Peer-reviewed version available at J. Sens. Actuator Netw. 2018, 7, 35; doi:10.3390/jsan7030035

Article

Improving Animal-Human Cohabitation with Machine Learning in Fiber-Wireless Networks

Sandeep Kumar Singh*, Francisco Carpio and Admela Jukan

Technische Universität Carolo-Wilhelmina zu Braunschweig, Germany
{sandeep.singh, f.carpio, a.jukan}@tu-bs.de
* Correspondence: sandeep.singh@tu-bs.de; Tel.: +49-531-391-9651

Abstract: In this paper, we investigate an animal-human cohabitation problem with the help of machine learning and fiber-wireless (FiWi) access networks integrating cloud and edge (fog) computing. We propose an early warning system which detects wild animals nearby road/rail with the help of wireless sensor networks and alerts passing vehicles of possible animal crossing. Additionally, we show that animals' detection at the earliest and the related processing, if possible, at sensors would reduce the energy consumption of edge devices and the end-to-end delay in notifying vehicles, as compared to the scenarios where raw sensed data needs to be transferred up the base stations or the cloud. At the same time, machine learning helps in classification of captured images at edge devices, and in predicting different time-varying traffic profiles– distinguished by latency and bandwidth requirements– at base stations, including animal appearance events at sensors, and allocating bandwidth in FiWi access networks accordingly. We compare three scenarios of processing data at sensor nodes, base stations and a hybrid case of processing sensed data at either sensors or at base stations, and showed that dynamic allocation of bandwidth in FiWi access networks and processing data at its origin leads to lowering the congestion of network traffic at base stations and reducing the average end-to-end delay.

Keywords: Fiber-wireless networks; edge (fog) computing; sensors; machine learning, ZigBee.

---

1. Introduction

With every passing year the animal habitat is shrinking, and new infrastructures like highways, railways, etc. are built across densely forested areas to connect cities. Thus, wild animals often stray into human habitats, cross roads or rails, and sometimes results into deaths of humans as well as animals due to various reasons, including non-intentional like animal-vehicle collision. Apart from Asia and Africa, where animal-human conflict is common [1], according to a statistic [2] in Germany alone 194,410 animals were involved in animal-vehicle collision in the years 2015-16. Even though the technological advancements in the computing and networking are today utilized to build smart cities, rails, etc., these systems have not been considered for animal-human cohabitation [3]. Various sensors such as infrared cameras, optical fiber sensors, radars, etc. could be added to the smart hot zones nearby the road/rail track or adjoining areas (to human habitation) to detect the movement of animals and warn humans/vehicles in advance for a likelihood of animals forage. Nevertheless, there are a few notable works that studied some aspects of the animal-human cohabitation problem [1,4–10]. A smart fault detection system presented by [4] in Sri Lanka was proposed to detect breakages in electric fences, which keeps wild elephants away from humans in order to protect both humans and animals. Similarly, a wireless sensor network (WSN) system that uses passive nodes and infrasonic sounds was deployed in India, as reported by [5] to deter elephants from crossing railways. The use of infrared sensors and seismic sensors was proposed in [6] for detecting wild elephants entering villages. A new WSN system proposed by [7,9] was developed to protect both animals and humans by alerting drivers about possible wildlife crossing. Recently, [10] uses a wireless sensor network to collect information about animals' behaviors using a neural-network-based classification algorithm, which leads to lower

Peer-reviewed version available at J. Sens. Actuator Netw. 2018, 7, 35; doi:10.3390/jsan7030035

power consumption and better behavior monitoring. Although past work [1,4–10] has tackled the issues of animal monitoring or a warning system, a study of resource usage at edge/access as well as higher layers such as a so-called fog and cloud layers (including base stations, mini clouds and the cloud), with a smart management system for animal-human cohabitation has not been considered yet.

To take advantage of today's smart systems to integrate early warning systems, we consider three recent trends: fog-to-cloud distributed computing, artificial intelligence methods, and fiber-wireless (FiWi) networks. With fog-to-cloud distributed computing, we can intelligently store, process and communicate among various layers of edge, fog and cloud [11]. The ability of fog computing to process data locally is critical for early warning system which is latency-sensitive. Artificial intelligence (AI) techniques, such as machine learning, would on the other hand help alleviate network congestion by processing the raw data at AI-enabled edge devices and transmitting only useful information to fog nodes (e.g., base stations) using its inbuilt capabilities of learning, analysis and decision making [3,12]. For instance, a small computing edge device, e.g., Raspberry Pi, integrated with global positioning system (GPS), camera and other sensors could run customized AI algorithms to transfer only relevant data to the cloud, which could reduce the overall energy consumption in the data capturing, processing and transmission. Finally, we propose to integrate the concept of passive optical and wireless networks, known as fiber-wireless (FiWi) networks, to build an early warning system network. This is because passive optical network (PON) is economical not only to provide broadband services, but also an access to the next generation (5G) mobile networks.

In this paper, we propose and study an early warning system using the "things"-fog-cloud system comprising of sensors or Internet-of-Things (cameras, optical fiber sensors), base stations (BSs) and the cloud. In the proposed system, we first detect movements of animals at sensor nodes, then transfer sensor data to fog nodes (at BSs) over wireless sensor aggregation networks, and finally to the cloud for further processing over passive optical network (PON). The information processed is then distributed to humans (or vehicles) that are living (or passing by) areas of animal habitats. Since an early warning system is a latency-critical application, thus the main question we try to answer is whether to transfer all data to higher layers for better inference (with respect to classification or alert accuracy) or to process data at edge devices before transferring to the fog nodes. At the same time, however, processing of data at its origin, i.e., edge devices (that would result in transferring lesser data) could result in false alarms being generated due to the limited store, compute and processing power of edge devices. Thus, first we try to find a trade-off between communication cost and computation/processing cost through an experimental setup by showing the average amount of current drawn (or energy consumed) during image capturing, processing and transmission. Then, we show the trade-off between end-to-end latency (time between animal appearance to notification) and data volume transferred for each animal detection event in our distributed computing and AI-enabled early warning system through a simulation.

The rest of this paper is organized as follows. Sections 2 and 3 briefly describe the suitability of FiWi networks and machine learning, respectively, in the proposed animal-human cohabitation system. Section 4 shows the communication and computation costs through experimental measurements. We describe implementation details of an early warning system in Section 5, and evaluate the performance through a simulation approach in Section 5.2. Finally, we conclude the paper in Section 6.

