transport-related emissions continue to challenge urban mobility and place increasing pressure on existing road infrastructure. This study investigates how smart traffic lights can improve traffic flow and support more efficient urban traffic management. The proposed architecture integrates radar sensors for vehicle and pedestrian detection and vehicle speed estimation with air quality, temperature and humidity monitoring. The CW (HB100) and FMCW (K-LC1a) radar front ends achieve speed resolutions of 0.089 m/s (0.322 km/h) and 0.125 m/s (0.45 km/h), respectively, and the FMCW unit achieves a range resolution of 0.834 m. The study combines simulation models developed in LabVIEW with experimental electronic circuits used to evaluate the system behavior under different traffic conditions. Several scenarios were analyzed in which the signal timing was varied according to the number and type of vehicles, the presence of pedestrians and the air quality. A fuzzy logic controller was also implemented to improve the conditions of pedestrians waiting at signalized intersections. The results demonstrate the operation of the proposed architecture in all investigated scenarios and illustrate how the signal timing adapts to changing traffic and environmental parameters. In 30 simulation runs of 15 min at 1200 veh/h, the proposed controller reduced the mean vehicle waiting time by 20% and the pedestrian waiting time by 13% compared with fixed-time control, whereas the estimated CO_2 reduction was small (about 3%). The system is compared with representative approaches from the literature, and its limitations and the remaining research gaps are discussed. The validation presented here is limited to laboratory experiments, LabVIEW simulations and two outdoor gas-sensor campaigns; field tests at real intersections have not yet been performed. Nevertheless, the results indicate that the proposed architecture can support adaptive traffic management and provide a basis for further development and validation in real urban environments.

Keywords: smart city; traffic signal timing; sensor networks; traffic flow; smart traffic light; artificial intelligence; radar sensors; fuzzy control

Check for updates

Academic Editor: Suchao Xie

Received: 17 September 2026
Revised: 18 October 2026
Accepted: 20 October 2026
Published: 22 October 2026

Copyright: © 2026 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license.

1. Introduction

Accelerating urbanization and the steady growth in the number of vehicles have made road traffic a major challenge for modern cities. Congestion, pollution and time lost in traffic directly affect the quality of life of citizens. Smart traffic lights are therefore a concrete step towards more efficient and environmentally friendly cities [1]. Communication technologies are also becoming integral components of intelligent traffic management systems.

One recent approach combines Visible Light Communication (VLC) with learning-based signal control to exchange information between the infrastructure and vehicles; using vehicle position, speed and lane occupancy as inputs, the system dynamically optimizes signal timing, improving intersection efficiency and reducing waiting times [2].

Traditional traffic light systems operate on rigid, predefined schedules, so their signal timings do not adapt automatically to traffic fluctuations. This causes congestion and longer travel times, especially in heavily trafficked areas and during peak periods. At large intersections, where traffic can be much heavier in one direction than in another, conventional traffic lights often perform poorly, leaving vehicles waiting unnecessarily or forming long queues. These inefficiencies not only inconvenience drivers and pedestrians but also contribute to air pollution by increasing the CO_2 emissions of idling vehicles [3].

Smart traffic light systems, in contrast, use sensors and data-processing algorithms to assess traffic conditions and adjust the signal timing accordingly. Instead of following a fixed schedule, they adapt their timing to the number of vehicles in each direction, their speed and other relevant parameters, thereby optimizing traffic flow.

Smart traffic lights collect and process data in real time using a variety of technologies, such as inductive loops, ultrasonic sensors, radar sensors, infrared sensors and video cameras with image analysis [4,5].

Smart traffic lights offer numerous benefits to the cities and regions that implement them:

- Reduced traffic congestion, as dynamically adjusted signal timings shorten waiting times;
- Lower CO_2 emissions, as smoother traffic flow reduces pollutant emissions and improves air quality;
- Increased safety, through real-time detection of dangerous vehicles and traffic situations;
- More efficient public transport, such as buses and trolleybuses, can be given priority at intersections, shortening their travel times;
- Lower operating costs, as traffic and resources are managed more efficiently, which reduces infrastructure maintenance costs.

However, deploying smart traffic light systems is not without challenges. Funding constraints, institutional barriers and the need for public engagement are significant obstacles to successful implementation. Moreover, integrating new technologies with existing infrastructure is complex and requires continuous evaluation and adaptation to ensure effectiveness [6].

Sensor accuracy directly determines how far the control algorithms can reduce waiting times. Camera-based solutions are therefore widely used: besides monitoring traffic, they enforce red-light compliance [7] and detect pedestrians, distinguishing between those who are crossing, waiting or approaching the crosswalk [8]. Passive detectors that require no push-button action, such as pressure-sensitive mats in front of the crossing, also help children, elderly and disabled pedestrians and allow the pedestrian green time to be shortened once everyone has crossed [9].

Other approaches to intelligent traffic lights also exist, but they share the same basic principle: multi-agent systems in which communication and coordination play a central role. Although many such systems have been implemented, no generally optimal solution has yet been found.

The simplest traffic lights rely on a fixed internal clock and cope only with predictable, predefined scenarios (rush hours, holidays); atypical situations require adaptive control based on measured traffic.

Successful implementations of adaptive traffic light systems include the following:

1. SCOOT (Split Cycle Offset Optimization Technique) [10] is an advanced urban traffic control technology that manages traffic lights and optimizes traffic flow in real time. It was developed to reduce the congestion, delays and pollution associated with

urban traffic by means of adaptive algorithms and real-time data. Deployed in cities such as London, SCOOT continuously adjusts the signal timings on the basis of measured traffic data. Its main drawbacks are the significant investment in detectors and software, the need for regular maintenance and calibration of the detectors and the difficulty of configuring them for complex road networks.

2. SURTRAC (Scalable Urban Traffic Control) [11], developed by Carnegie Mellon University and deployed in Pittsburgh, USA, uses cameras and sensors at each intersection and distributed artificial intelligence: decisions are taken locally and coordinated with the neighboring intersections. It has reduced waiting times by 25% and greenhouse gas emissions by 20%.
3. City-scale AI/IoT traffic management [12]. Large cities such as Beijing deploy camera- and IoT-based systems in which a control center adjusts the signal timing to the traffic density and gives priority to public transport or emergency vehicles. The first two systems (SCOOT and SURTRAC) are compared with the proposed system in Section Comparative Analysis.

Reinforcement learning (RL), which allows traffic controllers to learn optimal strategies directly from interaction with the environment, is currently the dominant research direction. Early RL-based traffic control systems used Q-learning to adapt signal timings to the observed traffic states, with rewards associated with reduced delays and queue lengths [13,14]. Deep reinforcement learning significantly improved scalability and decision-making capability: IntelliLight uses Deep Q-Networks (DQN) and traffic-state representations to optimize phase selection, achieving substantial delay reductions compared with traditional methods [15].

Several advanced RL frameworks have emerged in recent years. PressLight incorporates traffic engineering principles through the max-pressure control concept, using queue imbalances as rewards to optimize network performance [16]. CoLight extends this idea with graph-attention mechanisms that allow neighboring intersections to cooperate and share information, improving coordination across large urban networks [17]. DynSTGAT further uses dynamic spatial–temporal graph attention networks to capture both spatial and temporal traffic dependencies, providing greater robustness under highly variable traffic conditions [18].

Recent studies have also addressed the transferability and scalability of RL-based traffic controllers. CrossLight proposes an offline-to-online framework that allows models trained in one city to be adapted efficiently to another, reducing the need for costly retraining [19]. CoSLight goes further by jointly optimizing collaborator selection and decision-making among multiple intersections, outperforming previous cooperative RL methods in large-scale scenarios [20].

Fuzzy logic has been widely investigated as a means of handling uncertainty and nonlinearity in traffic environments. The pioneering fuzzy traffic controller of Pappis and Mamdani used queue lengths and vehicle demand as inputs to determine green-phase durations through linguistic IF–THEN rules [21].

Other intelligent control paradigms have also been explored. Self-organizing traffic lights operate without centralized coordination, adapting the signal phases to local traffic conditions and vehicle interactions [22]. Similar decentralized self-control approaches dynamically regulate traffic flows to prevent congestion from forming [23]. Swarm intelligence techniques inspired by collective behavior have also been applied to real-time signal control, yielding shorter waiting times and queues than conventional fixed-time systems [24].

Varaiya [25] established the theoretical foundation of the max-pressure controller for arbitrary networks of signalized intersections, and later learning-based methods have used

this principle to improve coordination and scalability. Yau et al. [26] reviewed RL models and algorithms for traffic signal control, Wei et al. [27] discussed recent advances in models and evaluation methods, and Agrahari et al. [28] recently reviewed AI-based adaptive traffic signal control, highlighting the growing importance of AI in this field.

Overall, current research shows a clear transition from conventional adaptive control towards AI-driven systems that integrate advanced sensing technologies and machine learning algorithms. Radar sensors provide accurate and privacy-preserving traffic measurements [29,30], while reinforcement learning offers superior adaptability and optimization performance in complex urban environments [3,24,26,29–31]. The most promising future architectures combine radar sensing, vehicle-to-infrastructure communication and multi-agent reinforcement learning to achieve efficient, sustainable and fully adaptive traffic signal control [2,16–20,26–28,30–35].

Although deep reinforcement learning offers high adaptability and strong optimization capability, it also poses several challenges: it requires large amounts of training data and long training times, its decisions are difficult to interpret, its deployment raises safety concerns, and simulated traffic conditions differ from real ones. Practical implementations therefore require robust sensing, reliable communication and validation in realistic urban scenarios.

Despite this progress, the reviewed controllers use only traffic variables (queue length, delay, flow) as inputs, and the exposure of the people waiting at the intersection to exhaust gases, temperature and humidity is not part of the control loop. The specific gap addressed in this paper is therefore the joint use of low-cost radar measurements (speed, distance) and local environmental measurements (gas concentration, temperature, humidity) in a single controller that acts on two actuators: the signal timing and a ventilation unit for the pedestrian waiting area (the gap and the limitations are discussed in detail in Section 4).

In this paper, a smart traffic light system equipped with several types of sensors is designed and simulated with the aim of reducing waiting times and improving air quality. Section 2 describes the proposed system, Section 3 discusses the results and compares the system with existing approaches, Section 4 states the research gap, the limitations and future work, and Section 5 presents the conclusions.

2. Materials and Methods

The proposed system, both simulated and implemented in hardware, includes several parameters that contribute to improving quality of life and comfort. It uses different types of sensors to detect vehicles and pedestrians and to measure exhaust gases, air temperature and humidity, while emergency vehicles request priority through a mobile application. The processed sensor data are used to adjust the traffic light timing and to control a fan motor that improves the air quality for pedestrians. The system is based on Internet of Things (IoT) technology: the data are stored in the cloud, and a controller unit processes them and controls the traffic lights, as shown in Figure 1.

The central controller receives the speed of the vehicle column on each road entering the intersection and correlates it with the pollutant gas concentration; the adaptive algorithm then adjusts the signal timing to minimize pollution and maximize vehicle speeds. Vehicle and pedestrian motion are assessed with continuous-wave (CW) Doppler or frequency-modulated continuous-wave (FMCW) radar sensors.

Compared with cameras, radar sensors require less digital processing and can operate in low-visibility conditions, such as heavy smoke, dust or low light. In addition, radar sensors do not acquire sensitive personal data, as video footage does [29].

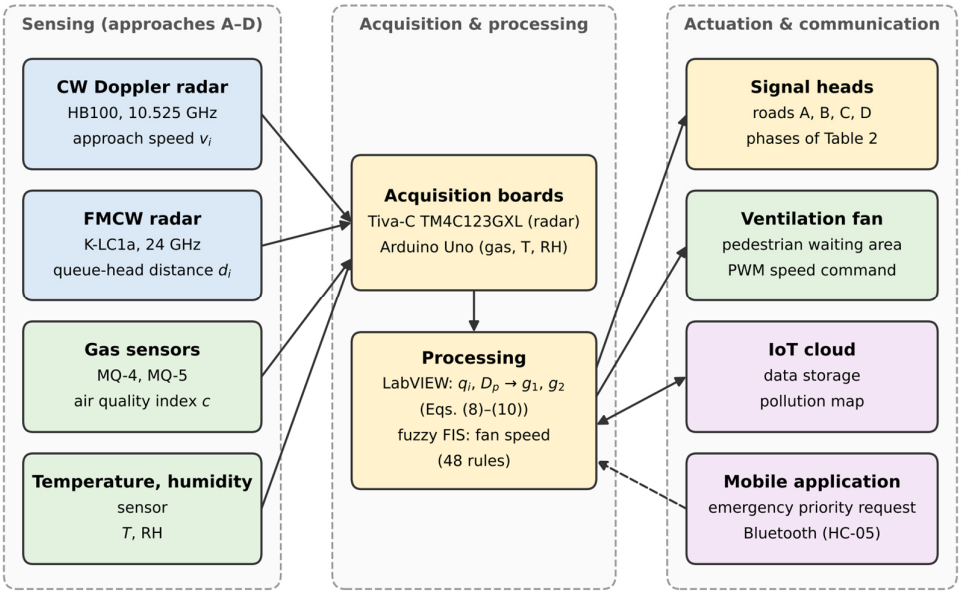


Figure 1. Block diagram of the proposed smart traffic light system: sensing layer (radar, gas, temperature and humidity sensors on each approach), acquisition and processing layer, and actuation and communication layer (signal heads, ventilation fan, IoT cloud and mobile application for emergency priority requests).

