3rd IEEE International Conference on Industrial Cyber-Physical Systems
(IEEE-ICPS) 10-12 Jun,2020

Towards SWARM: A Smart Water Monitoring System

Julian Kunze*, Vincent Mayer†, Lisa Thiergart*, Saqib Javed†,
Patrick Scheppe‡, Tam Tran*, Markus Haug§, Mariana Avezum*
* Department of Informatics, Technical University of Munich
† Department of Electrical Engineering and Information Technology, Technical University of Munich
‡ Department of Mechanical Engineering, University of Stuttgart
§ Department of Informatics, University of Stuttgart

E-Mail: julian.kunze@tum.de, vincent.mayer@tum.de, lisa.thiergart@tum.de, saqib7.javed@tum.de,
patrick@scheppe.info, tam.tran@tum.de, st143248@stud.uni-stuttgart.de, m.avezum@tum.de

***Abstract*—We introduce our Smart Water Monitoring System (SWARM), an interconnected system centered around unmanned aerial vehicles (UAVs). It is designed to gather environmental data in hard-to-reach areas in order to facilitate water quality management for surrounding communities. We focus specifically on the village of Ganvi on Lake Nokou in Africa. The lake suffers from severe pollution posing a threat to the ecosystem in and around the lake. The SWARM system consists of multiple drones with different purposes, including water sampling and aerial data collection. It utilizes a fog architecture to combine the advantages of centralized computing and storage with those of local data processing. The fog architecture is key to address connection issues around Lake Nokou. SWARM reduces the effort for researchers to effectively monitor and investigate water pollution issues by partially automating the process.**

***Index Terms*—Water Quality Monitoring, Pollution, UAV, Internet of Things, Fog Computing**

I. Introduction

A. Problem Statement

In past years, the village of Ganvi on Lake Nokou in Benin, West Africa has been increasingly struggling with water pollution problems. The lake's sediment and fish population have accumulated dangerous concentrations of lead and cadmium [1]. As the Lake Nokou region relies largely on fishing and tourism, fish death and decreasing water quality significantly affects the local economy and livelihood. Neighboring lakes suffer from similar problems, yet the cause remains unclear. Further research and monitoring of the water pollution state are required to "reconcile environmental protection and sustainable human development" [1].

B. Development Process

In September 2019, three professors, two doctoral candidates, and 23 students with diverse engineering and science backgrounds gathered in Sarntal, Italy for two weeks to work on a smart monitoring system. This summer school project called Ferienakademie[1] aimed to facilitate and simplify the research process currently ongoing in Benin, by enabling the researchers to more easily collect and analyze water samples from different regions of the lake. The project structure using the interdisciplinary system courses (ISC) teaching approach has previously been successfully applied in several similar projects involving customers requiring solutions from the aerospace industry [2].

C. Target and Development Environment

During the Ferienakademie project, Lake Durnholz in Italy served as the development and testing environment. Lake Durnholz spans a size of 0.12 km^2 with its maximal length being 0.9 km. The lake is part of a recreational area. It is low in nutrients and therefore not inhabited by many fish. The strong in- and outflow of water generate a high water quality[2].

The target environment Lake Nokou is situated in the South of Benin close to the Atlantic Ocean. It is approximately 20 km wide and 11 km long covering a total area of 49 km^2. The village of Ganvi is located in the northwestern part of the lake. The population of Ganvi mainly lives directly on the lake, therefore the major part of housing and economic activities are situated on the lake. The average maximum temperature over the year lies in the range between 28 to 32C [3]. Regarding its water quality situation, several different pollutants have been detected including phosphates, nitrate, sulfate, chloride and ammonium, as well as other chemical substances resulting from the dumping of organic matter, agricultural and industrial waste [1], [4].

