Unsupervised learning of wildlife behaviour for activity-driven opportunistic beacon networks

Fatjon Seraj, Eyuel Ayele, Nirvana Meratnia
Pervasive Systems Group
University of Twente, Enschede, The Netherlands
Email: {f.seraj, e.d.ayele, n.meratnia}@utwente.nl

***Abstract*—Monitoring wild animals in their natural habitat and in real time constitutes an essential aspect of biological and environmental studies. Monitoring is mainly conducted through wireless wildlife monitoring systems *(WMS)* due to their energy-efficiency and scalability properties. However, using WMS often involves the deployment of energy demanding wireless radio technologies and protocols that significantly increase energy consumption while tracking mobile animals. Thanks to the raise of IoT devices capable of sensing, computing, and wireless networking, WMS can become more efficient and overcome the initial drawbacks. This paper, describes an activity driven beaconing mechanism based on unsupervised activity classification scheme. The algorithm is evaluated for different parameters involving the sampling rate, processing window as well as different cluster sizes. The evaluation shows that use of lightweight algorithms and low sampling rates provides the possibility to reliably monitor the activity of the animal. The evaluation results showed that the proposed mechanism could reduce energy consumption by increasing communication sleep-time while the objects were stationary.**

***Index Terms*—Wildlife monitoring, IoT, sensor data, BLE, unsupervised learning, Self organizing maps**

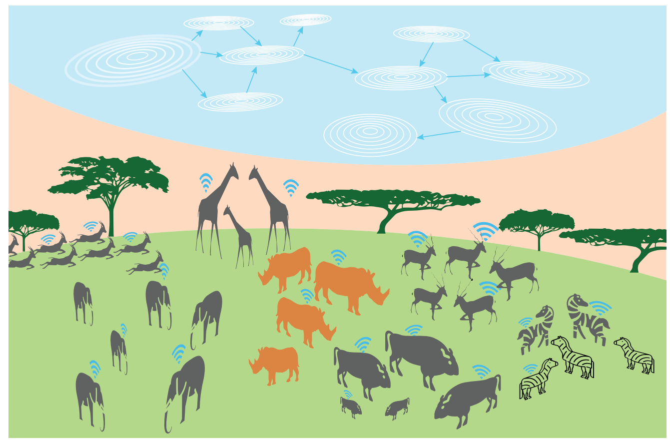

Figure 1. A wireless wildlife monitoring mock-up. Herds communicating together can be used as sentinels for endangered species in red.

I. Introduction

The past few years, have witnessed the advances in low power wireless network technologies while exploiting the potentials of the internet of things (IoT) in broad application domains. Wildlife monitoring is one of such trending IoT applications where a number of heterogeneous sensors (e.g. accelerometer and gyroscope, etc) are deployed in the form of collars, to monitor the activities of wild animals dwelling in remote and geographically large habitats [1]. In the context of wildlife monitoring systems where animals perform their actions in herds, detecting the exact activity of the individual is less important than knowing the whole herd state.

By nature, herds are associations of animals of the same type acting together. One main characteristic of the animals belonging to the herd is that the individuals behave according to the behavior of the majority of other members. This characteristic of the herd can be exploited to build a system that detects the behavior of each individual without high precision or accuracy but when placed in the context of the herd communications system, increases the accuracy by corroborating with the behavior of other individuals.

Albeit it might look like animals move in a chaotic manner, they do have a well-established mobility pattern, where the travel is directed toward points of interest (e.g., water, grazing, etc.) or necessities caused by distressing incidents (e.g. preying, running, etc.), often using the fastest path possible.

Therefore, wildlife monitoring solutions need to track the type of herd activities in real time as well as provide other services such as data processing and proximity detection. To these ends, any proposed solution is required to achieve: (i) high energy efficiency, since the sensors used will operate with a limited source of energy, (ii) high reliability to avoid false alarms, and (iii) low latency for a responsive system design.

An implementation scenario is the protection of endangered species like rhinos, where other animals that share the same habitat with rhinos are equipped with sensor nodes and act as live sentinels when poachers are in vicinity approaching their target, as shown in Figure 1.

