

Article

On the Use of Cameras for the Detection of Critical Events in Sensors-Based Emergency Alerting Systems

Daniel G. Costa ^{1,*}, Francisco Vasques ², Paulo Portugal ² and Ana Aguiar ³

¹ Department of Technology, State University of Feira de Santana, Feira de Santana 44036-900, Brazil

² INEGI/INESC-TEC—Faculty of Engineering, University of Porto, Porto 4200-465, Portugal; vasques@fe.up.pt (F.V.); pportugal@fe.up.pt (P.P.)

³ Instituto de Telecomunicações (IT), University of Porto, Porto 4200-465, Portugal; ana.aguiar@fe.up.pt

* Correspondence: danielgcosta@uefs.br

Received: 23 August 2020; Accepted: 5 October 2020; Published: 10 October 2020

Abstract: The adoption of emergency alerting systems can bring countless benefits when managing urban areas, industrial plants, farms, roads and virtually any area that is subject to the occurrence of critical events, supporting in rescue operations and reducing their negative impacts. For such systems, a promising approach is to exploit scalar sensors to detect events of interest, allowing for the distributed monitoring of different variables. However, the use of cameras as visual sensors can enhance the detection of critical events, which can be employed along with scalar sensors for a more comprehensive perception of the environment. Although the particularities of visual sensing may be challenging in some scenarios, the combination of scalar and visual sensors for the early detection of emergency situations can be valuable for many scenarios, such as smart cities and industry 4.0, bringing promising results. Therefore, in this article, we extend a sensors-based emergency detection and alerting system to also exploit visual monitoring when identifying critical events. Implementation and experimental details are provided to reinforce the use of cameras as a relevant sensor unit, bringing promising results for emergencies management.

Keywords: emergency alerting; visual sensing; Internet of Things; sensor networks; smart cities

1. Introduction

As urban populations grow and cities become highly populated, some challenging issues have been expected to play a central role in their dynamics. Actually, while some of such challenges are highly anticipated, such as air pollution, sustainability and mobility [1,2], other critical issues have been gaining attention in recent years, fostering new developments. Among them, emergencies management is expected to be a crucial element of the evolving smart cities landscape, directly impacting the perceived quality of life [3]. In fact, the particularities of smart cities scenarios, with many people in danger during emergencies, make them highly susceptible for the adoption of emergency alerting systems. This complex and emergency-prone scenario is assumed as the reference in this article.

Emergencies may happen anywhere and in an unpredictable way, with different reach, durations and potentials to cause damage in urban areas [4]. In this context, many systems have been developed to detect, to alert and to mitigate an emergency situation, covering one or more aspects of emergencies management. With the maturation of the Internet of Things (IoT) paradigm, some promising solutions to handle emergencies in smart cities have exploited interconnected sensors to detect critical events, allowing the fast and distributed management of potential emergencies [5]. Such sensors-based solutions should be an important trend when concerning how smart cities will evolve in this century [6].

In ref. [7], we proposed a comprehensive sensors-based emergency alerting system that is able to detect any number of critical events when issuing emergency alarms. The defined multi-tier

architecture in ref. [7] allows for the logical segmentation of “emergency detection” and “emergency alerting”, providing enough metadata for any emergency mitigation service that exploits the proposed system. In fact, the definition of multiple logical tiers leaves the actual detection of critical events to scattered sensors-based units named as Events Detection Units (EDUs). The EDUs are composed of any number of scalar sensors, one for each desired event to be detected.

A scalar sensor is able to retrieve numerical data within a defined scale, which is a function of the hardware characteristics of each sensor device [8]. In this sense, a temperature sensor will indicate the current measured temperature, while a humidity sensor will provide the measured relative humidity. Actually, for any type of scalar sensor, we can associate a sensor device to one or more types of events, with each event being assumed as “detected” when the measured numerical value is higher (or lower) than a defined threshold. As an example, a critical event of temperature may be assumed as “detected” if the measured temperature is higher than 80°, while a relative humidity lower than 10% may indicate that a critical humidity event is happening. Since an EDU may be composed of many sensor devices of different types, different critical events may be detected and alerted in the payload of emergency alarm messages.

The use of scalar sensors to detect events allows for the identification of many concurrent and independent critical situations, which are promptly reported to any requesting application in the form of emergency alarms. Although this model for emergency detection and alerting is promising when considering the big scope of smart cities (or any other emergency-prone scenario), the way critical events are detected can be improved when new sensing technologies are employed. Among the available sensing technologies for events detection, cameras can provide visual data about the monitored environment [9], which can potentially enrich the perception of the EDUs concerning critical situations. Hence, we consider the use of cameras as a new type of sensor device for the detection of critical events, enlarging the sensing capabilities of the EDUs in ref. [7]. Nevertheless, since visual sensors have different challenges and particularities when compared to scalar sensors [10,11], we analyze and discuss how cameras can contribute to emergency alerting in smart cities.

This article proposes the use of visual sensors to detect critical events in an emergency alerting system, complementing traditional scalar sensors and allowing for the identification of two different groups of events: *instance events* and *complex events*. The instance events are those that can be directly represented by a single type of numerical value, such as temperature, humidity, rain precipitation, luminosity, pressure, air pollution, UV radiation, and any other variable that can be assessed by scalar sensors. For that group of events, which are usually detected by scalar sensors, the use of cameras is valuable as an additional sensing device, for example when infrared/thermal images are used to indicate a temperature measure. On the other hand, cameras can also detect complex events, which can not be directly perceived by a single scalar sensor. Among complex events, Fire, Car Accident and Robbery are some common examples of events that can be perceived processing visual data, but raw scalar data alone are not sufficient to detect them with 100% confidence. Figure 1 depicts this principle when detecting these types of events.

Therefore, using cameras along with scalar sensors, the potential for the efficient detection of critical events by the EDUs is enhanced, which can be valuable for emergency alerting. Besides the identification of new types of events, cameras will typically sense “farther” than most scalar sensors, expanding the sensing reach of the EDU. Actually, since the idea is to define a highly flexible multi-sensory system that can be easily adjusted to detect any type of critical event, the proposed adoption of cameras as an “enhanced” sensing unit is intended to achieve a more efficient and generic solution, in a different way of event-triggered image sensors dedicated to a specific type of event [12]. In this context, this article proposes the use of visual sensors as a complement for our approach described in ref. [7], extending that model. Moreover, some algorithms are implemented for the practical exploitation of this idea, allowing for the assembling of camera-enabled sensors employing affordable off-the-shelf electronics components. Finally, experiments are conducted to validate the

detection of critical events when using both cameras and scalar sensors for emergency alerting. All these contributions are highlighted as follows:

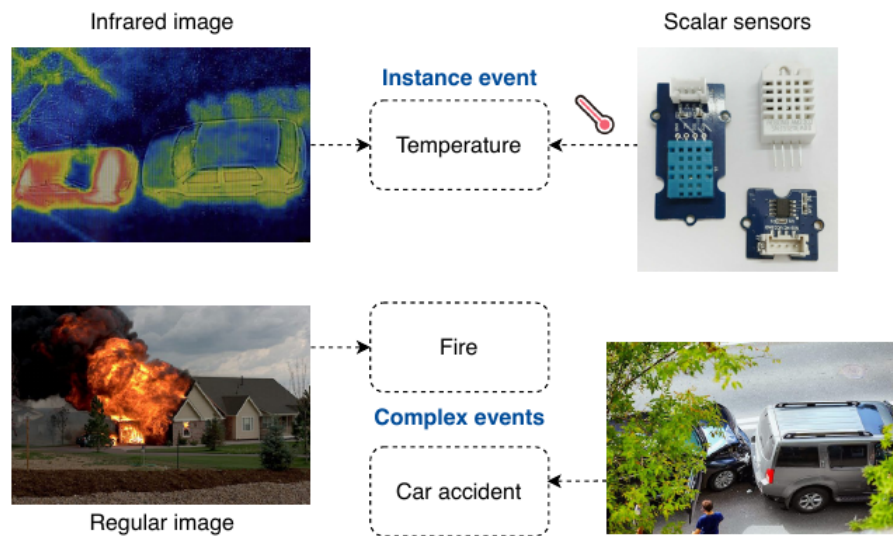


Figure 1. The use of visual data when detecting different types of critical events.

- Exploitation of scalar and visual sensors to detect emergencies in a combined way, employing for that a unified processing, storage and communication structure;
- Definition of two different types of events, instance and complex, which have different impacts when computing the severity level of an emergency alarm;
- A reference implementation for combined sensing using both scalar and visual sensors, which is ready to be used and available at a public repository (<https://github.com/lablara/cityalarmcamera.git>);
- A comprehensive discussion about failure conditions and practical implementation issues when detecting emergencies in smart cities.

The remainder of this article is organized as follows. Section 2 discusses related works that influenced this article. The fundamentals of critical events and emergency alarms are presented in Section 3. Section 4 presents the proposed camera-based emergency alerting approach. Implementation and evaluation results are described in Section 5. Section 6 discusses some practical issues when employing cameras for detection of critical events. Finally, conclusions and references are presented.

2. Related Works

Sensors-based events detection is not a novelty, with many works proposing some kind of sensing to detect an abnormal behavior that may indicate an emergency situation. In this context, some approaches have emerged to detect a particular emergency—for example, employing gas sensor [13], temperature sensor [14], sound sensor [15] or any sensor device to allow for a better perception of the environment. Although such specialized systems have contributed to the evolution of this research area, the unification of different detection systems in an integrated approach can be valuable for smart cities. In this context, we are particularly interested in how cameras can be used to detect events, allowing for their integration to scalar sensors monitoring.

The use of cameras as sensor devices has evolved in the last decades, with significant improvements in the way visual sensors are constructed [9], deployed [16] and interconnected [17] for different types of monitoring tasks. In this dynamic scenario, different issues have been investigated concerning sensing, coding and transmission issues, resulting in the maturation of visual sensing as an effective service for wireless sensor networks and more recently to the Internet of Things landscape.

As a result, many applications have relied on the processing of visual data retrieved from distributed interconnected cameras, influencing this article in different ways.

With the development of new sensing and communication technologies, many works have considered the use of cameras to detect different types of critical events, either processing images or using visual data as complements to emergency management approaches. Actually, for a group of them, the use of cameras can be worthwhile when providing additional information about an emergency, which could be valuable for many monitoring scenarios. In ref. [18], fire is detected considering the presence of smoke in the air, according to a defined threshold. Then, a camera is used to retrieve visual data from the monitored environment, which can be exploited to avoid false alarms when alerting about that emergency: a picture of the scene can be used by a human to confirm that a fire is happening. The work in ref. [19] also employs cameras after an event is detected through scalar sensors, providing images that can be used to corroborate the detection of events and avoid false alarms. Actually, we are particularly interested in works that process thermal or traditional (optical) visual data to identify critical situations, since they can more directly contribute to this article. In recent years, there have been interesting works in this sense. The work in ref. [20] employs a rotatable thermal camera for the detection of fire. Images are processed for the detection of fire flames, according to defined thresholds. Similarly, the work in ref. [21] employs a thermal camera to detect fire, potentially allowing a robot to mitigate to within it. In ref. [22], the authors employ cameras to detect when elderly people fall, which is also a critical situation that requires proper attention. Although such detection does not affect an area in a sense that requires urban emergency management, the detection of elderly falls is a good example of how complex events can be detected (wearable accelerometers could be used to indirectly detect a fall, but they require proper processing [23]). Such processing is similar to the use of cameras for home automation [24], which might be integrated to other systems when managing emergencies in cities.