D. Sustainability Constraints

Assessing core indicators pertaining to the sixth United Nations Sustainable Development Goal (SDG) "Clean Water and Sanitation" [5] can require a complex urban infrastructure for water monitoring. Currently, measuring indicators of water quality, as defined in the UN ISO-standards developed based on SDG 6, poses significant challenges at Lake Nokou. The application of these indicator definitions to lakes is challenging because they were proposed with cities in mind. Unlike in cities where green, grey and black water are separated, a lake

[1]https://www.ferienakademie.de/en/home-2/

[2]https://umwelt.provinz.bz.it/wasser/zustand-suedtiroler-seen.asp?news_action=4&news_article_id=626319

allows the spreading of potential water pollutants throughout different regions of the lake. For example, the stated indicators include the "percentage of the city population with potable water supply service" and the "percentage of population with access to improved sanitation" [6]. The second indicator is defined as the number of households with a pipe-based or similar connection to a public or communal grey and black water disposal system, which relays the wastewater to a water treatment facility, multiplied with the average number of people in such a household and divided by the city's total population [6]. Since the lake village does not contain a pipe-based sewerage system like a land-based city, measuring and managing this indicator in a representative way is challenging in Ganvi and other measurements have to be taken into consideration.

Furthermore, a lack of data due to limited resources complicates the investigation of water pollutant sources. Identification of pollution sources would simplify the monitoring of indicators and by this help to improve the water quality. This is what our system aims to do.

E. Paper Outline

In our foundations, essential terms and concepts for the context of the project are defined with respect to related and existing work. Subsequently, the requirement elicitation examines the environment context and from this derives functional and non-functional requirements. The resulting architecture and implementation of our system are outlined in the following sections. Finally, we evaluate our system and explore how well the requirements were fulfilled.

II. Foundations

This section provides definitions and gives an overview of related work. In order to be able to mitigate the effects of water pollution and to increase water quality, we first define water quality and water pollution, show how suitable water quality indicators can be derived from this, and lastly compare different modalities for collecting water samples. Subsequently, we illustrate a common architecture for cloud applications and highlight a few shortcomings of this architecture as they relate to our problem. We then describe an evolution of the common cloud architecture that addresses those shortcomings.

A. Water Quality Monitoring

Different sources vary in their exact specifications of water quality and there is no universal definition of the term. Boyd [7] describes it as all properties of water that have an impact on human water usage or influence natural ecological systems in general. These variables can either be biological, chemical or physical ones. In contrast, the work of Bartram and Ballance [8] defines water quality as a property of water that enables it to preserve various uses or processes, e.g. only the absence of toxic substances permits potability. Water pollution as opposed to quality directly deals with substances that "interfere with beneficial use of the water or with the natural functioning of ecosystems" [9] which can be both anthropogenic (e.g. domestic sewage, toxic waste) or natural (e.g. erosion).

Depending on the purpose that a body of water serves, different indicators are used to assess water quality and to identify pollutants. The critical values for each evaluated variable also differ widely according to water use. Therefore, in order to reduce the consequences of water pollution, one first has to identify the appropriate indicators [8]. Typically, the variables that are measured in water quality assessments are a combination of physical (e.g. temperature, pressure), chemical (e.g. pH value, hardness, dissolved substances) and biological (e.g. microorganisms) characteristics [7]. Section V-A deals with indicators that our system evaluates.

Another important aspect of water quality monitoring is the modality of sample collection. Besides taking samples manually, there exist multiple ways of automating the process: smart buoys can be distributed over the water body to track changes of parameters remotely and continuously [10]. However, they are limited to collecting data at fixed positions. In contrast, autonomous boats [11], [12] or submarines [13], [14] can be deployed flexibly to differing sites depending on where measurements are required. Also unmanned aerial vehicles (UAVs, in the following also referred to as *drones*) can be employed in the process of water quality monitoring [15]–[17]. They have an advantage over marine vehicles as they do not require the water body to be traversable itself but only demand that the landing location is unobstructed. Also, they are capable of collecting aerial data that can be used to determine points of interest and to supervise other vehicles. UAVs, however, are likely to complicate autonomy of the system as they require permanent compensation of gravity force to not crash. For our system, we decide to use UAVs despite their potential impediments to offer a solution as flexible and versatile as possible.

B. Fog Architecture

Typically, cloud applications use a layered architecture, consisting of at least two layers. The cloud layer provides centralized computing and storage services [18]. Client devices, such as smartphones, are located in the edge layer. Recent developments, such as the widespread deployment of devices in the Internet of Things (IoT), impose new requirements on applications. Among these requirements are low latency and mobility support, both of which are problems for the traditional cloud computing architecture [18].

Fog computing provides a new platform that is designed to meet the additional requirements of IoT and cyber-physical systems [18], [19]. To do this, a new layer, the fog layer, is introduced between the cloud layer and the client devices in the edge layer [20]. The fog layer extends the services provided by the cloud to the edge of the network by using geographically distributed fog nodes. This distribution reduces latency and increases location awareness [18].