Thus, to realize the design requirements, a mechanism to control the trade-off between energy versus latency is necessary, which is not practically achievable by using a wireless technology alone. Although researches have been conducted to address this issue by proposing an architecture for wildlife, to the best of the authors' knowledge, none of these works include sensor data classification mechanism to optimize the networking protocol and duty cycle, in their approach.

Therefore, in this paper, we address the importance of an efficient activity recognition algorithm for wildlife monitoring that exploits the actual accelerometer sensor data coupled with a new BLE beacon IoT network. This work shows that the

978-1-7281-4807-6/19/$31.00 ©2019 IEEE

accelerometer-based mechanism can be an accurate, robust and practical method for objectively monitoring the free movement of animals. Through the use of sensor fusion to beacon networks, the proposed architecture provides a wider control over the trade-offs between energy and latency, eventually making the system scalable and energy efficient. The main contributions of this work are listed below. (i) Light-weight and robust activity recognition algorithm for wildlife monitoring. (ii) Unsupervised Self Organizing Maps are proposed as reliable clustering mechanism for activity recognition. (iii) A new beacon IoT network architecture for wildlife monitoring is proposed. (iv) Design guidelines for beacon advertising is discussed.

II. Related Works

Biology researchers employ animal monitoring as tool to study the behavior mobility and habitat of specific animal or species. The study focuses on tracking the behaviour with special sensors attached to particular individuals of the herd. The collected data are either wirelessly broadcast or manually collected from the animal. Wireless data transmission in intractable environments is ponderous in terms of reliability and energy efficiency. Coke at al. [2] describes the preferable characteristic for a continuous bio-telemetry system for remote animal monitoring. They identified a set of limitations and challenges in the domain such (i) Understanding pattern in the data, (ii) Accuracy of energy estimates, (iii) Need for calibration, (iv) Burden on animal, (v) Availability and/or customization of the sensor nodes.

Recent IoT and machine learning scientific advancements in addition to wireless data broadcasting, provide the advantage to compute the data on the sensor node itself. Providing a broader real-time picture of the whole herd rather than particular individuals. Considerable research is conducted on behavioral classification using sensor data from animal collars.

Gonzales et al. [3] describe a setup for cattle behavioral monitoring with sensors rigidly secured on a known position on the collar. The nodes consists of a GPS sensor and 3axis accelerometer sampling at $4\&10Hz$. The data are processed every 10 second, computing the mean and the standard deviation for vibration and speed. Histogram analysis of the travel speed, identified 3 states corresponding to stationary, slow and fast travel behaviors. Accelerometer histograms of the mean from the gravity axis recognized states concerning the head position (up/down), while the standard deviation feature identified behaviors corresponding to the activity level (high, medium and low).

Juang et al. [4] describe ZebraNet, a setup in which the authors tend to address the shortcomings of the wireless WMS, identifying the power consumption elements such as GPS tracking systems, the duty cycle of wireless radio, sensing capabilities of the node, range of data collection setup, etc. The aim of this work was also to monitor the zebras' mobility pattern and build a suitable WSN protocol to fit this purpose. In ZebraNet, authors also identified three states for the animal activity graze, graze-walking, fast moving. Generalizing for most of our animals of interest in a simplified manner, the activity state of the animals in a herd can be described as a 1) passive(stationary) state, 2) active state and 3) panic state.

Different protocols specific to Wireless Sensor Network (WSN) are proposed and used for data broadcasting. Opportunistic beacon networks, on the other hand, present a robust energy efficient solution. To the best of our knowledge, this network typology has not been researched for WMS.

In their work Turkes et al. [5] describe a beacon opportunistic dissemination protocol based on WiFi beacon frames for vehicular pavement condition data. Taking a hint from this work and replacing the WiFi beacons with BLE beacons we can provide an efficient data dissemination protocol for animal activity. The data is then collected in a sink node with Lora radio to be conveyed on the cloud.