When considering a larger scope of emergencies, in an urban scale, cameras can also be used for the detection of critical events. In ref. [25], visual data are processed in order to identify abnormal behaviors, exploiting data from more than one camera in a collaborative way. In ref. [26], critical events are detected by scalar sensors, that trigger a "critical situation" associated to a computed severity level within a numerical scale. Then, all cameras in the event area are reconfigured, being assigned a numerical priority for sensing, coding and transmission that is compatible with the computed severity level of the event.

Besides the definition of computational systems to detect emergencies using cameras, some recent works have also employed open-source hardware platforms to construct affordable and highly programmable sensing devices and processing units [27]. Actually, the use of platforms such as Raspberry Pi and Arduino can bring significant results for the evolution of smart cities, reducing costs while allowing for efficient distributed processing. The works in ref. [18,21,28,29] are some examples of the practical exploitation of open-source hardware platforms that have influenced this article.

Finally, the literature has shown different approaches concerning the use of cameras as part of some events detection solution, but many challenges still remain. Actually, when concerning the bigger scope of smart cities and the management of any type of emergency, previous works have failed to provide a comprehensive and effective solution, to the best of our knowledge. Therefore, this article proposes a robust and highly configurable emergency alerting system for any emergency-prone environment, exploiting cameras as a relevant element for the enhanced detection of critical events.

3. Fundamentals of Emergency Alerting

As discussed before, emergency alerting systems can be developed in different ways and with different levels of automation, according to the desired services to be provided and the target monitoring scenario. Among the possibilities in this area, we chose as a reference the multi-tier system described in ref. [7] due to its high flexibility concerning the detection of events and the description of emergency situations in cities. That system defines a logical separation between events

of interest and emergencies alarms, which are processed by different units (logical layers) and that are composed of different types of metadata. Such a type of logical separation allows for the adoption of new sensing paradigms, as the detection of instance and complex events by visual sensing performed by cameras.

We defined that emergencies alerting is based on three basic elements, described as follows:

- **Events of Interest:** This is the fundamental concept when alerting about emergencies. An emergency happens when at least one event is detected, while an event is only detected when a measurable variable is above or below a given (configurable) threshold. In ref. [7], all events are detected through a scalar measure, but we propose to extend that behavior to include events that can be detected by cameras, defining new levels of events.
- **Events Reports:** An emergency may be associated with one or more events, and the number of detected events within a defined period of time is a relevant parameter when computing the potential of damage (severity level) of any emergency. When considering the use of cameras, the Events Reports, which are transmitted from the EDUs, may be adapted to accommodate this information.
- **Emergency Alarms:** The alarms are issued by an Emergencies Processor Unit (EPU) that computes the severity level of each alarm based on temporal and spatial information. When using cameras in the EDUs, the EPUs may employ a different algorithm when computing the severity of the alarms, in accordance with the proposals in this article.

The three elements described in ref. [7] are sufficient to issue alarms generated by events detected using scalar sensors, but they are inefficient when cameras are used to detect critical situations. Therefore, a new extension had to be proposed, as described in the next section.

Generally speaking, the use of sensors can resolve the uncertainty about whether some critical event exists or not, but the use of cameras and visual data as an important element of events detection can enhance how the overall process of emergency management is expected to be performed in smart cities. Figure 2 presents a general schema of this scenario, with events being detected and reported, and alarms being generated and transmitted to any requesting application (for warning or mitigation purposes).

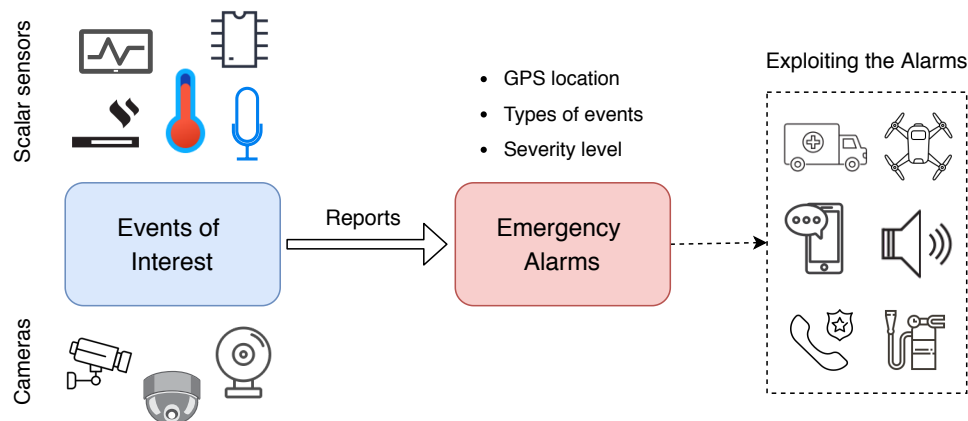


Figure 2. Emergency alerting based on the detection of events of interest by both scalar and camera sensors.

4. Proposed Approach

This article proposes the use of cameras as an effective type of sensor when detecting events, complementing the use of scalar sensors. For that, our work in ref. [7] was extended to incorporate this new idea, resulting in new specifications, implementations, experiments and discussions. We believe

that this extension can potentially achieve better results when detecting and alerting about emergencies, contributing to the development of this research area.

The main concept proposed in this article is the definition of two different types of events: instance events and complex events. Actually, both events are defined as logical parameters that may be active (TRUE/ON) or not (FALSE/OFF), depending on defined thresholds or baselines. This same idea is considered for camera-based monitoring, but while scalar sensors can accurately rely on numerical thresholds, events detection based on visual data require computer vision algorithms that may have different levels of accuracy, processing time and computational costs. Nevertheless, both types of events will follow the basic principle that they may be active or not, and any emergency alarm will comprise one or more (instance or complex) events.

4.1. Visual Data Processing

Images and video streams received by cameras will be constantly processed according to how the EDUs are configured. Actually, the expected operation of the EDUs when only scalar sensors are employed is to check all attached sensors, retrieving the measured value at a previously defined frequency of $f_{s(u)}$, for any EDU u . The retrieved data are then compared to all defined thresholds.

When exploiting cameras as sensors for events detection, the same frequency $f_{s(u)}$ can be used. In this case, when images are processed, a snapshot can be taken every $f_{s(u)}$. This is, in fact, a more feasible configuration for regular EDUs, requiring the processing of single snapshots at a known frequency. However, cameras may also capture video streams, continuously, which should be processed in real time for the identification of events of interest. Although this last scenario can be more effective, real time video processing may be very costly [30,31], being prohibitive when many EDUs have to be deployed over an area under some budget constraints. Actually, we believe that the processing of image snapshots and scalar sensed data under the same frequency is the best approach for practical events detection in smart cities, especially for low values of $f_{s(u)}$.

The use of visual sensors for events detection will deeply depend on the hardware characteristics of the cameras. There are different types of cameras and processing platforms when considering visual sensing [9]. Sometimes, thermal cameras will be employed—for example, allowing for the detection of the instance event "Heating". The thermal cameras use radiation from the infrared region of the spectrum, allowing for the detection of heat signatures. From a different perspective, optical cameras, which retrieve images in the visible human spectrum, may have a limited Field of View (FoV) or they may view 360° from its position. Moreover, different configurations concerning image resolution, Depth of Field (DoF) and zooming capabilities may also impact how the environment will be perceived when detecting events. Finally, some cameras may be rotatable, allowing for the repositioning of the cameras in response to some event or to avoid coverage failures.

Therefore, it is expected that visual data will be processed to the identification of instance or complex events, employing any type of computer vision algorithms [32,33]. Actually, the way visual data will be processed is out of the scope of this article. However, as patterns detection exploiting visual data has significantly evolved in last years, with some efficient and open algorithms being freely available, the immediate adoption of cameras for events detection is feasible for many scenarios.

4.2. Detecting Events and Sending Reports

This article proposes new approaches for the detection of critical events by sensor units, considering the use of scalar and visual sensors in an emergency alerting system. For that, some procedures in ref. [7] were extended, and new methodologies were proposed, potentially achieving a more robust and comprehensive solution for emergency management in smart cities.

The crucial element to be considered when managing emergencies is the "event of interest". Since the generation of emergency alarms results from the detection of at least one event, how events will be detected by cameras is of paramount importance and there are some important issues to be considered in this matter. First of all, the detection of any event is performed considering the existence

of two different types of events: instance events and complex events. Their processing by the system is defined as follows:

- **Instance Events:** They can be detected by both scalar and visual sensors. The defining characteristic of an instance event is that it is directly detected only employing a single numerical threshold. Each scalar sensor can detect one or more instance events and the same is true for visual sensors. Infrared/thermal images are examples of visual data that can be processed to identify instance events;
- **Complex Events:** They are supposed to be detected only by visual sensors since they are not associated to numerical thresholds. The detection of complex events is performed by the processing of visual data through computer vision algorithms, which must to be configured to detect critical situations, such as Fire, Smoking, Flooding, Accident, among other events-related emergencies.

As a second remark, each sensor unit may detect different types of events, but only one at a time. For example, at a given moment, a temperature sensor may detect an (instance) event of "Heating" (e.g., when the current temperature is higher than 100°) or an (instance) event of "Freezing" (e.g., for temperatures lower than -20°), but not both. This principle is also employed for visual sensors when detecting complex events, assuring an effective but still flexible perception of them. For example, a visual sensor may detect an (complex) event of "Fire" or an event of "Smoking", but if visual data are processed and fire and smoke are detected at the same image, the event "Fire with Smoke" is assumed as detected (and not two different events).

The third remark is about the impact of each detected event. Roughly speaking, instance events are easier detect and their impact on emergencies depends on the number of detected instance events. For example, an event of "Heating" may or may not indicate fire (a well-known emergency). In this case, the detection of both instance events of "Heating" and "Smoking" would give a stronger indication that something is on fire. However, a complex event of "Fire" is meaningful when indicating that some critical emergency situation (related to fire) is happening. Therefore, in order to leverage the use of cameras for the better perception of the environment, we define that a complex event will be twice as relevant as an instance event.

In this sense, we define that any instance event, ei , will be associated to a scalar data type y , resulting in the function $fy(ei) = y$, for $ei > 0$ and $fy(ei) > 0$. Additionally, any complex event, ec , will be associated with a visual data pattern w , $fw(ec) = w$, for $ec > 0$ and $fw(ec) > 0$. Of course, w can only be detected by visual sensors, while y can be detected by both scalar and visual sensors. Moreover, two events $ei1$ and $ei2$, for $ei1 \neq ei2$, may have the same type, $fy(ei1) = fy(ei2)$, and the same is true for complex events. Then, for instance and complex events, we define the detection functions $dy(ei) = 1$ and $dw(ec) = 1$ when the corresponding event is detected, respectively, giving 0 otherwise. Actually, these formulations indicate that two different events are processed in the same way if they are both detected and have the same type (y or w), even if they resulted from different sensor measures.

The definition of two different types of events requires the use of different decision parameters when processing the $dy(ei)$ and $dw(ec)$ functions. For instance events, a numerical threshold is defined, referred as $th_{(y)}$, which is configurable for each type of event y . However, for complex events, since visual data processing may be subjective, the decision parameter may be any type of pattern or configuration, resulting in the configurable parameter $pt_{(w)}$. Therefore, the detection of any given instance event will exploit the initial definitions in ref. [7], taking the current measured data in time t , $m(y, t)$, and comparing this value to the value of $th_{(y)}$. On the other hand, the detection of complex events occurs in snapshots defined as $v(w, t)$, and the pair $\{v(w, t), pt_{(w)}\}$ is processed when computing the value of the $dw(ec)$ function for any ec .