2. FiWi Networks

The dependency on cloud-based services is growing, and so is the amount of network traffic from Internet-of-Things (IoT) devices to the cloud, and vice versa. Thus, hybrid fiber-wireless networks presents a viable solution to ensure high capacity, high flexibility and low-cost broadband access to Cloud-based services [13]. A typical architecture of a FiWi network is an integrated passive optical network (PON) and wireless networks, where each optical network unit (ONU) is connected to several wireless nodes (e.g., sensors, smartphones) through a wireless gateway (e.g., base station). Fig. 1 illustrates the envisioned FiWi architecture that integrates passive optical network (PON) and wireless

Peer-reviewed version available at J. Sens. Actuator Netw. 2018, 7, 35; doi:10.3390/jsan7030035

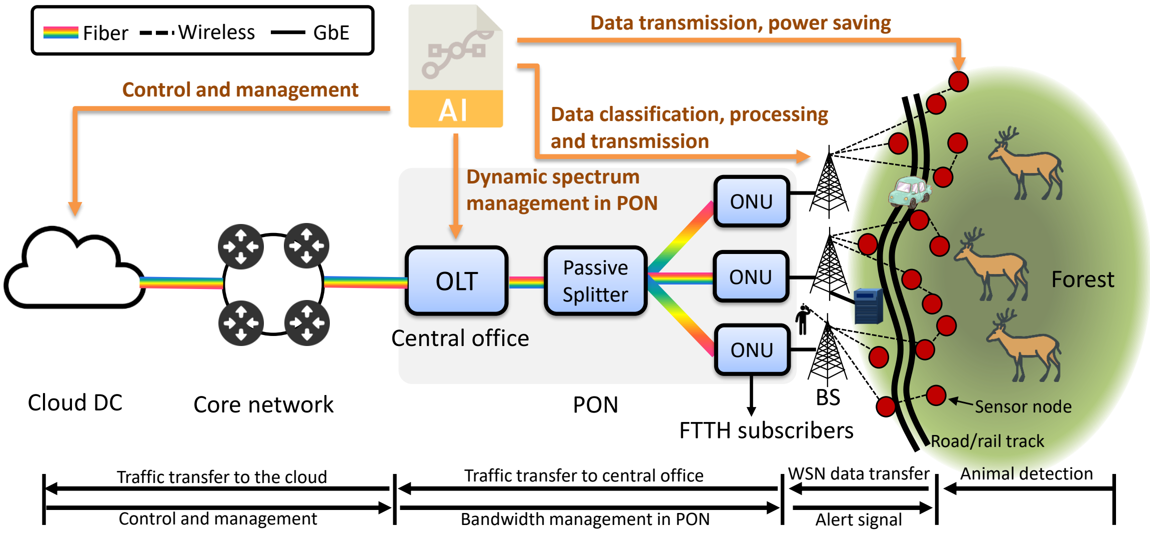

Figure 1. An architecture of FiWi network to carry cellular, FTTx, and animal traffic; and applications of AI techniques in different network subsystems are shown.

sensor networks (WSNs), and shows an example of network subsystems in all three layers: edge (sensors), fog (BSs, ONUs), and the cloud. As a main block of a FiWi network, a PON consists of an optical line terminal (OLT) at the service provider's central office, passive splitter and optical network units (ONUs). An OLT is a terminal equipment that includes a gateway router to connect PON to the fiber backbone or core networks. Additionally, its main function is to define transmission window and allocate bandwidth to optical network units (ONUs) which are near the end users. ONUs are expected to serve heterogeneous traffic of end-devices, since they would connect to wired, e.g., fiber-to-the-home (FTTH) subscribers, and wireless traffic through BSs in the 5th generation mobile networks. A single or a group of passive splitters allows to share a PON network among many end users by means of splitting power in Ethernet PON and/or demultiplexing wavelengths in wavelength-division multiplexing (WDM) PONs.

In this paper, we use orthogonal frequency-division multiplexing (OFDM)-PON architecture, where each ONU processes only its assigned fraction of the OFDM spectrum generated by the OLT. This is because of its known better performance in terms of latency and synchronization compared to other PON technologies [14]. Also, dynamic bandwidth allocation techniques used in OFDM-PON fully exploit the spectrum flexibility. This motivates us to use the system such that we allocate subcarriers' bandwidth based on the traffic volume of each class (cellular, FTTH and animal) within each ONU. At the same time, however, segregation of bandwidth requires the classification and prediction of traffic volumes, which in fact can be done through AI techniques. Various AI-based applications can be envisioned in Fig. 1, including the management of network devices and bandwidth allocation. For instance, instead of running a centralized bandwidth management of PON subsystems at the cloud, a central office (OLT) could dynamically allocate spectrum in PON subsystems based on the traffic predicted at its ONUs. This would help improve the bandwidth management and also latency.

In Fig. 1, wireless sensor nodes (edge devices) form an aggregation network along the road (e.g., to mimic a point of human-animal conflict) and they are wirelessly connected with the so-called sinks, denoted as base stations (BSs). Whenever a WSN node detects animals' presence, it tries to send sensed data (e.g., picture/video) of its surroundings to a nearest BS. The energy aspect is also relevant for systems that are not connected with power grid and are, for example, either battery-based, or solar-powered. Another advantage of AI techniques is that it could also relieve the network congestion by avoiding the unnecessary data transfers, which also improves the end-to-end delay of those transfers that need to go over the network. On the other hand, a limited processing and storage resource pool at

Preprints (www.preprints.org) | NOT PEER-REVIEWED | Posted: 18 June 2018 doi:10.20944/preprints201806.0281.v1

Peer-reviewed version available at J. Sens. Actuator Netw. 2018, 7, 35; doi:10.3390/jsan7030035

sensors would also have negative impact on the accuracy of classification and prediction. In contrast, a cloud-based infrastructure connected to the ONUs could deduce information from the animals' raw data with more accuracy in classification and prediction. This makes it interesting to investigate the role and placement of AI-based methods in animal welfare and human-animal cohabitation systems based on FiWi.