Figure 1 shows a typical fog architecture. The cloud provides almost infinitely scalable storage and computation capabilities. The fog layer, in turn, acts as a proxy providing

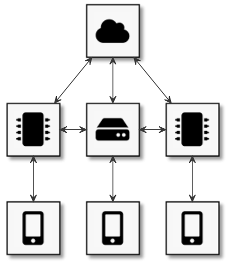

Fig. 1: Fog Architecture (adapted from [20])

additional computation and storage capabilities, but with a much more limited scope. A fog node only handles data and computation for clients in its proximity, which improves latency for the clients [20]. Additionally, the fog layer can provide fault tolerance in scenarios where the connection to the cloud layer is unreliable.

Use cases of fog computing include connected vehicles, smart grids, wireless sensor and actuator networks [18]. Additionally, Mohamed et al. deployed drones to use fog computing in areas where a reliable connection to the cloud is not possible [21].

III. Requirement Elicitation

This section first introduces the key differences regarding target and development environment as introduced in I-C. The comparison serves as a foundation for the following introduction of functional and especially non-functional requirements for the system.

A. Development Environment vs. Target Environment

Firstly, Lake Nokou covers more than 400 times the size of Lake Durnholz. A system able to cover Lake Durnholz could potentially be insufficient for Lake Nokou. Secondly, Lake Durnholz had very few static and no dynamic obstacles at testing time. In and around Ganvi, such would not be the case. The lake has many static objects, such as the homes of local people. Due to economic activities such as fishing or transportation, many dynamic obstacles including boats, fishing nets or creels would also be encountered. Thirdly, at Lake Durnholz the system was operated by the people engaged in the development process. When the system is transferred to Benin, it should be utilized and maintained by local authorities (researchers). Lastly, Lake Durnholz is not affected by any pollution, whereas Lake Nokou is strongly affected.

B. Functional Requirements

In order to monitor water quality appropriately, the system needs to be able to take water samples from the lake. During the process of taking a water sample, it measures the water temperature. Afterwards, the water samples need to be analyzed for contaminants and the water quality indicators need to be evaluated by the mobile laboratory. To fulfill the task of water quality measurements, the system needs to be able to take off and land autonomously on both lake and ground. After take-offs, it needs to be able to follow the geo-tagged mission waypoints. The altitude of the drones has to be high enough to avoid collision with objects on lake or ground level and low enough to avoid interference with air traffic. To ensure a collision-free flight, the system needs to scan the landing area and detect both static and dynamic objects such as boats.

Furthermore, the system has to provide aerial data that supports researchers in supervising the lake. This data needs to be sent to and stored in a repository along with relevant mission metadata, such as measurement values. Retrieval of requested portions of data should be possible. From the individual data points, a coherent map of the analyzed area should be created. This is necessary to enable manual and automated selection of possibly interesting areas using the latest data. Further analysis of the data includes pattern recognition techniques to detect changes over time and sources of pollution. They have to be applied to extract useful features from the image data that can then be integrated with the system.

C. Non-functional Requirements

In the following, the system's non-functional requirements are being discussed with a focus on the special target environment in Benin.

- *Environmental impact*: The system should have a minimal impact on the environment. Lake Nokou already suffers from severe pollution and it is, therefore, necessary that the system does not add contamination to the lake.
- *Flight duration*: As previously mentioned, Lake Nokou spans a large area. The flight duration of the individual drones needs to be sufficient to cover this lake size.
- *Extensibility*: The individual drones also need to be extensible in order to integrate more or different sensors, which might be required in the future.
- *Maintainability*: Assembly and disassembly of the individual system parts needs to be easy in order to fix possibly broken parts quickly.
- *Usability*: The system will be used by local authorities in Benin who were not involved in the development process. Hence it must be easy-to-use with a minimal amount of training required.
- *Scalability*: Due to the large size of Lake Nokou, covering the whole lake requires multiple drones. It must therefore be possible to add more drones to the system easily.
- *Security*: The access to the system must be authorized and communication between individual parts must be encrypted to guarantee a secure operation.

IV. System Architecture

We design our system architecture under the assumption that there is no sufficiently reliable internet connection at our target site since we do not have information on the internet coverage there. As a result of potential connectivity issues, we choose to implement the system with a fog architecture. To provide fault tolerance in case of a dropping internet connection our fog nodes must be able to operate without a connection to the cloud. We subdivided the system into several components that operate in different layers. Figure 2 shows the resulting subsystem decomposition.