The radio frequency (RF) section of the proposed CW radar sensor is an HB100 module, which operates at 10.525 GHz (X band) as a CW Doppler radar with a radiated power (EIRP) of 15 dBm. The intermediate frequency (IF) signal produced by the HB100 module is fed to a two-stage band-pass amplifier based on the MCP602 operational amplifier, shown in Figure 2.

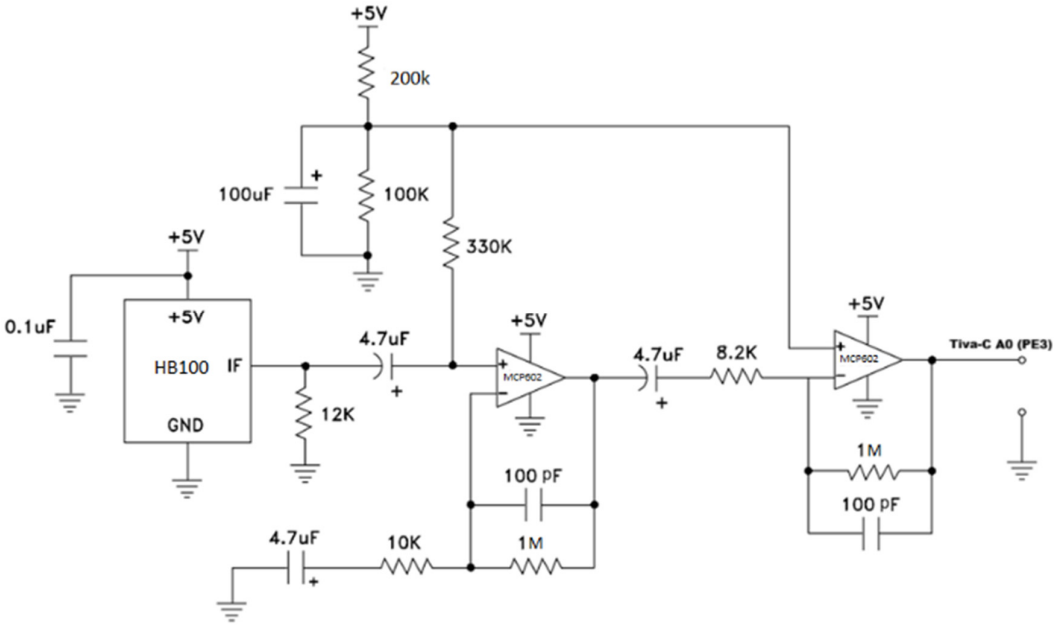


Figure 2. Schematic diagram of the RF and IF sections of the proposed CW Doppler radar sensor (HB100 module and two-stage MCP602 band-pass amplifier).

The amplifier has a bandwidth of 8–980 Hz and a mid-band gain of 82 dB (Figure 3), which corresponds to a maximum measurable target speed of 50 km/h.

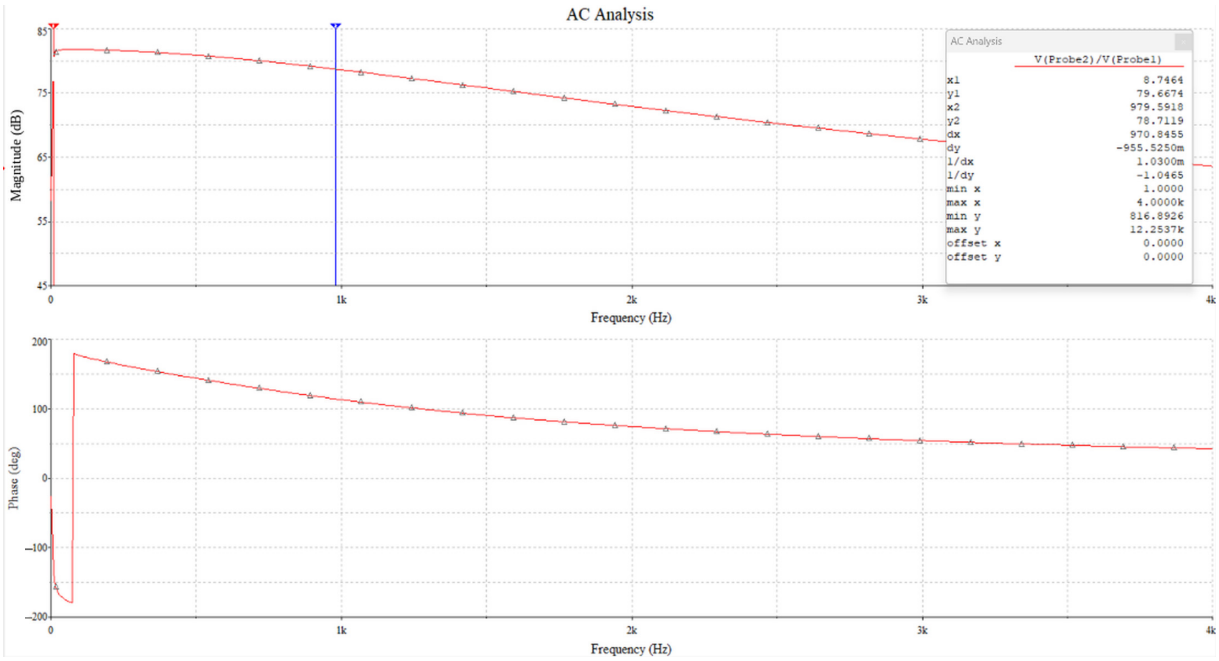


Figure 3. Simulated frequency response (AC analysis) of the IF band-pass amplifier of the CW radar in Figure 2: magnitude (**top**) and phase (**bottom**). The cursors mark the lower and upper cut-off frequencies (8.7 Hz and 980 Hz). The red trace is the simulated response; the red and blue vertical lines are the measurement cursors, whose readings are listed in the inset.

The amplified signal is sampled and buffered by a Tiva-C TM4C123GXL development board, which is built around an ARM Cortex-M4F microcontroller.

The sampling frequency is 125,628 Sa/s with 8-bit resolution, and the buffer depth is 20,000 samples. The acquired signal is transferred over a USB 2.0 connection to a computer running LabVIEW for further digital processing.

The Tiva-C firmware, written in C++ in the Energia IDE, implements the flowchart in Figure 4.

The LabVIEW program implements the flowchart in Figure 5.

The velocity V of the target (vehicle or pedestrian) is computed as

$$V = \frac{f_D \cdot c}{2 \cdot f_T} = \frac{f_D[\text{Hz}] \cdot 3 \cdot 10^8 \left[\frac{\text{m}}{\text{s}}\right]}{2 \cdot 10.525 \cdot 10^9 [\text{Hz}]} = \frac{f_D[\text{Hz}]}{19.49 \left[\frac{\text{Hz}}{\text{km/h}}\right]}, \quad (1)$$

where f_D is the Doppler frequency (the frequency of the dominant spectral component of the IF signal), c is the speed of electromagnetic waves in air, and f_T is the transmitted frequency of the HB100 module.

The frequency resolution ΔF of the FFT is the ratio of the sampling frequency F_s to the buffer depth N:

$$\Delta F = \frac{F_s}{N} = \frac{125,628 \left[\frac{Sa}{s}\right]}{20,000 [Sa]} = 6.28 [\text{Hz}]. \quad (2)$$

This gives a velocity resolution ΔV of

$$\Delta V = \frac{\Delta F \cdot c}{2 \cdot f_T} = \frac{\Delta F[\text{Hz}]}{19.49 \left[\frac{\text{Hz}}{\text{km/h}}\right]} = 0.322 \left[\frac{\text{km}}{\text{h}}\right] = 0.089 \left[\frac{\text{m}}{\text{s}}\right]. \quad (3)$$

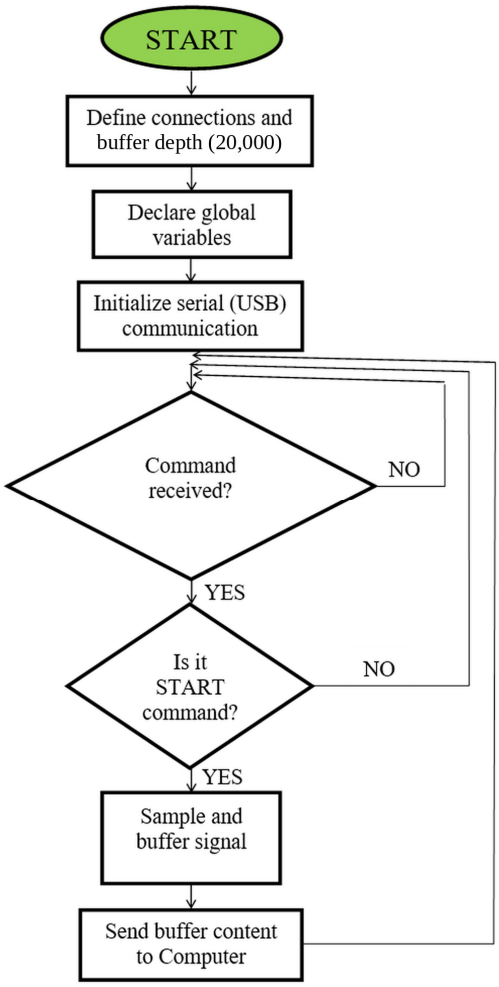


Figure 4. Flowchart of the Tiva-C firmware for the CW Doppler radar (signal acquisition and transfer to the computer).

Figure 6 shows the LabVIEW block diagram that implements the flowchart in Figure 5, including the signal pre-processing (band-pass filtering and FFT) built with virtual instrument (VI) tools. The model was tested in the laboratory through this interface to demonstrate the functionality and applicability of the system. LabVIEW was used as an intermediate step to validate the proposed approach before testing under real operating conditions.

In summary, the CW Doppler radar based on the HB100 RF module is a low-cost solution for detecting vehicles or pedestrians and measuring their velocity in the smart traffic light system. If the decision-making algorithm requires more information about vehicles and pedestrians, the distance between the radar sensor and the head of the vehicle or pedestrian column can be measured with a frequency-modulated continuous-wave (FMCW) radar module.

The K-LC1a module is a low-cost RF front end for an FMCW ranging radar. It operates in the K band at 24 GHz with an output power of 15 dBm EIRP, and its detection range is about 30 m for a car and 10 m for a person. Because it transmits a frequency-modulated continuous wave, the module allows simple, inexpensive radars with practically no blind zone.

The IF section of the proposed FMCW radar contains an amplifier for the 74 Hz–12 kHz range, designed according to the application note of the module, which amplifies the IF signal at the radar module output. It also acts as a band-pass filter that selects the useful signal from the output of the radar module mixer.

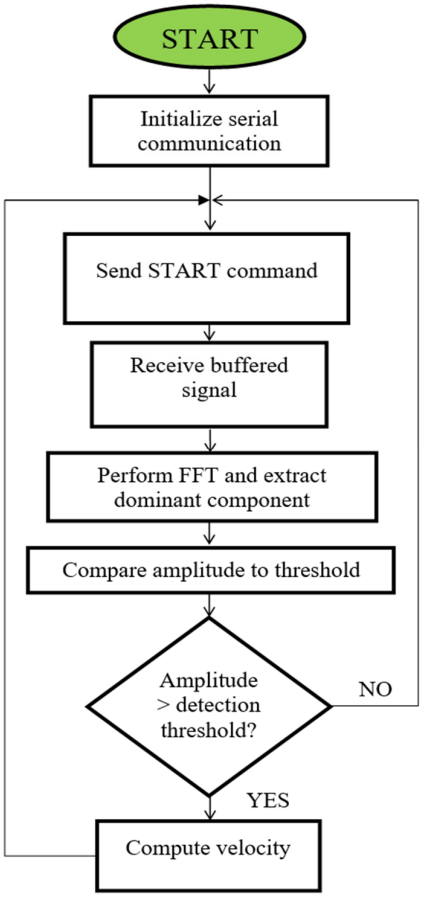


Figure 5. Flowchart of the LabVIEW program for the CW Doppler radar (FFT, detection threshold and velocity computation).