III. Beacon Protocol Overview

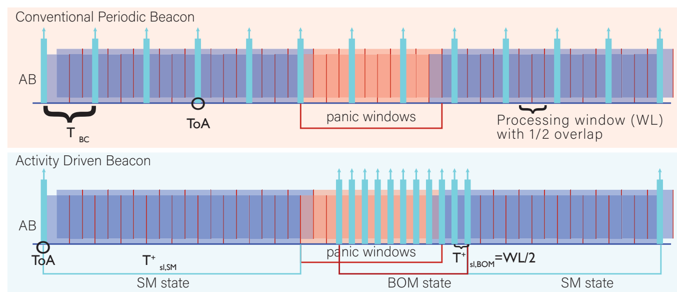

Figure 2. Mobility driven BLE AB-to-AS beacon advertising strategy. SM-Stationary state, BOM- Beacon on mobility. The system checks for movement activity every window length ($\frac{WL}{2}$). ToA is the time-on-air of BLE beacon data,

The entire approach for the event-driven opportunistic *(WMS)* is based on BLE bearer adaptive beacon advertising method. In WMS, a draw (or entire) from the herd population is collared with sensor nodes equipped with BLE radio. There are two network device roles that use short-range BLE to communicate beacon data in our WMS: (i) AS- Animal scanner is a BLE scanner node, which listens for BLE beacons in the surrounding, and (ii) AB- Animal Broadcaster is a BLE beacon broadcasting node, which sends BLE beacons to the surrounding AS. For details on BLE technology, the readers are referred to [6], [7]. In general, as far as wireless communication is concerned, animal mobility is classified as (i) SM-stationary state with no or relatively less mobility (e.g. grazing, graze-walking); and (ii) BOM- Active state with high mobility (e.g. running from predators or illegal poachers) Figure 1. Hence, instead of using conventional periodic beaconing, AB nodes send beacons in two state fashion: (i)- Stationary mode (SM), where beacons are transmitted in an optimal ADV interval (T_{BC}) for less frequent, and (ii)- Beacon-On-Motion (BOM), a highly active state (s), where each AB node detects own activity as either non-mobility (SM) or active state (BOM) based on the detected activity using accelerometer data as described in the next Section IV. In conventional BLE beaconing, each AB node periodically wakes-up every interval of (i.e. T^{+}_{BC}) for a very short duration (ToA) to send the data,

978-1-7281-4807-6/19/$31.00 ©2019 IEEE

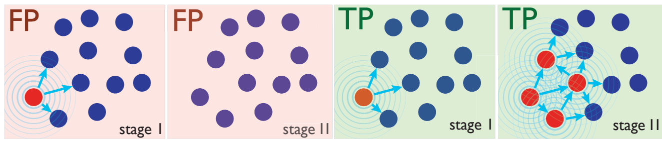

Figure 3. False positive FP scenario when other nodes do not confirm the detection and the true positive TP scenario when adjacent nodes corroborate the state change

which makes the process redundant if nothing happens in the animal realm as shown in Figure 2. Therefore, we propose to utilize activity detection and classification capabilities of the sensor node to optimize the BLE beacon advertising at AB node, thus decreasing substantially the energy consumption on the node itself.

Figure 3 shows two scenarios for the herd and network state, based on the activity classification. In stage I one node detects a panic state and transmits the state. If the state is a false positive FP the other nodes will confirm by not entering into a panic state, the opposite will happen if the detection is a TP.

AB nodes adjust their beacon advertising interval by decreasing or increasing the advertising interval depending on its mobility state. These intervals depend on the optimal data processing window length (WL) as determined by the classification algorithm. $T^{+}_{sl,BOM} = WL/2$, are $T^{+}_{sl,SM}$ the advertising interval for the SM and BOM states respectively.

IV. Unsupervised Animal Activity Classification

For the purpose of detecting the activity states of the animal the dataset from [8] is used. The data are generated by 5 goats outfitted with collars, where 6 sensors nodes are placed, around the collar, in different positions. The sensor data is sampled at 100 Hz and labeled for 16 types of activities performed by goats. In real life implementation the sampling rate, as well as the feature extraction and classification calculations for these many classes, performed in the node, might result too complex and raise energy concerns.

While monitoring zebras in savanna, researchers [4] identified three activity states: graze, graze-walking, fast moving. Therefore, in a simplified manner, the activity state of the animals in a herd can be described as a 1) passive(stationary) state, 2) active state and 3) panic state.

In the stationary state, the animals usually graze, sleep, stand, move their heads, scratch, rub etc. In the active state, animals walk around, fight with each other, climb up and down. In the panic state the animals usually trot and run, and this is a herd behavior rather than an individual one. According to these state, we can reduce the number of labeled activities in the dataset as shown in Table I.

