Challenges of Designing Smart Lighting

Manoël Dahan, Abdoul Aziz Mbacké, Oana Iova, Hervé Rivano
Université de Lyon, INSA Lyon, Inria, CITI
{manoel.dahan, adboul-aziz.mbacke, oana.iova, herve.rivano}@insa-lyon.fr

Abstract

Smart lighting is one of the main applications enabling the Smart Cities of today. Existing real life deployments have clearly shown that smart lighting drastically decreases the energy consumption of cities. However, with the plethora of solutions existing on the market today, it is difficult to find the one that fits the specific needs of each community. Our system can asses the gains of using a smart lighting system, and can use the collected data to give specific recommendations for the solution that should be implemented. We present here the main challenges and problems that we encountered during the design of our prototype and system, and we discuss some lessons learned.

1 Introduction

In the midst of all the possible use cases and applications for Smart Cities, one main concern remains energy efficiency, as cities account for 2/3 of world energy needs, where just city lighting is responsible for almost 20% of these needs [3]. Using a proper smart lighting solution can help reduce crime and antisocial behavior, revitalize abandoned areas, and improve pedestrian and driver safety. Smart lighting can also reduce light pollution, which induces biodiversity loss and impacts circadian rhythms. The scattering of artificial light at night in the atmosphere currently affects up to 83% of the population and prevents citizens from getting a clear view of the night sky [7].

It has been shown that the use of LED lights instead of common bulbs, combined with the integration of an intelligent management system can cut urban lighting budget by up to 30% without any inconvenience [5]. However, we identify several challenges for the successful deployment of smart urban lighting systems.

Hardware. Access to more and accurate information about the environment offers better opportunities for adaptability at runtime. Depending on the origin of the motion (animal, pedestrian, bike, car, etc.), of its characteristics (e.g., direction of movement, speed), and on the weather conditions (e.g., clear night, presence of fog, rainy day), the luminosity level should be set accordingly to provide better visibility. For example, to distinguish between pedestrians and vehicles, several technologies can be used: radar [11], video [6], or infrared [10]. They all held different tradeoffs between accuracy, cost, and energy consumption.

Software and communication. The communication of the data recovered from the sensors needs to be reliable, and properly analyzed and understood to produce a fast and accurate response. Inappropriate luminosity levels can result in disastrous outcomes. Regarding communication interfaces, several technologies are available: WiFi, GSM [12], ZigBee [14], Power-Line Communication (PLC), Low Power Wide Area Networks (LPWAN) [12], with different advantages and weaknesses.

Economical. For a smart lighting solution to be economically viable, the investment needs to be recovered in a short period. Both the capital expenses (hardware, software, manpower to produce and deploy the solution) and operational expenses (cost of maintaining the hardware and keeping a reliable service) need to be kept as low as possible.

In this paper we present the prototype that we developed for detecting and predicting urban mobility, which we tested in real conditions. Data collected from such a deployment can then be input into an economical model to compute the gains in energy consumption that a smart lighting solution would offer, and how long it will take to get a return on investment. Moreover, we can give recommendations for the characteristics and functionality that a future smart lighting system needs to have, in function of the type and amount of traffic in a specific area. For example, in an area with large traffic at night, it is not useful to invest into a complex detection system, as the lights will be on most of the time, in which case the investment should rather focus on the quality of the LEDs and a management of light color temperatures.

The main contributions of this paper are:

1. We design a prototype for motion detection and prediction, which takes into account weather conditions.
2. We discuss lessons learned (regarding both hardware and software) from the design of the prototype and the deployment of the system in real conditions.