Actually, the definition of the instance and complex events that are detected by a particular emergency alerting system depends on the characteristics of the considered urban scenario and the

intended monitoring services to be provided in terms of emergencies detection. Moreover, the available sensor units in each EDU play an important role in such decisions. Therefore, with the proposed exploitation of visual sensors, the definitions of the types of expected events becomes even more relevant, directly impacting on the performance of the implemented system.

Table 1 presents some examples for both instance and complex events, which may be used as a references for any implementation, according to the particularities of the target urban scenario.

Table 1. Examples of instance and complex events when managing emergencies in urban areas.

Instance Event	Scalar data	Type (y)	Threshold ($th_{(y)}$)
Heating	Temperature (thermometer)	1	$m(1, t) \geq 60\text{ }^\circ\text{C}$
Heating	Temperature (infrared image)	1	$m(1, t) \geq 60\text{ }^\circ\text{C}$
Freezing	Temperature (thermometer)	2	$m(2, t) \leq -20\text{ }^\circ\text{C}$
Freezing	Temperature (infrared image)	2	$m(2, t) \leq -20\text{ }^\circ\text{C}$
Low Humidity	Relative humidity	3	$m(3, t) \leq 10\%$
Smoke	CO ₂ concentration	4	$m(4, t) \geq 5000\text{ ppm}$
Toxic Gases (leakage)	Ammonia (NH ₃)	5	$m(5, t) \geq 35\text{ ppm}$
Heavy Rain	Rain precipitation	6	$m(6, t) \geq 10\text{ mm/h}$
Earthquake	Seismic magnitude	7	$m(7, t) \geq 6.5M_L$
Noise Pollution	Audio signal strength	8	$m(8, t) \geq 100\text{ dB}$
Radiation	Ionizing radiation (Geiger sensor)	9	$m(9, t) \geq 1000\mu\text{Sv/hr}$
Radiation	Ionizing radiation (gamma image)	9	$m(9, t) \geq 1000\mu\text{Sv/hr}$
Blast Wave	Air pressure (or wind speed)	10	$m(10, t) \geq 5\text{ psi}$
Complex Event	Dangerous pattern	Type (w)	Function $\{v(w, t), pt_{(w)}\}$
Fire	Active flames (any colors)	1	Detected flames
Smoke	Any type of smoke	2	Detected smoke (fire or chemical)
Fire and Smoke	Flames with smoke	3	Detected flames and smoke
Car Accident	Two or more cars abruptly stopped	4	Crash patterns (broken elements)
Injured people	People needing assistance	5	People laying on the floor
Explosion	Destruction patterns	6	Detected wreckage
Panic	People randomly running	7	Running but no historical behavior

The definition of the events of interest that will be handled by a system should not be static, allowing easy modifications. First of all, new hardware components may be created along the time and become affordable for large-scale deployment, for example as it is happening with gamma cameras [34,35]. Second, new monitoring requirements may arise, depending on the occurrence of critical events. Whatever the case, the proposed model allows easy and straightforward adaptations for the inclusion of new types of events of interest.

This adaptability principle may still present other relevant issues. In fact, new perceptions and algorithms may continuously evolve, changing the definitions of the events. For example, the detection of people falling can be performed using accelerometers (scalar sensor) and recent works have increased the efficiency of such detection [23,36]. However, cameras may also detect falls when processing visual data. Nevertheless, in order to assure the consistence of the solution, all events detected by scalar sensors should be processed as instance events, better supporting the interactions of the multiple tiers of the proposed emergency alerting approach.

After detecting events every $fs_{(u)}$ seconds, an Event Report (ER) has to be transmitted from the corresponding EDU to a previously configured EPU. As defined in ref. [7], an event detection unit u will issue an ER composed of the ID (y) of all detected events. Moreover, that ER will also comprise a timestamp ts (the moment the report was created) and the GPS coordinates of the EDU (la for latitude and lo for longitude). This complementary information is required to keep track of the reports and also when generating the alarms, which are the outcomes of the system. Concerning the proposed use of cameras, the timestamp and GPS information will not be altered, but the perception of the detected events must be extended to incorporate the proposed types of critical events. In this case, an event report i will be now represented by the tuple $(u, i, ts, la, lo, Ei, Ec)$, for $i > 0$. The Ei is a set containing the

identification type of all detected instance events in time ts by the EDU u , while Ec is the set with the types of all detected complex events. Doing so, $|Ei| \geq 0$ and $|Ec| \geq 0$, but they can not be both empty sets at the same time, since an event report must comprise at least one detected event.

The Events Reports are expressed in the JSON (JavaScript Object Notation) format, describing the (u,i,ts,la,lo,Ei,Ec) information. An example of an Event Report for two instance events ($y1$ and $y2$) and one complex event ($w1$) is described as follows:

```

{
  "edu": "u" ,
  "id": "i" ,
  "timestamp": "ts" ,
  "gps": {
    "latitude": "la" ,
    "longitude": "lo"
  } ,
  "ei": [
    y1 ,
    y2
  ] ,
  "ec": [
    w1
  ]
}

```

The transmission of the Events Reports remains unaltered: they are transmitted both asynchronously (every time some new event is detected) and synchronously (according to the transmission frequency $f_{x(u)}$). This hybrid transmission pattern assures high efficiency and flexibility, with direct impact on the time the events are detected and they remain assumed as being detected.

4.3. Computing the Severity Level When Employing Cameras

When an EPU receives an Events Report, an Emergency Alarm (EA) will be generated. Actually, the general procedure defined in ref. [7] is leveraged when generating alarms, which is divided in three basic definitions: (1) every event report results in the generation of a single emergency alarm; (2) every emergency alarm is associated with a unique event report; (3) all emergency alarms are independent. The EPU will then generate alarms and deliver them to any requesting applications.

An alarm a comprises temporal information (provided by the timestamp in the ER), spatial information (provided by the GPS position) and a severity level ($sl_{(a)}$). For this last information, three elements are combined to achieve a numeric index between 0 and r_{max} (usually 100). The computation of $sl_{(a)}$ is highly comprehensive, considering when the report was issued (the hour and day of the week), where it comes from (is it a risky zone?) and how many events are being reported. In fact, when using cameras and defining two different types of events, the computation of $sl_{(a)}$ has to be altered—although all events are equal when alerting people about a critical situation, instance and complex events will be differently processed.

For the computation of $sl_{(a)}$, the number (and relevance) of events to be considered will be limited to five, even though more than five events may be reported in an ER. In fact, an emergency situation with five concurrent events being detected is already extremely severe and such a high level of severity would be almost the same if more events were detected and reported. However, we propose that a complex event will be twice as relevant as an instance event when computing the numeric severity index. For that, there will be $n_{(i)}$ instance events in report i and $m_{(i)}$ complex events, for $0 < (n_{(i)} + (m_{(i)} \times 2)) \leq 5$. The value of $sl_{(a)}$ is computed as expressed in Equation (1).

$$sl_{(a)} = ((n_{(i)} + (m_{(i)} \times 2)) \times 20 \times f_e) + \left(\frac{r_{(z)} \times 100}{rmax} \times f_r \right) + \left(\frac{t_{(a)} \times 100}{tmax} \times c_{(t)} \times f_t \right) \quad (1)$$

The formulation presented in Equation (1) is normalized, resulting in a value for the severity level in the range $0 \leq sl_{(a)} \leq (rmax)$. In order to achieve flexibility, the constants f_e , f_r and f_t are used to tune the variables employed when computing $sl_{(a)}$, since $(f_e + f_r + f_t) = 1.0$. Hence, a “regular” implementation would define $f_e = f_r = f_t = 1.0/3$. Moreover, the values of $r_{(z)}$ and $c_{(t)}$ stand for the computed risk zone and temporal significance indexes, respectively, which will be computed using the same procedures of ref. [7]. In other words, the impact of spatial and temporal information is not altered, but only the interpretation of the detected events.

An emergency alarm a is also represented in the JSON format. An alarm will replicate most of the information that is already in the corresponding ER (including the types of the detected instance and complex events), plus two additional pieces of information: the ID of the alarm, a , and the computed severity level. With this information, any requesting application can exploit the alarms to perform any warning or countermeasure service.

Finally, any requesting application can receive alarms as soon as they are generated. For that, the MQTT (Message Queuing Telemetry Transport) protocol can be exploited (or any publish-and-subscribe protocol). The MQTT will be used as the basis for the experiments in this article.

5. Experiments and Results

The proposed approach defines a complete system to detect both instance and complex events when performing emergency alerting in scenarios, such as smart cities and industry 4.0, defining a comprehensive mechanism to distributively detect emergencies and to notify any requesting emergency-based application. Although this innovative approach was carefully described, it has to be validated to support its practical exploitation in real scenarios. However, since it may be worthless to compare the proposed approach with existing solutions, mostly due to the fact that the proposed approach addresses some problems that were not considered before, the validation was centered on the implementation of the modules of the proposed system, supporting its practical implementation. Actually, besides providing a consistent proof-of-concept, the developed experiments are an important reference when employing the proposed system in practical applications: all the codes can be easily modified to attend the characteristics of any target scenario.

5.1. Implementation Details

The implementation of the proposed approach is intended to be a proof-of-concept, supporting not only its validation but also practical usage in real cities. For that, all defined elements were implemented, being freely and openly available in a public repository at <https://github.com/lablara/cityalarmcamera.git>. All provided codes, which were implemented in the Python 3 programming language, are final and ready to be used, being a valuable resource when attesting the effectiveness of the proposed approach to provide the expected services.

The implementation codes of the Events Detection Unit designed to detect both instance and complex events were stored in the directory “EDU” of the project repository (*eduCamera.py*). All EDUs were implemented using the Raspberry Pi platform, model 3B+, which is a robust, energy-efficient and low-cost hardware platform to develop electronic devices for countless functions [9]. Additionally, the Raspberry Pi efficiently supports the use of cameras to retrieve visual information, providing a specific hardware interface for the official Raspberry camera. In fact, although an EDU may be implemented using any available hardware platform, the adoption of Raspberry Pi boards to construct them is reasonable for many scenarios and thus it is the base for this proof-of-concept.

The EDUs were implemented to sense both instance and complex events. Therefore, an EDU needed to be able to sense scalar and visual data, which can be implemented attaching different sensor

units to the Raspberry board. For that, sensor units may be attached to a Raspberry Pi board following different standards, but we decided to exploit the GrovePi+ hardware framework to support the easy attachment of sensors to the board. In fact, multiple heterogeneous sensors may be attached to a single Raspberry Pi board using the GrovePi+ hardware framework, following a plug-and-play paradigm and enabling easy communication to both analog and digital sensor devices. Doing so, an EDU may be easily assembled, better supporting straightforward deployment exploiting this implementation reference.

Figure 3 presents the electronic components that were used to create the proof-of-concept EDU (wires are also required). In the figure, we can see a Raspberry Pi 3B+ board, a GrovePi+ HAT (Hardware Attached on Top), a Raspberry Camera v2, a GrovePi+ GPS module and four different GrovePi+ scalar sensors (audio, water, air pollution and humidity). Finally, we used a 16 × 2 LCD to easily present the number of currently detected events.