3. The Role of Machine Learning

AI techniques, especially machine learning (ML), are mainly used in networking to make a distributed computing system intelligent enough to take decision on its own on various aspects, for example changing device setting based on collected and processed data; how much data to transmit from edge devices to fog nodes and/or to the cloud, etc. Machine leaning would help particularly in identification of animals' movement using image classification algorithms, and predicting animals' movement or traffic in general, correlate it with human movement and traffic, or classifying animals' behavior. Additionally, based on the network data traffic predicted, a hybrid approach of processing and communication could be used. In other words, animals' data could be processed at the source sensor (or end device) when the animal movement is predicted as more likely to happen. On the other hand, when animal movement is less likely to happen, during that period, most of the data collected could be transferred to mini-cloud (fog) nodes for more accurate processing. On the other hand, and as previously mentioned, attention needs to be paid to the accuracy of the classification and the prediction, which is a challenge.

Machine learning has also been successfully used in optical transmission systems and networks, as presented in a survey [12]. In our OFDM-PON architecture, machine learning could also split optical resources flexibly based on predicted traffic demands of heterogeneous services. For example, in PON, upstream flow scheduling at ONUs is more critical due to lower data rate (e.g., 2.5 Gb/s) as compared to downstream traffic at OLT with higher data rate (e.g., 10 Gb/s). Moreover, ONU needs to allocate OFDM subcarriers in OFDM-PON or time slot in Ethernet PON to various types of application demands keeping their requirements (latency, bandwidth) in mind. As observed in the literature, peak and low of various traffic patterns occur at different times during a day. Thus, ML-based traffic prediction could allocate just-enough bandwidth to data traffic demands [15]. In 5G, for example, cellular connections would need to satisfy latency in the order of milliseconds. Therefore, a set of dedicated but flexible frequency bands and slots must be assigned for latency-sensitive applications. More importantly, lightweight ML image classification models, such as MobileNet [16], can be run on power and resource constrained end devices, and they could be helpful in identifying animals' presence or absence, and other related information.

To address the issue of the said granularity of traffic patterns, we consider multi-class (three) traffic scenarios and use machine learning-based traffic prediction to allocate bandwidth as per traffic demands of each class. Specifically, we use a long short-term memory (LSTM) network model– a recurrent neural network with LSTM units [17], for the time-series traffic prediction, since it is capable of learning long-term dependencies. More importantly, it avoids vanishing gradients, which is a common problem in machine learning algorithms where the gradient of error functions decreases quickly without improving the learning process. To get a good prediction, a machine learning algorithm needs to be trained with many and different traffic patterns, and depending on the number of hidden layers, neurons per layer and training datasets, it predicts (also classify) the future events. In our approach, we first generate a time-varying traffic (real data) using the superposition of sinusoidal functions with different frequency components for training (67%), and for validation and testing (33%) of LSTM networks. We used an input layer, a hidden layer with 4 LSTM blocks or neurons, and an output layer for prediction.

In Fig. 2, a small fraction of real data and predicted data during training and testing are shown to illustrate the design. We can see that the LSTM neural network-based prediction is very good, as root mean square errors during training and testing are 3.04 and 3.08 respectively. Thus, LSTM

Preprints (www.preprints.org) | NOT PEER-REVIEWED | Posted: 18 June 2018 doi:10.20944/preprints201806.0281.v1

Peer-reviewed version available at *J. Sens. Actuator Netw.* **2018**, *7*, 35; doi:10.3390/jsan7030035

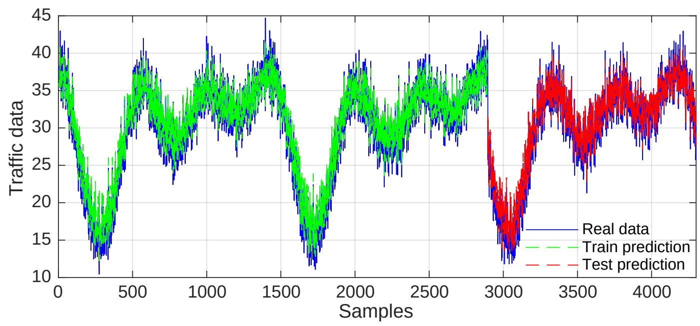

Figure 2. A fraction of real data and predicted data.

networks can learn various traffic patterns to predict the future traffic, which is the reason why we propose to use it in allocating spectrum proportionately. Additionally, for animal identification on end devices (Raspberry Pi) connected with sensors, we use a convolutional neural network-based MobileNet model, which is described in the next section.

4. Computation vs. Communication: an Exprememtal Approach

With the advances in edge or fog computing and acknowledging the fact that end devices are mostly power and resource constrained, even though they are capable of some amount of computing and processing, an important aspect in a Internet-of-Things (IoT)-fog-to-cloud system is to analyze the trade-off between computation cost and communication cost in terms of energy consumption at end devices. In this section, we show this trade-off through an experimental setup and measurement results. Throughout this paper, we assume that an edge (end) device, e.g., Raspberry Pi, has some computational capability and it can also communicate to other edge devices or upper layer nodes, e.g., base stations or servers. The end devices are integrated with various sensors.

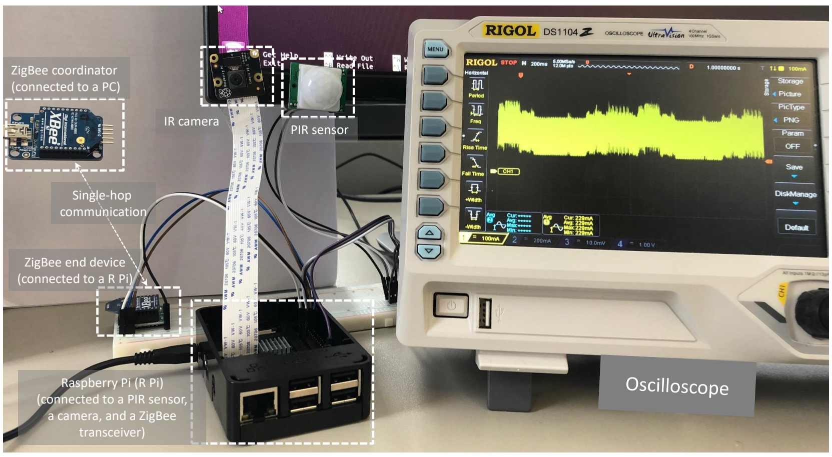

Figure 3. An experimental setup of a single-hop communication between two ZigBee (XBee) transceivers, one configured as an end device (WSN node) and another as a coordinator (sink). A Raspberry pi (R Pi) as an end device is integrated with an IR camera, a PIR sensor, and a XBee transceiver.