There are two subsystems in the cloud layer: the *mission repository* and the *image repository*. The mission repository provides an archive of all past and active missions. It also stores flight data and measurements that were collected during those missions. To accomplish these tasks, it provides an interface that allows other subsystems to create and query missions as well as to upload flight data. The image repository is responsible for storing RGB and thermal images collected during missions. To do this, it provides an interface to upload new images and download and existing images, optionally restricted to a specific geographic region. The two repositories were designed as separate subsystems to allow them to be scaled independently of each other.

The fog layer only contains the *mobile proxy* subsystem. This subsystem consists of a *mission repository proxy*, an *image repository proxy*, and the *drone base*. The mission repository proxy and the image repository proxy provide the same services to their clients as the corresponding repository in the cloud layer. Additionally, they synchronize their local data with the cloud repositories. The drone base is designed to interface with all types of drones in the system. It allows the drones to register themselves, update their status, and upload flight data, measurements, and images. The data is then forwarded to the correct repository proxy. Furthermore, flight plans for missions are generated here and assigned to the drones. To provide the location sensitivity that is typical of a fog architecture, all parts of the mobile proxy can be assigned an area of responsibility. The subsystems will only provide services for their assigned area of responsibility.

The edge layer contains four subsystems: the *SWARM app*, the *camera drone*, the *thermal camera drone*, and the *scoop drone*. The SWARM app allows an operator in the field to control the system. It can visualize missions, their flight plans, and the assigned drones, flight data, measurements, and images collected during missions. Additionally, it allows other subsystems to create new missions and to add measurements and images to the repositories. The camera drone subsystem models every drone that has access to an RGB camera. It interfaces with the drone base to register for missions and provide status updates. When the drone base assigns a flight plan to it, the drone will follow the flight path and take images at specified locations along the way. After completing the flight plan, the drone will upload the flight data and the captured images to the drone base. A thermal camera drone consists of two components. The flight controller fulfills the same tasks as the camera drone subsystem. However, in contrast to the camera drone, the flight controller in the thermal camera drone subsystem does not control the thermal camera directly. Instead, it only instructs the thermal camera component on when to take a picture. A scoop drone, responsible for the water sampling, also consists of multiple components. The flight controller component again handles the tasks related to status management and flight control. This component can also instruct an active extraction mechanism on when to do the water sampling. Additionally, the flight controller can collect readings from sensors while the drone is floating on water. In our architecture, we use a temperature sensor to represent this functionality.

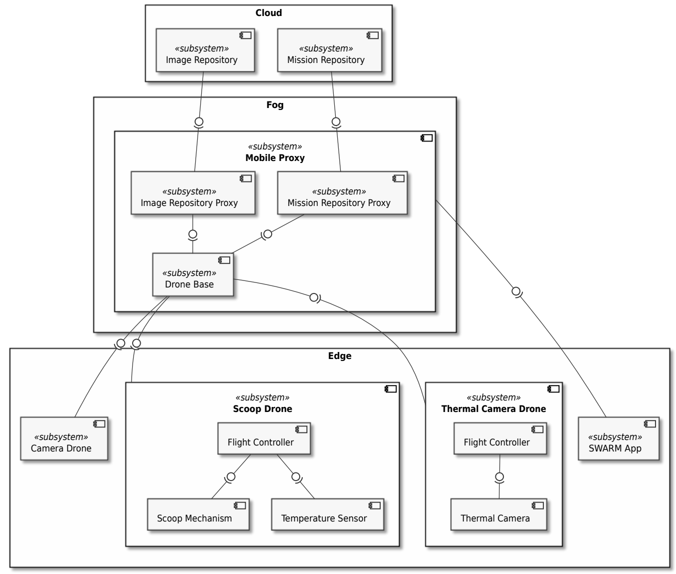

Fig. 2: Subsystem Decomposition for SWARM

V. Implementation

Our SWARM system consists of three parts: mobile laboratories, UAVs carrying different equipment, and back end systems in a fog and cloud layer (see section IV).

The mobile laboratory houses the laboratory for analysis of collected water samples and serves as the base for the three types of drones described in section IV. Researchers select a specific area of the lake to be analyzed in order to start a mission. Its plan is saved in the corresponding repository for later execution. Each mission can either be initiated manually or according to a pre-planned schedule.