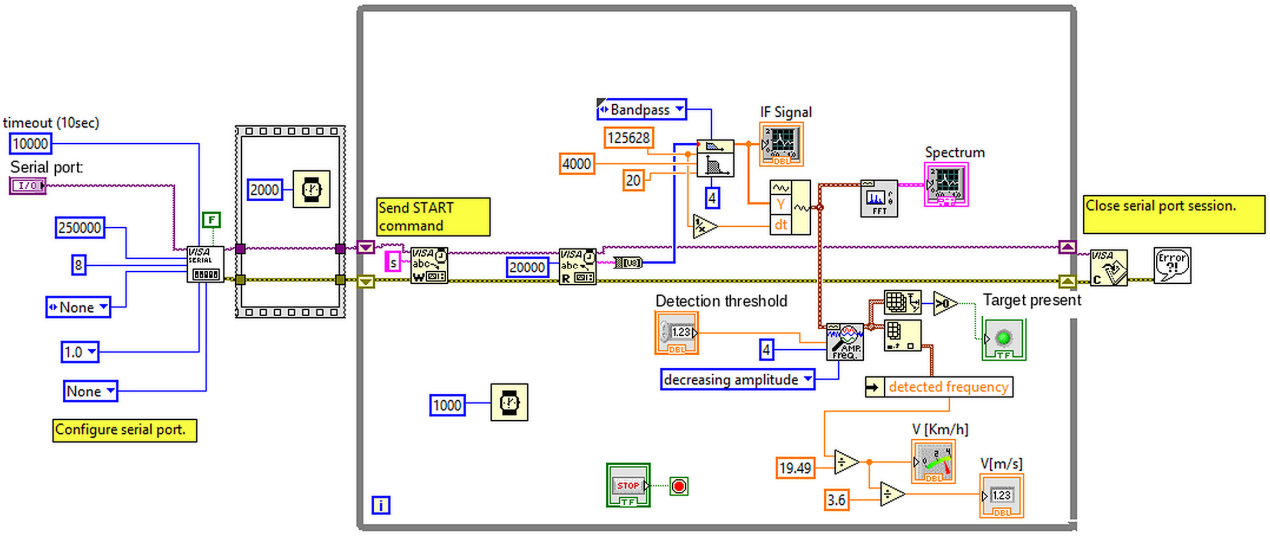


Figure 6. LabVIEW block diagram of the CW radar signal processing: serial port configuration, acquisition of the 20,000-sample buffer, band-pass filtering, FFT, extraction of the dominant frequency, threshold-based target detection and velocity computation in km/h and m/s (Equation (1)). Wire colors follow the LabVIEW data-type convention (orange: floating-point; blue: integer; pink: string; green: Boolean; purple: VISA resource; dark yellow: error cluster).

A digital-to-analog converter (MCP4725) generates the ramp signal that modulates the radar module.

Figure 7 shows the circuit diagram of the IF section described above, and Figure 8 shows the frequency response of the amplifier.

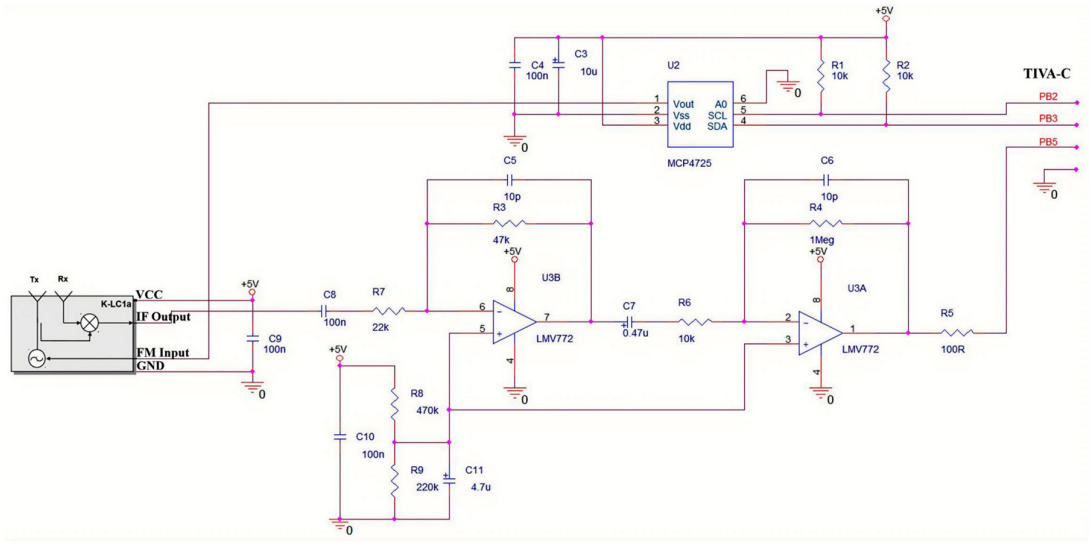


Figure 7. Schematic diagram of the RF and IF sections of the proposed FMCW radar sensor (K-LC1a module, two-stage LMV772 band-pass amplifier and MCP4725 DAC that generates the modulation ramp). The numbers next to the integrated circuits are pin numbers; component labels and values are shown in blue, the Tiva-C pin names in red and the wire junctions in magenta.

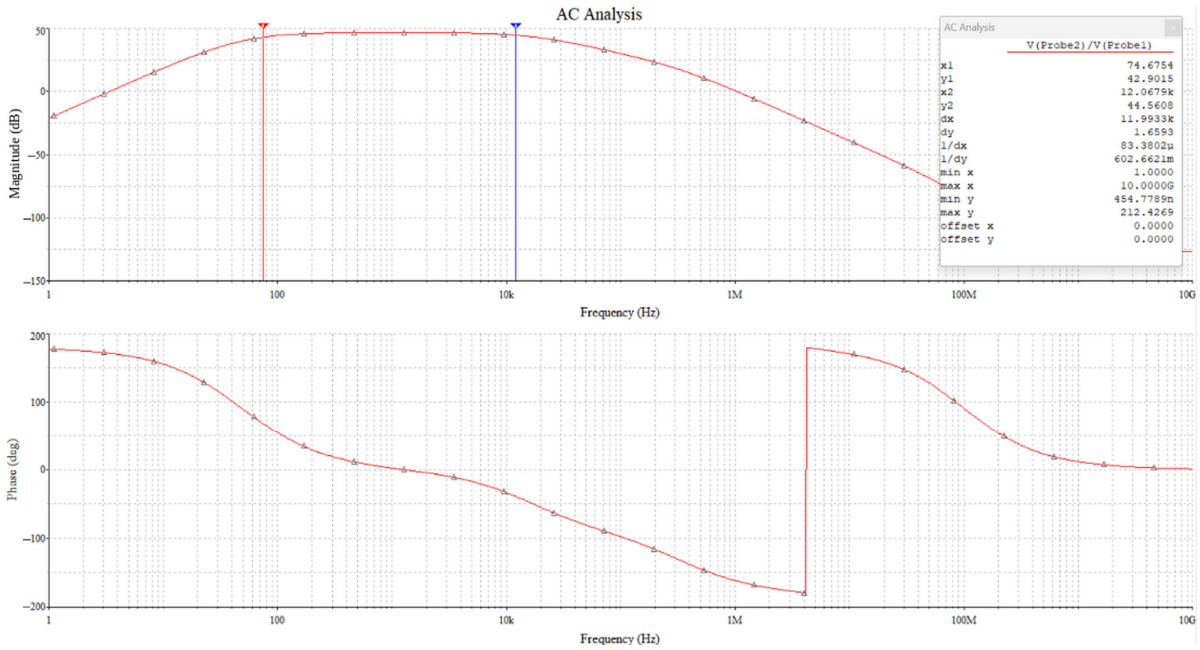


Figure 8. Simulated frequency response (AC analysis) of the IF band-pass amplifier of the FMCW radar in Figure 7: magnitude (**top**) and phase (**bottom**). The cursors mark the lower and upper cut-off frequencies (75 Hz and 12 kHz). The red trace is the simulated response, the dashed lines are the plot grid and the two vertical lines are the measurement cursors, whose readings are listed in the inset.

The IF signal is digitized and processed by a Tiva-C TM4C123GXL development board at a sampling frequency of 20,525 Sa/s with 12-bit resolution and a buffer depth of 820 samples. The acquired signal is transferred over USB 2.0 to the computer running LabVIEW for further digital processing.

The range resolution ΔR of the proposed FMCW radar is

$$\Delta R = \frac{c}{2 \cdot B} \cdot \frac{N}{N_{FFT}} = \frac{3 \cdot 10^8 \left[\frac{\mathrm{m}}{\mathrm{s}}\right]}{2 \cdot 144 \cdot 10^6 [\mathrm{Hz}]} \cdot \frac{820}{1024} = 0.834 [\mathrm{m}] \,, \tag{4}$$

where B = 144 MHz is the bandwidth of the transmitted signal, N = 820 is the IF buffer depth, and N_{FFT} = 1024 is the number of FFT points.

The target velocity is determined from the Doppler effect using a separate constant-frequency scan interleaved between the FM scans, with a resolution of

$$\Delta V = \frac{F_s}{N_{FFT}} \cdot \frac{c}{2 \cdot f_T} = \frac{20,525 \left[\frac{Sa}{s}\right]}{1024 [Sa]} \cdot \frac{3 \cdot 10^8 \left[\frac{m}{s}\right]}{2 \cdot 24 \cdot 10^9 [Hz]} = 0.125 \left[\frac{m}{s}\right] = 0.450 \left[\frac{km}{h}\right]. \tag{5}$$

The Tiva-C firmware, written in C++ in the Energia IDE, implements the flowchart in Figure 9.

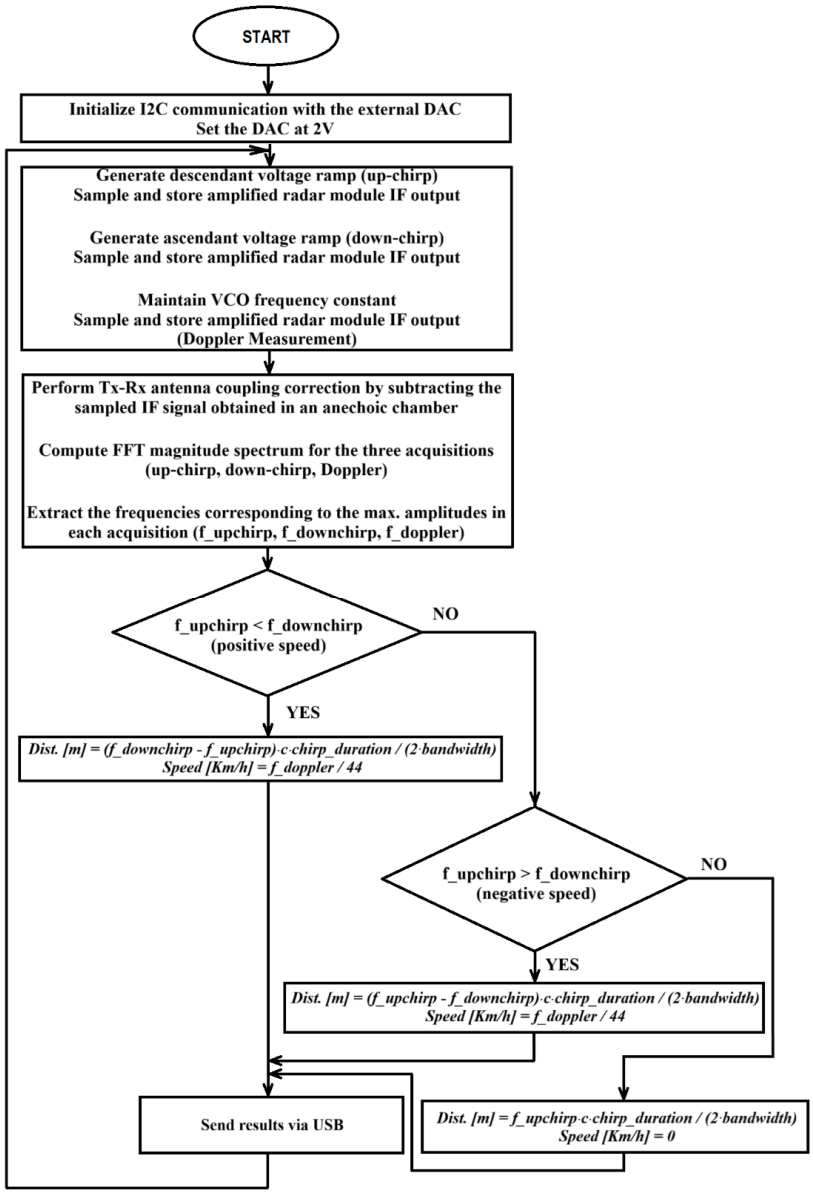


Figure 9. Flowchart of the Tiva-C firmware for the FMCW radar (up-chirp, down-chirp and constant-frequency acquisitions; range and speed computation).

