A New System for Real-time Video Surveillance in Smart Cities Based on Wireless Visual Sensor Networks and Fog Computing

Afaf Mosaif and Said Rakrak
Laboratory of Computer and Systems Engineering (L2IS), Faculty of Science and Techniques, Cadi Ayyad University, 40000 Marrakesh, Morocco
Email: afaf.mosaif@edu.uca.ac.ma; s.rakrak@uca.ma

Abstract —Nowadays, public security is becoming an increasingly serious issue in our society and its requirements have been extended from urban centers to all remote areas. Therefore, surveillance and security cameras are being deployed worldwide. Wireless Visual Sensor Networks nodes can be employed as camera nodes to monitor in the city without the need for any cables installation. However, these cameras are constrained in processing, memory, and energy resources. Also, they generate a massive amount of data that must be analyzed in real-time to ensure public safety and deal with emergency situations. As a result, data processing, information fusion, and decision making have to be executed on-site (near to the data collection location). Besides, surveillance cameras are directional sensors, which makes the coverage problem another issue to deal with. Therefore, we present a new system for real-time video surveillance in a smart city, in which transportations equipped with camera nodes are used as the mobile part of the system and an architecture based on fog computing and wireless visual sensor networks is adopted. Furthermore, we propose an approach for selecting the camera nodes that will participate in the tracking process and we simulated three different use cases to test the effectiveness of our system in terms of target detection. The simulation results show that our system is a promising solution for smart city surveillance applications.

Index Terms—Fog computing, smart city, target detection, target tracking, video surveillance, wireless visual sensor networks

I. INTRODUCTION

The rapid development of information and communication technologies has allowed the development of smart city applications, which can improve the citizen daily life and makes urban planning and city governance more efficient. According to [1], the main goal of smart cities is to make citizens happier by utilizing information technologies. However, security is a prerequisite for happiness. In other words, a citizen will not be happy if he doesn't feel secure, especially in this world where crime rates and terrorist attacks are increased. Therefore, for a very long time, video surveillance systems, such as Closed-circuit television (CCTV) systems, are being used to monitor arias of interest in the city and to deliver the collected data to a central facility, where it is visualized and analyzed by a human operator. However, the attention of most individuals decreases below acceptable levels after only 20 min [2]. As a result, the live person watching the monitor cannot detect and keep tracking all information, especially those that are occurring at the same time in different areas.

Therefore, Wireless Visual Sensor Networks (WVSNs) are used in our proposed surveillance system in order to benefit from its advantages such as: - WVSNs consist of a large number of tiny visual sensor nodes called camera nodes, which integrate an image sensor, an embedded processor, and a wireless transceiver[3], - The camera nodes can be deployed more easily without the need of new cables installation [4], - The camera nodes can process the collected data locally (on-board) which reduces the total amount of data communicated through the network and it can provide different levels of intelligence depending on the used processing algorithms [3], - Based on exchanged information, the camera nodes can collaborate and reason autonomously [3] and send just the useful information to the Base Station (BS) for further analysis.

However, WVSNs are constrained in processing, memory, and energy resources. Therefore, they are still facing several issues, such as finding efficient collaborative image processing, finding efficient coding techniques and finding how to reliably send the relevant visual data from the camera nodes to the aggregation nodes or the BS in an energy-efficient way [5]. Furthermore, differing from scalar WSN nodes, the camera nodes in WVSN are characterized by a limited directional view, called Field-of-View (FoV), defined by the camera direction, its angle of view, its depth of view and its location, which affects the sensing coverage of the network. For that reason, coverage and connectivity have to be considered in this type of network.

To ensure public safety in the city, the camera nodes generate every day a massive amount of data that must be analyzed in real-time, especially in emergency situations, which request quick response and low latency. Therefore,

---

Manuscript received September 20, 2020; revised April 13, 2021.
Corresponding author email: afaf.mosaif@edu.uca.ac.ma.
doi: 10.12720/jcm.16.5.175-184

the well-known "Cloud computing" solutions have become inefficient for analyzing and processing streaming video in real-time due to its associated latency challenges, the network availability and the huge volume of data that need to be transferred to the cloud for decision making. Thus, fog computing, an extension of the Cloud Computing paradigm, is a promising solution that is used in our proposed surveillance system as it brings the benefits of cloud computing to the edge of the network [6]. As a result, information fusion, video analytics algorithms and quick decisions can now be located and performed on fog nodes close to the cameras, allowing for faster security decisions and more reliable image/video data transmissions. Furthermore, by using fog computing we can benefit from the service of the fog storage [7] and other characteristics summarized in Fig. 1 [8].

As video surveillance requires highly-reliable connectivity and bandwidth, 5G technology can be adopted in our proposed system as a technology of wireless communication because it enables much lower latency in service delivery and higher signal capacity and speed communication compared to existing cellular systems [9].

In summary, the main contribution of the present work is to present a new wireless system for video surveillance in smart cities, in which connected transportation equipped by wireless camera nodes are used as mobile nodes to improve the coverage of the system and an architecture based on fog computing is used to have real-time video surveillance, instant decision making, rapid reaction to emergency situations and benefit from its storage services. In addition, an approach for selecting the camera nodes that will participate in the tracking process is presented and 3 different use cases are simulated to test the effectiveness of the system in terms of target detection.

The remainder of this paper is organized as follows. Section II presents the related works, followed by Section III that describes in detail our proposed system, its camera nodes operations, and the method used to select the camera nodes that will participate in the tracking process. Section IV presents three use cases with the details and results of the simulation experiments. Finally, conclusions and future works are presented in Section V.