The drone base pulls mission data from the mission repository, creates a flight plan accordingly and assigns it to the camera drones. The camera drones collect aerial data and upload it to the repository. The data is stored on a Python REST API server which was built using Flask-RESTful[3]. Upon successful upload, the system creates a map from the aerial data of the lake. Pattern recognition algorithms identify sites with anomalous characteristics; thereupon suitable locations for taking water samples get highlighted on the map. We are using OpenCV functions [22] for that. Based on this information researchers can identify suitable locations for taking water samples. Again, a flight plan is created and sent to the scoop drones. Containers that hold the water sample are attached manually to the landing gear before the start of the mission. The scoop drones fly to their designated location, land on the water, collect the probe, measure water temperature and return to the drone base.

[3]https://github.com/flask-restful/flask-restful

Upon arrival at the mobile laboratory, all metadata, including temperature values and sampling location, is transmitted to the drone base. The water samples get analyzed manually; only the concentration analysis of dissolved substances is partially automated. All findings are uploaded to the image repository and merged with aerial data. Due to the central storing, all test results are bundled and easily accessible for the analysis of the water pollution's source and can be compared to the results of other missions.

A. Hardware and Data Sources

As a result of the findings in section II-A, water quality indicators are to be chosen dependent on the water body's purpose. For our target site Lake Nokou, this means that the water quality indicators should enable researchers to find the source for water pollution in order to ensure the perpetuation of life in and around the lake. However, due to the reason that the prototype tested at Lake Durnholz should rather be a proof-of-concept than a fully functional system, we had to limit our choice of sensors. Fast availability and low initial cost were important factors. Therefore, we decided to work with water test strips and combined temperature, pressure and humidity sensors to measure water characteristics directly. Furthermore, RGB cameras and a thermal camera are used for more coarse-grained and extensive supervision of the water body.

The test strips are employed as they offer a range of different measurements, i.e. pH value, alkalinity, calcium hardness, bromine and chlorine concentration. Such test strips are widely available and low-cost. Moreover, the manual measurement is straightforward as the test strips' indicator substances change color upon a change in concentration and therefore are easy to handle without requiring any expertise. An application is built to partially automate and simplify the process of analyzing results in order to speed up evaluation and ease the further analysis of the data. The application extracts color values from the test strips for each indicator and compares them to the reference value. With this reference, the test image's values can be mapped to the respective concentrations.

Additionally, a combined temperature and pressure sensor is mounted on the scoop drone which allows direct measurement of the water body's temperature. A second sensor is mounted on the drone's upper part which allows comparing air temperature directly with the water temperature for further analysis. These sensors also have low initial costs and their digital output can be handled easily by Arduino boards to evaluate and integrate with the system.

These two direct ways of analyzing the water provide accurate readings at individual locations. A more extensive analysis can be ensured by making use of the drones' RGB camera which can fulfill a range of different tasks. With the images provided by the sensor, a map of the environment is generated in which the test strips' and temperature sensors' readings are integrated. Furthermore, the images can also be used for flight control, i.e. obstacle avoidance and object tracking, as well as visual supervision of a mission. Another use case for these images is detecting anomalies in the water body's visual appearance in order to find potential illegal activities that harm the water quality.

Further information is gained by overlaying RGB images with readings from thermal camera sensors which enables a more extensive water surface temperature analysis. This can be used to discover local temperature gradients that could hint at an illegal discharge of waste into the lake or determine sites that require deeper analysis. The thermal camera is mounted on a mobile phone which is attached to the camera drone. Temperature values are encoded into color values that can be evaluated both manually and digitally but require similar approaches as for the color images to extract meaningful features.

Moreover, a Raspberry Pi is used to control the scoop drone.

VI. System Evaluation

We evaluate the SWARM system by conducting unit tests on its subcomponents, and hence address the fulfillment of its functional requirements. Subsequently, we discuss to what extent its non-functional requirements were fulfilled.

Our field tests at Lake Durnholz mainly aimed to evaluate the *functional requirements*. The test involved both remote control of the drone as well as automated operation, both to land on and depart from ground and water surfaces. The drones altitude during the flight was 25 m and thereby within an altitude interval void of air traffic and ground objects, ensuring a collision-free flight. The landing procedure of the scoop drone was successful, partly because the RGB camera drone verified the landing spot to be free of any obstacles. The sampling of water was successful.