The Tiva-C development board was chosen because of its much higher processing power than the more popular Arduino Uno or Mega boards. Compared with the Arduino Due, which runs at a similar clock frequency (approximately 80 MHz), the Tiva-C offers a more advanced processor with a floating-point unit and a built-in debugger. Moreover, although both boards operate at 3.3 V, the Tiva-C I/O pins are 5 V tolerant, whereas those of the Arduino Due are not.

A limitation of the implementation described above is that it handles only a single dominant target. In the present implementation, the K-LC1a module returns the range of the strongest (dominant) reflector, i.e., the head of the approaching column; the queue length is estimated from this distance, and vehicles are not counted individually. If multiple targets must be analyzed, the XENSIV™ BGT60TR13C module can be used instead. This 60 GHz FMCW radar module has a detection range of 15 m and, thanks to its three receive channels, can determine the distance, velocity and angular position of multiple simultaneous targets, so that the number of vehicles or pedestrians near the sensor can be computed. For easy implementation, the manufacturer offers a demo board with USB connectivity and the Radar Fusion GUI interface shown in Figure 10; however, the cost is significantly higher than that of the K-LC1a system. The BGT60TR13C module was only exercised with the manufacturer's interface; it was neither integrated into the controller nor tested, so multi-target detection and vehicle counting are capabilities of the module, not results of this study.

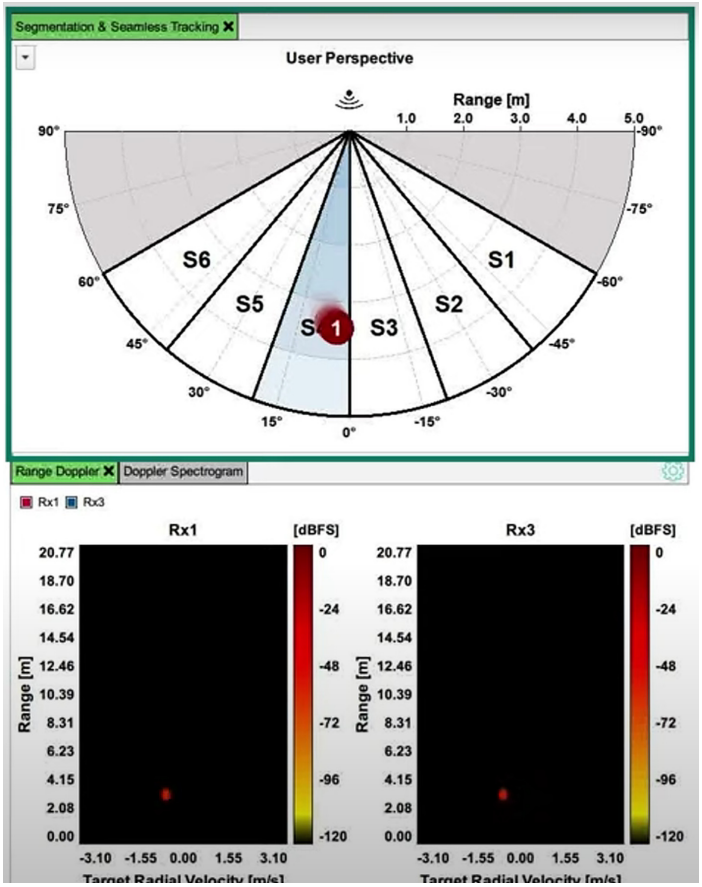


Figure 10. Radar Fusion GUI test interface for the BGT60TR13C FMCW radar with three receive channels [36]: target position in the field of view (**top**) and range–Doppler maps of two receive channels (**bottom**).

Vehicle exhaust gases are known to be harmful to the health of pedestrians. The pollutant gas sensing system of the proposed traffic light is based on two analog sensors,

MQ-4 and MQ-5, connected to an Arduino Uno board. The PuTTY and PLX-DAQ v2 applications were used to communicate with the Arduino board and to display the collected data in real time over a serial interface at 9600 bit/s; PLX-DAQ records the serial data directly into an Excel spreadsheet. This approach facilitated both the verification of the sensors and the analysis of the system behavior in various operating scenarios.

In the first test, measurements were performed in the city of Năvodari, Romania, close to the Petromidia refinery, where pollution levels are higher. The data were collected in real time at 5 s intervals, which allowed the temporal variation in the gas concentrations to be analyzed.

Figure 11 shows the evolution of the LPG and methane concentrations during the experiment. The correlation coefficient between the LPG and methane (CH_4) values is 0.9993. This very high value is a consequence of the processing model of Equation (7): both concentrations are computed from the same R_s/R_0 ratio through monotonic power laws (Table 1), so the two series are almost perfectly correlated regardless of the actual gas composition. The correlation therefore reflects the sensor model and the cross-sensitivity of the sensors rather than an independent physical relationship between methane and LPG emissions.

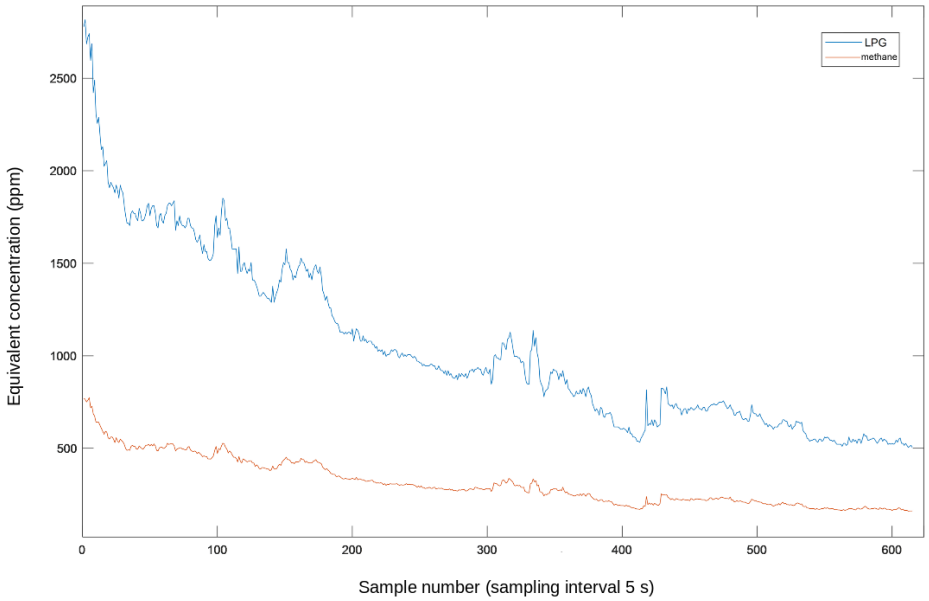


Figure 11. Time evolution of the equivalent LPG and methane (CH_4) concentrations measured near the Petromidia refinery (Năvodari, Romania), with one sample every 5 s. The values are uncalibrated equivalent concentrations (see the discussion of cross-sensitivity below).

The meteorological conditions on the day of the experiment, i.e., a low temperature (10 °C), relatively low humidity (41%), an atmospheric pressure of 1021 hPa and a south-westerly wind, may partly explain the observed gas concentrations:

1. High atmospheric pressure (above 1020 hPa) favors the confinement of pollutant gases close to the ground, which explains the high values initially detected in the urban area.
2. Low temperature and low humidity limit the evaporation and dispersion of some volatile compounds, contributing to a temporary accumulation of gases in low-lying areas.

Table 1. Power-law models used to estimate the equivalent gas concentrations from the R_s/R_0 ratio of the MQ-4 and MQ-5 sensors, with the coefficient of determination R^2 of each fit.

Sensor	Gas	Coefficient k	Exponent m	R^2
MQ-4	CH_4	983.96	−2.837	0.9999
	LPG	3701.5	−3.14	0.9995
	Alcohol	6×10^{10}	−13.93	0.9982
	CO	2×10^{15}	−20.72	0.955
MQ-5	LPG	75.352	−2.528	0.9994
	CH_4	183.49	−2.547	0.9993
	Alcohol	66,308	−4.501	0.9929
	CO	3×10^6	−7.358	0.9661

Model: ppm = $k \cdot (R_s/R_0)^m$, equivalent to Equation (7) with $a = m$ and $b = \log_{10} k$. The values are equivalent concentrations (see the discussion of cross-sensitivity).

When the sensors were moved to a less exposed area, the values decreased significantly, indicating a clear difference between the polluted and the cleaner environment. The overall trend is decreasing, with fluctuations that reflect the real dynamics of gas emissions in the urban environment. The reference resistance R_0 of each sensor is calibrated as

$$R_0 = \frac{R_{S_AIR}}{R0_CLEAN_AIR_FACTOR} \quad (6)$$

where R_{S_AIR} is the average of several measurements of the sensor resistance in clean air, and *R0_CLEAN_AIR_FACTOR* is a sensor-specific coefficient (the R_s/R_0 ratio in clean air given in the datasheet).

The calibration uses 50 samples acquired at 500 ms intervals in clean outdoor air, after the heater pre-heating period recommended by the manufacturer, at an ambient temperature of 20 °C and a relative humidity of 41%. The measurement campaign of Figure 11 took place at 10 °C and 41% RH, i.e., outside the datasheet reference conditions of the MQ sensors (20 ± 2 °C, 65 ± 5% RH), which shifts the R_s/R_0 curves of Table 1 [37].

After calibration, the system computes the gas concentrations in ppm from the ratio R_s/R_0, where R_s is the measured sensor resistance, using a power-law (log–log) relationship specific to each MQ sensor and gas:

$$\text{ppm} = 10^{(a \log_{10}((\frac{R_S}{R_0}))+b)} \quad (7)$$

where the coefficients a and b, which differ for each gas and sensor, were obtained by logarithmic regression of the sensitivity curves provided by the manufacturer.

Table 1 lists the resulting relationships between the gas concentration (ppm) and the ratio of the sensor resistance in the presence of gas (R_s) to the reference resistance in clean air (R_0). The fits have coefficients of determination R^2 between 0.955 and 0.9999, which shows that the regression models reproduce the datasheet curves well and can be used in the acquisition code on the Arduino platform for real-time estimation. Note that a high R^2 only describes the agreement with the datasheet curves, not the measurement accuracy, which is discussed below.

The different exponents reflect the different sensitivities of each sensor to the analyzed gases.

Cross-sensitivity and validation. MQ-4 and MQ-5 are metal-oxide sensors that are not gas-specific: MQ-4 responds mainly to CH_4 but also to LPG, H_2, CO, alcohol and smoke, while MQ-5 responds to LPG and CH_4 but also to CO, H_2 and alcohol; both are affected by temperature and humidity [37]. The concentrations obtained in Table 1 are therefore equivalent concentrations (the datasheet curve of one target gas applied to the measured

R_s/R_0), not the concentrations of the individual gases. This effect was handled as follows: (i) no gas-specific value is used for control; the air quality input of the fuzzy controller is a single dimensionless index derived from both sensors, so that the decision depends on the total load of reducing gases; (ii) the two sensors, which have different CH_4/LPG sensitivity ratios, are used as redundant indicators of the same load; (iii) the readings are presented as relative pollution indicators. Validation against certified reference gas mixtures or a reference gas analyzer and compensation of R_s for temperature and humidity have not yet been carried out; this is a limitation of the present study and is part of the planned work (Section 4).

The second experiment with the gas sensors aimed to map pollutant concentrations in the city of Constanța, Romania. After the data had been sampled at different locations across the city, the 3D pollution map in Figure 12 was created from the raw distribution matrix. No spatial interpolation (kriging, inverse-distance weighting or similar) was applied: the surface in Figure 12 is a direct 3D rendering of the raw matrix of measured values, in which each sampling location is one matrix element and neighboring points are joined by the plotting tool. The map therefore shows only the sampled points and is not an estimate of the concentration field between them. To relate the concentration map to the city map, the graphical interface has a dedicated button that toggles between the concentration graph and an image of the city map.