4.1. Experimental Setup and Measurement Results

The experimental setup is shown in Fig. 3. A Raspberry Pi3 (model B) as an end device is connected to a passive infrared (PIR) sensor to detect the movement of animals and used to wake up a 5 megapixel (MP) IR camera module, which captures animals' images. Additionally, We utilize a low-power ZigBee communication protocol to transfer data between an end device and a server (PC). For which, we use two XBee transceivers, one is configured as an end device and connected to the Raspberry Pi, and another is configured as a coordinator and it is connected to a server situated a few meters apart. Since the Raspberry Pi can compute and process images with the help of an ML image classification algorithm, we evaluate two scenarios: i) capture-transfer, and ii) capture-process-transfer. The first scenario captures an image and then transfers its binary data to the server. On the other hand, the later scenario captures an image, processes it to detect whether animals are present or not and their types, and finally it transfers only a message about the animals' presence in its vicinity. For the image processing and classification we install a pre-trained model (trained on a PC) using TensorFlow [16] on Raspberry Pi and run a convolutional neural network (CNN) algorithm whenever an image classification is needed (e.g., second scenario). The current measurement setup is shown in Fig. 3. We use an oscilloscope to measure the amount of current drawn by the Raspberry Pi module. The current is measured from the voltage drop across a very low shunt resistor (1 Ohm), which is connected in series to the Raspberry Pi module.

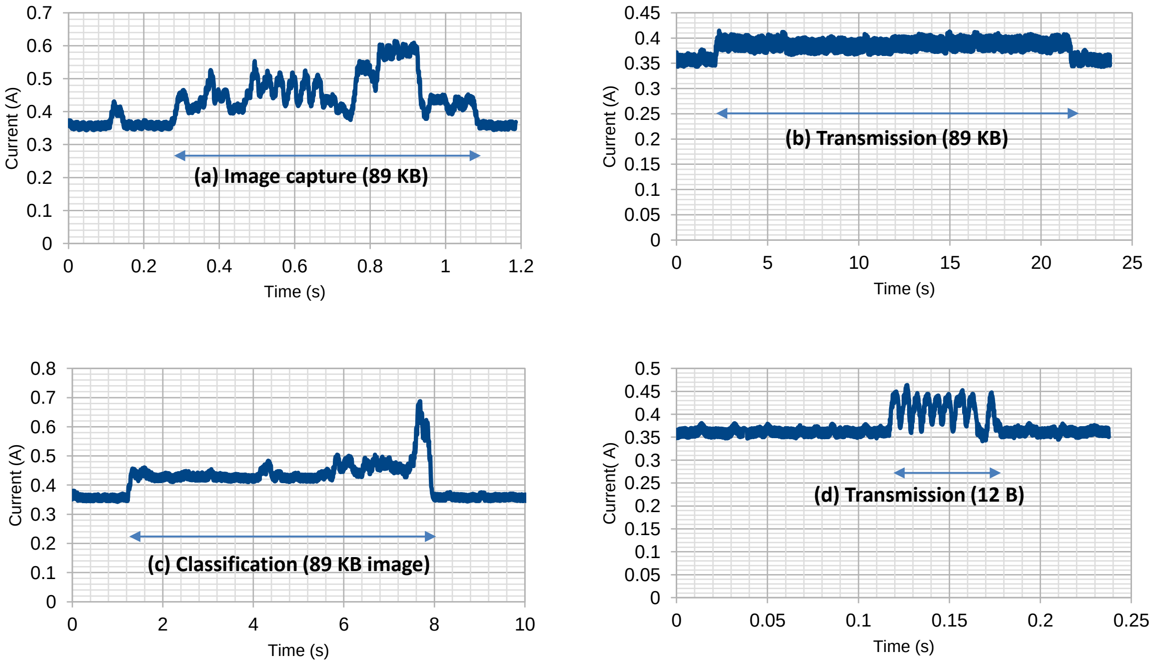

Figure 4. Average drawn current: (a) in capturing an 89 Killo bytes (KB) image, (b) for transmission of 89 KB image, (c) for classification of a captured image of the same image, (d) for transmission of processed data of size 12 bytes.

To find the trade-off between communication and computation costs in terms of average consumption of energy, let us evaluate individual parts associated with these scenarios. Fig. 4(a) shows the average current required to capture an 89 Kilobytes (KB) image, which includes the triggering by a PIR sensor. The amount of current drawn depends on various parameters: camera resolution, picture size, color quantization, image compression (raw, jpeg, png), etc. For this measurement, we used a 5 MP camera and set it up to capture 299×299 pixel color images. We used jpeg as compression format. The average time required for the camera to capture an 89 KB image is around 0.84 second. The second part in this scenario is transmitting the image captured in the previous step. The energy consumed (or current drawn) by the system during transmission varies depending on the amount of data transmitted, distance between the sender and the receiver, and the selected working mode since the XBee transceiver module may work on different modes: *coordinator, router* or *end device*. In this setup we evaluate a single-hop transmission, i.e., one XBee as an end device (integrated with the Raspberry Pi) and another as a coordinator (attached to a PC by a USB connector). The right part of the Fig. 4(b)

Preprints (www.preprints.org) | NOT PEER-REVIEWED | Posted: 18 June 2018 doi:10.20944/preprints201806.0281.v1

Peer-reviewed version available at J. Sens. Actuator Netw. 2018, 7, 35; doi:10.3390/jsan7030035

shows the average current drawn by the Raspberry Pi module by transmitting an 89 KB image. The transmission time for this operation is around 19.33 seconds, which corresponds to an average data rate of $\sim$ 37 Kbit/s, which is lower than the maximum data rate that a ZigBee (or IEEE 802.15.4) end device can transfer, i.e., 250 Kbit/s in 2.4 GHz spectrum band. Therefore, the energy consumed in the first scenario, the capture-transfer, at $5V$ is approximately given as $5 \times (0.44 \times 0.84 + 0.39 \times 19.33) = 39.5J$, which uses the average current $0.44A$ for image capturing, and $0.39A$ for transmission.