International Conference on Embedded Wireless Systems and Networks (EWSN) 2020
17–19 February, Lyon, France
ISBN: 978-0-9949886-4-5

3. We offer pointers towards improving the lifetime of the deployment through both hardware and software modifications.

2 Prototyping a Smart Lighting Device

As our goal is to create an autonomous and easily deployable system that can asses the energy gains of deploying a smart lighting system, we needed to create a device that has all the functionalities of a smart lighting device, but which does not need to actually control the light. Consequently, our prototype has the following characteristics:

- Sensors can detect passing-by pedestrians / vehicles, and measure the surrounding noise and weather conditions. The collected data can afterwards be used to refine the accuracy of lighting, and even predict which lights to switch on in function of the traffic conditions.
- Data from sensors can be retrieved both offline (using an SD card) and online (using a wireless communication) at different granularity.
- The prototype is battery powered, in order to be easily deployable, and not depend on cables or bureaucracy (e.g., for asking access to the power grid of the city).
- We used energy efficient electronic components to have a reasonable autonomy, so that we can collect a significant data set.
- The prototype is designed to be easily maintained. All the electronic components are connected via pin headers and can be accessed without much effort.
- The casing of the prototype is weather proof and was designed to allow quick access to the main components: battery, SD card, and main board.

The remaining of this section presents into details the hardware and software choices that we made.

2.1 Hardware

We chose to build our prototype based on a low energy consumption micro-controller that has a sleep mode, enabling important energy savings. The Arduino MKRWAN 1300 board was our first pick, as it is powerful and energy efficient at the same time (see Table 1). Indeed, it integrates both an ARM Cortex-M0+ 32bit low power micro-controller and a Murata board, which in turn incorporates a Semtech SX1276 chip for LoRa wireless connectivity. Furthermore, the MKRWAN board is also very compact, while being able to handle all the sensors needed by our smart lighting device.

As it can also be seen in Fig. 1, we chose the following electronic components to go with our Arduino board:

- *DFRobot SEN0018 PIR* sensor for motion detection. The sensor can detect movement up to 7 meters, and has a detection angle of 110 degrees. Its output is a binary response when a thermal motion is detected.
- *SparkFun SEN-12642* sound sensor that captures the surrounding sound and returns an audio recording, a binary value, or the amplitude of the signal (in our case).
- *Adafruit BME280* environmental sensor that measures temperature, relative humidity, and barometric pressure.

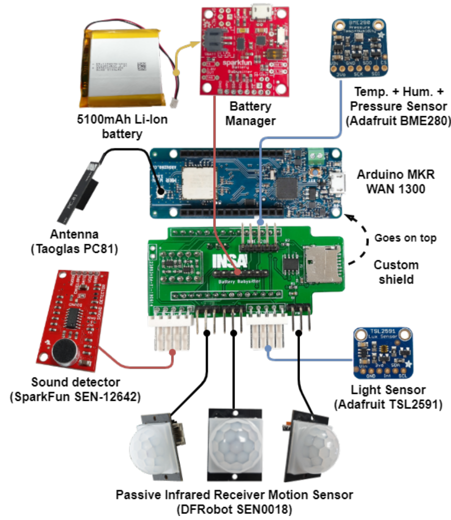

Figure 1: Overview of the smart lighting prototype

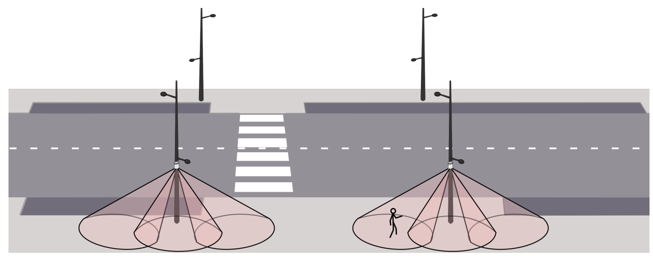

Figure 2: Smart lighting detection system using PIR sensors

- *Adafruit TSL2591* light sensor that can detect luminosity from 188 µlx up to 88, 000 lx, with an infrared and a full spectrum diodes.
- *Taoglas PC81 868MHz ISM Mini PCB* omnidirectional antenna used for the wireless communication.

Motion Detection. There are several technologies that can be used for motion detection (e.g., infrared, laser, cameras, etc.) with different tradeoffs between precision of detection, energy consumption, size, and cost. As Light Detection And Ranging technology (LIDAR) does not correspond to the requirements of low energy consumption and wide detection area, and video cameras are too power hungry, we chose to use a sensor based on Passive Infrared Receiver (PIR) technology. In order to be able to predict the direction in which pedestrians / vehicles are headed, we decided to use 3 PIR sensors oriented at different angles. This also enables us to cover a larger area (see Fig. 2), and better detect the direction when the density of pedestrians increases.