Upon the drones return to land, we verified that the sensor was able to correctly measure the temperature of the water and the air. To assess the test strip analysis subsystem, we manually evaluated the water sample with test strips and compared it to the system's result. The mobile laboratory app provided an accurate measurement result. We validated the proper storage and retrieval of data on our servers by sending REST-based requests. RGB and thermal images along with the corresponding metadata were successfully stored and retrieved. From these images, a coherent map of the scanned section of Lake Durnholz was created. Pattern recognition techniques could not yet be tested due to a lack of data.

Among the *non-functional requirements*, SWARM does not purport negative environmental impacts. It is able to collect water samples and perform online analyses, hence no traces on the environment are made. It is also possible to enlarge the hardware load on the drone so that its sensing capabilities can be extended. However, the additional load would reduce flight duration and must stay in an admissible range. Notably, while the flight duration of our drones was sufficient to handle the unit tests, it has not yet been tested to the full extent of large lakes like Lake Nokou. Until this point the system was operated by the development team. Conclusions on the maintainability and usability could therefore not be drawn yet. Since SWARM in its current state has remaining steps for integration, its scalability also could not be evaluated yet.

VII. Conclusion

In this paper, we introduced our Smart Water Monitoring System (SWARM) that facilitates research on water pollution in remote areas. We designed our system to deal with water quality issues that people face at Lake Nokou, Benin. Functional and non-functional requirements have been formulated that allow verification of our system: It has been found that the onsite situation requires an architecture that does not rely on permanent internet connection but can handle data processing via local networks. Therefore, we decided on a fog architecture approach that uses one or multiple fog nodes managing the exchange of data. The fog nodes synchronize with the cloud layer whenever possible but the functioning of the system does not constantly depend on it. Moreover, it has been shown that utilizing drones for the collection of water samples and supervision of the process has benefits over other methods. Thus, drones have been equipped with sensors and gear to sample water that allow for effective water quality assessment. Our prototype's sensor set, however, does not align with what our target site demands as the focus for the prototype lay on building a proof-of-concept and therefore we concentrated on factors like cost and availability for choosing our hardware.

Our system evaluation showed that SWARM is capable of sampling water and returning it to a mobile laboratory. Furthermore, water temperature measurements have been successfully conducted as well as concentration measurements for different substances in the water. Also, we were able to confirm effective communication between subsystems. However, there are some aspects that require further work in order to transfer our results to Lake Nokou. The already mentioned choice of sensors has to be aligned with the formulated requirements. Moreover, the system has to be scaled up to the size of the target environment by potentially increasing the number of drones, mobile laboratories, and drones' maximum flight duration. In the end, an implementation test in Benin has to be executed in cooperation with system developers as well as the locals.

References

[1] A. Hounkpatin, A. Edorh, M. Sezonlin, P. Guédénon, B. Elégbédé, G. Boni, V. Dougnon, S. Montcho, E. Kéké, M. Boko *et al.*, "Pollution of aquatic ecosystems by heavy metals at Ganvié's lacustrine city (Benin)," *International Research Journal of Biotechnology*, vol. 3, no. 6, pp. 81–87, 2012.

[2] A. Seitz, M. Avezum, B. Bruegge, and S. Wagner, "Interdisciplinary system courses-teaching agile systems engineering," *CEUR Workshop Proceedings*, 2019. [Online]. Available: http://elib.uni-stuttgart.de/handle/11682/10378

[3] C. Atti-Mama, A. Normann, J. Nielsen, and S. Sverdrup-Jensen, "Co-management in continental fishing in benin: The case of lake nokoué hydraulic department, cotonou-benin," in *Fisheries Co-Management in Africa: Proceedings From a Regional Workshop on Fisheries Co-Management Research [1998]*, 1998, pp. 18–20.

[5] United Nations, "Transforming our world: the 2030 agenda for sustainable development," *UN General Assembly*, vol. A/RES/70/1, 2015. [Online]. Available: https://EconPapers.repec.org/RePEc:ess:wpaper:id:7559

[4] B. Schmidt, "'The Venice of Africa' sinks into the rubbish - Lake Nokoué in Benin is Threatened Lake of the Year 2019," *Global Nature Fund*, Feb 2019. [Online]. Available: https://www.globalnature.org/en/threatened-lake-2019

[6] ISO, *Sustainable cities and communities - Indicators for city services and quality of life*. Beuth Verlag, Berlin, 2018, no. ISO 37120.