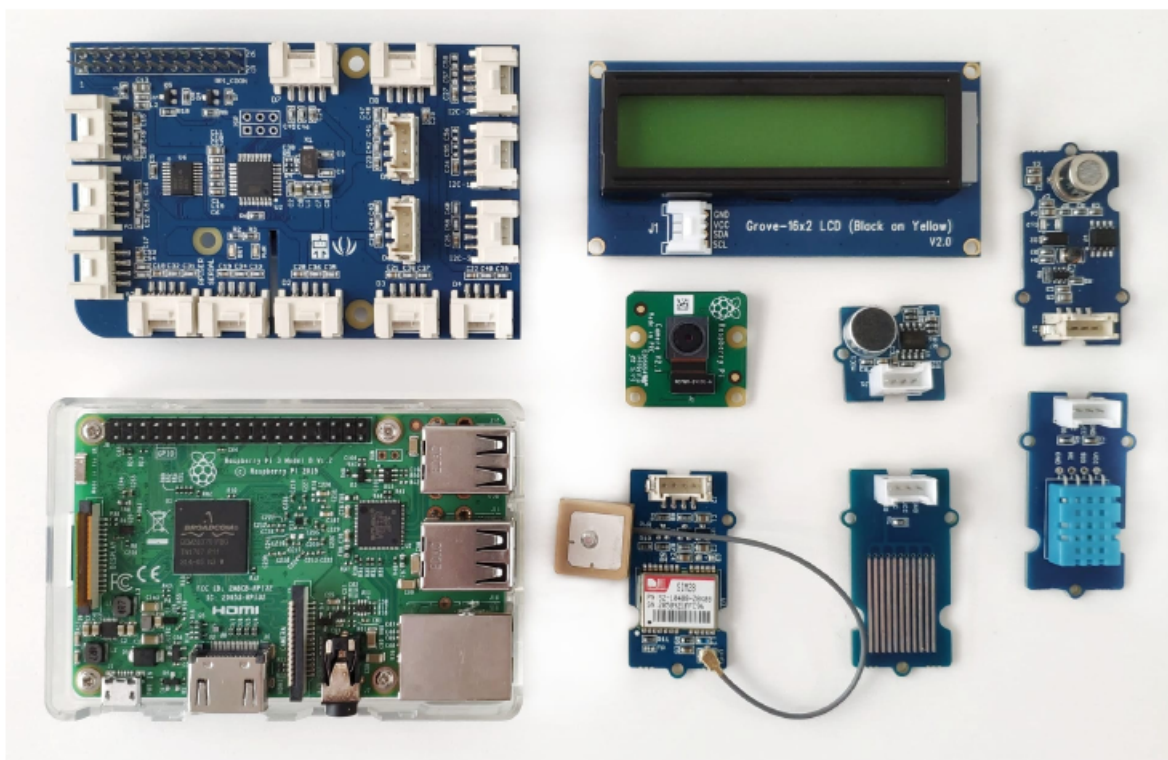


Figure 3. Electronic components to create an EDU.

The detection of events based on scalar sensors was performed considering the sensed data and the defined thresholds, as defined in the example in Table 1. However, the detection of complex events is not straightforward, requiring visual data processing. In fact, the last decades have seen a tremendous evolution of visual computing algorithms, allowing fast and more confident pattern detection through the processing of images. This trend has even benefited from the development of artificial intelligence algorithms based on the deep learning paradigm, opening new possibilities in this area [37,38]. Actually, when considering the proposed emergency alerting system, we are concerned with the structured detection of new events and the generation of emergency alarms, but how new events will be detected is out of the scope of the proposed systems, assuring high flexibility to it. In other words, any hardware components and any processing algorithms may be employed, since events can be detected (ON) or not (OFF).

For the developed proof-of-concept, an algorithm was created to detect the complex event of “Fire”, demonstrating how instance and complex events may be simultaneously detected by an EDU. In fact, there are plenty of solutions for such processing, which has evolved considerably [39,40].

For this particular implementation, we employed the OpenCV framework [41], which is an open-source multi-platform library that supports a great number of functions to process visual data. The OpenCV is used to detect a flame on pictures provided by the Raspberry Camera, performing for that some basic transformations on them—the provided code (*fireCamera.py*) can be easily adjusted or replaced to detect fire in different scenarios.

Figure 4 presents the experimental EDU created to perform initial tests to validate the practical operation of the proposed system. The components were arranged on a box just for experimental purposes, but any configuration and hardware arrangement is possible.

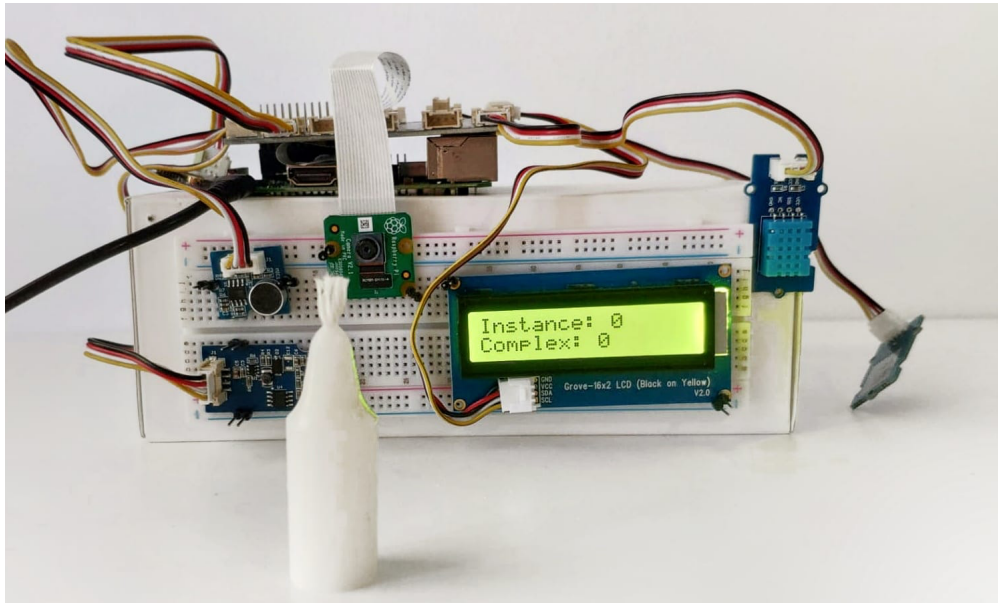


Figure 4. The experimental EDU when no event is being detected.

For the experimental EDU, we created a continuous loud noise, raising the detection of 1 instance event every 5 s ($f_{s(1)} = 5$). The detection of that event is shown on the LCD display and an Events Report (ER) is transmitted to the EPU—the EPU was executing on an Intel i5 8 GB RAM server. That scenario is demonstrated in Figure 5.

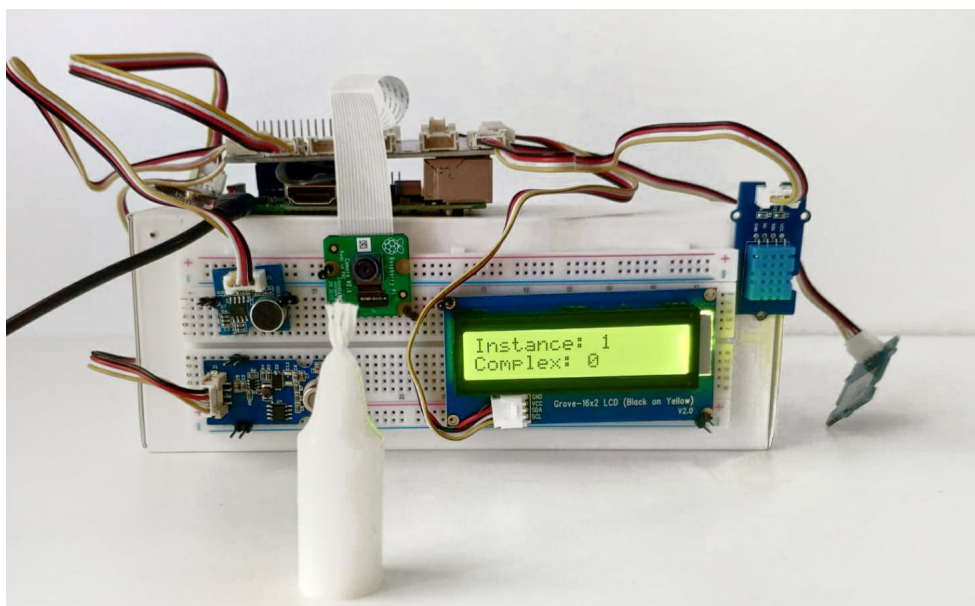


Figure 5. The experimental EDU when detecting 1 instance event (noise).

After that, we lit a candle to simulate a fire situation, as presented in Figure 6. The candle simulated the presence of fire as an emergency that must to be promptly alerted.

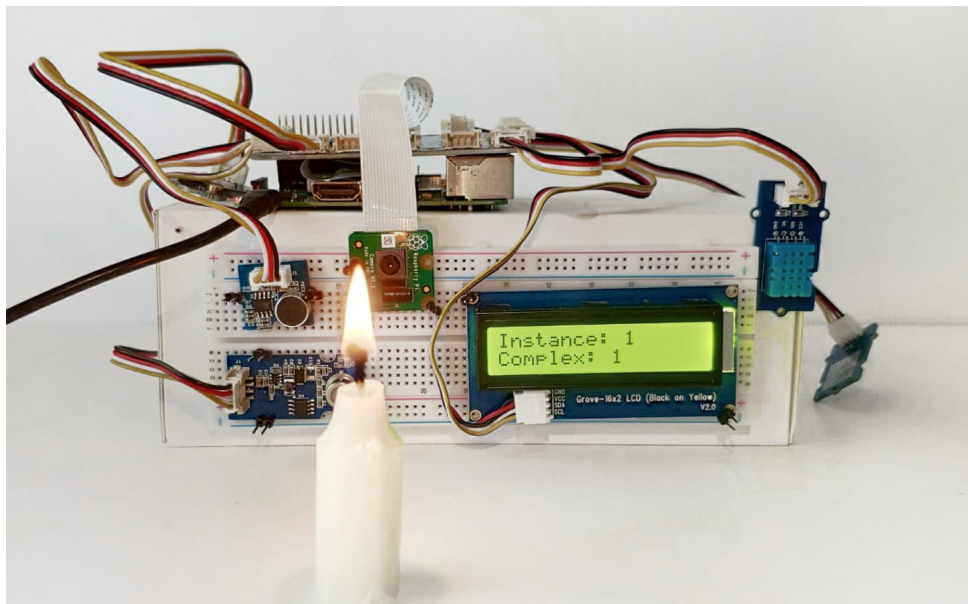


Figure 6. The experimental EDU when detecting 1 instance event (noise) and 1 complex event (fire).

Finally, we immersed the water sensor into a cup with water, generating a new instance event, as presented in Figure 7. Then, the numbers of the detected events are presented on the LCD display and a new ER is transmitted to the EPU.