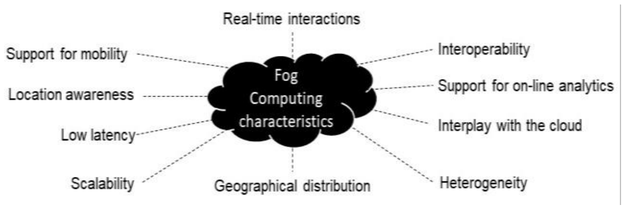

Fig. 1. Characteristics of fog computing

II. RELATED WORKS

WVSN is used in many researches to ensure safety in smart cities. Authors in [10] proposed a wireless system solution that uses existing public bus transit systems in future smart cities to collect video data from the cameras and physically deliver it for aggregation with a maximum median delay of 5 minutes, to be uploaded then to the data center. In this work, the future smart city is supposed to be equipped with video surveillance high definition cameras in every intersection of the city to cover 360° of vision in every street corner. In [4], Peixoto and Costa presented a relevance-based algorithm to position multiple mobile sinks in WVSN deployed on vehicles along roads and streets of a smart city in order to collect data from scalar sensors and cameras. The proposed algorithm detects forbidden and disconnected zones and dynamically computes the optimal positions of the sinks in order to position them in permitted areas closer to source nodes with higher sensing relevance. These presented approaches use also vehicles as mobile nodes of the network but as a mobile sink, not as node source like in our proposed approach. Generally, the idea behind using a mobile sink in WVSN is to reduce the energy consumption of the network. In our case, the energy preservation is not necessarily an issue as the mobile nodes are the cameras installed and powered by vehicles and the static nodes are powered by solar panels. Besides, mobile sinks can be used more in delay-tolerant applications, or our proposed system is designed for real-time surveillance.

In order to monitor an area of interest, WVSN nodes are deployed in a random or in a planned way to cover this area. As explained in the introduction, the camera nodes in WVSN are characterized by a limited directional view (FoV) that affects the sensing coverage of the network. Therefore, various coverage optimization methods are proposed and presented in the literature to maximize the network coverage while minimizing the coverage holes and the covered overlapping areas.

In the deployment step, strategies based on Voronoi-diagram or based on an algorithm such as Virtual Force Algorithm are proposed in the literature to enhance the network coverage (e.g. in [11] and [12] respectively). Whereas, the motility (rotating the node or the lens of the node) and/or the mobility of nodes are used in other works such as [13] to optimize sensing coverage of random deployed wireless multimedia sensor networks. In our case, these methods can be applied in the static part of our proposed system to maximize the coverage, while the mobile part will ensure the coverage of the public transportation interior and some uncovered areas in the network such as the places where the mobile nodes are stationed or rural zones in mobile nodes roads.

Since high coverage results in high tracking accuracy, the new areas covered in our proposed system will enhance the tracking of mobile targets in the network. Target tracking is an important application in WVSN, which involves the detection and localization of targets that move in the monitored area. In the literature, the prediction is used to activate or deactivate camera sensors during the tracking process to preserve the energy of the network such as [14]–[16]. In our case, the prediction is used to switch sensors from mode to mode to facilitate the tracking process and have an efficient collaboration between the network nodes**.** It is very important to note that mobile nodes in our approach don’t follow the target like in [17], it just collaborate with the static nodes in order to give more information about the targets and to track it when it is detected in uncovered zones. Thus, the system can obtain the position of the target and track it in a map and have slices of video describing its activity.

To deal with the latency challenges caused by the fusion and the processing of the large amount of video data generated by the city's camera nodes in real-time, Fog Computing is proposed as a solution in many researches. In [6], the authors proposed an urban speeding traffic monitoring system using the fog computing paradigm. In which, a drone is used to monitor the vehicles on the roads and sends the raw stream back to the ground control station where it is displayed on a screen. Instead of forwarding the whole video frame, the sub-area including the suspicious vehicle is identified by the human operator and extracted from the original frame and sent then to the Fog Computing node where a dynamic, real-time tracking algorithm is executed. Finally, the tracking result is sent back to the end-users. Authors in [18], presented the hardware options and key software components of their novel proposed architecture for a smart surveillance system, which is based on edge and fog computing concepts. The Edge computing is realized by a camera embedded system while for the fog computing a Video Content Management Software was developed for processing and logging of multimedia/heterogeneous content.

In our proposed system, an architecture based on fog computing concept is used to have on-site and instant decision making to manage emergency situations. Fog Computing nodes in our architecture not only provides the computing resource but also the storage space to store the mobile nodes collected data.

III. OUR PROPOSED SYSTEM

A. System Description

We propose in this paper a new system that aims to have real-time video surveillance in smart cities by using wireless camera nodes connected to fog computing nodes and the Cloud. As shown in Fig. 2, the system is composed of hybrid nodes: static and mobile ones.

The static camera nodes are organized in static cluster-based architecture, where the network is divided into smaller groups called Clusters. In each Cluster, there are Cluster Members (camera nodes) and Cluster Heads (fog nodes).