[7] C. E. Boyd, *Water Quality*. Springer International Publishing, 2020. [Online]. Available: https://doi.org/10.1007/978-3-030-23335-8

[8] J. Bartram and R. Ballance, Eds., *Water Quality Monitoring - A practical guide to the design and implementation of freshwater quality studies and monitoring programmes*. E & FN SPON, Jan 1996.

[9] J. A. Nathanson, "Water pollution," *Encyclopædia Britannica*, May 2019. [Online]. Available: https://www.britannica.com/science/water-pollution

[10] H. B. Glasgow, J. M. Burkholder, R. E. Reed, A. J. Lewitus, and J. E. Kleinman, "Real-time remote monitoring of water quality: a review of current applications, and advancements in sensor, telemetry, and computing technologies," *Journal of Experimental Marine Biology and Ecology*, vol. 300, no. 1, pp. 409 – 448, 2004, vOLUME 300 Special Issue. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S0022098104001066

[11] M. D. Dunbabin, A. Grinham, and J. Udy, "An autonomous surface vehicle for water quality monitoring," in *ICRA 2009*, 2009.

[12] G. Ferri, A. Manzi, F. Fornai, F. Ciuchi, and C. Laschi, "The hydronet asv, a small-sized autonomous catamaran for real-time monitoring of water quality: From design to missions at sea," *IEEE Journal of Oceanic Engineering*, vol. 40, no. 3, pp. 710–726, July 2015.

[13] M. Kumagai, T. Ura, Y. Kuroda, and R. Walker, "A new autonomous underwater vehicle designed for lake environment monitoring," *Advanced Robotics*, vol. 16, no. 1, pp. 17–26, 2002.

[14] H. Yu, A. Shen, and L. Peng, "A new autonomous underwater robotic fish designed for water quality monitoring," in *2012 Proceedings of International Conference on Modelling, Identification and Control*, June 2012, pp. 561–566.

[15] A. S. Woodget, R. Austrums, I. P. Maddock, and E. Habit, "Drones and digital photogrammetry: from classifications to continuums for monitoring river habitat and hydromorphology," *Wiley Interdisciplinary Reviews: Water*, vol. 4, no. 4, p. e1222, 2017. [Online]. Available: https://onlinelibrary.wiley.com/doi/abs/10.1002/wat2.1222

[16] A. Horcher and R. J. Visser, "Unmanned aerial vehicles: applications for natural resource management and monitoring," *Proceedings of the Council on Forest Engineering Proceedings*, 2004.

[17] L. Ma, M. Li, L. Tong, Y. Wang, and L. Cheng, "Using unmanned aerial vehicle for remote sensing application," *International Conference on Geoinformatics*, pp. 1–5, 06 2013.

[18] F. Bonomi, R. Milito, J. Zhu, and S. Addepalli, "Fog computing and its role in the internet of things," in *Proceedings of the First Edition of the MCC Workshop on Mobile Cloud Computing*, ser. MCC 12. New York, NY, USA: Association for Computing Machinery, 2012, p. 1316. [Online]. Available: https://doi.org/10.1145/2342509.2342513

[19] I. Stojmenovic, "Fog computing: A cloud to the ground support for smart things and machine-to-machine networks," in *2014 Australasian Telecommunication Networks and Applications Conference (ATNAC)*, Nov 2014, pp. 117–122.

[20] A. Seitz, F. Thiele, and B. Bruegge, "Fogxy," in *Proceedings of the 23rd European Conference on Pattern Languages of Programs - EuroPLoP '18*. ACM Press, 2018.

[21] N. Mohamed, J. Al-Jaroodi, I. Jawhar, H. Noura, and S. Mahmoud, "UAVFog: A UAV-based fog computing for internet of things," in *2017 IEEE SmartWorld, Ubiquitous Intelligence & Computing, Advanced & Trusted Computed, Scalable Computing & Communications, Cloud & Big Data Computing, Internet of People and Smart City Innovation (SmartWorld/SCALCOM/UIC/ATC/CBDCom/IOP/SCI)*. IEEE, aug 2017.

[22] G. Bradski, "The OpenCV Library," *Dr. Dobb's Journal of Software Tools*, 2000.