For the second scenario, the capture-process-transfer, we process a captured image before transmitting some relevant information, thus, Fig. 4(c) shows the average current required to process an image. For the image classification (processing), we used one of the widely used CNN models on constrained devices: MobileNet. MobileNet is a small size CNN that can support different input image resolution. The classification accuracy improves with higher resolution, however, at the expense of more processing time. The model is retrained to recognize two classes of images: the presence or the absence of an animal inside the captured image. The classification time for an 89 KB image is around 6.8 seconds. Finally, a 12 bytes data message about the presence or absence of animals and their types are transferred from the end device to the server, for which the average drawn current is shown in 4(d), which takes 0.05 second. The total runtime for the three operations shown in Figs. 4(a), (c) and (d) is around 7.6 seconds, and the energy consumed during the second scenario, the capture-process-transfer, at $5V$ is around $5 \times (0.44 \times 0.84 + 0.44 \times 6.8 + 0.40 \times 0.05) = 16.9J$, which results in $\sim$ 57.2% energy saving as compare to the previous scenario. Thus, we can conclude that the communication cost in power constrained devices could be very high if they require to transfer more data. Therefore, it is better to compute and process at the end devices, if possible, and transfer only relevant information to the server. It should be noted that the energy consumption can be further reduced by processing data near the sensor nodes using lightweight ML algorithms, such as support vector machine, however, the classification accuracy would be reduced, which might result in generating more false alarms [18].

5. An Early Warning System: a Simulation Approach

In this section, we present a framework for an early warning system, describing its implementation details, and evaluate it by a simulation approach.

5.1. Implementation Details

To analyze the performance of the proposed network system, we design the following subsystems, earlier illustrated in Fig. 1: (i) animal detection; (ii) message transfer to fog nodes (BSs); (iii) alert transmission to vehicles/humans, when necessary; and (iv) bandwidth allocation subsystem in PON, located between fog nodes and the cloud. Let us now present and discuss each subsystem one-by-one.

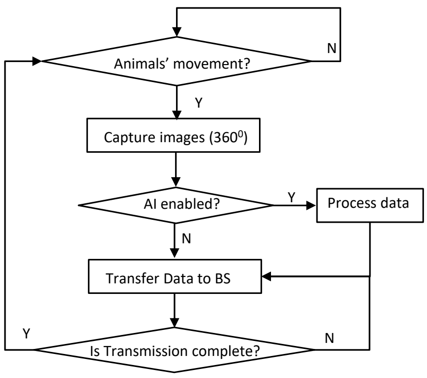

Figure 5. An implementation of program flow chart.

Animal detection subsystem is the most crucial part of the animal-human cohabitation system. As energy is a major issue in remote sensors, only a few sensors need to continuously sense the effects

Peer-reviewed version available at J. Sens. Actuator Netw. 2018, 7, 35; doi:10.3390/jsan7030035

for detection of animal crossing, like pressure, vibration etc. At the same time, disturbances caused by the wind and other events that can create false alarms (such as large birds) need to be differentiated from relevant animal events. It has been shown earlier that fiber- or infrared-based sensors are most suitable for this purpose [1]. Once animals are detected, the animal triggering sensors would wake up other sensors to capture mainly images, which then needs to be processed to find animals' size, position, speed, numbers, direction and other specific features. As we discussed before, the processing of raw data either at origin (end devices) or at higher layers (fog or cloud) depends on the capability of sensor nodes as well as on network ability to transfer data without congestion. Even though a capable end-device with various sensors could process the raw data, in case of images, the processed data need to be send to fog nodes (see Fig. 5), which can reprocess it to decide whether to generate an alert message or not.

The second subsystem is responsible for transfer of the processed or unprocessed data to fog (BS) nodes. For this purpose, whenever a sensor detects movements of animals, it tries to send raw data (e.g., pictures, GPS signal) or just relevant (processed) data to a sink node, i.e., BS. Whether to send raw or only relevant data depends on the AI running capability of end devices. We utilize a low-power ZigBee communication protocol to transfer data from sensor nodes to BSs, which specifies a maximum 250 kb/s data rate and transmits 128 bytes per packet over unlicensed spectrum of 2.4 GHz. As the sensor nodes are assumed to be deployed along rail/road track, and ZigBee utilizes the shortest path to send a packet from a sensor to a BS, we use tree topology in Riverbed (earlier known as OPNET) Modeler simulator [19]. The simulator models three different kind of nodes: *End Devices*, *Routers* and *Coordinators*. End Devices only act as sensor nodes without routing capabilities, while routers can play both sensor and forwarding roles. Finally, coordinators, acting as sinks, are responsible for the connection among all nodes, and are also capable to act as sensor or router nodes.

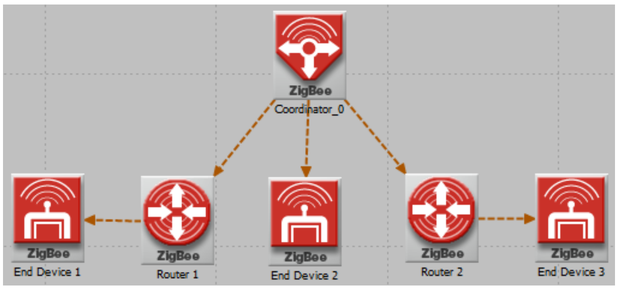

Figure 6. WSN topology per BS simulated by Riverbed Modeler [19]

The third subsystem is dedicated to alert transmission to humans, and to this end, it needs to be designed to process the data at fog nodes (BSs) and to trigger the alarm (notification) signal, when necessary. Although the alert signal is generated at BSs, animals' related traffic still needs to be sent to the cloud for long-term statistics and efficient execution of notification messages by sharing them with neighboring BSs. Therefore, we assume that the fog nodes (BSs, ONUs) locally connect to servers or a mini-cloud for processing and classification of various traffic (cellular, FTTH, animal). In this way, the mini-cloud is also the part of a fog computing system, able to replacing the cloud functions in the system, which is fog's salient feature.

As can be seen in Fig. 1, each BS is connected directly to an ONU through Gigabit Ethernet interface, and the OLT schedules the ONU's traffic by allocating required bandwidth as per their overall traffic demand. However, it should be noticed that various types of traffic share the ONU upstream bandwidth. Thus, as mentioned in Section 3, the forth subsystem utilizes the predicted traffic to dynamically allocate bandwidth of OFDM subcarriers per traffic class at ONUs. A new service request is routed over an existing lightpath that has free sufficient bandwidth, otherwise a new lightpath is created to carry traffic of a new request if sufficient resources (spectrum, transceiver) are free. It should also be noted that the cellular and other connection requests are blocked if resources

Preprints (www.preprints.org) | NOT PEER-REVIEWED | Posted: 18 June 2018 doi:10.20944/preprints201806.0281.v1

Peer-reviewed version available at J. Sens. Actuator Netw. 2018, 7, 35; doi:10.3390/jsan7030035

are not sufficient to satisfy the demand. The exception to this is animals' traffic, as it needs to be transferred to the cloud for long-term statistics. In our approach, animal's traffic is transferred either immediately or whenever resources are free, therefore we assume that a queue can be maintained at ONUs/BSs with sufficiently large size to store animals' data.