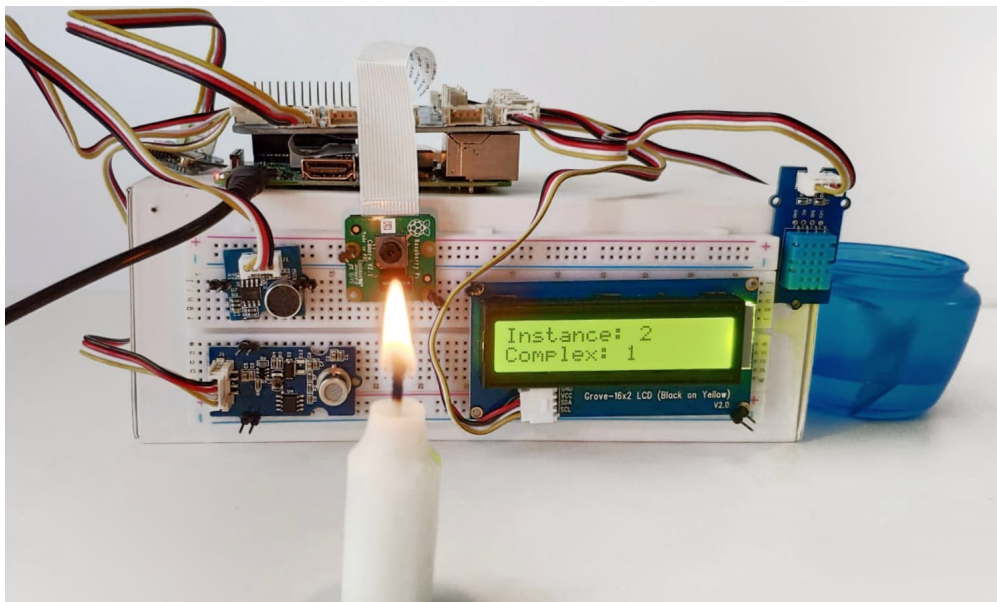


Figure 7. The experimental EDU when detecting 2 instance events (noise and water) and 1 complex event (fire).

The employed algorithm to detect fire extracts the flame pattern to try to identify if something is on fire. This is a “simple” but effective algorithm, but more efficient and robust approaches can be used depending on the available processing, memory and energy resources of the implemented EDU. For the instance events, they are detected in a simpler way, comparing the retrieved data with the defined thresholds (easily configurable in the codes).

The Emergencies Processing Unit receives the Events Reports transmitted by the EDU and generate emergency alarms, which are published in a MQTT broker. The alarms contain the computed severity level according to the received data from the EDU, the time relevance and the defined risk zones. In order to facilitate the interaction with the EPU, information is displayed to the user. Figure 8 presents one of the outputs of the EPU.

```

EPU — pi@raspberrypi: ~ — Python epu.py — 95x33
New EDU connected: 192.168.1.129 : 35814
Received ER from EDU n. 1
Computing the magnitude of the EA...
Sl of EA: 77
EA id: 8 , Timestamp: Fri May 22 18:40:26 2020 , Latitude: 41.176898 , Longitude: -8.585529 SL
: 77
This EA has the following EI types:
Type: 8
Type: 16
This EA has the following EC types:
Type: 1

Transmitting the Emergency Alarm 8
EA in the JSON format:
{
  "gps": {
    "la": 41.176898,
    "lo": -8.585529
  },
  "id": 8,
  "sl": 77,
  "timestamp": "Fri May 22 18:40:26 2020",
  "typesComplex": [
    1
  ],
  "typesInstance": [
    8,
    16
  ]
}
Publishing an Emergency Alarm (EA) to the MQTT Broker: 192.168.1.131
The Emergency Alarm was published!

```

Figure 8. An example of interaction with the EPU.

The provided software implementation of the EDU and EPU, as well as the hardware assembling of the EDU, worked properly and with no errors. The events were correctly detected and the Events Reports and Emergency Alarms were transmitted and received as expected. Besides initially demonstrating that the proposed approach works as defined, although additional tests may sustain more incisive conclusions, the provided implementation can support practical implementation in real cities, which just need to adapt the provided code to perform the desired functions.

5.2. Detecting Emergencies in A City

The developed EDU for initial experiments is an important reference for practical exploitation of the proposed emergency alerting system, supporting prompt deployment of EDU units in real cities. The EPU module is also operational and available for use. Besides those implementations, we also created a client to demonstrate how emergency alarms can be retrieved and processed.

The communication between the EPU and any emergency-oriented application is performed through the MQTT protocol. This approach allows for different clients receiving the same Emergency Alarms through a uniform publish-and-subscribe paradigm—the EPU publishes alarms to the MQTT Broker, which notifies all registered clients. The MQTT protocol is largely employed in Internet of Things projects and there are open source libraries to support easy communications through it [42].

Figure 9 depicts the topology of the performed experiments. Three EDUs are in operation, transmitting Emergency Reports to the same EPU. Then, the EPU communicates to the open-source Mosquitto MQTT Broker, which was deployed on a Raspberry Pi 3B+ board. Doing so, all Emergency Alarms created by the EPU are published to the MQTT Broker, which automatically transmits them to all registered clients.

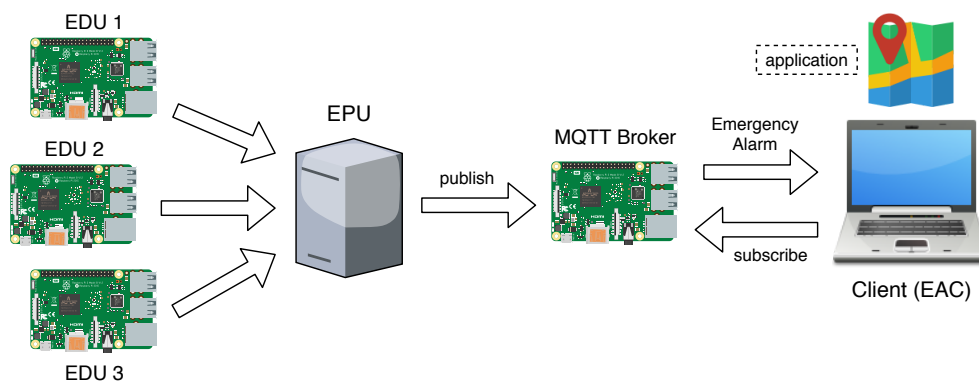


Figure 9. The defined experimental scenario.

Due to the COVID-19 pandemic, which imposed many restrictions for movement, including in research centers, we could not safely spread the EDUs over the target city (Porto, Portugal). Therefore, we executed all units and servers in the same room, only simulating the GPS coordinates. Nevertheless, since the units communicate using Internet connections, such physical configuration are not expected to compromise the achieved results.

The implemented client for this experiment (EAC—Emergency Alarm Client) retrieves Emergency Alarms from the MQTT Broker and plots the results on a map. For that, the popular open-source Foliom Python library was exploited to allow for the plotting of information over real maps, using for that GPS coordinates as a reference. Doing so, the developed client retrieves EA from the MQTT Broker (in the JSON format), processes them and plots alert pins over the map—each of those pins also indicates the computed severity level of the alarm and the associated detected (instance and complex) events. Therefore, the client program constantly generates a ".html" file containing the map with the pins, which can be directly visualized or easily embedded into any web-based applications.

Figure 10 presents the map when there is only one EDU detecting an event. In order to facilitate the easy perception of the way an alarm was generated, we defined the following classification for the pins on a map: "red" for alarms generated by both instance and complex events, "purple" for only complex events and "orange" for only instance events. Actually, although the type of the detected events only affects the computed severity level of the associated alarms ($sl_{(a)}$), the developed application exhibits the alarms considering the type of the detected events for visualization purposes. Nevertheless, any processing of the received alarms is possible, depending on the expected services of the final applications.

Figure 11 presents the map with two active alarms. An EDU in a different (simulated) position is reporting a new alarm comprising only instance events (orange).

Both alarms are being reported under the same Risk Zone and with the same temporal significance, and therefore the severity level of them is only differentiated by the number of detected events. In this case, they have the same severity since a detected complex event is twice as relevant as a single instance event. Figure 12 depicts this information, which is displayed when clicking on the plotted pin.

The third deployed EDU did not report any event during the experiment, generating no alarms, which is reasonable in real scenarios. For the other EDUs, we employed candles, matches and a noise generator (cellphone) to simulate critical events.

In order to assure robustness to the overall system, the Events Reports are constantly refreshed by the EDUs, resulting in a refreshing frequency for the alarms. This operation was proposed by us in ref. [7] and it consists in "positive feedback", informing only about detected events. Doing so, any event is indirectly assumed as undetected when it is not being reported, and the same is true for the alarms. For the developed application, which is also available in the open repository at <https://github.com/lablara/cityalarmcamera.git>, the pins are only presented when alarms are received, being removed if they are not being received after 60 s (it is a configuration parameter).

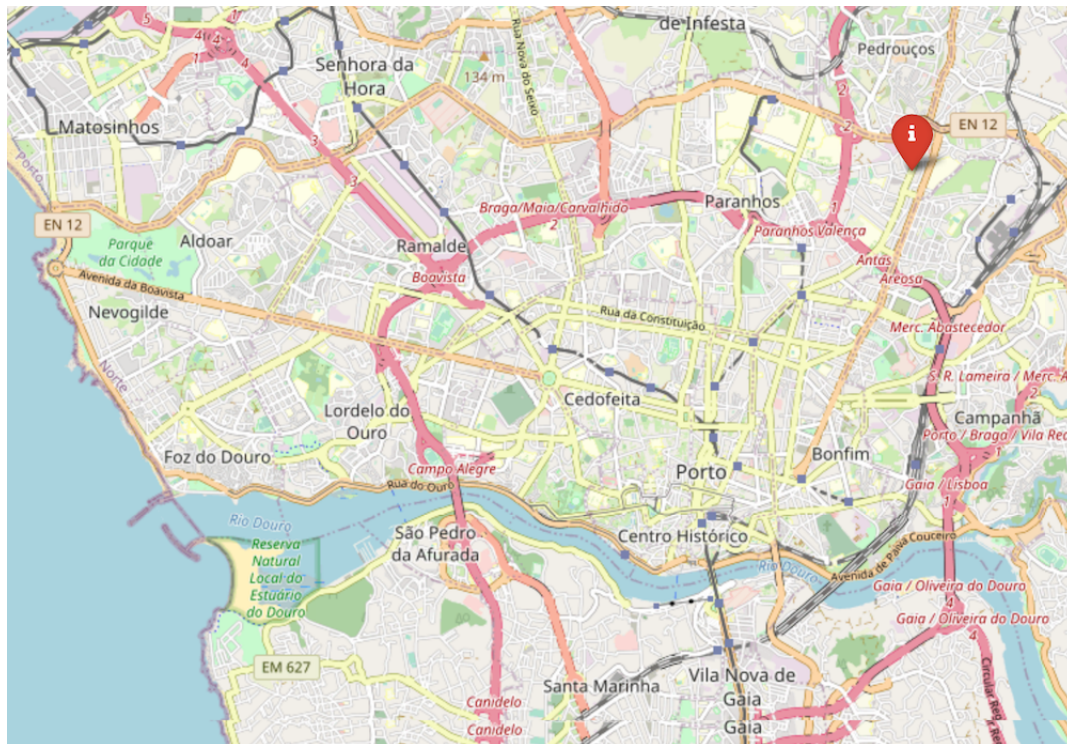


Figure 10. One alarm is being generated when fire and noise are being detected.