Environmental Data. The environmental sensor has three roles. First, its data can be used to increase the intensity of light during bad weather. Second, it is useful to analyze the wireless network behavior, especially w.r.t. battery discharge phenomena [4], and weather conditions [1]. Third, environmental data is of main interest for smart cities, and can be used in related projects such as environmental modelling [2] and urban heat island studies [13].

Table 1: Power consumption of the electronic components

Electrical component	Type	Active mode		Sleep Mode	
		Measured value	Theoretical value	Measured value	Theoretical value
MKRWAN 1300	Main board	16.6 mA without LED up to 30.6 mA with LED	/	1.15 mA	/
PRT-13777	Battery manager	1 mA	/	No sleep mode	
PIR	Motion Detection	85 µA when detecting 34 µA when output at 0	50 µA	No sleep mode	
BME280	Temp, humidity, atmospheric pressure	470 µA	3.6 µA @ 1Hz	6.5 µA	0.1 µA
TSL2591	Light	270 µA	275 µA	9 µA	2.3 µA
SEN-12642	Sound	855 µA without LED up to 2.24 mA with LED	320 µA	No sleep mode	

Power Supply. To power up all the electronic components, we used a 5100mAh / 3.8 V Li-ion battery. However, the Arduino board is powered through its VIN port, which requires 5V input, and some sensors have to be supplied with 5 V (the PIRs and the sound sensor), while others with 3.3 V (the environment and light sensors). Hence, we used a battery manager board coupled with a 5 V regulator (the Polulu S7V7F5) and a shield extension that we developed in-house to redirect the proper voltage to each electronic component[1].

2.2 Casing

The casing of any sensor device is very important, as it has to protect all the electronic components, without impeding their proper functioning. It has been repeatedly shown that the choice of the casing can make or break a deployment [8, 9]. More specifically, in our case, we have to account for the following factors:

- Different weather conditions (rain, wind, UV radiation, etc.) can trigger short circuit and destroy the electronic components, so the casing has to be weatherproof.
- Humidity can accumulate inside the casing and destroy the electronic components, so the casing needs to allow the air to circulate inside.
- In a wireless environment, the material of the casing may interfere with the signal and cause attenuation, so it has to be carefully chosen.

Besides these factors, our casing needed to be tailored on the specific sensors that we use. The BME280 environmental sensor needs a proper ventilation, and the PIR motion sensors have to be placed at a certain angle on the horizontal and vertical planes based on the detection cone and the height at which the device will be attached.

Alltogether, it was basically impossible to use an off-the-shelf casing, so we decided to design and 3D print our own casing. We used Fusion-360[2], a 3D modeling software to design the casing. We then 3D printed the model with Nylon agglomerated powder in white (to reduce the accumulation of heat), with a printing resolution of 16 µm. The casing is composed of three main parts:

- The bottom part, which contains the three PIR motion sensors, the sound sensor, and the luminosity sensor. Besides the motion sensors, all the other sensors are situated at the interior of the casing. An aperture in the bottom of the casing allows for the light sensor to properly estimate the outside luminosity.
- The middle part, which contains the micro-controller, the battery manager board, the extension shield, and the battery. This part has a trap on the side that allows quick access to the main board, the SD card, and the battery.
- The top part, which is the environmental measure chamber that contains the BME280 sensor.

Figure 3: Fully assembled sensor and its casing

The assembled sensor (see Fig. 3) has a size of 180x115x90mm, and is completely weatherproof, while allowing airflow to circulate from the aperture in the bottom part to the environmental measure chamber, through the middle part, avoiding this way the accumulation of moisture. The total cost of one device is apx. 500€ (apx. 250€ for the electronics and 250€ for the casing).

2.3 Software

As the goal of our system is to switch on / off the lights on the street when there is not enough environmental light, our device has two different behaviours depending on the outside luminosity. Indeed, we use the data from the light sensor to detect day / night cycles. From here on, "night" indicates the part of the day when there is not enough environmental light and the street lights need to be switched on.