Also, the activities tend to happen for a certain amount of time with a certain frequency. We assume that the activity will last for at least 6 seconds hence we decide for an optimal window length for the feature calculation Δt seconds. Considering the above assumption we decide to reduce drastically the sampling rate of the sensor and the number of class activities.

Detecting activities using sensor data requires the implementation of some learning algorithms that can classify or

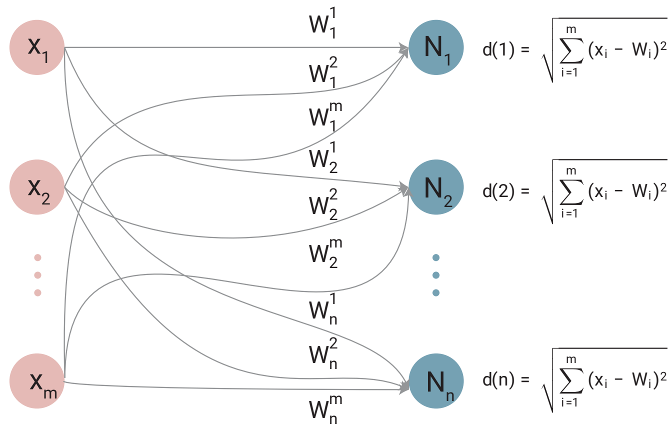

Figure 4. Self Organizing Maps

group the activities into distinguishable states. There are two main learning algorithm groups, supervised and unsupervised learning algorithms. Supervised learning algorithms learn by example, thus requiring a huge amount of labeled data representing the example being learned. Unsupervised learning algorithms, on the other hand, tend to group the sample into clusters with similar features. Such algorithms are for example: K-means, Gaussian Mixture Models or Self-Organizing Maps.

Table I
Reduction of initial activity

	Passive L1	Active L2	Panic L3
1	shaking	walking	trotting
2	standing	fighting	running
3	lying	food_fight	
4	grazing	climbing_up	
5	eating	climbing_down	
6	brest_feeding		
7	scratch_biting		
8	rubbing		
9	standing_up		
10	null		
	Distribution of activities as percentage		
	68.55	29.07	2.37

A. Self Organizing Maps

Self-organizing maps (SOM) learn to classify the input feature vectors based on how they are grouped in the input space. SOM is a single layer feed-forward neural network where all the inputs are connected to all the output neurons. Neurons in the SOM learn to recognize neighboring sections of the feature space and learn both the distribution and topology of the input vectors they are trained on.

The SOM algorithm is based on unsupervised, competitive learning. The proposed method is appropriate for clustering problems, i.e. grouping different elements according to the similarity in pattern and feature set. SOM is inspired by the way the brain stores and organizes the information, by

978-1-7281-4807-6/19/$31.00 ©2019 IEEE

storing the correlated information in close by area. SOM creates a bi-dimensional map of neurons, a neuron lattice, in which the input features are grouped through a neighborhood function that calculates the degree of similarity between them. Features representing similar information will be closed on the neuron map. Fig.4 shows a typical visualization of a SOM neuron network lattice. The features are recursively shown to each of the neurons. The neighborhood is characterized by the distance between neurons. The behavior of a SOM network can be summarized in the following steps: Consider, the input features D^i are calculated every given time Δt^i, $D^i = \{x_1, x_2, ..., x_m\}$ where m is the number of features. We set the initial weights vector W for each neuron N , randomly close to zero. Each time the feature vector D^i is shown to the neuron N, the distance d_i is calculated as Euclidean distance, between the feature vector and the weights of the neuron. Once, all distances are calculated, the neuron with the shortest distance will be selected as Best Matching Unit (BMU).

$$BMU = \operatorname{argmin}(d) \tag{1}$$

The wining neuron will adjust the weights using the neighbourhood function $\Theta(u, n, s)$ and the update rule $\alpha(s)$, where s is the present iteration or epoch.

$$W_n(s+1) = W_n(s) + \Theta(u, n, t)\,\alpha(s)\left(D^i(m) - W_n(s)\right) \tag{2}$$