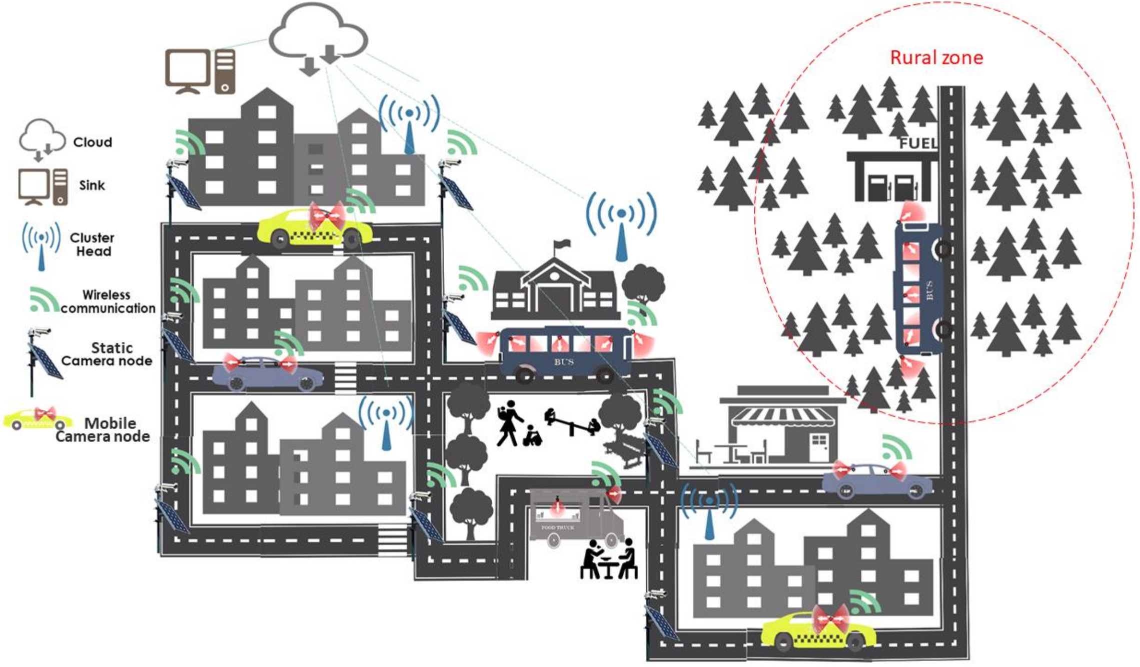

Fig. 2. The proposed system

The Cluster Members (CMs) detect important objects/events and send the data to their Cluster Heads for further processing (e.g. recognition, tracking, information fusion, etc.). The Cluster Heads (CHs) can exchange information with other neighboring CHs and they serve as relays for transmitting the data to the BS (Cloud).

The mobile camera nodes of our surveillance system are presented by the wireless camera nodes installed in public or private transportation like buses, taxis, etc., and are connected to the system network. Therefore, these camera nodes present the mobile part of the system and each one presents, temporary, a mobile CM of a CH in the static part of the system. As shown in Fig. 3, the vehicles can be equipped with one or more wireless camera sensors that record from the vehicles’ exterior and/or interior. For example, Fig. 3 (a) shows a car equipped with a node that has 2 camera sensors, one of which records in the front and the other in the rear.

The use of mobile camera nodes in our proposed video surveillance system enhances the coverage by monitoring non-covered areas. Therefore, isolated regions are related and the connectivity constraint of the network is relaxed. Also, it enhances the target-tracking application by giving a close-up view of the target in order to have more information about it.

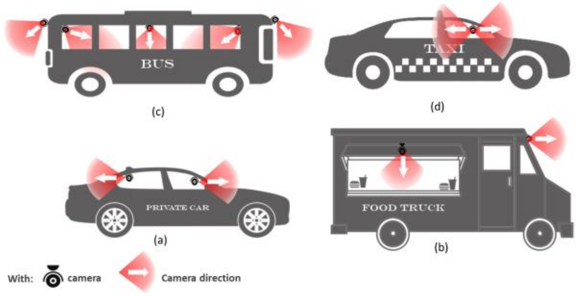

Fig. 3. Types of vehicles with attached camera nodes

To sum-up, the camera nodes of our system are composed of Static Cluster Members (SCMs) and Mobile Cluster Members (MCMs). The SCMs are composed of the static WVSN nodes. The MCMs are composed of mobile WVSN nodes classified into 3 types are: - Mobile nodes with a known trajectory, such as public buses. - Mobile nodes with random mobility, such as cars. - Mobile nodes with a long time of stops, such as food trucks.

B. System Architecture

Fig. 4 presents our proposed system architecture, which consists of three units: a *data collection unit* composed of static and mobile cluster members, a *fog computing unit* composed of static Cluster Heads that represent the fog nodes, and a *remote cloud center and data analysis* unit. The network computation and storage capabilities are distributed through the three units.

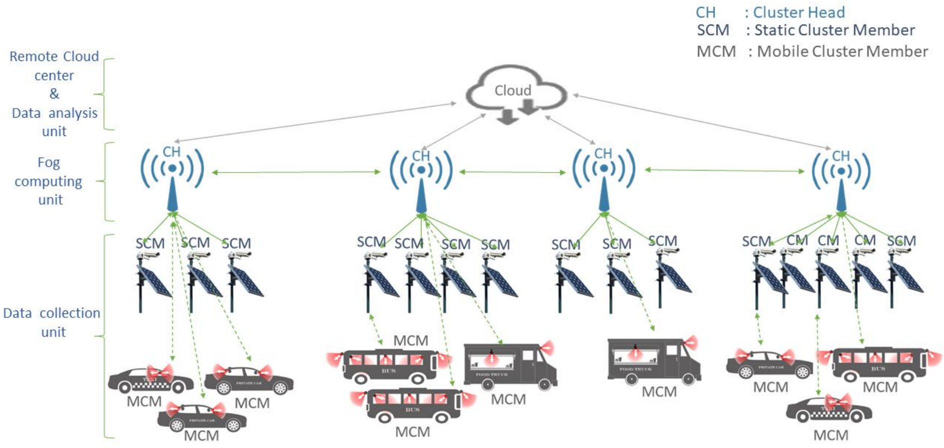

Fig. 4. Proposed system architecture

The *data collection unit* is responsible for image/video data collection. The SCMs in this unit are responsible just for lightweight video analytics (e.g. intrusion detection) because there are always connected to the CHs where complex video analytics can be done. Unlike the SCMs, the MCMs can be not connected all the time to the network (e.g. in rural zones). Therefore, they are powerful than the SCM in terms of computation and storage and are equipped with some complex algorithm (e.g. object recognition) that can be used even if the camera is out of the connected area of the city.