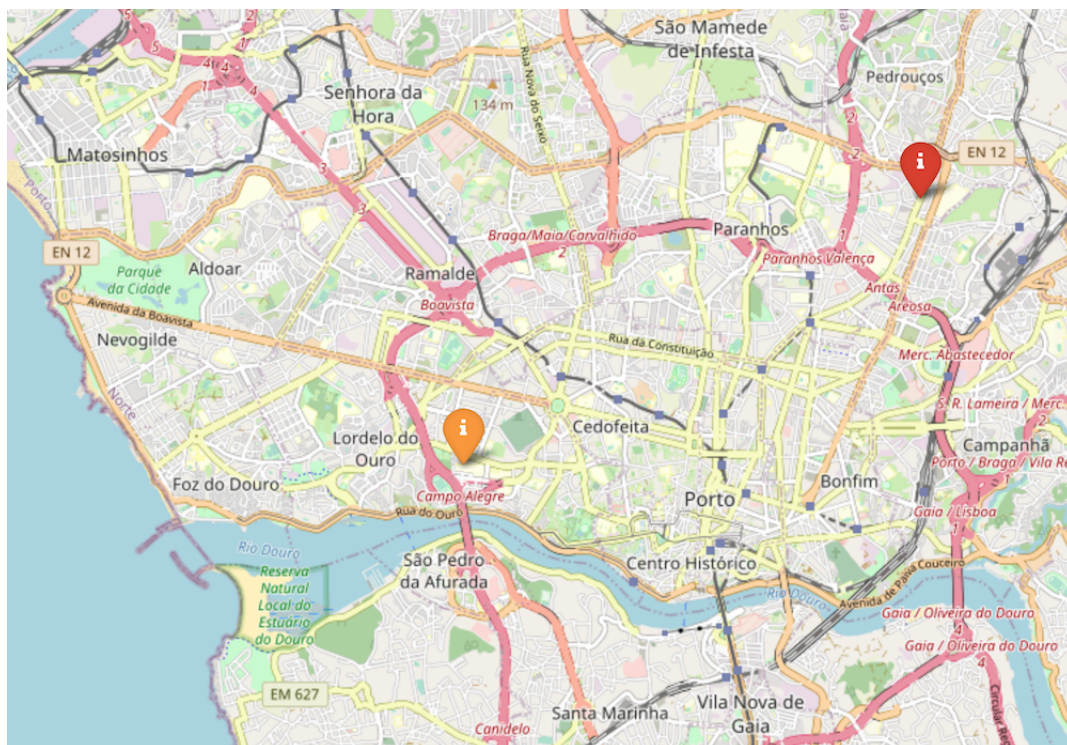


Figure 11. A new alarm is being reported in a different location, indicating noise, smoke and heating instance events.

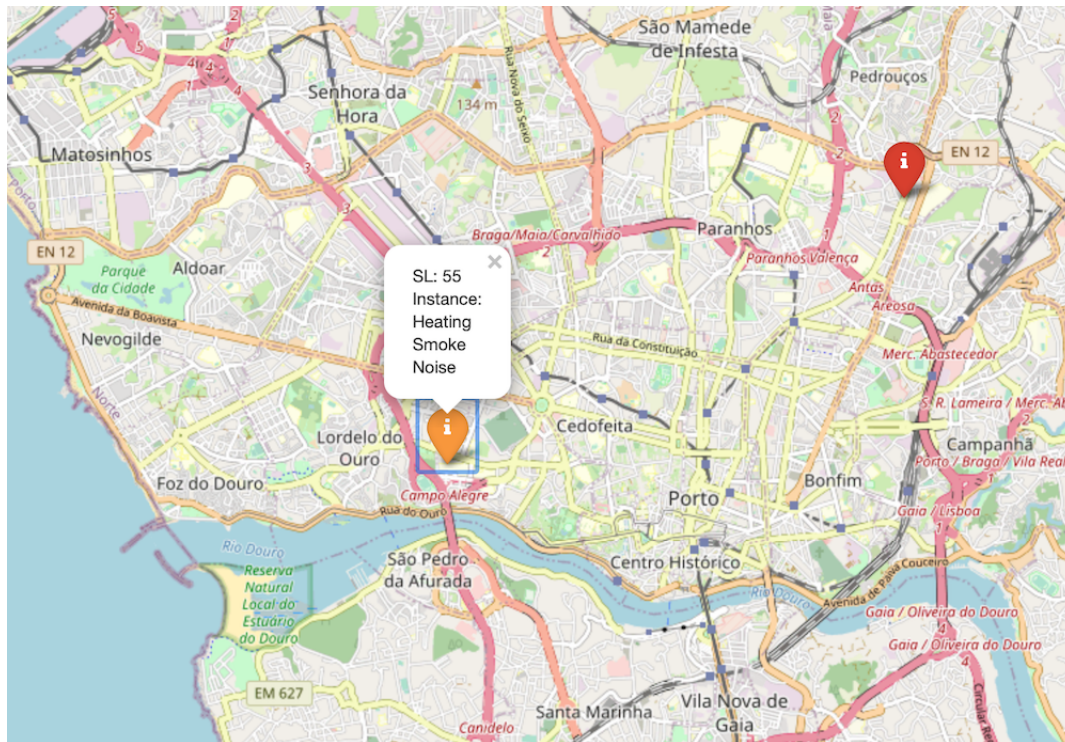


Figure 12. The events that comprise the alarm are displayed when we click on the pin.

5.3. Failures and Dependability of the System

The detection of a single emergency in a smart city is an inherent complex task that is related to different important issues. In first place, the emergency should be detected in a timely manner, providing relevant information for alerting and mitigation in order to reduce the probability of causalities and economic losses. Secondly, the detection system itself is vulnerable to failures that may compromise the detection of the intended emergency, specially when in loco sensors are deployed. Therefore, besides the designing and deployment of any emergency detection system, some important issues concerning errors and fault tolerance should be properly considered.

Actually, a specialized emergency detection system created to detect a single type of critical event, as already proposed in the literature, is naturally vulnerable to different errors and fault conditions. However, since the proposed system is a highly flexible and multi-sensory solution to detect virtually any type of critical event, its vulnerability to failures is increased as there are more sensors that may fail. Additionally, although the use of a single integrated system to handle all types of emergencies may significantly increase the efficiency of emergencies management in smart cities, a failure in the system may impair the entire response to emergencies. Therefore, errors and fault tolerance should be considered as important quality issues to be pursued.

Errors management and fault tolerance in critical systems are related to different aspects that have been addressed by research works in the last decades. In general, the impact of failures to the expected services of critical systems has been addressed as the dependability concept, which combines the availability, reliability and maintainability aspects of any system. When assessing dependability, the probabilities of failures and the required time for repairing are considered in different metrics, as well as the probable system downtime, providing important information about the resistance of critical systems to a set of anticipated failure conditions [43]. These principles may be also applied to emergencies detecting and alerting.

When implementing sensors-based monitoring applications, an initial concern that affects its dependability is energy depletion. Although it is common to consider battery-operated sensor units in usual sensor networks, the target scenario of the proposed system will be smart cities, with easy

and affordable unlimited energy supply. However, since some emergencies may interrupt the regular energy supply (e.g., destroying electric power poles and wires), batteries may be used to power the EDUs. In this case, the energy depletion of the batteries has to be modelled and accounted for. Nevertheless, even though the battery replacement time might be considered as an important dependability parameter in this case, the lack of regular energy supply may be also modelled as an emergency situation itself, triggering alerting and mitigation actions.

The other important error condition is hardware failure. The sensor units and the processing devices that compose the EDUs are prone to hardware failures, resulted from natural component wearing. In this case, the Mean Time To Failure (MTTF), Mean Time Between Failures (MTBF) and Mean Time To Repair (MTTR) are useful metrics that have been used to assess hardware failures and that are useful when implementing and deploying the EDUs.

Since the proposed system is composed of different elements that communicate through some networking infrastructure, the existence of permanent and transient communication failures should be also accounted for. In fact, the use of reliable protocols is an important element in this sense, but other aspects are also relevant when concerning emergencies detection. For example, since an EDU will “cover” a particular area, its disconnection may result in emergencies that are not being properly detected: in fact, it may be detected, but the EPU is not being informed. Some previous works have addressed this issue, employing different formalisms for that [44,45], but with some limitations. However, some strategies can be used to enhance reliability even in the presence of emergencies, usually employing wireless communication technologies since they are more resistant to critical situations [46]. Additionally, the proposed system could be extended to provide an active synchronous probe service to transmit a “ping” message from the EPU to all EDUs, checking if they are active. Such a service will be addressed in future works.

Still considering the most relevant failure conditions for the proposed system, the use of cameras to detect critical events makes the EDUs susceptible to coverage failures [47]. If the positioning of the camera is not planned considering all obstacles, visual sensors may become faulty (occluded) [48], reducing the effective Field of View of the camera and potentially compromising the detection of critical events. Considering the proposed approach that attaches a camera to each EDU, occlusion is only one of the concerns that should be accounted for, since cameras may also face other problems, such as poor luminosity and dust on the lens. Actually, the same availability issues that may affect any camera network are also relevant for camera-based events detection.

Finally, when detecting critical events, sometimes the EDUs may be damaged by “aggressive” events, such as fire and flooding. In fact, it is expected that the EDU will detect and report a critical event before the EDU is destroyed by it. In such a way, the same probe service that may be used by the EPU to detect permanent communication failures may also check if the EDU is still active. Whatever the case, since such failure is not easily predicted, the detection of critical events should also trigger some proactive maintenance action of the EDUs in the area of the detected event. Doing so, eventually damaged sensors or electronic components could be identified and replaced.

Table 2 summarizes the most common failure conditions that may compromise the effectiveness of the proposed emergencies alerting system.

Table 2. Common failures and potential solutions for the proposed approach.

Failure	Causes	Impacts	Solutions
Energy	Current energy supply is interrupted	An EDU or EPU gets offline. For EPU disabling, the impact is more severe	The use of backup batteries will enhance the resistance to energy failures
	An EDU uses only batteries and their energy is depleted	An EDU gets offline	Batteries recharging or replacement is required
Hardware	Components wearing out	Electronic components may malfunction or stop working	Periodic checking of the components
	Errors in the fabrication process	Electronic components may malfunction or stop working	Periodic checking of the components
Communication	Permanent communication failures	Disconnection of an EDU, an EPU or the MQTT broker	Backup communication links can be used
	Transient communication failures	Packet losses in EDU/EPU and EPU/MQTT-broker communications	Retransmission of lost packets or transmission of redundant packets
Coverage	Camera’s FoV is occluded by an obstacle	Critical events may be not detected	Pan-Tilt-Zoom cameras can be used
	Low luminosity	Critical events may be not detected	Infrared cameras can complement “conventional” cameras
Damage	Critical events may damage an EDU	The EDU may malfunction or stop working	Protective cases or deployment on “safer” spots

6. Practical Issues When Employing Cameras for Emergency Alerting

The use of cameras as an affordable and efficient sensor unit has been intensified in the last decade, with many applications being designed to perform comprehensive and sometimes creative monitoring by the processing of visual data. This vibrating scenario has opened many possibilities for camera-based emergency alerting in urban areas, but some issues should still be properly considered for such applications. This section discusses how such issues may impact the development of visual sensing for emergencies management.

