Cow estrus detection with low frequency accelerometer sensor by unsupervised learning

Tran Viet Toan
Ho Chi Minh city University of Technology
Ho Chi Minh, Vietnam
111111@hcmut.edu.vn

Rin Nishikawa
Tokyo University of Agriculture and Technology
Tokyo, Japan
nishikawa@nj.cs.tuat.jp

Le Tien Thanh
Ha Noi University of Science and Technology
Ha Noi, Vietnam
20144075@student.hust.edu.vn

Masashi Takemoto
BeatCraft, Inc
Tokyo, Japan
lesser@beatcraft.jp

Tran Van Hoai
Ho Chi Minh city University of Technology
Ho Chi Minh, Vietnam
hoai@hcmut.edu.vn

Hynh Thi Thanh Binh
Ha Noi University of Science and Technology
Ha Noi, Vietnam
binhht@soict.hust.edu.vn

Hironori Nakajo
Tokyo University of Agriculture and Technology
Tokyo, Japan
nakajo@cc.tuat.ac.jp

ABSTRACT

Recently with the explosion of internet of things and data mining, a lot of research has applied successfully in agriculture help increase productivity and reduce labor. One of challenges in IoT (internet of things) is energy of devices. In this study, we developed a method use unsupervised learning to detect cow estrus event with low sample data frequency from the cow collars. Estrus is the special period during reproductive cycle when female animals become sexually accessible. In artificial propagation of cows, the detection of cow estrus is very important. If it isn't detected, the cow will not be able to breed. If it is detected false, then after insemination the cow will not be pregnant, all cost for female cow farming is considered to be zero. In this paper, our method tries to extract feature and cluster next we will calculate activity index level then we will predict estrus or not. The method shows us the activity index level of hour happen estrus will be unusually high.

(thanh-comment) Overall, the paper only specify the pipeline of the method. Each part of the pipeline did not include the detail explaination of why the author choose the component (why use GMM, FFT, etc). IMHO, the pipeline can be found easily on many related paper, and you can describe it briefly.

Instead, you should focus on your main and novel contribution to the pipeline.

- One suggestion is you can focus on the post processing with some heuristic on timing constraint of estrus event happened in only 12 hours, repeat within 18 to 24 days, and reduce the false positive of the system.
- Besides, another contribution could be dimensionality reduction for sparse time series (low-frequency problem for energy saving).

If you finally determined that it is clearly your new contribution to the field, then you tailor all of the part of the paper toward that

- fix the title of the paper to highlight on your novel contribution
- related work show that the problem/shortcoming has not been tackled in the past
- method clearly explain the method with longer details than other part of the detection pipeline
- design the experiment following the that main contribution

CCS CONCEPTS

• **Applied computing** → **Agriculture**.

KEYWORDS

estrus detection

ACM Reference Format:
Tran Viet Toan, Rin Nishikawa, Le Tien Thanh, Masashi Takemoto, Tran Van Hoai, Hynh Thi Thanh Binh, and Hironori Nakajo. 2019. Cow estrus detection with low frequency accelerometer sensor by unsupervised learning. In *SoICT 2019: Symposium on Information and Communication Technology, Dec 4-6, 2019, Hanoi - Halong Bay, VN*. ACM, New York, NY, USA, 8 pages. https://doi.org/10.1145/1122445.1122456

SoICT 2019, June 03–05, 2018, Hanoi - Halong Bay, VN

ACM ISBN 978-1-4503-9999-9/18/06…$15.00
https://doi.org/10.1145/1122445.1122456

1 INTRODUCTION

From the past until now cattle breeding has always been an important manufacturing industry. With the development of science and technology, the productivity of cattle breeding has been improved. Recently, IoT (internet of things) has proved effective in cattle breeding applications. The cow is one of the most important cattle that provide humans with beef and milk. Estrus detection is the first step to get a cow pregnant. Estrus detection is necessary for the planned insemination process and is the key to successful use of artificial insemination. But, cow estrus detection is a challenging job for farmers. Cow estrus detection involved 300 million dollars in American dairy farming in 1994 [Senger 1994]. The estrus cycle occurs on average 21 days within a normal range of 18 - 26 days, yet it lasts only for range 7 - 18 hours for one time [Diskin and M. Sreenan 2000]. Farmers must observe cows 30 minutes for a time, three times a day to detect estrus in cows. Each missed estrus is equivalent to 21 days loss in production [Rao et al. 2013]. Although researchers introduced a rich variety of methods for the detection of estrus, due to the inaccurate and ineffective of the current algorithm, a more accurate and practical method is still required. Thus, any method estrus detection with high accuracy will bring great value for breeding industry.