Instead of communicating directly to servers in the cloud, SCMs and MCMs can communicate to an intermediate layer represented in the present architecture as the *fog computing unit*, which is responsible for on-site data processing, information fusion, and instant decision making.

As useful video content collected by cameras is sparse, fog computing nodes send just relevant and filtered frames to the cloud unit for further analysis. In addition, these nodes are presented as the static CHs in the present architecture and not only provide the computing resource, but also the storage space.

Remote cloud center and data analysis unit is responsible for long-term analysis that requires powerful computation capability to extract valuable information that can be used to enhance the city security and to build historical records.

C. Assumptions

Throughout this paper, we make the following assumptions:

- Cluster Head nodes have great processing power
- Each camera node in the system knows its coordinates using Global Positioning System (GPS) or other localization methods
- Each camera node in the system can return the exact location of every object in its field of view

- Mobile camera nodes have more storage and processing power than the static ones
- All mobile camera nodes have the same proprieties such as video quality, field of view, etc.

- To deal with the energy problem in WVSN nodes, the SCMs are powered by solar panels and the mobile camera nodes are powered by batteries and rechargeable by the vehicles

D. Camera Nodes Operations

The Static Cluster Members

As the system is for real-time video surveillance, the SCMs cannot be in sleep mode to not lose targets and record special events. The authors in [6], declared that the video processing time at the Fog nodes is an issue for real-time processing. Therefore, the output frame rate should be equal to or higher than the input frame rate to meet the real-time video stream processing requirement. One solution mentioned by the authors is to decrease the resolution of surveillance video, which reduces its data size. However, this led to sacrifice the details in the video stream which can be a big loss of information. To address this problem and inspired by the work in the literature [19], the SCMs in our proposed system will operate under two modes: *low-resolution mode* and *high-resolution mode*.

To save energy SCMs operate with *low-resolution mode* and use lightweight algorithms to detect important events (e.g. abnormal event, wanted person detection) and send the data back to the CH for further analysis. The SCMs switch *high-resolution mode* if they are requested by the CH to participate in a tracking process or an event monitoring.

The Mobile Cluster Members

Unlike the SCMs that use just lightweight algorithms to detect important events, the MCMs are more powerful in terms of computation and storage and are equipped with some complex algorithms (e.g. faces and vehicles recognition) that can be used even if the MCM is out of the connected area of the city.

The MCMs operate under three modes: *normal mode*, *sleep mode*, and *real-time video surveillance mode.*

In *normal mode*, the MCMs store locally its collected data in groups (minimum of 2 groups) of video sequences (with the same size) where each group has at least one file. Each file contains the recorded video with some additional information such as group id, file id, time of record, vehicle position and speed.

As shown in Fig. 5, the groups are used in a circular fashion when one group fills up, it will be switched to the next group. If the archive mode is enabled and the MCM is connected to the system network, the filled group will be archived on the CH (Fog node) and its space will be freed. With this option, the collected data will remain on the fog even if the camera node is stolen or damaged.

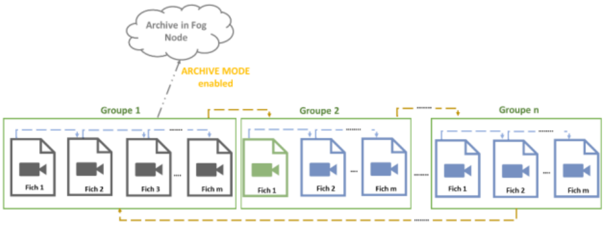

Fig. 5. MCM storage operation

In the case where the archive mode is not enabled or if the MCM is outside the connected area of the system, the camera node will record in files one by one locally until it runs out of space. Then, it will start from the first files recorded, and delete the files from oldest to newest, one by one.

If the MCM is outside the connected area, it will automatically lock and save any video file that was recorded when an important event was detected and it will assign a priority to these locked files, in order to be the first to be sent in case of a reconnection. In order to win time, other mobile nodes that have as destination the connected area can serve as a relay to transport a copy of these locked files to the system.

Once the MCM is connected to the network, it sends a periodic message with information about its position, speed, etc., to inform the nearest CH that it will be temporary its MCM. This message contains the information about the vehicle Id, number of cameras in the vehicle, the vehicle position (x; y) (that is expressed as GPS coordinates), and the speed of the vehicle. It sends also to its CH the data collected when it is out of the connected area.

In the case where the MCM is connected to the network and it detects an important event or a wanted target, it notifies its associated CH by sending the video sequences where the event is detected with additional information (e.g. position, type of target, speed, etc.). If the CH is occupied (all channels are used), a SCM in *low-resolution mode* will serve as a relay to transmit the MCM data that has a higher priority.

After five minutes of inactivity, the camera goes into *sleep mode* and as soon as it detects a motion, it turns on and starts recording in *normal mode.*

The *real-time video surveillance mode* is activated only if the MCM is connected to the network and is requested, by its associated CH, to participate in a tracking process or an event monitoring. In this mode, the MCM sends the collected data to the associated CH in real-time (live streaming).

It is very important to note that the mobile nodes do not follow the targets in the tracking process but they just cooperate with the SCMs to have a multi-view point and a close-up view of the target or to monitor the target in the non-covered area. Fig. 6 summarizes the MCMs operations.