To decorrelate the feature input we apply Zero Component Analysis (ZCA) whitening. Whiteningin is a data transformation where the covariance matrix Σ is the identity matrix. We first estimate the covariance matrix $\Sigma \in D^{m\times m}$

$$\hat{\Sigma} = \frac{1}{m-1}\sum_{i=1}^{m}(x_i - \bar{x}) \cdot (x_i - \bar{x})^T \tag{3}$$

Then we calculate the singular value decomposition for the covariance matrix Σ to calculate the singular values unitary $u \in D^{m\times m}$ and the diagoanl $s \in D^{m\times m}$ of Σ. Then we compute ZCA as follows:

$$ZCA = u \cdot \frac{1}{\sqrt{s+\epsilon}} \cdot u^T \tag{4}$$

Some of the s values might be close close to 0, and to avoid the scaling moment where we divide by $\sqrt{s}$ we add a constant $\epsilon = 10^{-5}$ The whitened data is a product of input data with the ZCA.

$$D^i_{white} = D^i \cdot ZCA \tag{5}$$

B. Training SOM

For the SOM training we calculate the feature vector by windowing the data into windows with length Δt and half window overlap. We first remove the gravity vector from accelerometer data by applying a low-pass filter to calculate the gravity, then removing the gravity component from the sensor data as shown in:

$$\begin{aligned} G(x) = (a \times G(x-i)) + (1-a) \times Acc(x) \\ LAcc = G(x) - Acc(x) \end{aligned} \tag{6}$$

Sequentially, the magnitude of the linear acceleration and rotation is computed, followed by feature extraction. On account of simplicity, only the time-domain analysis is considered, computing five features for each sensor: mean, variance, root mean square, kurtosis, and skewness, resulting into a 5 and 10 feature long array D^i respectively, for accelerometer alone or a combination of accelerometer and gyroscope. Features $D^i = \{x_1, x_2, ..., x_{10}\}$ are calculated every given time Δt^i, resulting into our feature matrix.

$$D = \begin{bmatrix} x_1^1 & x_2^1 & ... & x_{10}^1 \\ ... & ... & ... & ... \\ x_1^n & x_2^n & ... & x_{10}^n \end{bmatrix}$$

V. Evaluation

Evaluation, of the activity clustering based on SOM learning algorithm, is performed for different parameters influencing both the efficiency and accuracy of the results. These parameters are introduced during the data processing phase and the training phase as follows:

- Sampling rate (SR). The original data set sampled at 100HZ is downsampled to $\{$*10Hz, 5Hz, 2Hz, 1Hz, 0.5Hz*$\}$. This parameter allows to establish the minimal SR able to detect most of the animal activities.
- Window length (WL). Defining the optimal WL is imperative to decrease the computation frequency for the feature extraction and the classification. Additionally, the window must be wide enough to capture the entire pattern of the performed activity. Therefore, the SOM is trained with $WL = \{2sec, 4sec, 6sec\}$ for SR above 1Hz, and $WL = \{4sec, 8sec\}$ for SR below 1Hz.
- Lattice size (LS). In an effort to find the appropriate dimensions for the SOM neuron lattice, the training phase is conducted with $LS = \{4, 9, 16, 25\}$ neurons.
- Epochs (EP). The epochs is fixed at 1000 iterations.
- Training instances (Ti). From the initial 17 labeled classes, 1000 instances are randomly drawn.

This will result into 72 different models, 4 for each window length. The evaluation is initially executed on the accuracy of the clustering algorithm, for each LS. Followed by the examination of the effects of WL both in processing efficiency as well as in accuracy.

A. Clustering results

Considering that the obtained data set is labeled, the clusters can easily be converted into classes by feeding the feature set of the same class and observing the cluster it will fall into. Resulting in a sort of confusion matrix $m \times n$, where m is the lattice size and n is the class number. Since the classes distribution is not balanced, the number of elements in the cluster are shown as the percentage of the class.