The proposed approach is a multi-sensory system that can detect any type of event as long as they are properly defined. In practical means, all events detection systems that are dedicated to a particular event could be replaced by the proposed approach, which would act as a unified detection system to support the general goal of emergencies management in urban areas. Therefore, the previously mentioned complexities associated to a single-emergency alerting system is potentially increased in the proposed system, since its disabling may compromise the entire response of a city to new emergencies.

Besides the dependability issues discussed in the previous section, the adoption of the proposed approach to detect all types of emergencies may raise some efficiency concerns related to the computation cost. In a rough analysis, the addition of new sensor units to detect new types of events will also increase the processing and memory demands of the system as a whole. Since the EDUs will be scattered over a target city, with dozen or hundreds of those units deployed over a single urban area, energy consumption may also be an issue. Although the processing costs of the EDUs and EPU were not evaluated in this article, we believe that each city will have a particular demand that will require a specific analysis of the available resources. Such analyses should consider all required sensor units and the core processing devices (such as a Raspberry Pi or a BeagleBone board). Nevertheless, since the new generations of embedded electronic devices have achieved high energy, processing and memory efficiency, we believe that the overall performance of the deployed system will be satisfactory in most cases.

When employing cameras to detect critical events, a reasonable initial consideration is about the quality of the used camera hardware, since it may significantly impact the visual data processing algorithms and the computational cost and energy consumption of the EDUs. Actually, different models and types of cameras are available and the choosing of the best option is not straightforward. First of all, cameras with a fixed focus are cheaper, but they may provide low-quality visual data for

emergency detection. On the other hand, cameras with zooming capability are more expensive options but they can better adjust the FoV according to the monitored area. This difference in the characteristics of the cameras can go further, since cameras may have different resolutions and embedded codecs. Additionally, rotatable cameras may also be employed when obstacles have to be avoided. Since the proposed system is not dependent of any hardware specification, the choosing of the camera is impacted by the available budget for the system and the “precision” of the visual data processing algorithms.

The development of new cameras is a trend that should also impact the construction of events detection systems. This should have stronger impacts in some applications. For example, wildfire monitoring performed by drones has been allowed by recent technologies at affordable prices. When those drones are equipped with more accurate thermal and optical cameras, the possibility of effective detection of fire can be significantly improved [49].

Overall, a promising approach is to combine scalar and visual sensors, achieving higher performance when detecting critical events [26]. Actually, while scalar sensors can be valuable for the fast detection of critical situations, camera-enabled sensors can provide visual data for a better perception of the environment in different occasions. Additionally, visual sensors can reach farther and enlarge the coverage area of any event detection unit, while scalar sensors will provide data that are only accurate for the current location of the sensor. Therefore, we believe that the best approach is to combine both types of sensors.

Once again, the addition of multiple scalar sensors to the EDUs have some impact on energy consumption, since it is expected that all sensors will be operating continuously. In this scenario, the attachment of one or more cameras to the EDUs will put additional pressure on energy consumption, specially if high-resolution adjustable cameras are employed. Although it is expected that the EDUs will be powered by wired energy supplies in most cases, energy efficiency is an important design issue in smart cities and thus the emergencies detection units should be designed considering this issue, especially when backup batteries are used to increase the EDUs resistance to failures. For the new smart cities being developed, energy efficiency will be a recurrent quality parameter.

Another relevant issue is the prevention or forensics of critical events, which may be very important in crowded areas to avoid future emergencies and to identify criminal acts. In such cases, the storage and retrieving of images and videos is required, demanding proper storage infrastructure [25]. In such cases, some issues as storage space and transmission bandwidth may become relevant for some applications, demanding additional attention [50].

Since emergency alerting systems in urban areas are subject to false alarms, the use of cameras may also be valuable to reinforce detection decisions performed by scalar sensors, increasing the accuracy of the system. In such cases, assuming that systems may be fully automated or semi-automated, some measures can be adopted to reduce the probability of false alarms, such as the use of multiple cameras or data mining to support decisions. In this sense, since the acquisition of images and videos can also be performed from different sources of data, social media could be employed as a complementary source of data when detecting events, exploiting the idea that people may act as independent sensors in smart cities [51,52]. Moreover, for semi-automated systems, any detected event could require validation by a human, increasing accuracy at the cost of higher decision times for the systems.

Another promising issue for visual sensing is machine learning [37], which can also be exploited along with data mining from social media and supportive IoT systems. When data from different sources are considered, the probability of issuing correct and fast emergency alarms is increased, which is the objective of emergencies management in urban areas. In fact, such procedures are already being adopted by very sophisticated video monitoring systems in some large cities around the world [53], which promise the fast identification of people, vehicles, animals and even some patterns that can be classified as emergencies. In this sense, the same algorithms and cloud-based data sources could be employed in the proposed system, potentially achieving similar results. Whatever the case,

the proposed approach is intended to provide flexibility and a unified structure for any type of event detection and alerting, and thus any visual data processing algorithm can be incorporated to enhance the detection of emergencies.

Finally, the dramatic events of the COVID-19 pandemic have put huge pressure on the development of monitoring systems supported by artificial intelligence, with the processing of visual data at the center of such solutions. The development of active monitoring to identify people with fever, for example, has already been adopted in many airports and large cities, indicating an important development trend for emergency alerting.

7. Conclusions

The development of new monitoring technologies with reasonable processing power and affordable prices has the potential to transform the way we perceive our environment and detect critical events, potentially saving lives and reducing properties damages. Together with the maturation of communication protocols and the Internet of Things paradigm, the resulting scenario is fertile for the adoption of new services to improve our quality of life. In this scenario, with most people living in urban areas and with the proliferation of large cities around the world, the development of emergency alerting systems has become even more relevant, as critical events have the potential to affect a great number of people.

In this scenario of increasing urbanization, which is subject to a large variety of (critical) events, the adoption of cameras for visual monitoring can bring significant results, potentially enhancing emergencies detection systems. Cameras can be valuable when performing monitoring tasks, allowing distant monitoring from the EDU when the FoV is not obstructed. Moreover, visual data can provide rich information from the monitored area that could not be easily perceived by only scalar sensors. Such characteristics can be leveraged in any emergency alerting and management solution, with promising results.

The proposed system was designed to exploit visual monitoring by cameras along with scalar sensors to define a comprehensive, flexible and consistent solution that can be employed on any emergency-prone environment. The idea is to integrate all detection mechanisms in a single unified approach, which could be easily adapted to detect any type of event. Therefore, the presented definitions are important contributions in this sense.

This emergency detection system was evaluated in different ways. Initially, an implementation reference was defined, providing information about a hardware scheme that could be used to create an EDU. Moreover, we also openly provided communication and control software for both EDU and EPU elements, making the proposed system ready to be used. After that, some experiments were performed in the city of Porto, Portugal, demonstrating a practical application of multiple events detection that are displayed on a map. Finally, an extensive discussion about failures conditions and practical implementation issues was done, raising important concerns about emergencies detection in the real world. Together, all these discussions and experiments were important to demonstrate the effectiveness of the proposed system, even in an initial stage of evaluation.

As future works, new codes will be created and openly provided to allow for the detection of new complex events, further supporting the adoption of the proposed system. Doing so, new experiments should be designed to assess the detection of new types of events that have not been considered yet. Finally, large scale deployment of EDUs will be performed for more complete results.

Author Contributions: Authors contributed equally to the development of this work. D.G.C. and F.V. contributed with experiences and ideas for the definitions and fundamental concepts of the proposed approach. D.G.C., P.P. and A.A. contributed with discussions and reviews of related works. The experiments were implemented by D.G.C. All authors contributed to the writing of the manuscript and performed revisions. All authors have read and agreed to the published version of the manuscript.

Funding: This work was partially funded by the Brazilian CNPq Agency with grants 204691/2018-4, 407903/2018-6 and 303527/2018-8.

Conflicts of Interest: The authors declare no conflict of interest. The founding sponsors had no role in the design of the study; in the collection, analyses, or interpretation of data; in the writing of the manuscript, or in the decision to publish the results.

References

1. Yigitcanlar, T.; Kamruzzaman, M.; Foth, M.; Sabatini-Marques, J.; da Costa, E.; Ioppolo, G. Can cities become smart without being sustainable? A systematic review of the literature. *Sustain. Cities Soc.* **2019**, *45*, 348–365. [[CrossRef](#)]
2. Nagy, A.M.; Simon, V. Survey on traffic prediction in smart cities. *Pervasive Mobile Comput.* **2018**, *50*, 148–163. [[CrossRef](#)]
3. Alazawi, Z.; Alani, O.; Abdaljabar, M.B.; Altowaijri, S.; Mehmood, R. A Smart Disaster Management System for Future Cities. In *Proceedings of the 2014 ACM International Workshop on Wireless and Mobile Technologies for Smart Cities; WiMobCity '14*; Association for Computing Machinery: New York, NY, USA, 2014; pp. 1–10. [[CrossRef](#)]
4. Soyata, T.; Habibzadeh, H.; Ekenna, C.; Nussbaum, B.; Lozano, J. Smart city in crisis: Technology and policy concerns. *Sustain. Cities Soc.* **2019**, *50*, 101566. [[CrossRef](#)]
5. Shah, S.A.; Seker, D.Z.; Rathore, M.M.; Hameed, S.; Ben Yahia, S.; Draheim, D. Towards Disaster Resilient Smart Cities: Can Internet of Things and Big Data Analytics Be the Game Changers? *IEEE Access* **2019**, *7*, 91885–91903. [[CrossRef](#)]
6. Costa, D.G.; de Oliveira, F.P. A prioritization approach for optimization of multiple concurrent sensing applications in smart cities. *Future Gener. Comput. Syst.* **2020**, *108*, 228–243. [[CrossRef](#)]
7. Costa, D.G.; Vasques, F.; Portugal, P.; Aguiar, A. A Distributed Multi-Tier Emergency Alerting System Exploiting Sensors-Based Event Detection to Support Smart City Applications. *Sensors* **2020**, *20*, 170. [[CrossRef](#)] [[PubMed](#)]
8. Karray, F.; Jmal, M.W.; Garcia-Ortiz, A.; Abid, M.; Obeid, A.M. A comprehensive survey on wireless sensor node hardware platforms. *Comput. Netw.* **2018**, *144*, 89–110. [[CrossRef](#)]
9. Costa, D.G. Visual Sensors Hardware Platforms: A Review. *IEEE Sensors J.* **2020**, *20*, 4025–4033. [[CrossRef](#)]
10. Simonjan, J.; Rinner, B. Decentralized and resource-efficient self-calibration of visual sensor networks. *Ad. Hoc. Netw.* **2019**, *88*, 112–128. [[CrossRef](#)]
11. Peixoto, J.; Costa, D.G. Wireless visual sensor networks for smart city applications: A relevance-based approach for multiple sinks mobility. *Future Gener. Comput. Sys.* **2017**, *76*, 51–62. [[CrossRef](#)]
12. Akararai, M.; Margotat, N.; Sicard, G.; Fesquet, L. A Novel Event Based Image Sensor with spacial and temporal redundancy suppression. In *Proceedings of the 2020 18th IEEE International New Circuits and Systems Conference (NEWCAS)*, Montreal, QC, Canada, 16–18 June 2020; pp. 238–241.
13. Fan, H.; Hernandez Bennetts, V.; Schaffernicht, E.; Lilienthal, A.J. Towards Gas Discrimination and Mapping in Emergency Response Scenarios Using a Mobile Robot with an Electronic Nose. *Sensors* **2019**, *19*, 685. [[CrossRef](#)]
14. Roque, G.; Padilla, V.S. LPWAN Based IoT Surveillance System for Outdoor Fire Detection. *IEEE Access* **2020**, *8*, 114900–114909. [[CrossRef](#)]
15. Min, K.; Jung, M.; Kim, J.; Chi, S. Sound Event Recognition-Based Classification Model for Automated Emergency Detection in Indoor Environment. In *Advances in Informatics and Computing in Civil and Construction Engineering*; Mutis, I., Hartmann, T., Eds.; Springer International Publishing: Cham, Switzerland, 2019; pp. 529–535.
16. Yap, F.G.H.; Yen, H. Novel Visual Sensor Deployment Algorithm in Occluded Wireless Visual Sensor Networks. *IEEE Syst. J.* **2017**, *11*, 2512–2523. [[CrossRef](#)]
17. Li, S.; Kim, J.G.; Han, D.H.; Lee, K.S. A Survey of Energy-Efficient Communication Protocols with QoS Guarantees in Wireless Multimedia Sensor Networks. *Sensors* **2019**, *19*, 199. [[CrossRef](#)]
18. Bin Bahrudin, M.S.; Kassim, R.A.; Buniyamin, N. Development of Fire alarm system using Raspberry Pi and Arduino Uno. In *Proceedings of the 2013 International Conference on Electrical, Electronics and System Engineering (ICEESE)*, Kuala Lumpur, Malaysia, 4–5 December 2013; pp. 43–48. [[CrossRef](#)]
19. Lloret, J.; Garcia, M.; Bri, D.; Sendra, S. A Wireless Sensor Network Deployment for Rural and Forest Fire Detection and Verification. *Sensors* **2009**, *9*, 8722–8747. [[CrossRef](#)]