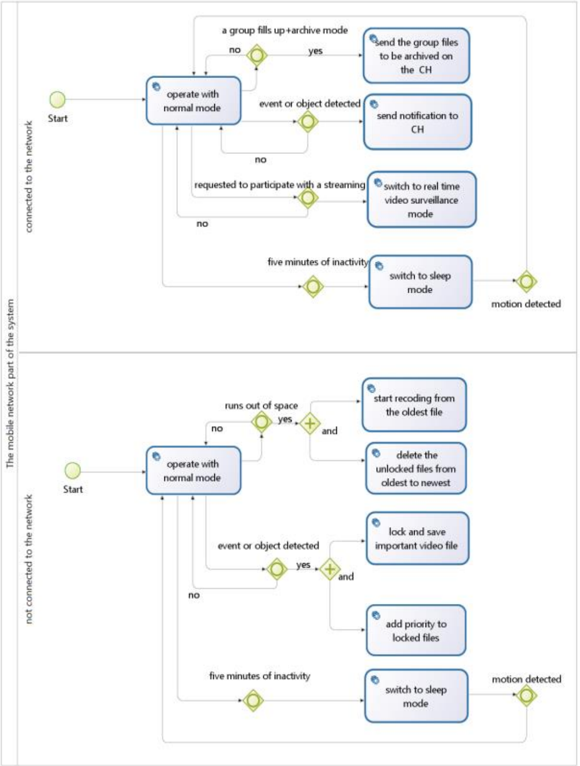

Fig. 6. MCMs operations

The Cluster Heads

In our architecture, the CHs represent the fog nodes and are responsible for on-site data processing, information fusion, future target position prediction, and decision making. These nodes can collaborate and share data with its neighboring CHs for data aggregation and target features handover. Also, they not only provide the computing resource but also the storage space to keep on-site relevant MCMs and SCMs videos data and send just important filtered frames to the BS (Cloud unit) for further analysis.

Fig. 7. SCM/MCM message information

Each SCM/MCM in the network sends their messages to their CH. These messages contain the information shown in Fig. 7 and described in Table I.

The CHs are connected to a *Shared Database* (SD) that stores the list of the processed message ids, the event ids, the resource states of each CH, the MCMs information, and the searched targets information. The history of the processed message-ids is important to not process the same message twice, for example, a MCM out of the connected zone can send a copy of its urgent messages with another MCM and when it is connected to the network it will send again these messages to its associated CH. Therefore, the CH will check first if the received messages are already processed or not by searching their message_id in the SD to process just the unprocessed ones. The event ids stored in the SD are also important to have a history of all the camera nodes that have participated in an event monitoring or a target tracking and in which CH the data is stored. The MCMs information stored in the SD gives the CHs the information about the cameras installed in MCM theirs type (records from the vehicles exterior or interior, faces forward or behind the vehicle) and the searched targets information.

TABLE I: SCM/MCM MESSAGE INFORMATION MEANINGS

Field	*Meaning*
message_id	message identifier (is unique)
message_class	type of message: urgent, normal, etc.
node_type	type of node : MCM or SCM
node_id	camera node identifier (is unique)
message_time	time when the message is generated at the camera node
Other information	this field differs from a message to another, for example, it can contain information about the event/target such as its speed and position, or information about the speed and the position of the MCM.

The CH can communicate with each other for efficient resource allocation and load balancing. If there is an insufficient resource available in the CH, it checks in the SD the resource available in its neighbor CHs to reassign some messages to them in order to serve the critical messages within its deadline.

The CHs have also other functions such as choosing the MCMs and the SCMs that will participate in a tracking process or an event monitoring in real-time as described in next subsection.

E. Our Proposed Approach for Camera Nodes Selections

To track a detected wanted target or to monitor an important event, the CH chooses the SCMs that will switch to the high-resolution mode to participate in the tracking or event monitoring process and selects the MCMs candidates that will provide more information about the event/target and its neighbors. However, if multiple mobile cameras are present and at the same time detect the same event/target as shown in Fig. 8, the CH have to choose the best node that will be temporary a MCM that operates in real-time mode to record in streaming the event or to participate in the tracking process.

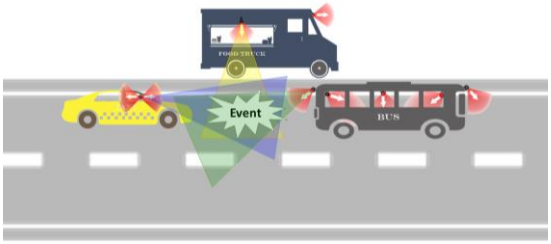

Fig. 8. MCMs that detect at the same time the same event/target

Therefore, some parameters have to be considered depending on the type of the target and the type of the MCM and its mobility such as:

- The angle of view and the direction, of the camera node, that are required to know if the target/event is in the FoV of the camera or not.
  The type of the mobile node, which is necessary to choose the best node to use depending on the type of event (e.g. mobile nodes with a long time of stop as food trucks will be the best choice to record a fixed event/target).
- The speed of the mobile node, which is useful for knowing the best node allowing a long duration of recording of the event.
- The distance between the mobile node and the target/event which is useful for choosing the best node that gives a close-up view of the event.

If the target is mobile, other parameters must be considered such as:

- The next position of the target/event, which is required to know the nodes that can participate in the tracking process.
- The speed of the target/event, which is needed to be compared to the speed of vehicles to choose the best one that well gives a long recording time of the event.
- Type of the target/event (vehicle or pedestrian), which can give information about the next position of the target and it can be useful in the prediction step.

In this paper, we assume that all MCMs have the same proprieties. Otherwise, more other parameters such as video quality, processing power, data rate, network signal strength, etc. need to be considered.