Figure 6 shows the results from data sampled at 10Hz and 2 sec window for different cluster sizes. Clearly the *panic* class creates distinctive clusters with some overlapping with an active state, which is expected as the walking and running activities only differ in frequencies when measured from the

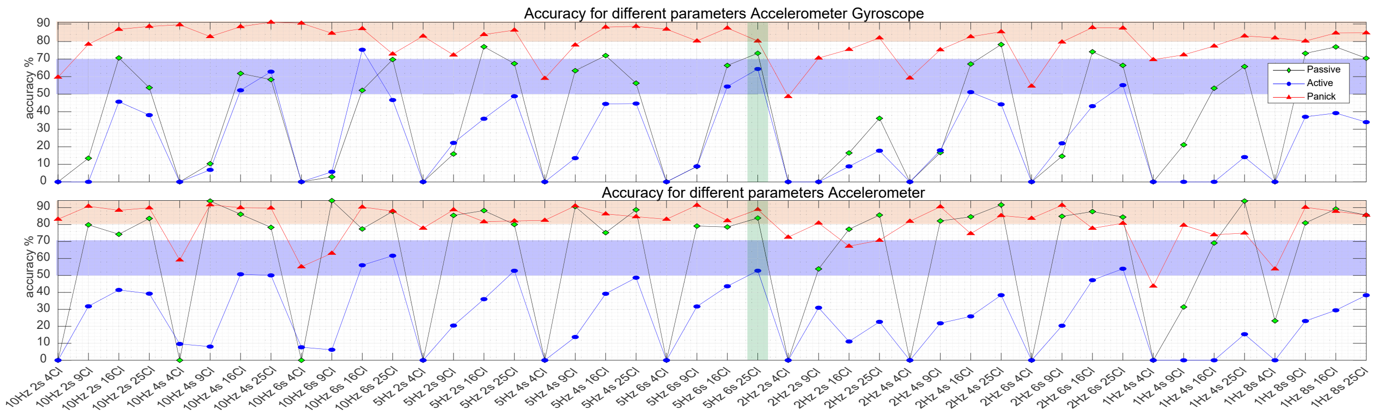

Figure 5. Accuracy for different parameters

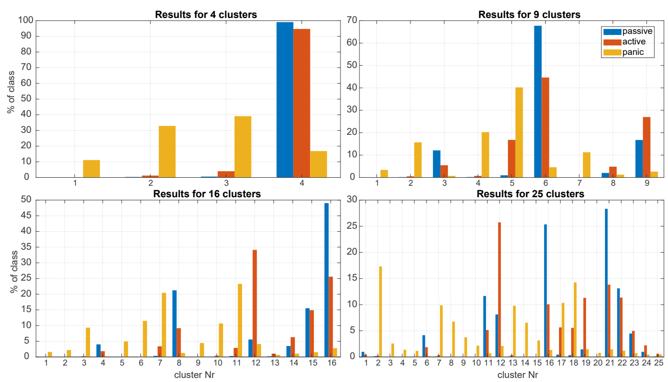

Figure 6. Classification results for different lattice size, 10Hz, 2 sec window

neck of the animal. Nevertheless, the overlap is very small. However, is more challenging to distinguish the *passive* and *active* classes with less clusters. *Active* class is only clustered distinctively for only 35% of instances in cluster Nr.12 $(4 \times 4)SOM$ and cluster Nr.12&19 $(5 \times 5)SOM$.

The probability that a cluster C^n belongs to one of three classes C_i^n is calculated as follows:

$$P(C^n) = \frac{C_i^n}{\sum_{i=1}^{3}(C_i^n)} \tag{7}$$

The cluster is considered a representative of the given class if the $P(C^n) > 0.5$. From here the accuracy of each class Cl_i is calculated as

$$Acu(Cl_i) = \sum P(C_i) > 0.5 \tag{8}$$

Thus we can identify the clusters that belong to a class with a certain level of probability. This method results in a better cluster distribution toward different activities that show similarities, such are *walking* and *running*. Once the accuracy is computed for all the parameters we pick the maximum pair of accuracy for all the classes. Figure 5 shows the accuracy for different parameters of the sampling rate, window length and cluster size when accelerometer alone is used, or when both the accelerometer and gyroscope sensors are used. It can be observed that when gyroscope sensor is used to compute the magnitude of the rotation for the goat collar it helps to distinguish better the running activity when the animal does not rotate the neck that much, however, it brings uncertainties in distinguishing the active state with miss-classifications that result in a drop of accuracy.