Table 1. Different parameters for the simulation

Parameters	Value (for different line rates)		
	Cellular	FTTH	Animal
Bit rate (in Gb/s)	1	10	0.00025
Arrival rate (Requests per second)	$exp(\lambda_1)$	$exp(\lambda_2)$	$exp(\lambda_3)$
Holding-time (in minutes)	$exp(\mu_1 = 10)$	$exp(\mu_2 = 10)$	0.00067

5.2. Simulation Results

The scenario we analyze using the Riverbed Modeler simulator consists of three end-devices and two routers per coordinator (BS), as shown in Fig. 6. End-devices act as sensor nodes, and routers act as relay as well as sensors, detecting the animal presence and capturing images of surrounding to deduce animal information, including animal position, movement direction, speed etc. Data is split into packets of 128 bytes payload (ZigBee), which are queued for transmission. The packets are consecutively forwarded to the coordinator acting as a sink (BS). We assume OFDM-PON as an optical access network with 16 ONUs, and a WSN network with 5 sensors is connected to each base station with shortest routing path, and each base station is directly connected to an ONU. The WSN topology is an aggregation-based network, as shown earlier in Fig. 1, where the BSs act as sink nodes. Wireless link capacity is assumed to be 250 kilobits per second (Kbps) in ZigBee protocol. We assume fiber capacity as 4 THz in OFDM-PON, which is divided among 320 spectrum slices, each with granularity of 12.5 GHz. Each spectrum slice could support a maximum of 25 Gb/s using QPSK modulation format. All 16 ONUs share equal proportion of fiber bandwidth, i.e., each ONU can reserve 20 spectrum slices.

We evaluate the performance of our animal-human cohabitation system by means of latency observed (i) with AI, that transfers the processed data from sensors to BSs (ii) without AI, that transfers raw sensed data, and (iii) the so-called *hybrid approach* that transfers most of the data during low traffic condition and only relevant information during busy hour.

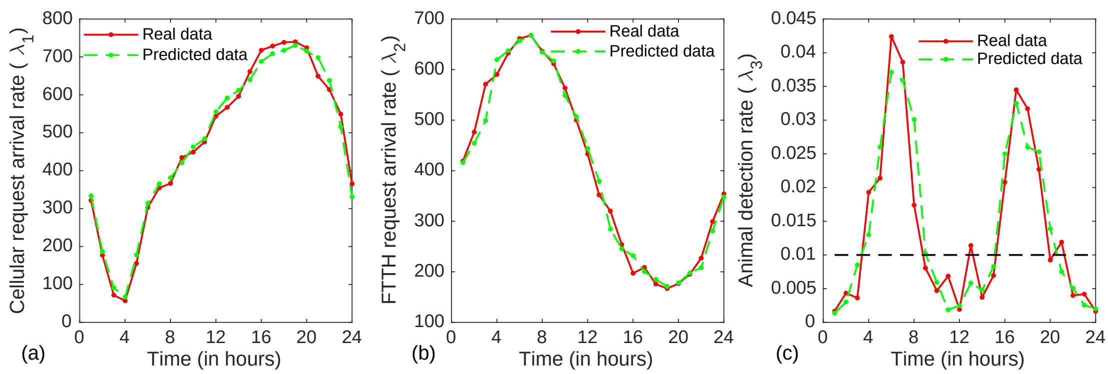

Figure 7. Real traffic vs. predicted traffic of three classes of services, namely cellular, FTTH and animal are shown in (a), (b) and (c), respectively.

We consider three classes of services: cellular (class-1), FTTH (class-2) and animal traffic (class-3). Fig. 7 illustrates real and predicted traffic of these three classes. The simulation parameters are listed in Table 1. It should be noted that traffic arrival rate varies with time t, and the traffic per class is

Preprints (www.preprints.org) | NOT PEER-REVIEWED | Posted: 18 June 2018 doi:10.20944/preprints201806.0281.v1

Peer-reviewed version available at J. Sens. Actuator Netw. 2018, 7, 35; doi:10.3390/jsan7030035

calculated as $Tr_k(t) = (N * \lambda_k(t)/\mu_k)$, where N is the number of class-k traffic sources, λ_k is the class-k arrival rate at the source of traffic, and μ_k is mean holding (service) time of class-k connections at ONUs. Notice that we kept the mean holding time of animal traffic arriving from sensors as constant (40 milliseconds), since the data volume (processed) of each animal connection that needs to be transferred from ONUs to the central office is fixed as 1 KB and the bit rate is 250 Kbit/s. However, remember that the latency required to transmit sensor data from edge devices to BSs mainly depends on the congestion in WSN networks. Furthermore, we are focused on upstream reservation, therefore, all results presented here are based on upstream reservation from edge devices to BSs to central office. We maintain a sufficiently large fixed size queue to store data packets of animal class at each ONU. Other latency-sensitive classes of traffic (cellular and FTTH) are either served immediately or blocked.

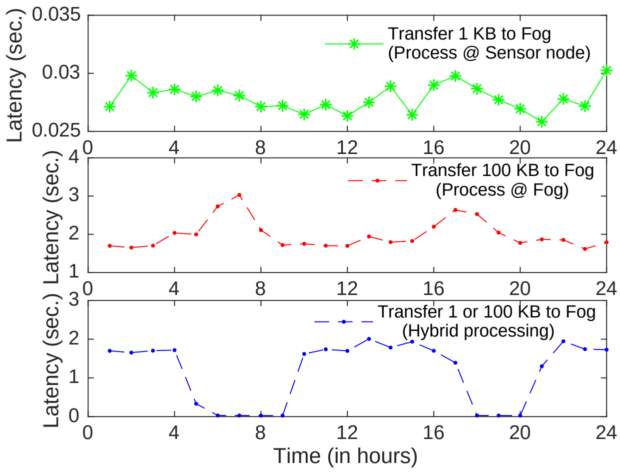

Figure 8. Average end-to-end packet delay (latency) from sensors to BSs.