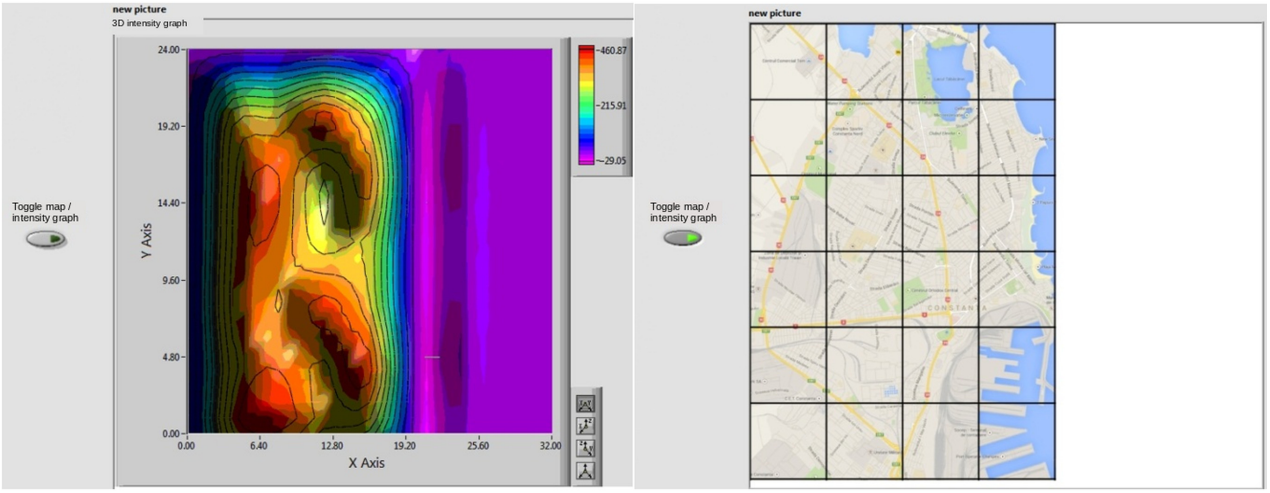


Figure 12. LabVIEW interface for the pollution map of Constanța: 3D intensity graph of the equivalent gas concentration over the sampling grid (**left**); the vertical axis and the color scale give the concentration in ppm and the city map with the same grid (**right**). The toggle button switches between the two views.

To test the proposed traffic light solution, a LabVIEW model of an intersection controller was implemented; its graphical interface is shown in Figure 13.

The implemented system gives priority to the direction with heavier traffic. It also handles priority requests sent by emergency services (police, ambulance) from a mobile application. The application, developed in MIT App Inventor, connects to the controller through a Bluetooth link based on an HC-05 module; its program is shown in Figure 14 and its interface in Figure 15. The HC-05 module implements the Bluetooth Serial Port Profile (a transparent UART link at 9600 baud), and the phone sends one command byte per request. The latency and reliability of this link were not measured in this study; they are listed as limitations in Section 4.

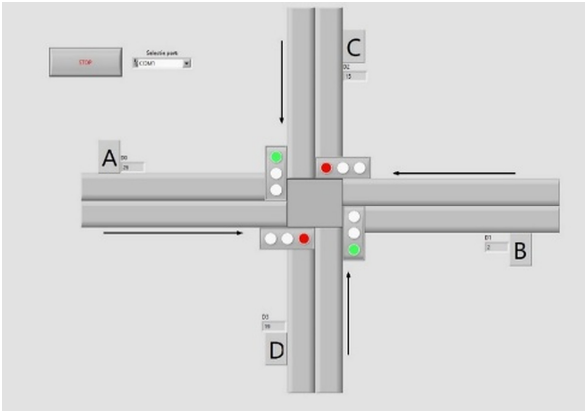


Figure 13. Graphical interface of the LabVIEW intersection model, showing roads A–D, their signal heads and the queue-distance indicator of each approach; the drop-down list selects the serial (Bluetooth) port. The arrows show the direction of traffic on each road; the lit lamps (red or green) show the current state of each signal head.

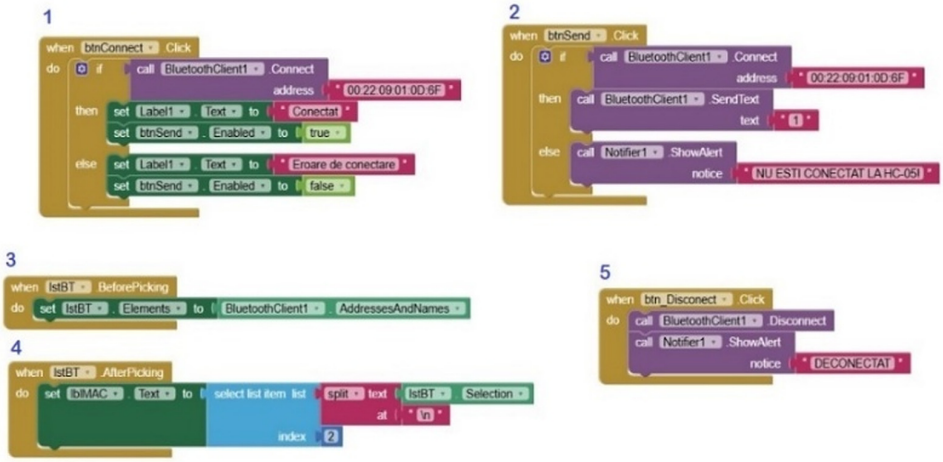


Figure 14. Block-based program of the mobile application in MIT App Inventor: (**1**) connection to the HC-05 module, (**2**) transmission of the priority command, (**3**,**4**) selection of the paired Bluetooth device and (**5**) disconnection. The interface messages are in Romanian (e.g., "Conectat" = connected; "NU EȘTI CONECTAT LA HC-05" = not connected to HC-05).



Figure 15. Interface of the mobile application for emergency priority requests (labels in Romanian: "Semafor Urgență" = emergency traffic light; "Conectare" = connect; "Trimite" = send; "Selectează" = select device).

The intersection simulated in Figure 13 is an intelligent traffic light system that controls the traffic flow in four directions, denoted A, B, C and D: road A (left to right), road B (right to left), road C (top to bottom) and road D (bottom to top). Each road has a three-color traffic light (red, yellow, green).

The "Select COM port (COM1 active)" control indicates that the application is connected through the HC-05 Bluetooth serial port. The traffic lights operate in time cycles that depend on the congestion of each road. For example, if a long queue forms on road D, the system immediately detects it with the FMCW radar distance sensor, switches the light for road D to green and switches the conflicting roads to red to prevent congestion.

Signal control model. Let P_1 = {A, B} and P_2 = {C, D} be the two conflicting phase groups of Table 2. For each approach i, the controller uses three measured variables: the distance d_i from the FMCW radar to the head of the column (queue length), the mean approach speed v_i measured by the HB100 Doppler radar and a normalized pollutant index $c_i \in [0, 1]$ derived from the gas sensors (if a single gas unit serves the whole intersection, $c_i = c$ for all i). The queue length is estimated as

$$q_i = \frac{d_i}{\ell_v} \tag{8}$$

where $\ell_v = 7$ m is the average vehicle spacing in a queue. The demand index of phase P_p is

$$D_p = \frac{1}{|P_p|}\sum_{i\in P_p}\left[w_q \cdot \frac{q_i}{q_{\max}} + w_v \cdot \left(1 - \frac{v_i}{v_f}\right) + w_c \cdot c_i\right],\ w_q + w_v + w_c = 1 \tag{9}$$

with the weighting factors $w_q = 0.5$, $w_v = 0.3$ and $w_c = 0.2$ (priority is given to queue length and speed, i.e., to delay and idling, and a smaller weight to the pollutant index; no sensitivity analysis of the weights was performed), the saturation queue $q_{\max} = 12$ veh and the free-flow speed $v_f = 50$ km/h (the maximum speed measurable by the HB100 channel), so that $D_p \in [0, 1]$. The green time of phase P_p is

$$g_p = \min\{g_{\max}, \max[g_{\min} + (g_{\max} - g_{\min}) \cdot D_p, \xi_p \cdot g_{\text{ped}}]\} \tag{10}$$

where $g_{\min} = 10$ s and $g_{\max} = 60$ s, $\xi_p = 1$ if pedestrians are detected at the crossing served by P_p (0 otherwise), and $g_{\text{ped}} = t_0 + L_c/v_{\text{ped}}$ is the minimum pedestrian green time ($t_0 = 5$ s, crossing length $L_c = 10$ m, walking speed $v_{\text{ped}} = 1$ m/s). The objective is to minimize the demand-weighted red time

$$J = \sum_p D_p \cdot (C - g_p),\ C = g_1 + g_2 + 2y \tag{11}$$

subject to the following constraints: (a) $g_{\min} \le g_p \le g_{\max}$ (the pedestrian minimum overrides the lower bound when $\xi_p = 1$); (b) a yellow interval $y = 3$ s between phases (cases 1 and 3 in Table 2); (c) $C \le 2g_{\max} + 2y$; (d) an emergency request received from the mobile application interrupts Equations (9) and (10): the running phase is terminated through its yellow interval, and the requested direction (or the all-red state of case 5) is maintained until the request is cleared. Because J is linear in g_p, its exact minimizer would assign all the available green time to the phase with the larger D_p; Equation (10) is a bounded proportional rule that approximates this minimizer while preventing starvation of the other phase, i.e., it is a heuristic rather than an optimal solution. The green times behave as follows: (1) light traffic on both phase groups ($D_1, D_2 \approx 0$) gives $g_1 = g_2 = g_{\min}$; (2) heavy, balanced traffic ($D_1, D_2 \approx 1$) gives $g_1 = g_2 = g_{\max}$; (3) unbalanced traffic, e.g., $D_1 = 0.1$ and $D_2 = 0.8$, gives $g_1 = 15$ s and $g_2 = 50$ s; (4) a pedestrian request on a phase with a short green

time extends it to g_{ped} = 15 s; (5) an emergency request overrides all previous cases. The numerical values in cases (3) and (4) are illustrative.

Table 2. Signal states of the four approaches in each phase of the control cycle.

Case	Road A	Road B	Road C	Road D	Phase
1	Yellow	Yellow	Red	Red	A–B clearance
2	Red	Red	Green	Green	C–D green
3	Red	Red	Yellow	Yellow	C–D clearance
4	Green	Green	Red	Red	A–B green
5	Red	Red	Red	Red	All red (emergency)

Cell shading repeats the signal state given in each cell (green, yellow or red).

Table 2 lists the sequence of traffic light phases for the intersection with four roads (A, B, C and D).

The first column gives the case number, columns 2–5 give the signal color shown on each road, and the last column describes the phase.

In case 1, roads A and B are in transition (yellow, warning that the signal is about to turn red), while roads C and D are already stopped. In case 2, roads A and B are stopped and roads C and D have the right of way. In case 3, roads C and D are preparing to stop (yellow) and roads A and B are still stopped. In case 4, roads A and B have the right of way and roads C and D are stopped, after which the cycle repeats. Case 5 is the emergency state, in which all signals are set to red simultaneously.

The right of way therefore alternates between the road pairs A–B and C–D, with yellow transition phases between them and an all-red phase (case 5) for emergencies.

3. Discussion and Comparative Analysis

To improve the air quality for pedestrians waiting at the intersection, a fan that circulates the air can be mounted on the traffic light pole. The fan motor controller implements a fuzzy logic system whose inputs are provided by the temperature, air quality and humidity sensors.

In several situations, the sensors may give very different readings: for example, one sensor may show a medium-to-high value, the second a low value and the third a medium-to-low value. The system must interpret such readings correctly and respond appropriately; therefore, the rule base contains 48 rules that cover these combinations (Figure 16). In the first scenario, the temperature is very low, the humidity is very low and the air quality is good; under these conditions, the corresponding fuzzy rule sets the ventilation speed to a low level. In another scenario, the temperature is low, the humidity is medium and the air quality is good, and the corresponding rule sets a medium (normal) ventilation speed. The three inputs are processed jointly by a single fuzzy inference system (FIS) with three inputs and one output; the rules are not split into three separate two-input models. The three panels of Figure 17 are two-dimensional slices of the same three-input surface, in each of which the third input is held constant.

Fuzzy controller design. The controller has the standard structure of a fuzzy regulator (fuzzification, rule base with inference engine, defuzzification). Its inputs are the temperature (universe of discourse $-10 \ldots 40$ °C), the relative humidity ($0 \ldots 100\%$) and the air quality index (dimensionless, up to at least 350; see Section 2); its output is the fan speed command on the 8-bit PWM scale (the maximum value reached by the rules is $\approx$225). The FIS was designed and simulated with the MATLAB Fuzzy Logic Toolbox; Figure 16 shows its rule viewer.



Figure 16. Rule viewer of the fuzzy fan motor controller (MATLAB version R2025b Fuzzy Logic Toolbox). Inputs: temperature (°C), relative humidity (%) and air quality index (dimensionless); output: ventilation speed (PWM units). Each row is one rule; the yellow areas show the activated part of each membership function for the current input vector, and the last column shows the aggregated output.