B. Processing time evaluation

The proposed classification algorithm has a straightforward mechanism starting with filling a buffer with sensor data, calculating features over that buffer, normalizing the feature set, followed by feeding the normalized feature set to the SOM model. For a thorough evaluation, we measure the average time required to complete one full task operation for each parameter value. The objective is to optimize the parameter selection that satisfies the restrictions set by the wildlife monitoring task. Figure 8 shows the average time required by the algorithm to calculate the feature set for one sensor for different window lengths and different sampling rate. The higher the sampling rate the higher the processing time, also the wider the window length the higher the processing time but less frequently the number of processing.

The average time required to normalize the feature set of length 5 features is $0.067\mu s$ if accelerometer alone is used and $0.1\mu s$ for 10 features if booth accelerometer and gyroscope sensors are used.

Figure 8 shows the processing time required to apply SOM model over the feature set. As expected, fewer clusters will result in fewer weights, thus shorter processing time.

C. Results for chosen parameters

Once the best parameter tuple is defined the data clusters are tested against the stream of labeled data. The purpose of this step is to see how the misclassified data affect the overall classification. First we need to calculate the True Positive Rate (TPR) as the ratio of True Positives (TP) over Positives (P)

$$\mathrm{TPR} = \frac{\mathrm{TP}}{P} = \frac{\mathrm{TP}}{\mathrm{TP} + \mathrm{FN}} \tag{9}$$

It is also important to know the False Detection Rate (FDR), ie. the rate of falsely detecting the window as positive, which

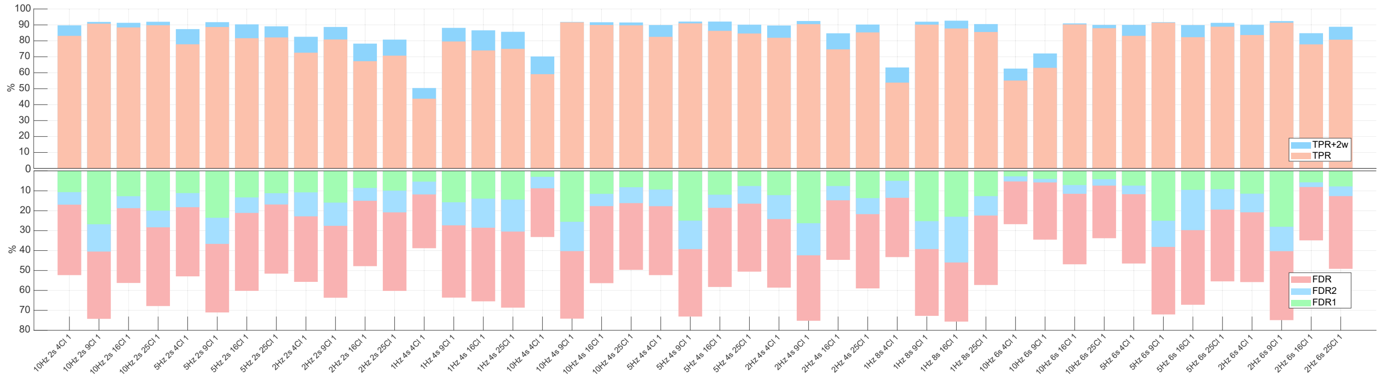

Figure 7. TPR and FDR for accelerometer results. TPR+2w when 2 consecutive false negatives are excluded from false negatives. FDR1, FDR2 when up to 2 and 3 consecutive false positives windows are excluded from the results