Collecting Data. During the day, only the light and environment sensors are activated, taking measures every 30 minutes. The rest of the time, they enter into a sleep mode where energy consumption is minimal (see Table 1). This also al-

[1] The SparkFun battery manager output can vary from 3 to 5.5V depending on the state of the battery, this is why we use a 5 V step up/step down voltage regulator.

[2] `https://www.autodesk.com/products/fusion-360/`

lows the MKRWAN board to use a low power mode, which greatly improves the autonomy of our device.

During the night, the light, environment, and sound sensors are activated, taking measures every minute. The sleep mode is seriously reduced, increasing energy consumption. Moreover, the PIR sensors are configured to always sense the environment and wake up the micro-controller from its sleep state through an interruption if a movement is detected. If no other movement is detected in the next seconds, the micro-controller returns to sleep mode.

Retrieving Data. Data from the sensors is sent to an application server using the LoRaWAN protocol every 10 minutes during the night. More specifically, we send the averaged values for the temperature, relative humidity, atmospheric pressure, sound level, and luminosity, as well as the battery level and the number of movements detected by each PIR sensor. We use a confirmed uplink transmission in which a device will retry sending the data up to 3 times if no acknowledgment is received, before giving up. Communication is disabled during daytime to reduce energy consumption.

As LoRaWAN works in the 868MHz band, it is submitted to the 1% duty cycle restriction, which limits the amount of data that can be transmitted. To overcome this constraint, all sensor data is also saved locally on a micro SD card. Since we store all the measured values, not just the average, this also gives us a better temporal granularity of the data for post-deployment analysis. Data is saved both during daytime (every 30 minutes, when the measure is taken) and during the night (every 10 minutes, with the LoRaWAN transmission).

3 Deployment

In order to validate the proper functioning of our system, we did a test deployment in real conditions. Although the hardware, the casing, and the software were each tested and validated individually, it is of uttermost importance to validate the system as a whole. This section presents the environment where we did the deployment, and some preliminary results regarding the autonomy of the prototype.

3.1 Setup

The location of our test deployment is on the street of an university campus, where several university buildings, student residences, and a restaurant are present. This should have enough pedestrian traffic at night to test our system.

The deployment took place on November 2019 and was composed of 5 prototype nodes deployed in a straight line on light poles on the same side of the street, at a height of 3 meters. This height has been determined based on the maximum distance that the PIR motion sensor can detect movement, and takes into consideration the human factor in order to avoid theft and degradation. This results in a detection area of around 5 meters on each side of the light pole. The street lights on our test street are situated at a distance of 25 meters from each other, which means that there is a "blind" detection area between two poles. While this is not ideal, the use of inexpensive and energy-efficient sensors is a compromise in order to have a large deployment at a reduced cost. Indeed, create a proficient algorithm to predict the direction of movement and provide a better data analysis is part of our future work.

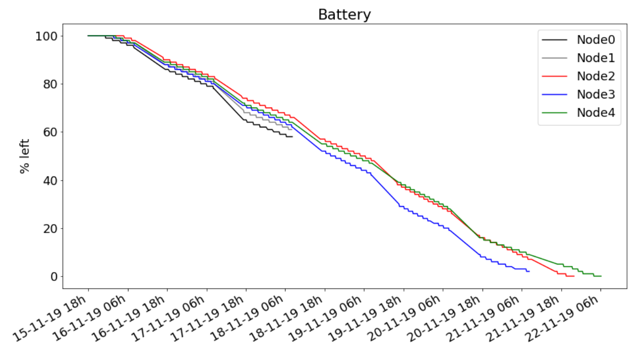

Figure 4: Battery depletion of the deployed prototypes

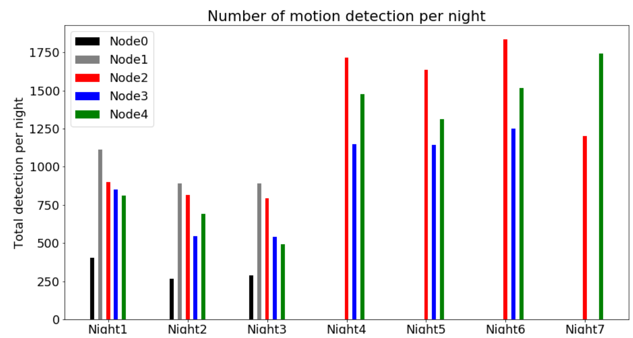

Figure 5: Motion detected during the deployment

We used the existing LoRaWAN network that covers the campus and that is part of The Thing Network (TTN) platform to recover the data from our sensors. The TTN platform allows us to decode the data received, send it to a local database over the MQTT protocol, and manage our sensors.