20. Hwang, J.; Jun, S.; Kim, S.; Cha, D.; Jeon, K.; Lee, J. Novel fire detection device for robotic fire fighting. In Proceedings of the ICCAS 2010, Gyeonggi-do, South Korea, 27–30 October 2010; pp. 96–100. [\[CrossRef\]](#)
21. Perumal, K.A.; Ali, M.A.M.; Yahya, Z.H. Fire Fighter Robot with Night Vision Camera. In Proceedings of the 2019 IEEE 15th International Colloquium on Signal Processing Its Applications (CSPA), Penang, Malaysia, 8–9 March 2019; pp. 270–274. [\[CrossRef\]](#)
22. Nguyen, V.D.; Le, M.T.; Do, A.D.; Duong, H.H.; Thai, T.D.; Tran, D.H. An efficient camera-based surveillance for fall detection of elderly people. In Proceedings of the 2014 9th IEEE Conference on Industrial Electronics and Applications, Hangzhou, China, 9–11 June 2014; pp. 994–997. [\[CrossRef\]](#)
23. Sucerquia, A.; López, J.D.; Vargas-Bonilla, J.F. Real-Life/Real-Time Elderly Fall Detection with a Triaxial Accelerometer. *Sensors* **2018**, *18*, 1101. [\[CrossRef\]](#)
24. Alonso, M.; Brunete, A.; Hernando, M.; Gambao, E. Background-Subtraction Algorithm Optimization for Home Camera-Based Night-Vision Fall Detectors. *IEEE Access* **2019**, *7*, 152399–152411. [\[CrossRef\]](#)
25. Shao, Z.; Cai, J.; Wang, Z. Smart Monitoring Cameras Driven Intelligent Processing to Big Surveillance Video Data. *IEEE Trans. Big Data* **2018**, *4*, 105–116. [\[CrossRef\]](#)
26. Costa, D.G.; Guedes, L.A.; Vasques, F.; Portugal, P. Adaptive Monitoring Relevance in Camera Networks for Critical Surveillance Applications. *Inter. J. Distrib. Sens. Netw.* **2013**, *9*, 836721. [\[CrossRef\]](#)
27. Costa, D.G.; Duran-Faundez, C. Open-Source Electronics Platforms as Enabling Technologies for Smart Cities: Recent Developments and Perspectives. *Electronics* **2018**, *7*, 44. [\[CrossRef\]](#)
28. Sogi, N.R.; Chatterjee, P.; Nethra, U.; Suma, V. SMARISA: A Raspberry Pi Based Smart Ring for Women Safety Using IoT. In Proceedings of the 2018 International Conference on Inventive Research in Computing Applications (ICIRCA), Coimbatore, India, 11–12 July 2018; pp. 451–454. [\[CrossRef\]](#)
29. Kumar, K.N.K.; Natraj, H.; Jacob, T.P. Motion activated security camera using raspberry Pi. In Proceedings of the 2017 International Conference on Communication and Signal Processing (ICCSP), Chennai, India, 6–8 April 2017; pp. 1598–1601. [\[CrossRef\]](#)
30. Khurana, R.; Kushwaha, A.K.S. Deep Learning Approaches for Human Activity Recognition in Video Surveillance-A Survey. In Proceedings of the 2018 First International Conference on Secure Cyber Computing and Communication (ICSCCC), Jalandhar, India, 15–17 December 2018; pp. 542–544. [\[CrossRef\]](#)
31. Abu-Bakar, S.A.R. Advances in human action recognition: An updated survey. *IET Image Process.* **2019**, *13*, 2381–2394. [\[CrossRef\]](#)
32. Kim, H.; Lee, S.; Kim, Y.; Lee, S.; Lee, D.; Ju, J.; Myung, H. Weighted joint-based human behavior recognition algorithm using only depth information for low-cost intelligent video-surveillance system. *Expert Syst. Appl.* **2016**, *45*, 131–141. [\[CrossRef\]](#)
33. Chen, C.; Shao, Y.; Bi, X. Detection of Anomalous Crowd Behavior Based on the Acceleration Feature. *IEEE Sensors J.* **2015**, *15*, 7252–7261. [\[CrossRef\]](#)
34. Vetter, K.; Barnowski, R.; Cates, J.W.; Haefner, A.; Joshi, T.H.; Pavlovsky, R.; Quiter, B.J. Advances in Nuclear Radiation Sensing: Enabling 3-D Gamma-Ray Vision. *Sensors* **2019**, *19*, 2541. [\[CrossRef\]](#)
35. Tomono, D.; Tanimori, T.; Kubo, H.; Takada, A.; Mizumoto, T.; Mizumura, Y.; Sawano, T.; Matsuoka, Y.; Komura, S.; Nakamura, S.; et al. First application to environmental gamma-ray imaging with an electron tracking compton camera. In Proceedings of the 2013 IEEE Nuclear Science Symposium and Medical Imaging Conference (2013 NSS/MIC), Seoul, Korea, 27 October–2 November 2013; pp. 1–5. [\[CrossRef\]](#)
36. Santos, G.L.; Endo, P.T.; Monteiro, K.H.d.C.; Rocha, E.d.S.; Silva, I.; Lynn, T. Accelerometer-Based Human Fall Detection Using Convolutional Neural Networks. *Sensors* **2019**, *19*, 1644. [\[CrossRef\]](#)
37. Govil, K.; Welch, M.L.; Ball, J.T.; Pennypacker, C.R. Preliminary Results from a Wildfire Detection System Using Deep Learning on Remote Camera Images. *Remote Sens.* **2020**, *12*, 166. [\[CrossRef\]](#)
38. Xie, D.; Zhang, L.; Bai, L. Deep learning in visual computing and signal processing. *Appl. Comput. Intell. Soft Comput.* **2017**, *2017*. [\[CrossRef\]](#)
39. Bu, F.; Gharajeh, M.S. Intelligent and vision-based fire detection systems: A survey. *Image Vis. Comput.* **2019**, *91*, 103803. [\[CrossRef\]](#)
40. Muhammad, K.; Ahmad, J.; Mehmood, I.; Rho, S.; Baik, S.W. Convolutional Neural Networks Based Fire Detection in Surveillance Videos. *IEEE Access* **2018**, *6*, 18174–18183. [\[CrossRef\]](#)
41. Howse, J.; Joshi, P.; Beyeler, M. *OpenCV: Computer Vision Projects with Python*; Packt Publishing Ltd.: Birmingham, UK, 2016.

42. Yassein, M.B.; Shatnawi, M.Q.; Aljwarneh, S.; Al-Hatmi, R. Internet of Things: Survey and open issues of MQTT protocol. In Proceedings of the 2017 International Conference on Engineering MIS (ICEMIS), Monastir, Tunisia, 8–10 May 2017; pp. 1–6.
43. Jesus, G.; Casimiro, A.; Oliveira, A. A Survey on Data Quality for Dependable Monitoring in Wireless Sensor Networks. *Sensors* **2017**, *17*, 2010. [[CrossRef](#)]
44. Jesus, T.C.; Portugal, P.; Vasques, F.; Costa, D.G. Automated Methodology for Dependability Evaluation of Wireless Visual Sensor Networks. *Sensors* **2018**, *18*, 2629. [[CrossRef](#)]
45. Deif, D.; Gadallah, Y. Reliability assessment of wireless sensor network deployments. In Proceedings of the 2016 Wireless Days (WD), Toulouse, France, 23–25 March 2016; pp. 1–6.
46. Pervez, F.; Qadir, J.; Khalil, M.; Yaqoob, T.; Ashraf, U.; Younis, S. Wireless Technologies for Emergency Response: A Comprehensive Review and Some Guidelines. *IEEE Access* **2018**, *6*, 71814–71838. [[CrossRef](#)]
47. Costa, D.G.; Silva, I.; Guedes, L.A.; Vasques, F.; Portugal, P. Availability Issues in Wireless Visual Sensor Networks. *Sensors* **2014**, *14*, 2795–2821. [[CrossRef](#)]
48. Jesus, T.C.; Costa, D.G.; Portugal, P.; Vasques, F.; Aguiar, A. Modelling Coverage Failures Caused by Mobile Obstacles for the Selection of Faultless Visual Nodes in Wireless Sensor Networks. *IEEE Access* **2020**, *8*, 41537–41550. [[CrossRef](#)]
49. Allison, R.S.; Johnston, J.M.; Craig, G.; Jennings, S. Airborne Optical and Thermal Remote Sensing for Wildfire Detection and Monitoring. *Sensors* **2016**, *16*, 1310. [[CrossRef](#)]
50. Yan, H.; Li, X.; Wang, Y.; Jia, C. Centralized Duplicate Removal Video Storage System with Privacy Preservation in IoT. *Sensors* **2018**, *18*, 1814. [[CrossRef](#)]
51. Ning, H.; Li, Z.; Hodgson, M.E.; Wang, C.S. Prototyping a Social Media Flooding Photo Screening System Based on Deep Learning. *ISPRS Int. J. Geo-Inf.* **2020**, *9*, 104. [[CrossRef](#)]
52. Costa, D.G.; Duran-Faundez, C.; Andrade, D.C.; Rocha-Junior, J.B.; Just Peixoto, J.P. TwitterSensing: An Event-Based Approach for Wireless Sensor Networks Optimization Exploiting Social Media in Smart City Applications. *Sensors* **2018**, *18*, 1080. [[CrossRef](#)]
53. Muhammad, G.; Alsulaiman, M.; Amin, S.U.; Ghoneim, A.; Alhamid, M.F. A Facial-Expression Monitoring System for Improved Healthcare in Smart Cities. *IEEE Access* **2017**, *5*, 10871–10881. [[CrossRef](#)]



© 2020 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 (<http://creativecommons.org/licenses/by/4.0/>).