Fig. 8 illustrates the average per packet delay (latency) between sensors and BSs of animal traffic under three data transmission scenarios: 1 KB, 100 KB and hybrid (i.e., a mix of 1 and 100 KB) using ZigBee protocol. These scenarios also depict the case of the data processing at different layers: edge (sensors), fog (BS) and a hybrid approach, i.e., processing either at the edge or at BSs. We could also compare these scenarios with the cloud processing, where raw sensed data is sent to the cloud for the processing and decision making. However, we ignore this scenario, as we did not simulate the core networks and the cloud. But it is obvious that the latency obtained in the cloud processing will be much higher than that of the other three scenarios presented here. Furthermore, we assume that the given specified data volume is transmitted for every animal detection event at sensors. In the first scenario, we assume that sensors are attached to some processing unit, thus they could run a simple machine learning algorithm to detect and classify animals' related data and transmit only 1 KB of data (or 8 packets) for every detection. In the second scenario, we assume that sensors leave the processing task to the fog layer (BSs) and transmit 100 KB data (or 800 packets) to BSs. The hybrid case, on the other hand, takes predicted animal traffic into the account to decide whether to process data at the edge layer (sensors) or at the fog layer (BSs). The decision threshold is shown by a black dotted line in Fig. 7(c), and it could be set or even varied by monitoring the congestion or by predicted animal traffic. In the hybrid case, we enforce processing at sensor nodes (1 KB transmission) when the predicted arrival rate is higher than the threshold, otherwise transmit 100 KB data per animal detection. From Fig. 8, we observe that when the raw data is processed at end devices, and only 1 KB data is transferred to BSs for per animal detection event then the average packet latency varies between 26 to 30 milliseconds, thus it can be deduced that WSNs are not congested. However, the effect of congestion on packet latency can be seen in the 100 KB transmission case, and the latency increases by two order of magnitude than that of the prior scenario (1 KB transmission). Interestingly, the average packet latency in the hybrid scenario can be reduced from a few seconds to milliseconds

Preprints (www.preprints.org) | NOT PEER-REVIEWED | Posted: 18 June 2018 doi:10.20944/preprints201806.0281.v1

Peer-reviewed version available at J. Sens. Actuator Netw. 2018, 7, 35; doi:10.3390/jsan7030035

during the congestion hours of the day by applying an intelligent control mechanism using the traffic prediction.

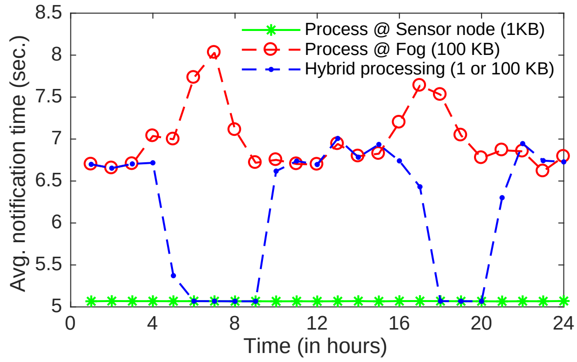

Figure 9. Average notification time in seconds to alert vehicles.

Fig. 9 shows the approximate average notification time, which is the time difference between animal detection and the alert message disbursed to the passing vehicles/humans. The notification time is calculated as the sum of data transmission time, average packet latency and processing time. We ignore the delay in disbursing alert messages from BSs to passing vehicles, since it is negligible as compared to other delays and remains same in all scenarios. Furthermore, we assume that processing times at sensors and BSs are 5 seconds and 1 second, respectively. ZigBee transmits 128 bytes data per packet at a maximum rate of 250 Kbit/s, thus the total transmission times of 1KB and 100 KB data are approximately 0.04 second and 4 seconds, respectively. As can be seen in Fig. 9, the notification time in case of transferring raw data (100 KB) varies from 6.5 seconds to 8 seconds as compare to transferring just relevant information (1 KB), which is around 5 seconds. It should be noted that even though the processing time at the edge layer (5 seconds) is selected four times higher than that of the fog layer (1 second), it is advisable to process data at the edge layer and transfer only relevant data to higher layers during peak hours (dawn and dusk) as done in the hybrid processing, especially in WSNs with low power communication protocol (e.g., ZigBee), since the packet latency and the transmission time are the main contributors in the notification time.

Finally, Fig. 10 illustrates the normalized fraction of dropped requests due to congestion of cellular and FTTH traffic with and without machine learning-based traffic prediction and bandwidth allocation, i.e., dynamic and static bandwidth allocation (DBA & SBA) in OFDM-PON. Without machine learning, the bandwidth allocated to each ONU is statically divided among the three classes of traffic based on the proportion of their long-term average traffic per day. With machine learning, the system divides bandwidth based on the predicted traffic for every hour, thus it exploits the dynamic assignment of subcarriers. Furthermore, the animal traffic is very low as compare to cellular and FTTH, thus animal (class-3) packets are stored in enough memory when they could not be sent to the central office immediately. Thus, there is no dropped packets for the class-3 traffic at ONUs/BSs. The normalized fraction of dropped requests of a class in an hour is calculated as the ratio of blocked requests in an hour and the maximum requests generated per hour during a day. In Fig. 10, we can see that AI-enabled dynamic management of optical resources decreases the fraction of dropped connections during the peak hours of both traffic classes by assigning proportional bandwidth required for the traffic classes, thus effectively utilizing the statistical multiplexing gain. More importantly, the gain for one class (FTTH) could come at the cost of increasing the fraction of dropped connections for other class (cellular) during a short-interval in a day, and it depends on the accuracy of predicted traffic.

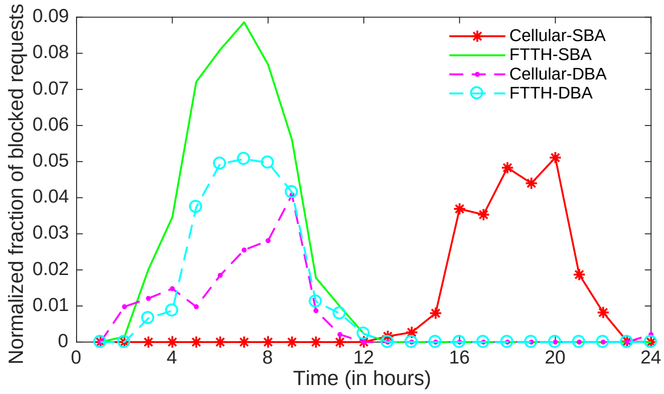

Figure 10. Fraction of blocked requests of cellular and FTTH traffic.