MCM Nodes Selection Procedure

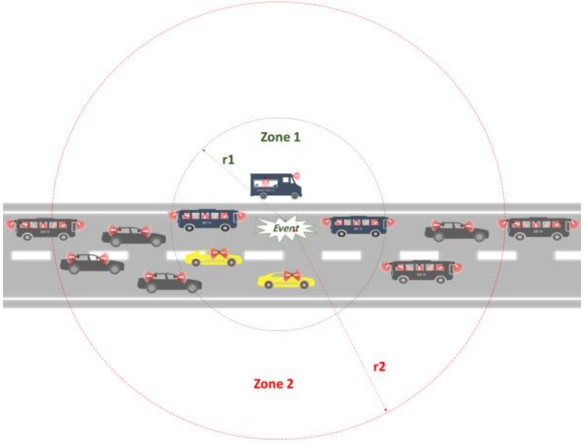

Fig. 9. MCMs selection zones

Once the event/target is detected for the first time by a SCM or MCM, the CH receives the information about the type of the event/target, its speed, and its position. Then, the CH attributes an identifier (*target_id*) to this event/target, and selects a list of camera nodes that can have a close-up view of the target/event at an instant t_0 (zone1 in Fig. 9) and the instant $t+t_0$ (zone2 in Fig. 9). These candidates can also give more information about the situation of the neighbor area of the event. In Fig. 9, ***r1*** and ***r2*** can be adjusted basing on the position, the type of the event/target and its mobility.

$$\begin{cases} if\ \ dist(target, MCM) < r1 \quad the\ MCM \in\ candidates\ at\ instant\ t_0 \\ if\ \ r1 < dist(target, MCM) < r2 \quad the\ MCM \in\ candidates\ at\ instant\ t + t_0 \end{cases}$$

After the selection of the candidates, the CH uses our proposed *MCM election algorithm to* choose from zone1 a candidate as a source node to send streams about the event in real-time, this node is called the Elected MCM (E-MCM).

In order to reduce system upgrades, calculation and communication costs, the E-MCM re-election procedure is non-periodic and is only invoked when the E-MCM no longer meets the eligibility criteria requested by the CH such as:

- Interruption of the connection
- The event is no more in the FoV of the E-MCM
- An obstacle (object/structure) is blocking the E-MCM view
- E-MCM failure due to hardware failures, physical disasters…
- E-MCM camera has bad viewing conditions such as illumination (external conditions)

At the end of the tracking or the event monitoring process, the CH archives the list of candidates in which the target was detected, to use it in the future to have more information and a different view of the event.

MCM Election Algorithm

The CH choose the E-MCM based on the following steps:

Step 1: Each candidate **k** in zone1 that has the event in its FoV calculates locally (for the first time) its selection index at an instant **t** using the following equations:

$$\begin{cases} \xi_k(t) = \alpha_1 \times \left|(Vspeed_k(t) - Tspeed(t))\right| + \alpha_2 \times VdistanceToTarget_k(t) + \alpha_3 \times Vtype_k(t) \\ \alpha_1 + \alpha_2 + \alpha_3 = 1 \end{cases}$$

where

$VdistanceToTarget_k(t)$: is the Euclidean distance between the node k and the target at the instant **t**

$Vspeed_k(t)$ **:** is the speed of the mobile node **k** at the instant **t**

$Tspeed(t)$ **:** is the speed of the target at the instant **t**

$Vtype_k(t)$ **:** takes a value **0** or **1** depends to the **type of the node k** and the **type of the target** at the instant **t**, as shown in the Table II.

TABLE II: VTYPE VALUES

		Target type	
		Fix (stationary)	mobile
Mobile node type	Fix (stationary)	0	1
	mobile	1	0

Step 2: Each candidate **k** sends then its selection index to the CH in the form of a message shown in Fig. 10. The meaning of the message fields is described in Table III.

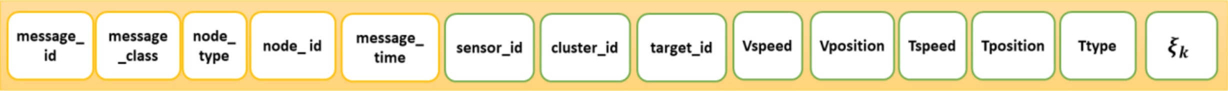

Fig. 10. a MCM candidate message information

TABLE III: A MCM CANDIDATE MESSAGE INFORMATION MEANINGS

Fields	Meanings
sensor_id	camera sensor identifier (is unique)
cluster_id	the Cluster Head Id
target_id	Target/event identifier (is unique)
Vspeed	vehicle speed
Vposition	and position (x; y) (that is expressed as GPS coordinates) information
Tspeed	target speed
Tposition	target position
Ttype	target type (abnormal event or target etc)
ξ_k	the vehicle selection index

Step 3: The CH sort the candidates according to their index of selection in ascending order, and the first element in the list is chosen E-MCM to send streams about the event in real-time.

$$E\_MCM = \arg\min_{k \in \Omega} \xi_k \quad \text{where} \quad \Omega = \{list\ of\ the\ candidates\ at\ instant\ t_0\}$$

The E-MCM re-election procedure is non-periodic. Therefore, the candidates recalculate their index of selection and send the message described in Fig. 10 only when it is requested by the CH.

Algorithms 1 describes the steps running at the CH in order to choose the E-MCM.

```
Algorithm 1: Algorithm running at the CH
1: Receive the Event/target location and its speed.
2: Select the MCM that has $dist(target, MCM) < r1$ and add them to a list of candidates.
3: Receive the calculated index of selection of each candidate k
4: Sort the candidates according to their index of selection in ascending order
5: Choose the first candidate to be the E-MCM that will provide a live streaming
6: Verify, in real-time, if E-MCM meets the eligibility criteria requested by the CH
7: if E-MCM no longer meets the eligible criteria requested by the CH:
    Repeat the steps 2,3,4,5,6
```

IV. PERFORMANCE EVALUATION

In this section, 3 scenarios are simulated which are: wanted person detection, wanted vehicle detection, and abnormal event monitoring. The general simulations parameters are summarized in Table IV. A section of Marrakech is imported from OpenStreetMap[20] then is used by SUMO[21] to generate vehicles paths and movement. The vehicles movement generated by SUMO is then imported into ns-3 simulator, using our visual node module presented in [22], to simulate the 3 scenarios in a wireless visual sensor network.