3.2 Prototype Autonomy

We focus here on the results[3] concerning the lifetime of the deployment, which lasted seven days, as it can be seen in Fig. 4. First, we notice that two out of the five nodes stopped transmitting during the third night of deployment, a problem we already encountered in the past. Second, we notice that the batteries did not deplete at the same time, a delay of 12 hours can be observed between Node3 and Node4. This is explained by the relative position of each node in the street and the environmental impact this has. Indeed, a node that detects more movement and sound, is more power hungry than a more secluded one. For example, Node2 was located over a bike parking area where noise pollution was very abundant impacting its battery life. This was confirmed also by the data recovered from the sound sensors.

We can also notice that the energy depiction is more steep after the third night. If we look at Fig. 5, which shows the number of motion detected per night and per node, it is clear that this is mainly due to increase traffic. This is explained by the fact that the nodes were deployed on a Friday evening, when most of the students are off campus and activity is re-

[3]The results presented in this section are obtained from the data sent using the LoRaWAN communication.

duced. Starting on the forth night, activity quickly increases during Monday to Tuesday night and remains stable until the end of the week.

The energy consumption seems to be slightly increased during the day, while the main board and part of the sensors should spend most of the time in sleep mode. However, the PIR and the sound sensors do not have a sleep mode, so they continue to detect motion and sound, hence activating more often the main board. Indeed, it is not possible to completely disable them using this hardware. This aspect will be further discussed in the next section.

4 Problems encountered

During the development of the sensor and its deployment, we faced several problems and challenges. In this section, we make a detailed description of these difficulties, and we discuss the solutions that we adopted.

4.1 Design of the Casing

The design of the casing was a main challenge in this project because of the specific needs of each sensor, and of the outdoor environment where they would be used. We did several iterations both in the design as well as in the 3D printing itself. We passed from 50 hours of printing for each casing using Fused Deposition Modeling technique, to just a few hours using Nylon agglomerated powder.

Weatherproof. Being deployed in an outdoor environment, the casing needs to be weatherproof, so we tested it in the laboratory, by reproducing heavy rain within a shower. The test consisted of positioning the casing on a small pole and sprinkle water above the casing at different flows and angles (from 0 to 15 degrees, corresponding to an Ingress Protection level of 2 for water). After just a few tests, it appeared that water could get in the casing at the junction between any two parts, and in the environmental measure chamber. Therefore we beveled the top flat surfaces, reduced the interstices of the measure chamber's layers, increased the contact surface between the junction points, and finally increased the top cover layer of the measure chamber to better protect the node.

Ease of assembly. Because we wanted the prototype to be compact, but with easy access to the important parts, we had to design a modular casing. However, because it is composed of multiple pieces and screws, the first casing took a large amount of time to assemble. In consequence, we needed to find the right balance between reducing the number of screws and adding a few mounting clips while keeping a good solidity.

Location of the sensors. In the first iteration of the prototype, we decided to put the light sensor in the measurement chamber, to get a better sense of the outside luminosity. However, the first tests showed that not enough light was passing between the venting layers, and therefore we were not able to properly detect the night and day cycles. We had to change its position, so that it measures correctly, while still keeping it protected from the weather. The only viable solution was to move the light sensor in the bottom part of the device. We consequently needed to modify the design of this part by adding an aperture that would permit enough light to reach the sensor, while keeping possible water splashes away. The secret is to find the correct inclination of the sides of the aperture to get the maximum of light and prevent capillary effect of water drops.

Casing color. Another important rework on the early design was the color of the nodes, which were initially printed in a dark gray color. Based on interactions that we had with a researcher that studies air pollution, we found out that the measurement chamber has to be white near the environmental sensors (temperature, humidity), because the heat accumulated by the gray plastic can distort measurements. Consequently, we printed the final casing in white.