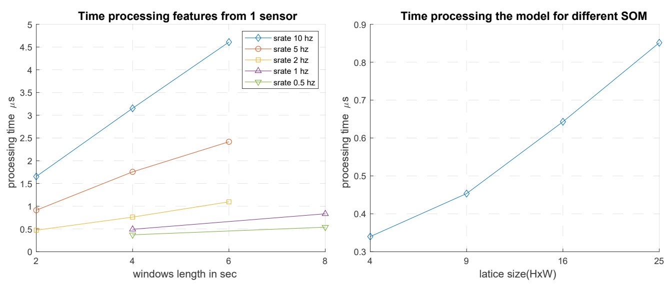

Figure 8. Calculating processing time for different parameters

is the rate of False Positives (FP) over all the positive detection $FP + TP$

$$\mathrm{FDR} = \frac{\mathrm{FP}}{\mathrm{FP} + \mathrm{TP}} \tag{10}$$

Figure 7 shows a high FDR, nevertheless most of the FDR are just single miss-classified windows. Furthermore to establish the right ratio of detection before the panic flag is raised the algorithm requires at least 3 consecutive *TP* windows before the radio is turned on and broadcasts the activity. The rationale behind this reasoning is that the animal cannot change the states only for the duration of one window length. If the animal runs, it will run for an amount of time greater than the window length (WL). The same argument is valid for the TPR, 2 consecutive false negative windows that happen while the panic flag is raised will be considered true positives. FDR1 and FDR2 show the improvement if up to 2 and 3 consecutive false positives are excluded from the detection results. Given the half window overlap, the algorithm excludes false positives that lasted for $1\frac{1}{2}WL$ or $2WL$ respectively.

VI. Conclusions

The present study show that low sampling data rate from sensor attached on the animal collar can provide sufficient information to detect different activity states. The ability to classify three distinctive activity states (stationary, active and panic state), consents us to apply this knowledge to improve the data communication between sensor nodes without sacrificing the most critical requirement for WMS, the power consumption of the sensor node. Applying an unsupervised learning algorithm such as SOM, provides substantial benefits over other classification methods such are supervised machine learning algorithms. By using SOM, the activities are clustered into distinctive groups without prior knowledge, thus reducing the need of labeling the data for each activity or animal type. In fact, the paper showed that with a very small number of samples, 1000 for each class, the detection accuracy reaches 90% TP and reduce the FP to 10%. When placed into the context of activity driven opportunistic beacon network, the classification accuracy translates into a drastic increase of radio sleep time without compromising the quality of service. The decrease of energy consumption allows for the introduction of additional sensors as well as monitor the entire herd for longer periods of time, study their mobility patter, points of interests, social behaviour, health etc.

References

[1] W. Cooper and D. Blumstein, *Escaping From Predators: An Integrative View of Escape Decisions*. Cambridge University Press, 2015.

[2] S. J. Cooke, S. G. Hinch, M. Wikelski, R. D. Andrews, L. J. Kuchel, T. G. Wolcott, and P. J. Butler, "Biotelemetry: a mechanistic approach to ecology," *Trends in Ecology & Evolution*, vol. 19, no. 6, pp. 334–343, Jun. 2004. [Online]. Available: http://linkinghub.elsevier.com/retrieve/pii/S0169534704001120

[3] L. González, G. Bishop-Hurley, R. Handcock, and C. Crossman, "Behavioral classification of data from collars containing motion sensors in grazing cattle," *Computers and Electronics in Agriculture*, vol. 110, pp. 91–102, 2015.

[4] P. Juang, H. Oki, Y. Wang, M. Martonosi, L. S. Peh, and D. Rubenstein, "Energy-efficient computing for wildlife tracking: Design tradeoffs and early experiences with zebranet," *SIGARCH Comput. Archit. News*, vol. 30, no. 5, pp. 96–107, Oct. 2002.

[5] O. Turkes, F. Seraj, H. Scholten, N. Meratnia, and P. J. M. Havinga, "An Ad-Hoc Opportunistic Dissemination Protocol for Smartphone-Based Participatory Traffic Monitoring," in *(VTC2015-Fall)*. Boston, MA, USA: IEEE, Sep. 2015, pp. 1–5.

[6] "Ble core specifications," September 2018, [online] https://www.bluetooth.com/specifications/bluetooth-core-specification.

[7] E. D. Ayele, K. Das, N. Meratnia, and P. J. M. Havinga, "Leveraging ble and lora in iot network for wildlife monitoring system (wms)," in *2018 IEEE 4th World Forum on Internet of Things (WF-IoT)*, Feb 2018, pp. 342–348.

[8] J. W. Kamminga, D. V. Le, J. P. Meijers, H. Bisby, N. Meratnia, and P. J. Havinga, "Robust sensor-orientation-independent feature selection for animal activity recognition on collar tags," *UBICOMP*, vol. 2, no. 1, pp. 15:1–15:27, Mar. 2018.