Fig. 11 presents a screenshot of the simulation with the target position in the 3 scenarios at an instant t different to 0.

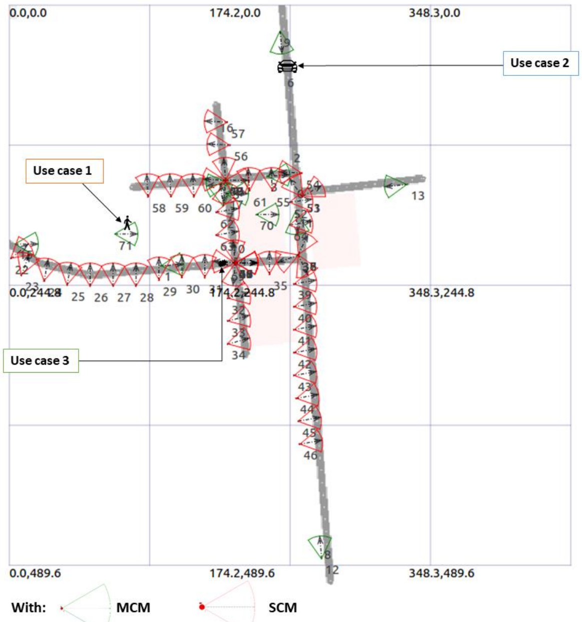

Fig. 11. Simulation screenshot

TABLE IV. SIMULATION PARAMETERS

Simulation time [s]	100
Road network	Section of Marrakech
Number of static sensors (SCM)	49
Number of mobile sensors (MCM)	12
Angle of view of vi (2 α) [rad]	π/3
Depth of view (R_V) of the sensors [m]	20
Distance between the SCM [m]	30
Number of targets	1
Radius of zone1 (r1) [m]	50

A. *Use Case 1: Wanted Person Detection*

Tracking a wanted person is a difficult task, which needs the collaboration of all camera nodes to detect the target and to track its movement positions. The proposed system in this article permits communication between camera nodes attached in public and private vehicles and static surveillance cameras in the streets. Therefore, tracking a wanted person in this system will be faster than the traditional systems (that contains just SCM) and more areas will be covered such as the interior of public transportations.

In this first use case, the search for a wanted person moves randomly in the zone of interest is simulated. The target speed corresponds to a pedestrian (1,38m/s) and the mobility model used is the Random Walk 2d.

The results of the simulation show that the target was moving in an area not covered by static cameras. However, it was detected 1 time after 10 seconds from the beginning of the simulation by MCM which is a camera node installed in a vehicle parked in this area.

Therefore, compared to the traditional systems, there are more chances to detect the target with our proposed surveillance system.

B. Use Case 2: Wanted Vehicle Detection

In this second use case, the search for a wanted vehicle is simulated. The movement of this target is generated also by SUMO.

The results of the simulation show that the target was moving first in an area which is not covered by SCM, however, it was detected by 1 MCM. Then at t=23s the target entered the covered area and was detected by another MCM and 8 SCM. As a summary, the target is detected by 10 nodes where 2 are MCM and 8 are SCM.

Therefore, our proposed system can be useful too in the case of vehicle detection and tracking, as it provides a close-up view of the target, which gives more information about vehicle driver, vehicle direction, etc. Also, there are more chances to detect the wanted vehicles even in non-covered and in rural zones (by the MCM).

C. Use Case 3: Abnormal Event Monitoring (accident...)

In this third use case, a fixed event monitoring is simulated. The results of the simulation show that the event is detected for the first time by a SCM, which sends its position to the network. After 19s, the event is detected by 1 MCM (node 1). At t=27s it is detected then by another MCM (node 11) and at t=33s the event is detected by 2 MCM (node 1 and node15). Finally, at t=62s it is detected by 1 new MCM (node 13). To sum up, the event was detected by 5 nodes, 4 MCM and 1 SCM which monitor the fix event during all the simulation time.

The MCMs in our proposed system will give more details and a close up view of the monitored event. In addition, the MCM candidates will give more information about the situation of the neighbor area of the event, which will be helpful in decision making such as which road must be closed, etc.

V. CONCLUSIONS

In this paper, a new system of video surveillance in the smart cities was presented. This system aims to have real-time video surveillance, instant decision making, rapid reaction to emergency situations and cover more zones in the city and its suburban areas. Therefore, transportations equipped by camera nodes are used as the mobile part of the system and an architecture based on fog computing and wireless visual sensor networks is adopted.

In addition, an approach for selecting the camera nodes that will participate in the tracking process was proposed in the current paper, and 3 different use cases are simulated, which are: wanted person detection, wanted vehicle detection, and abnormal event monitoring. The results of the simulations show that there is more chance to detect the target with our proposed surveillance system compared to the traditional one.

The proposed system will help law enforcement to get information quickly, deal with some emergency situations and find missing people or wanted targets in a more efficient way.

For future work, we will propose a load balancing algorithm at the fog nodes to process the received messages according to their levels of importance and delay tolerance.

CONFLICT OF INTEREST

The authors declare no conflict of interest.

AUTHOR CONTRIBUTIONS

Mosaif conducted the research and wrote the paper under the guidance and supervision of the Prof. Rakrak. All authors had approved the final version.