4.2 Electrical Components

Using jumpers to connect the sensors to the microcontroller is useful in the first steps of prototyping but a long deployment can lead to their disconnection or oxidation. Another option was to solder the connectors on an Arduino MKR MEM shield[4] and make wire connections between them and the corresponding MKRWAN pins. However, soldering cable is very time consuming (it can take up to 8 hours to finish one shield), so we decided to create a replica of the MKR MEM shield with the connections already made. Moreover, this allowed us to evolve the design easier, place the connectors to optimal positions, and make the electrical circuit more compact.

The design of the electrical extension board needed several iterations, as we encountered multiple problems, which were detected over different periods of time. The first problem we encountered was a trace that went around the micro SD card reader, making the board one millimetre longer and not allowing it to get fixated in its slot inside the casing. Another problem was a trace going to the wrong pin of the MKRWAN that we had to reroute. Finally we had to change the voltage input of the motion sensors because although the data sheet mentions a voltage input range, we experienced inconsistency in motion detection with the 3.3 voltage input. So we had to change their input voltage and power them with 5 volts instead.

4.3 Autonomy

As we chose third party components for our electrical circuit, we were not able to fully control the consumption of some sensors, which had unused functionalities that still consumed energy. As it can be seen in Table 1, the data sheet values of most of the electronic components is greatly underestimated. For example, for the sound and environmental sensors, the difference between the measured values and the ones specified in the data sheet is one order of magnitude.

Once we had our first prototype and we started testing it, we realized that its autonomy was barely a day, so we had to investigate different energy saving solutions. The main components that consume energy are the Arduino MKRWAN board, the 5 volt regulator and the power manager board. If we wanted to save energy, we needed to keep the main board (and the components that would allow it) in sleep mode as much as possible. In consequence, we designed the measuring periods in the day/night cycles with this in mind.

Finally, we realized that some of the electronic components were using LEDs, and removing or disabling them al-

[4]https://store.arduino.cc/mkr-mem-shield

most doubled the autonomy of the prototype, reaching to 7 days. We now consider that this is the maximum that we can reach with this hardware, as the motion detection sensor have to be constantly powered, and hardware restriction does not allow the main board to enter a deep sleep mode while waiting for motion detection interruptions.

5 Lessons learned

Throughout the journey of prototyping a sensor, it is usual to come over a few challenges that were not foreseen in the early development. It points out how important it is to elaborate a strong specification at the beginning of the project.

For example, in our case, in what concerns the casing, we think that with more understanding and safety margin, we could have spent less time in designing its weatherproof capability. For short deployment projects, designing a complex casing can be avoided by using third party casings or by using less removable parts with simpler design, but it is a more complex task for long-term projects like this one.

Another key point is that building a battery-powered device with sensors that work with interruptions is more challenging than it seems, because they have to be constantly powered and cannot be in a deep sleep mode. If some electrical components are not used during daytime (for example our PIR and sound sensors), a possible solution that would bring a great improvement is to switch them off with a transistor, and make sure to respect their heating time.

We noticed during the deployment that some devices would stop transmitting through their LoRa chip, but we have yet to identify the cause of this behavior (battery issue, communication problem or vandalism), as debugging such a behaviour is not at all straightforward, and is difficult to reproduce. We think that it is essential to have a backup solution to restart sensors remotely, especially if the deployment is for a long period of time, or at least have a detailed log to debug post deployment (which is more energy consuming). As our solution has a strong SD logging system, data can still be retrieved offline in case of connectivity issues.

Overall, we can extract the following engineering insights from our experience:

- *Start simple*: one should start with a simple yet modular prototype than can afterwards be easily upgraded;
- *Keep in mind the environment*: the targeted deployment environment influences the design of the casing and a particular attention should be given to expected conditions (rain, sun and wind can impact the shape, color, and positioning of sensors);
- *Test before commit*: one should test several components of the same type before choosing the final one, in order to ensure that the most appropriate ones are chosen, as they will not only impact the data that is retrieved but also the autonomy of the overall device.