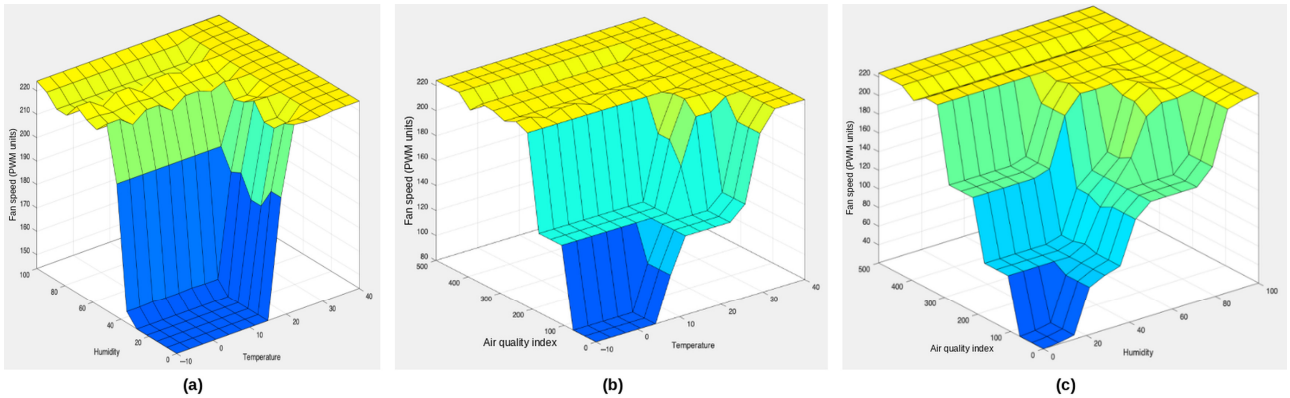


Figure 17. Control surfaces of the fuzzy fan controller (ventilation speed in PWM units): (**a**) temperature (°C) and relative humidity (%), with the air quality index held constant; (**b**) temperature (°C) and air quality index, with the humidity held constant; (**c**) relative humidity (%) and air quality index, with the temperature held constant. The surface color encodes the fan speed (blue: low; yellow: high).

Membership functions. Each input is described by linguistic terms (e.g., very low, low, medium and high for temperature and humidity; good, medium and poor for air quality) with triangular and trapezoidal membership functions (Figure 16), which are well suited to variables with a simple physical meaning. The terms and their breakpoints were chosen from expert knowledge of the comfort ranges and of the working range of the sensors, and were tuned by inspecting the control surfaces (Figure 17) until the output was monotonic and free of abrupt jumps.

Rule base. The 48 rules have the form IF temperature is T_j AND humidity is H_k AND air quality is Q_l THEN fan speed is S_m. They were formulated by expert reasoning according to three design principles: (i) the fan speed increases monotonically with the air quality index, because diluting the pollutants has priority; (ii) temperature and humidity modify the speed according to their distance from the comfort range; (iii) low temperature

and low humidity combined with good air quality keep the speed low to avoid unnecessary airflow and energy use, whereas high temperature, high humidity and poor air quality give the maximum speed. With four terms for temperature, four for humidity and three for air quality, the rule base covers all $4 \times 4 \times 3 = 48$ combinations, including the conflicting cases mentioned above.

Inference and defuzzification. Mamdani inference [38] was used, with the min operator for AND and for implication, the max operator for aggregating the rule outputs and centroid defuzzification. The crisp output is the center of gravity of the aggregated output set:

$$u^* = \frac{\int y \cdot \mu_{\text{agg}}(y)\mathrm{d}y}{\int \mu_{\text{agg}}(y)\mathrm{d}y} \tag{12}$$

Comparison with simpler control and evaluation. A threshold controller (fan switched on when a variable exceeds its limit) switches abruptly at the limits, needs hysteresis to avoid repeated switching and cannot weigh a high value of one variable against low values of the others, whereas the fuzzy rule base produces a smooth output for conflicting readings (Figure 17). This advantage has not yet been quantified: the energy consumption, response time, number of fan switching events and pedestrian exposure of the fuzzy controller have not been compared with those of a threshold controller on the same recorded data; this comparison is part of the planned work (Section 4). These metrics require long-term data recorded with the ventilation unit operating at an intersection; they will be collected in the pilot installation described in Section 4.

Figure 17a shows the ventilation speed as a function of temperature and humidity. If the temperature is below 20 °C and the humidity is below 35%, the ventilation speed is about 100, whereas if the temperature is about 25 °C and the humidity is 35% or higher, the ventilation speed is between 215 and 225.

Figure 17b shows the ventilation speed as a function of temperature and air quality. When the temperature is at most 4 °C and the air quality index is below 70, the ventilation speed is set to 80, since only low ventilation is needed when the air quality is good and the temperature is low. As the temperature rises to 10–30 °C and the air quality index varies between 100 and 200, the ventilation speed increases from approximately 120 to 190, reflecting the combined effect of higher temperatures and deteriorating air quality, which require stronger ventilation to maintain a comfortable and healthy environment.

When the temperature exceeds 35 °C and the air quality index exceeds 250, the ventilation speed reaches its maximum (215–225) to improve thermal comfort and air quality.

Figure 17c shows the ventilation speed as a function of humidity and air quality. The relationship has four distinct operating levels that reflect the combined influence of the two inputs: as humidity and air pollution increase, the controller progressively raises the ventilation speed to maintain a comfortable and healthy local environment.

When the air quality index is below 80 and the humidity is below 17%, the ventilation speed is 0, as no additional ventilation is required. For an air quality index of 100–200 and a humidity of 20–40%, the ventilation speed increases to 50–120. When the air quality index reaches 250–350 and the humidity rises to 50–85%, the ventilation speed increases further to 120–200. Finally, when the air quality index exceeds 350 and the humidity exceeds 85%, the ventilation speed approaches its maximum of approximately 225, ensuring maximum air renewal under critical conditions.

Comparative Analysis

The proposed system differs from existing approaches in that it combines real-time IoT sensor fusion (radar and environmental sensors), adaptive fuzzy logic control and measures that improve quality of life (air quality management for pedestrians). Evolutionary and

swarm-based approaches, including genetic algorithm (GA)-based signal timing optimization [39], fuzzy control combined with differential evolution [1] and swarm-optimization-based real-time control [24], optimize signal timing plans by iteratively searching for near-optimal cycle lengths and phase splits, but they typically rely on offline optimization and do not respond to real-time environmental conditions. Vision-based systems using YOLO [33] enable fast, real-time detection of vehicles and pedestrians, improving responsiveness, but generally do not incorporate air quality or thermal comfort data into control decisions. Fuzzy-logic-based controllers [1,21] classify traffic states and adjust signal timing through rule-based inference; they are interpretable but adapt poorly to multi-variable environmental inputs. More recently, multi-agent reinforcement learning (MARL) has been widely applied to decentralized signal coordination across intersections [3,14,26,32,34,35]; it performs strongly in simulation but requires extensive training data and computational resources, which limits low-cost real-world deployment. In parallel, IoT-based traffic infrastructures incorporating wireless sensor networks and learning-based vehicle management [2,5,6] have introduced connectivity for centralized monitoring, although most of them focus on congestion metrics alone, without jointly addressing pollution, temperature and humidity or pedestrian comfort.

To evaluate the performance of the proposed system, a peak-hour traffic scenario (1200 veh/h) was modelled. The results of 30 simulation runs of 15 min are summarised in Table 3.

Table 3. Performance of the proposed controller compared with fixed-time control (simulated in this study: 1200 veh/h, 15 min) and with values reported for other methods in the literature.

Method	Avg. Vehicle Wait (s)	Queue Length (veh)	Throughput (veh/h)	CO_2 Reduction	Pedestrian Wait (s)
Simulated in this study (1200 veh/h, 15 min, N = 30)					
Fixed-time (baseline)	13.1 ± 0.8	3.6 ± 0.2	1195	0%	16.4 ± 1.6
Proposed system	10.5 ± 0.9	2.8 ± 0.2	1189	2.7 ± 4.4%	14.3 ± 1.7
Reported in the literature (different networks and demand)					
GA-based [39]	28.4 ± 6.1	7.9 ± 2.1	1189	18.7%	59.2 ± 11.2
MARL [3]	26.7 ± 5.3	7.2 ± 1.9	1203	21.3%	61.4 ± 10.8
Fuzzy + DE [1]	34.5 ± 6.3	9.2 ± 2.5	1142	13.8%	55.7 ± 10.5

Values are mean ± SD. The literature values are quoted from the cited works, which used different road networks and demand levels; they are not directly comparable with the simulated values and are given for orientation only. GA = genetic algorithm; MARL = multi-agent reinforcement learning; DE = differential evolution. At 1200 veh/h, the intersection is undersaturated (degree of saturation ≈ 0.44 under fixed-time control), so the throughput equals the demand for both controllers. Rows in italics are group headings separating the values simulated in this study from those reported in the literature.

Simulation set-up. The LabVIEW model of Figure 13 demonstrates the operation of the controller with the sensor data received over Bluetooth, but it does not generate traffic. For a quantitative evaluation, the control law of Equations (8)–(10) and a fixed-time baseline were implemented in a discrete-time intersection simulator written in Python [v2025] (time step 0.5 s; the code is provided as Supplementary Material). Both controllers were simulated with the same demand: a total of 1200 veh/h distributed over approaches A–D (360 veh/h on each of approaches A and B and 240 veh/h on each of approaches C and D, i.e., 30/30/20/20%; one lane per approach and Poisson arrivals), a simulated time of 15 min after a 5 min warm-up, and N = 30 runs with different random arrival sequences. The values in Table 3 are the mean ± standard deviation over these runs (the 95% confidence interval is $\pm 1.96 \cdot \mathrm{SD}/\sqrt{N}$). Queued vehicles discharge with a saturation headway of 2 s (1800 veh/h per lane) after a start-up lost time of 2 s; a vehicle that arrives

on green with an empty queue passes without stopping. The fixed-time baseline uses $g_1 = g_2 = 30$ s (C = 66 s). For the proposed controller, the green time of each phase is computed at the start of the phase with Equations (8)–(10): the queue distance measured by the FMCW radar is limited to its 30 m range and quantized to its 0.834 m range resolution, the approach speed is assumed to decrease linearly from v_f to 0 as the queue grows from 0 to q_max, and the pollutant index is constant (c = 0.3). Pedestrians arrive at each of the two crossings as a Poisson process of 120 ped/h; they cross immediately if they arrive during the green of the phase that serves them with at least 10 s of green remaining, and otherwise at the start of the next green. The same 30 arrival sequences were used for both controllers. The queue length in Table 3 is the maximum queue of each cycle, averaged over the cycles and the four approaches. The GA, MARL and fuzzy rows are taken from the cited works as reported, for different networks and demand levels, and are shown for qualitative orientation only.

CO_2 estimation. For each run, the CO_2 emitted at the intersection was estimated with an idling-and-stop model, $E = e_{idle} \cdot T_{idle} + e_{stop} \cdot N_{stop}$, where T_{idle} is the total time spent stationary by all vehicles (obtained from the simulated waiting times), N_{stop} is the number of stops, e_{idle} is the idling emission rate (0.6 g CO_2/s, corresponding to an idle fuel flow of 0.6 L/h per litre of engine displacement for a 1.6 L petrol engine and 2.3 kg CO_2 per litre of fuel [40]) and e_{stop} is the additional emission of one stop–start cycle (40 g CO_2/stop, estimated from the kinetic energy of a 1300 kg car at 50 km/h (≈125 kJ) and an engine efficiency of about 22% during acceleration, i.e., ≈17 mL of petrol). The CO_2 reduction in Table 3 is $1 - E_{proposed}/E_{fixed}$. With these factors, the 20% reduction in the idling time is partly offset by a slightly larger number of stops (+2.5%), caused by the shorter cycles of the proposed controller, so the estimated CO_2 reduction is small.

The main findings are as follows:

- Compared with fixed-time control, the proposed system reduces the mean vehicle waiting time by 20% (from 13.1 s to 10.5 s), the queue length by 24% and the pedestrian waiting time by 13%, whereas the estimated CO_2 reduction is only 2.7%;
- MARL shows the highest reported throughput but does not consider pedestrian comfort (the pedestrian waiting times reported in the literature were obtained for other crossings and demand levels and are not directly comparable with the simulated values);
- GA-based methods perform well in pure vehicle optimization but provide no environmental monitoring;

The proposed system balances vehicle throughput, pedestrian comfort and air quality.

The differences with respect to the GA, MARL and fuzzy methods are indicative only: the system has not been validated at a real intersection, and the pedestrian waiting time and CO_2 reduction depend on the assumptions listed in the simulation set-up. With a balanced demand (25% per approach), the results were similar: the vehicle waiting time decreased by 21%, the pedestrian waiting time by 9% and the estimated CO_2 emission by 3.8%.

Feasibility analysis. Table 4 compares the estimated deployment time, cost per intersection and power consumption of the proposed system with those of other approaches. These figures are estimates rather than measured deployment data (see limitation 5 in Section 4).

Power budget. The 45 W value in Table 4 is an estimate for one four-approach intersection based on datasheet values: four HB100 modules (5 V, 30–40 mA each; ≈0.8 W in total) [41], four K-LC1a modules (5 V, 35–65 mA each; ≈1.3 W) [42], eight MQ-4/MQ-5 sensor heaters (≤0.75 W and ≤0.8 W each; ≈6.2 W), the Tiva-C and Arduino boards with the IF amplifiers (≈2.5 W) and the HC-05 Bluetooth module (≈0.15 W), i.e., about 11 W

for sensing and acquisition. The remaining ≈34 W is allocated to the edge computer that runs the LabVIEW control application and to the network link. The ventilation fan is not included because its consumption depends on the selected motor and on the duty cycle set by the fuzzy controller; the estimate will be replaced by the measured consumption in the pilot installation.