REFERENCES

[1] C. T. Yin, Z. Xiong, H. Chen, J. Y. Wang, D. Cooper, and B. David, "A literature survey on smart cities," *Sci. China Inf. Sci.*, vol. 58, no. 10, pp. 1–18, 2015.

[2] M. W. Green, "The appropriate and effective use of security technologies in U.S. schools: A guide for schools and law enforcement agencies," 1999.

[3] S. Soro and W. Heinzelman, "A survey of visual sensor networks," *Adv. Multimed.*, vol. 2009, p. 21, 2009.

[4] J. P. J. Peixoto and D. G. Costa, "Wireless visual sensor networks for smart city applications: A relevance-based approach for multiple sinks mobility," *Futur. Gener. Comput. Syst.*, vol. 76, pp. 51–62, 2017.

[5] M. M. Y. Charfi, B. Canada, and N. Wakamiya, "Challenging issues in visual sensor networks," *IEEE Wirel. Commun.*, vol. 16, no. 2, 2009.

[6] N. Chen, Y. Chen, Y. You, H. Ling, P. Liang, and R. Zimmermann, "Dynamic urban surveillance video stream processing using fog computing," in *Proc. IEEE 2nd Int. Conf. Multimed. Big Data, BigMM 2016*, 2016, pp. 105–112.

[7] J. Kharel, H. T. Reda, and S. Y. Shin, "An architecture for smart health monitoring system based on fog computing," *J. Commun.*, vol. 12, no. 4, pp. 228–233, 2017.

[8] H. Atlam, R. Walters, and G. Wills, "Fog computing and the internet of things: A review," *Big Data Cogn. Comput.*, vol. 2, no. 2, p. 10, Apr. 2018.

[9] R. Mahmud, R. Kotagiri, and R. Buyya, "Fog computing: A taxonomy, survey and future directions," *Internet of Everything*, pp. 103–130, 2018.

[10] W. Zhou, D. Saha, and S. Rangarajan, "A system architecture to aggregate video surveillance data in smart cities," in *Proc. IEEE Glob. Commun. Conf. GLOBECOM 2015*, 2015.

[11] T. W. Sung and C. S. Yang, "A voronoi-based sensor handover protocol for target tracking in distributed visual sensor networks," *Int. J. Distrib. Sens. Networks*, vol. 2014, no. 1, 2014.

[12] I. Boulanouar, S. Lohier, A. Rachedi, and G. Roussel, "A collaborative tracking algorithm for communicating target in wireless multimedia sensor networks," in *Proc. 7th*

IFIP Wireless and Mobile Networking Conference (WMNC), 2014, pp. 1–7.

[13] J. Pournazari, M. Alaei, and F. Yazdanpanah, "An energy efficient autonomous method for coverage optimization in wireless multimedia sensor networks," *Wirel. Pers. Commun.*, vol. 99, no. 2, pp. 717–736, 2018.

[14] M. Sabokrou, "Mobile target tracking in non-overlapping wireless visual sensor networks using neural networks," in *Proc. 3rd International Conference on Computer and Knowledge Engineering (ICCKE 2013)*, 2013, pp. 1–6.

[15] J. Amudha and P. Arpita, "Multi-Camera activation scheme for target tracking with dynamic active camera group and virtual grid-based target recovery," *Procedia Comput. Sci.*, vol. 58, pp. 241–248, 2015.

[16] I. Boulanouar, S. Lohier, A. Rachedi, and G. Roussel, "PTA : A predictive tracking algorithm in wireless multimedia sensor networks," in *Proc. Global Information Infrastructure Symposium-GIIS 2013*, 2013, pp. 1–6.

[17] T. Wang, *et al.*, "Following targets for mobile tracking in wireless sensor networks," *ACM Trans. Sens. Networks*, vol. 12, no. 4, pp. 1–24, Sep. 2016.

[18] G. Kioumourtzis, M. Skitsas, N. Zotos, and A. Sideris, "Wide area video surveillane based on edge and fog computing concept," in *Proc. 8th Int. Conf. Information, Intell. Syst. Appl. IISA 2017*, vol. 2018–Janua, pp. 1–6, 2018.

[19] O. Alaoui-Fdili, P. Corlay, Y. Fakhri, F. X. Coudoux, and D. Aboutajdine, "A Cross-layer approach for video delivery over wireless video sensor networks," in *TarXiv preprint arXiv:1501.07362*, 2015.

[20] J. Vargas, S. Srivastava, D. Tuia, and A. Falcao, "OpenStreetMap: Challenges and opportunities in machine learning and remote sensing," *IEEE Geosci. Remote Sens. Mag.*, pp. 1–13, Jul. 2020.

[21] M. Behrisch, L. Bieker, J. Erdmann, and D. Krajzewicz, "SUMO - Simulation of urban mobility - an overview," in *Proc. 3rd Int. Conf. Adv. Syst. Simul.*, no. c, pp. 63–68, 2011.

[22] A. Mosaif and S. Rakrak, "Visual node module: An open-source extension to the ns-3 network simulator," *Int. J. Commun. Syst.*, vol. 33, no. 12, pp. 1–18, 2020.

Mosaif Afaf is an engineer in Computer science, Network and Information Systems and Ph.D. student in the computer science department at Computer and Systems Engineering Laboratory (L2IS), Faculty of Science and Techniques, Cadi Ayyad University, Marrakech, Morocco. Her main research interests are related to Wireless Visual Sensor Networks and the Internet of Things.

Rakrak Said is a full professor at the Faculty of Sciences and Techniques, Cadi Ayyad University, Marrakech, Morocco. He is a researcher in the computer science department and head of the Computer and Systems Engineering Laboratory (L2IS). His main research interests are related to the Internet of Things, wireless sensor networks and cloud computing.