6. Conclusion

In this paper, we proposed a novel early warning system framework to improve animal-human cohabitation. The proposed system is fiber-wireless-based and could detect wild animals nearby road/rail and transfers sensed data to base stations in order to generate alert messages for passing vehicles or humans, if necessary. We compared three scenarios of processing data at sensor nodes, BSs and a hybrid case of processing sensed data at either sensors or at BSs depending on the congestion in WSN, and showed that dynamic allocation of bandwidth in access networks and processing data at its origin could eventually lead to lowering the congestion of network traffic at base stations and, most importantly, reducing the average end-to-end delay. Furthermore, we showed that the energy consumption can be reduced by processing data at the sensor nodes. We believe that this kind of system can be integrated into ongoing smart city and smart transportation infrastructure at no significant additional cost, while improving quality of life for humans and animals.

Author Contributions: This work has been carried out in collaboration between all authors. S. K. Singh conceived and designed the idea, the architecture and the experiments; S. K. Singh and F. Carpio developed the experimental setup and simulation framework for the early warning system; A. Jukan supervised the overall development; and S. K. Singh and F. Carpio with the supervision and coordination of A. Jukan wrote the paper.

Funding: This research received no external funding.

Acknowledgments: The authors would like to thank Mortadha Dhkar for setting up the experiments.

Conflicts of Interest: The authors declare no conflict of interest.

1. Suman, P.; Gupta, P.; Kassey, P.B.; Saxena, N.; Choudhary, Y.; Singh, V.; Radhakrishna, M. Identification of trespasser from the signatures of buried single mode fiber optic sensor cable. India Conference (INDICON), 2015 Annual IEEE. IEEE, 2015, pp. 1–6.
2. http://www.jagdverband.de/content/wildunfallstatistik.
3. Jukan, A.; Masip-Bruin, X.; Amla, N. Smart Computing and Sensing Technologies for Animal Welfare: A Systematic Review. *ACM Computing Surveys (CSUR)* **2017**, *50*, 1–27.
4. Tennakoon, E.; Madusanka, C.; De Zoysa, K.; Keppitiyagama, C.; Iyer, V.; Hewage, K.; Voigt, T. Sensor-based breakage detection for electric fences. Sensors Applications Symposium (SAS), 2015 IEEE. IEEE, 2015, pp. 1–4.
5. Mathur, P.; Nielsen, R.H.; Prasad, N.R.; Prasad, R. Wildlife conservation and rail track monitoring using wireless sensor networks. Wireless Communications, Vehicular Technology, Information Theory and Aerospace & Electronic Systems (VITAE), 2014 4th International Conference on. IEEE, 2014, pp. 1–4.
6. Nakandala, M.; Namasivayam, S.; Chandima, D.; Udawatta, L. Detecting wild elephants via WSN for early warning system. Information and Automation for Sustainability (ICIAfS), 2014 7th International Conference on. IEEE, 2014, pp. 1–6.

7. Viani, F.; Rocca, P.; Lizzi, L.; Rocca, M.; Benedetti, G.; Massa, A. WSN-based early alert system for preventing wildlife-vehicle collisions in Alps regions. Antennas and Propagation in Wireless Communications (APWC), 2011 IEEE-APS Topical Conference on. IEEE, 2011, pp. 106–109.
8. Zhang, J.; Luo, X.; Chen, C.; Liu, Z.; Cao, S. A wildlife monitoring system based on wireless image sensor networks. *Sensors & Transducers* **2014**, *180*, 104.
9. Viani, F.; Polo, A.; Giarola, E.; Robol, F.; Benedetti, G.; Zanetti, S. Performance assessment of a smart road management system for the wireless detection of wildlife road-crossing. Smart Cities Conference (ISC2), 2016 IEEE International. IEEE, 2016, pp. 1–6.
10. Dominguez-Morales, J.P.; Rios-Navarro, A.; Dominguez-Morales, M.; Tapiador-Morales, R.; Gutierrez-Galan, D.; Cascado-Caballero, D.; Jiménez-Fernandez, A.; Linares-Barranco, A. Wireless sensor network for wildlife tracking and behavior classification of animals in Doñana. *IEEE Communications Letters* **2016**, *20*, 2534–2537.
11. Masip-Bruin, X.; Marín-Tordera, E.; Tashakor, G.; Jukan, A.; Ren, G.J. Foggy clouds and cloudy fogs: a real need for coordinated management of fog-to-cloud computing systems. *IEEE Wireless Communications* **2016**, *23*, 120–128.
12. Mata, J.; de Miguel, I.; Durán, R.J.; Merayo, N.; Singh, S.K.; Jukan, A.; Chamania, M. Artificial intelligence (AI) methods in optical networks: A comprehensive survey. *Optical Switching and Networking* **2018**.
13. Ranaweera, C.; Wong, E.; Lim, C.; Nirmalathas, A. Next generation optical-wireless converged network architectures. *IEEE Network* **2012**, *26*.
14. Habel, K.; Koepp, M.; Weide, S.; del Rosal, L.F.; Kottke, C.; Jungnickel, V. 100G OFDM-PON for Converged 5G Networks: From Concept to Realtime Prototype. Optical Fiber Communication Conference. Optical Society of America, 2017, pp. W1K–4.
15. Singh, S.K.; Jukan, A. Machine-Learning-Based Prediction for Resource (Re) allocation in Optical Data Center Networks. *Journal of Optical Communications and Networking* **2018**, *10*, D12–D28.
16. https://www.tensorflow.org/mobile/tflite/.
17. Hochreiter, S.; Schmidhuber, J. Long short-term memory. *Neural computation* **1997**, *9*, 1735–1780.
18. Mahdavinejad, M.S.; Rezvan, M.; Barekatain, M.; Adibi, P.; Barnaghi, P.; Sheth, A.P. Machine learning for Internet of Things data analysis: A survey. *Digital Communications and Networks* **2017**.
19. Riverbed. Riverbed Modeler Academic Edition release 17.5 A PL7, 2014.