Table 4. Estimated deployment effort, cost and power consumption of the proposed system and of other approaches.

Approach	Deployment Time	Cost per Intersection (USD)	Power Consumption	Main Cost Items
Proposed system	2–3 days	3500–5000	45 W (continuous, edge processing)	Radar, sensors, controller, installation
GA/PSO-based	1–2 weeks	–	–	Sensor installation, parameter optimization
Vision-based (YOLO)	–	8000–12,000	≥250 W	Cameras, GPU servers, training
DNN/MARL (cloud)	4–8 weeks	15,000–25,000	≥150 W	Sensors, edge/cloud infrastructure, data collection, training, subscription
SCOOT	2–4 weeks	12,000–18,000	–	Inductive loops, controller, configuration

All values are estimates (see limitation 5 in Section 4); – = not estimated. PSO = particle swarm optimization; DNN = deep neural network.

Compared with existing intelligent traffic signal solutions, the proposed architecture introduces a multi-sensor approach that combines radar-based traffic monitoring with environmental sensing and pedestrian comfort management (Table 5).

Table 5. Comparison of the proposed system with representative intelligent traffic signal control approaches.

Ref.	Approach	Sensing	Control Technique	Input Parameters	Main Objective
[1]	Fuzzy traffic signal control	Traffic detectors	Fuzzy logic + differential evolution	Density, queue length, traffic flow	Minimize delay and queue length
[2]	VLC-based intelligent intersection	VLC devices, vehicle communication	Learning-based control	Position, speed, lane occupancy	Improve intersection efficiency
[3]	Adaptive traffic signal control	Traffic detectors	Dual-objective reinforcement learning	Traffic state; CO_2 emissions (indirect)	Reduce delays and CO_2 emissions
[10]	SCOOT	Loop detectors	Adaptive optimization	Flow, occupancy, speed	Traffic flow optimization
[11]	SURTRAC	Cameras, traffic sensors	Distributed artificial intelligence	Vehicles, pedestrians	Waiting-time reduction
This work	Radar- and IoT-based smart traffic light	CW and FMCW radar; gas (MQ-4, MQ-5), temperature and humidity sensors	Adaptive control (Equations (8)–(10)) + fuzzy logic + IoT monitoring	Speed, queue-head distance; equivalent CH_4/LPG concentration, temperature, humidity	Traffic optimization, pollution reduction, pedestrian comfort

Whereas previous studies focus mainly on traffic flow optimization through fuzzy logic [1], reinforcement learning [3] or adaptive control systems such as SCOOT [10] and SURTRAC [11], the proposed system integrates vehicle detection, vehicle speed estimation, air quality assessment and temperature and humidity monitoring within a unified IoT framework. Furthermore, the low-cost CW and FMCW radar sensors provide privacy-preserving traffic monitoring together with speed and distance measurement; vehicle counting would require the multi-target module, which was not tested (Table 6).

Table 6. Features of the proposed system compared with those of the reviewed approaches.

Feature	[1]	[2]	[3]	[10]	[11]	This Work
Real-time adaptive signal timing	✓	✓	✓	✓	✓	✓
Radar-based vehicle detection	✗	✗	✗	✗	✗	✓
Vehicle speed estimation	✗	✓	✓	✓	✓	✓
Vehicle distance (queue) estimation	✗	✗	✗	✗	✗	✓
Multi-target radar capability	✗	✗	✗	✗	✗	(✓) [a]
Pedestrian detection	✗	✗	✗	✗	✓	✓
Air quality monitoring	✗	✗	Partial	✗	✗	✓
Pollution map generation	✗	✗	✗	✗	✗	✓
Temperature and humidity integration	✗	✗	✗	✗	✗	✓
Pedestrian comfort control	✗	✗	✗	✗	✗	✓
Fuzzy environmental controller	Partial	✗	✗	✗	✗	✓
Emergency vehicle prioritization	✗	✗	✗	✗	✗	✓
Mobile application interface	✗	✗	✗	✗	✗	✓
Low-cost implementation	Partial	✗	✗	✗	✗	✓
IoT cloud integration	✗	✓	✗	✗	✗	✓
Validated in real deployment	✗	✗	✗	✓	✓	✗

✓ = supported; ✗ = not supported; Partial = partially supported. [a] Capability of the BGT60TR13C module only; the module was not integrated into the controller or tested in this study.

Tables 5–7 compare features rather than demonstrated performance: SCOOT and SURTRAC have been validated in real deployments for many years, whereas the proposed system has so far been evaluated only in the laboratory and in simulation.

A further novel aspect is the fuzzy-controlled ventilation subsystem designed to improve pedestrian comfort at intersections, a feature not addressed in the reviewed studies. The integration of emergency vehicle prioritization through a mobile application further extends the practical applicability of the proposed architecture.

Table 7 compares the sensing performance of the proposed system with that of other sensing approaches. Together with Table 5, it shows that the proposed system extends the traffic light with gas sensors, distance sensors and an IoT platform in order to shorten waiting times and reduce pollution.

Table 7. Comparison of sensing methods and their performance.

Ref.	Method	Measured Parameters	Reported Accuracy/Resolution	Limitations
[4]	Single magnetic sensor	Vehicle presence, classification	Vehicle classification only	No distance or speed estimation
[30]	FMCW radar + group tracking	Pedestrian position, speed, count	Robust pedestrian detection	Focused only on pedestrian crossings
[29]	Radar + deep learning	Human activity recognition	High recognition accuracy	No traffic optimization
[41]	HB100 Doppler radar	Vehicle speed	Doppler speed estimation	No distance measurement
[42]	K-LC1a FMCW radar	Distance and speed	Range ≈ 30 m for vehicles	Single-target oriented
This work	HB100 + FMCW radar + IoT sensors	Speed, distance, pedestrian detection, pollution, temperature, humidity	Speed resolution: HB100 0.089 m/s (0.322 km/h); FMCW 0.125 m/s (0.45 km/h). Range resolution: 0.834 m	Laboratory validation only; no field tests

4. Research Gap, Limitations and Future Work

Existing traffic signal control systems focus primarily on reducing congestion through adaptive optimization, fuzzy control, reinforcement learning or vehicle-to-infrastructure communication [1–3,10–12]. However, most studies consider only traffic-related parameters such as vehicle flow, density, queue length or travel time. To the best of the authors' knowledge, and without claiming an exhaustive systematic review, none of the approaches reviewed here simultaneously integrates radar-based vehicle detection, vehicle speed and distance estimation, environmental monitoring (air quality, temperature and humidity), pedestrian comfort enhancement, emergency vehicle prioritization, IoT connectivity and fuzzy environmental control within a single traffic management architecture.

The proposed system addresses these gaps with a multi-sensor, IoT-based smart traffic light platform that combines traffic optimization with environmental awareness and pedestrian well-being. In addition to adaptive traffic control, the system uses radar sensors for privacy-preserving vehicle and pedestrian detection, generates pollution maps and dynamically controls a ventilation subsystem through fuzzy logic to improve pedestrian comfort. Furthermore, emergency vehicles can request priority through a dedicated mobile application, a feature rarely integrated into existing smart traffic light solutions. The proposed architecture thus extends the functionality of current intelligent traffic control systems beyond traffic efficiency towards a more comprehensive smart city framework.

Contribution of the integration. The new capability lies not in a single component but in closing the loop between traffic sensing and environmental sensing in one low-cost node: (i) the radar data that drive the signal timing are read together with the local gas, temperature and humidity data, so that the exhaust exposure of waiting pedestrians becomes a control variable rather than only a post hoc CO_2 metric; (ii) one shared state drives two actuators, the signal timing and the local ventilation, which none of the reviewed systems does; (iii) the same sensors produce a pollution map that can be used to rank intersections. Beyond the sum of its parts, this study demonstrates, with stated resolutions and a joint simulation, that these functions can share one low-cost sensing and processing chain. Whether the combination actually reduces pedestrian exposure remains to be quantified (Section 5).

Limitations and validation. (1) No field test was carried out at a real intersection. The LabVIEW model was used as an intermediate step because a test on a public road requires the approval of the road administrator and safety measures that were not available; the radar and gas hardware were tested in the laboratory and outdoors, but the controller was run in simulation. (2) The CW radar handles a single dominant target, and the multi-target module was not tested. (3) The MQ sensors were not validated against a reference analyzer and provide equivalent concentrations. (4) The fuzzy fan controller was not compared quantitatively with a threshold controller. (5) The energy, cost and installation figures are estimates. (6) The latency and reliability of the Bluetooth link were not measured. (7) The detection accuracy of the radar sensors (probability of detection, false-alarm rate and speed/range error with respect to a reference instrument) was not measured; only the theoretical resolutions are reported. Simulation-only validation of the controller is appropriate at this stage because it allows the proposed and the fixed-time controllers to be compared under identical, repeatable demand, which is not possible on a public road; it does not, however, replace the field evaluation planned below. The planned validation consists of hardware-in-the-loop tests of the radar and gas units connected to the LabVIEW model; calibration of the gas sensors with certified gas mixtures at controlled temperature and humidity; a quantitative comparison of the fuzzy and threshold controllers; and a pilot installation at one intersection, using the PCI and MOS indicators defined in Section 5 as evaluation metrics.

Future research will extend the proposed architecture with advanced artificial intelligence techniques and additional sensing capabilities. A first direction is to feed radar-derived traffic parameters, such as vehicle count, queue length, traffic density and pedestrian density, into the fuzzy inference system, so that the controller can consider traffic efficiency, environmental quality and pedestrian comfort simultaneously.

Furthermore, the current rule-based fuzzy controller could be upgraded to an adaptive neuro-fuzzy inference system (ANFIS), which would optimize the membership functions and fuzzy rules automatically from real-world traffic and environmental data.

Another important direction is multi-objective decision-making that jointly optimizes vehicle waiting time, pedestrian waiting time, air quality and energy consumption. In addition, machine learning algorithms could predict short-term traffic conditions and pollution levels, enabling proactive rather than reactive control. Multi-agent reinforcement learning would further allow neighboring intersections to cooperate, leading to network-wide optimization and coordinated green-wave operation.

Future work will also include deploying the system in real urban environments and evaluating its long-term performance under different traffic and weather conditions. Large-scale field trials would quantify the impact of the architecture on traffic congestion, pollutant exposure, pedestrian comfort and emergency vehicle response times. Ultimately, the combination of radar sensing, IoT connectivity, environmental monitoring, fuzzy control and predictive analytics could turn the proposed solution into a comprehensive smart city platform that supports safer, healthier and more sustainable urban mobility.

In particular, a new machine learning algorithm for traffic control will be developed. The key parameter is time, which can be predicted from information about the current traffic light and its neighbors. A region-of-interest (ROI) approach covering the relevant traffic lights could therefore help predict the travel time of ordinary vehicles and of emergency vehicles such as police cars and ambulances.

5. Conclusions

The intelligent traffic light system proposed in this paper uses radar sensors for vehicle detection together with air quality, humidity and temperature sensors. Its control algorithm

aims to optimize traffic and reduce pollution, and the system improves the comfort of pedestrians waiting at the traffic light by circulating the air according to local conditions. Quantitatively, the CW radar achieved a speed resolution of 0.089 m/s (maximum measurable speed 50 km/h), and the FMCW radar achieved a speed resolution of 0.125 m/s and a range resolution of 0.834 m (range $\approx$ 30 m for cars). In 30 simulation runs of 15 min at 1200 veh/h, the proposed controller reduced the mean vehicle waiting time from 13.1 s to 10.5 s (-20%), the queue length from 3.6 to 2.8 vehicles (-24%) and the pedestrian waiting time from 16.4 s to 14.3 s (-13%), while the estimated CO_2 emission decreased by only 2.7%, compared with fixed-time control. These are simulation results: the system has not been tested at a real intersection, the gas readings are uncalibrated equivalent concentrations, and the benefit of the fuzzy-controlled ventilation for pedestrians has not been measured.

The main expected benefits of the system are the following:

1. Shorter waiting times: vehicles and public transport spend less time at intersections, which reduces congestion.
2. Lower CO_2 emissions: smoother traffic reduces idling at intersections and thus pollution.
3. Increased safety: intelligent intersection management with real-time monitoring reduces the risk of accidents.
4. Improved comfort for pedestrians waiting at the intersection.

To evaluate the effectiveness of the proposed fuzzy pedestrian comfort controller quantitatively, a set of objective and subjective performance indicators should be considered. Objective metrics include air-quality comfort, thermal comfort, pollutant exposure and average pedestrian exposure time, while subjective indicators can be obtained from user questionnaires through Mean Opinion Score (MOS) analysis. Furthermore, a composite Pedestrian Comfort Index (PCI) can be defined by combining air-quality, thermal and exposure-related parameters into a single metric. Such an approach would allow a comprehensive assessment of the impact of the proposed system on pedestrian well-being and a direct comparison with conventional traffic management solutions. These indicators (PCI and MOS) were not computed in the present study; they define the evaluation protocol of the planned fieldwork.