6 Conclusions

We proposed here an autonomous and easily deployable smart lighting system that can asses the gains of using a smart lighting system through data collection and provide a forecast of specific recommendations for the solution that will be implemented. For example, the results from our deployment show that energy efficiency could be greatly improved by dimming lights during weekend due to reduced activity, which might be counter-intuitive at first.

We also presented the challenges that we faced in creating a smart lighting prototype, both software and hardware related. We looked back at our design process and tried to offer some pointers for future prototyping projects.

Future work will be focused on combining all the data collected from the sensors into an algorithm for the detection of the origin of the motion (animal, pedestrian, bike, car, etc.) and for the prediction of lighting based on the identified motion characteristics (direction of movement, speed).

7 Acknowledgments

We acknowledge support of this work by the ESMARTCITY project (3MED171.1M2022), which is implemented under the Interreg Mediterranean programme, co-financed by the European Regional Development Fund (ERDF) and the Instrument for Pre-Accession Assistance (IPA).

8 References

[1] C. A. Boano, M. Cattani, and K. Romer. Impact of Temperature Variations on the Reliability of LoRa - An Experimental Evaluation. In *Int. Conf. on Sensor Networks*, pages 39–50, 2018.

[2] A. Boubrima, A. Boukerche, W. Bechkit, and H. Rivano. WSN Scheduling for Energy-Efficient Correction of Environmental Modelling. In *IEEE Int. Conf. on Mobile Ad Hoc and Sensor Systems*, pages 380–387, 2018.

[3] M. Castro, A. J. Jara, and A. F. Skarmeta. Smart Lighting Solutions for Smart Cities. In *Int. Conf. on Advanced Information Networking and Applications Workshops*, pages 1374–1379. IEEE, 2013.

[4] M. Ceriotti and et al. Is There Light at the Ends of the Tunnel? Wireless Sensor Networks for Adaptive Lighting in Road Tunnels. In *ACM/IEEE Int. Conf. on Information Processing in Sensor Networks*, pages 187–198, 2011.

[5] R. Elberg and E. Woods. Smart street lighting as a smart cityt platform. Technical report, Navigant, 2017.

[6] G. Gagliardi et al. A Smart City Adaptive Lighting System. In *IEEE Int. Conf. on Fog and Mobile Edge Computing*, pages 258–263, 2018.

[7] M. Gandy. Negative luminescence. *Annals of the American Association of Geographers*, 107(5):1090–1107, 2017.

[8] R. Hartung, U. Kulau, B. Gernert, S. Rottmann, and L. Wolf. On the Experiences with Testbeds and Applications in Precision Farming. In *ACM Int. Workshop on the Engineering of Reliable, Robust, and Secure Embedded Wireless Sensing Systems*, pages 54–61, 2017.

[9] G. Jackson, S. Gallacher, D. Wilson, and J. A. McCann. Tales from the Wild: Lessons Learned from Creating a Living Lab. In *ACM Int. Workshop on the Engineering of Reliable, Robust, and Secure Embedded Wireless Sensing Systems*, pages 62–68, 2017.

[10] E. Juntunen, E.-M. Sarjanoja, J. Eskeli, H. Pihlajaniemi, and T. Österlund. Smart and Dynamic Route Lighting Control Based on Movement Tracking. *Building and Environment*, 142:472–483, 2018.

[11] M. Kuusik, T. Varjas, and A. Rosin. Case Study of Smart City Lighting System with Motion Detector and Remote Control. In *IEEE International Energy Conference*, pages 1–5, 2016.

[12] LeapCraft. CPH Sense - Ambient Sensing for Smart Cities, 2019.

[13] C. L. Muller, L. Chapman, C. S. B. Grimmond, D. T. Young, and X. Cai. Sensors and the City: A Review of Urban Meteorological Networks. *Int. Journal of Climatology*, 33(7):1585–1600, 2013.

[14] N. Yoshiura, Y. Fujii, and N. Ohta. Smart Street Light System Looking Like Usual Street Lights based on Sensor Networks. In *Int. Symp. on Comm. and Information Technologies, IEEE*, pages 633–637, 2013.