Supplementary Materials: The following supporting information can be downloaded at https://www.mdpi.com/article/10.3390/app162110456/s1, the LabVIEW VI files for radar signal processing and for the intersection model, the Tiva-C and Arduino source code and the Python code of the intersection simulation.

Author Contributions: Conceptualization, M.H.; methodology, M.H. and D.-I.U.; software, M.P.; validation, M.H. and M.P.; formal analysis, M.P.; investigation, M.H. and D.-I.U.; resources, M.P.; data curation, M.H.; writing—original draft preparation, M.H.; writing—review and editing, M.P. and D.-I.U.; visualization, M.P.; supervision, M.H.; project administration, M.H.; funding acquisition, M.H. All authors have read and agreed to the published version of the manuscript.

Funding: This work was supported by the Research Funds of Constanta Maritime University, 2026.

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: This work is supported by "Research Funds of the Constanta Maritime University", 2026. The original contributions presented in this study are included in the article and its Supplementary Materials. Further inquiries can be directed to the corresponding author.

Acknowledgments: The authors would like to thank the students who helped during the development and testing of the proposed systems.

Conflicts of Interest: Author Diana-Ioana Urum is employed by the company USONIA S.R.L. The remaining authors declare that the research was conducted in the absence of any commercial or financial relationships that could be construed as a potential conflict of interest.

References

1. Lin, H.; Han, Y.; Cai, W.; Jin, B. Traffic Signal Optimization Based on Fuzzy Control and Differential Evolution Algorithm. *IEEE Trans. Intell. Transp. Syst.* **2023**, *24*, 8555–8566. [CrossRef]
2. Vieira, M.A.; Galvão, G.M.; Vieira, M.; Louro, P.; Vestias, M.; Vieira, P. Enhancing urban intersection efficiency: Visible light communication and learning-based control for traffic signal optimization and vehicle management. *Symmetry* **2024**, *16*, 240. [CrossRef]
3. Zhang, G.; Chang, F.; Huang, H.; Zhou, Z. Dual-objective reinforcement learning-based adaptive traffic signal control for decarbonization and efficiency optimization. *Mathematics* **2024**, *12*, 2056. [CrossRef]
4. Cheung, S.Y.; Coleri, S.; Dundar, B.; Ganesh, S.; Tan, C.W.; Varaiya, P. Traffic measurement and vehicle classification with a single magnetic sensor. *Transp. Res. Rec.* **2005**, *1917*, 173–181. [CrossRef]
5. Chen, W.J.; Chen, L.F.; Chen, Z.L.; Tu, S.L. A wireless sensor network for intelligent transportation system. In Proceedings of the First International Multi-Symposium of Computer and Computational Sciences (IMSCCS—06), Hangzhou, China, 20–24 June 2006; Volume 2, pp. 635–641.
6. Djahel, S.; Doolan, R.; Muntean, G.-M.; Murphy, J. A Communications-Oriented Perspective on Traffic Management Systems for Smart Cities: Challenges and Innovative Approaches. *IEEE Commun. Surv. Tutor.* **2015**, *17*, 125–151. [CrossRef]
7. Cohn, E.G.; Kakar, S.; Perkins, C.; Steinbach, R.; Edwards, P. Red light camera interventions for reducing traffic violations and traffic crashes: A systematic review. *Campbell Syst. Rev.* **2020**, *16*, e1091. [CrossRef]
8. Ahmed, S.; Huda, M.N.; Rajbhandari, S.; Saha, C.; Elshaw, M.; Kanarachos, S. Pedestrian and Cyclist Detection and Intent Estimation for Autonomous Vehicles: A Survey. *Appl. Sci.* **2019**, *9*, 2335. [CrossRef]
9. Larson, T.; Wyman, A.; Hurwitz, D.S.; Dorado, M.; Quayle, S.; Shetler, S. Evaluation of dynamic passive pedestrian detection. *Transp. Res. Interdiscip. Perspect.* **2020**, *8*, 100268. [CrossRef]
10. Bing, B.; Carter, A.T. SCOOT: The World's Foremost Adaptive Traffic Control System. In *Traffic Technology International '95*; UK and International Press: Dorking, UK, 1995; pp. 176–180.
11. SURTRAC: Scalable Urban Traffic Control. Available online: https://www.ri.cmu.edu/pub_files/2013/1/13-0315.pdf (accessed on 6 July 2026).
12. Zhang, J.; Wang, F.-Y.; Wang, K.; Lin, W.-H.; Xu, X.; Chen, C. Data-Driven Intelligent Transportation Systems: A Survey. *IEEE Trans. Intell. Transp. Syst.* **2011**, *12*, 1624–1639. [CrossRef]
13. Abdulhai, B.; Pringle, R.; Karakoulas, G.J. Reinforcement Learning for True Adaptive Traffic Signal Control. *Transp. Eng.* **2003**, *129*, 278–285. [CrossRef]
14. Wiering, M.A. Multi-Agent Reinforcement Learning for Traffic Light Control. In Proceedings of the International Conference on Machine Learning (ICML), Stanford, CA, USA, 29 June–2 July 2000; pp. 1151–1158.
15. Wei, H.; Zheng, G.; Yao, H.; Li, Z. IntelliLight: A Reinforcement Learning Approach for Intelligent Traffic Light Control. In Proceedings of the ACM SIGKDD International Conference on Knowledge Discovery & Data Mining (KDD), London, UK, 19–23 August 2018; pp. 2496–2505. [CrossRef]
16. Wei, H.; Chen, C.; Zheng, G.; Wu, K.; Gayah, V.; Xu, K.; Li, Z. PressLight: Learning Max Pressure Control to Coordinate Traffic Signals in Arterial Network. In Proceedings of the ACM SIGKDD (KDD), Anchorage, AK, USA, 4–8 August 2019; pp. 1290–1298. [CrossRef]
17. Wei, H.; Xu, N.; Zhang, H.; Zheng, G.; Zang, X.; Chen, C.; Zhang, W.; Zhu, Y.; Xu, K.; Li, Z. CoLight: Learning Network-level Cooperation for Traffic Signal Control. In Proceedings of the ACM International Conference on Information and Knowledge Management (CIKM), Beijing, China, 3–7 November 2019; pp. 1913–1922. [CrossRef]
18. Wu, Y.; Xu, H.; Wu, J.; Ma, T. DynSTGAT: Dynamic Spatial-Temporal Graph Attention Network for Traffic Signal Control. In Proceedings of the ACM CIKM, Gold Coast, QLD, Australia, 1–5 November 2021. [CrossRef]
19. Ruan, S.; Wang, L.; Xie, X.; Li, Z.; Yu, B. CrossLight: Offline-to-Online Reinforcement Learning for Cross-City Traffic Signal Control. In Proceedings of the ACM SIGKDD (KDD), Barcelona, Spain, 25–29 August 2024; pp. 2765–2774. [CrossRef]
20. Jiang, H. CoSLight: Co-optimizing Collaborator Selection and Decision-making to Enhance Traffic Signal Control. In Proceedings of the ACM SIGKDD (KDD), Barcelona, Spain, 25–29 August 2024; pp. 2500–2511. [CrossRef]
21. Pappis, C.P.; Mamdani, E.H. A Fuzzy Logic Controller for a Traffic Junction. *IEEE Trans. Syst. Man. Cybern.* **1977**, *7*, 707–717. [CrossRef]
22. Gershenson, C. Self-Organizing Traffic Lights. *Complex Syst.* **2005**, *16*, 29–53. [CrossRef]

23. Lämmer, S.; Helbing, D. Self-Control of Traffic Lights and Vehicle Flows in Urban Road Networks. *Stat. Mech. Theory Exp.* **2008**, *2008*, P04019. [CrossRef]
24. Celtek, M. Swarm optimization-based real-time traffic signal control. *Measurement* **2020**, *166*, 108206. [CrossRef]
25. Varaiya, P. The Max-Pressure Controller for Arbitrary Networks of Signalized Intersections. *Transp. Res. Part C Emerg. Technol.* **2013**, *36*, 177–195. [CrossRef]
26. Yau, K.-L.A.; Qadir, J.; Khoo, H.L.; Ling, M.H.; Komisarczuk, P. A survey on reinforcement learning models and algorithms for traffic signal control. *ACM Comput. Surv.* **2017**, *50*, 1–38. [CrossRef]
27. Wei, H.; Zheng, G.; Gayah, V.; Li, Z. Recent Advances in Reinforcement Learning for Traffic Signal Control: A Survey of Models and Evaluation. *ACM SIGKDD Explor. Newsl.* **2021**, *22*, 12–18. [CrossRef]
28. Agrahari, A. Artificial Intelligence-Based Adaptive Traffic Signal Control: Review. *Electronics* **2024**, *13*, 3875. [CrossRef]
29. Diraco, G.; Gabriele, R.; Leone, A. Radar-Based Activity Recognition in Strictly Privacy-Sensitive Settings Through Deep Feature Learning. *Biomimetics* **2025**, *10*, 243. [CrossRef]
30. Nimac, P.; Krpič, A.; Batagelj, B.; Gams, A. Pedestrian Traffic Light Control with Crosswalk FMCW Radar and Group Tracking Algorithm. *Sensors* **2022**, *22*, 1754. [CrossRef]
31. Behrisch, M.; Bieker, L.; Erdmann, J.; Krajzewicz, D. SUMO—Simulation of Urban MObility: An Overview. In Proceedings of the Third International Conference on Advances in System Simulation SIMUL 2011, Barcelona, Spain, 23–29 October 2011; pp. 63–68.
32. Zhang, H.; Feng, S.; Liu, C.; Ding, Y.; Zhu, Y.; Zhou, Z.; Zhang, W.; Yu, Y.; Jin, H.; Li, Z. CityFlow: A Multi-Agent Reinforcement Learning Environment for Large Scale City Traffic Scenario. In Proceedings of the Web Conf. (WWW), The Web Conference 2019 (WWW '19), San Francisco, CA, USA, 13–17 May 2019; pp. 3620–3624. [CrossRef]
33. Redmon, J.; Divvala, S.; Girshick, R.; Farhadi, A. You Only Look Once: Unified, Real-Time Object Detection. In Proceedings of the 29th IEEE Conference on Computer Vision and Pattern Recognition CVPR 2016, Las Vegas, NV, USA, 27–30 June 2016. [CrossRef]
34. Prabuchandran, K.J.; Kumar, A.N.H.; Bhatnagar, S. Multi-agent reinforcement learning for traffic signal control. In Proceedings of the IEEE 17th Int. Conf. Intelligent Transportation Systems (ITSC), Qingdao, China, 8–11 October 2014; pp. 2529–2534. [CrossRef]
35. Chu, T.; Wang, J.; Codecà, L.; Li, Z. Multi-Agent Deep Reinforcement Learning for Large-Scale Traffic Signal Control. *IEEE Trans. Intell. Transp. Syst.* **2020**, *21*, 1086–1095. [CrossRef]
36. Radar Fusion GUI. Available online: https://softwaretools.infineon.com/tools/com.ifx.tb.tool.radarfusiongui (accessed on 6 July 2026).
37. Fine, G.F.; Cavanagh, L.M.; Afonja, A.; Binions, R. Metal Oxide Semi-Conductor Gas Sensors in Environmental Monitoring. *Sensors* **2010**, *10*, 5469–5502. [CrossRef]
38. Mamdani, E.H.; Assilian, S. An experiment in linguistic synthesis with a fuzzy logic controller. *Int. J. Man.-Mach. Stud.* **1975**, *7*, 1–13. [CrossRef]
39. Mao, T.; Mihăiță, A.-S.; Chen, F.; Vu, H.L. Boosted Genetic Algorithm Using Machine Learning for Traffic Control Optimization. *IEEE Trans. Intell. Transp. Syst.* **2022**, *23*, 7112–7141. [CrossRef]
40. Natural Resources Canada. Idling Reduction: Arriving at Calculations. Available online: https://www.nrcan.gc.ca/energy/efficiency/communities-infrastructure/transportation/idling/4457 (accessed on 18 October 2026).
41. Agilsense. *HB100 Microwave Sensor Module Datasheet*, Version 1.02; Agilsense: Singapore. Available online: https://www.theremino.com/wp-content/uploads/files/HB100_Microwave_Sensor_Module_Datasheet.pdf (accessed on 18 October 2026).
42. RFbeam Microwave GmbH. *K-LC1a Radar Transceiver, Datasheet, Revision 2.1*; RFbeam Microwave GmbH: St. Gallen, Switzerland, 2020.

Disclaimer/Publisher's Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.