Until now, they have so many technologies that can detect cow estrus events, but almost of methods with low accuracy, high material cost, high labor cost. The simple method is visual observation and tail-painting, which requires considerable skill and experience of farmers. So this method will be a labor waste and impossible to handle a large farm. The higher accuracy method is doing an examination test with milk and blood to monitor the hormone, but this method is so expensive. Another method is the basal body temperature monitoring. However, the basal body temperature of cows changes significantly during the day-depends on weather conditions, activities and health status of cows. A few methods use accelerometers and odometers for the detection, but the frequency is high, so they will interfere with the device's energy problem when applied [Miciakova et al. 2018].

When cows in estrus period, three distinct patterns are observed such as males like mounting, rise in spontaneous activity and mating responses [Rao et al. 2013]. Lately year, a few researchers used GPS data and accelerometer sensors to monitor cow estrus events. This paper [Hanson and Mo 2014] indicates the different moving behavior when cows in estrus period. This paper provides solid reasons for using an accelerometer sensor to detect estrus events for extremely feasibility.

Our main contribution are:

- Develop the method suitable for low frequency, less information data to detect cow estrus events by unsupervised learning.
- Implement the algorithm to find estrus events from activity index level with unlabel estrus data, also it help to reduce false positive of the system.âĂŃ

In our study, we develop a method uses accelerometer data with only one dimension by unsupervised learning. Because of estrus period happens in a short time and artificial insemination must be done a few hours before estrus start time until a few hours after estrus finish, so predicting estrus day is not serviceable for application. Thus, we try to predict the early hours during the estrus period in our study. Devices use low frequency and Bluetooth low energy (Bluetooth LE, BLE) technology suitable for IoT. We use slide windows to split segments from data time series, then design feature extraction and have the feature vector for each segment, next we use K-mean [Lloyd 1982] to cluster and computing activity index level. The result of this study shows points in estrus detected have unusually high activity index level.

The remaining of the paper is organized as follows. We shortly discuss the related work in Section 2. In Section 3, we will describe the unsupervised learning method and how it works, before delving into details about the result of this method in Section 4.

2 RELATED WORK

In this section we will mention to a few related methods and discuss some problems of those methods.

In [Vanrell et al. 2014], an supervised learning method is used to detect estrus with 1Hz frequency and three dimensions of accelerometer data. It guarantees a high accuracy output but the cow behaviors level in estrus period is various depend on cows behaviour, so supervised learning in this situation it may not work clearly for all cows.

The unsupervised learning method is presented [Shahriar et al. 2016] with high frequency (10Hz) and three dimensions accelerometer data. This method with high frequency will be easier because of more information but it will be waste energy in devices and resources on the centralized servers in farms. A more recent study [Thanh et al. 2018] uses Dynamic Time Warping and Iterative-K-Means to solve the problem.

In [Fukumoto et al. 2018] they proposed the method to detect estrus concerning the approaching behavior between cattle with GPS (Global Positioning System) 0.2 Hz frequency. This method only suits for big farm, outdoor farm, with small farms or indoor farms data from GPS can be incorrect and distance from two times will very small.

In the process of understanding the problem, we found an idea to use sound to detect estrus events [Chung et al. 2013], this method can detect the sound of the estrus period, but recording all sound on a farm is a difficult job and it can't detect the specific cow in estrus period.

3 PROPOSED METHOD

This section is the main content of paper. We will present specific usage methods.

3.1 Data collection

This study work base on BC-IoT-Kit [Takemoto 2015]. This kit includes a PIC microprocessor unit, a nRF52832 module and an ADXL362BCCZ-RL sensor. The kit has open source for developer to do with IoT system. The nRF52832 is an ultra-low power 2.4 GHz wireless system on chip (SoC) integrating the nRF52 Series 2.4 GHz transceiver and an ARM® Cortex®-M4 CPU with flash memory [Semiconductor 2015]. The widespread using nRF52832 proved this module is suitable for IoT because of the stability and ultra-low power consumption. The ADXL362BCCZ-RL sensor is

3-Axis digital output MEMS accelerometer [mouser 2012]. Devices attached to collars and wear on the neck of cows.

In this study, devices collect 3-Axis accelerometer data with 0.25 Hz frequency and combine to only one dimension is the signal magnitude vector (SMV or a_{rms}) with formula $a_{rms} = \sqrt{acc_x^2 + acc_y^2 + acc_z^2}$ in cm/s^2. After that, devices will send a_{rms} to gateway through BLE network. We set up the system described in Figure 1 for this study.

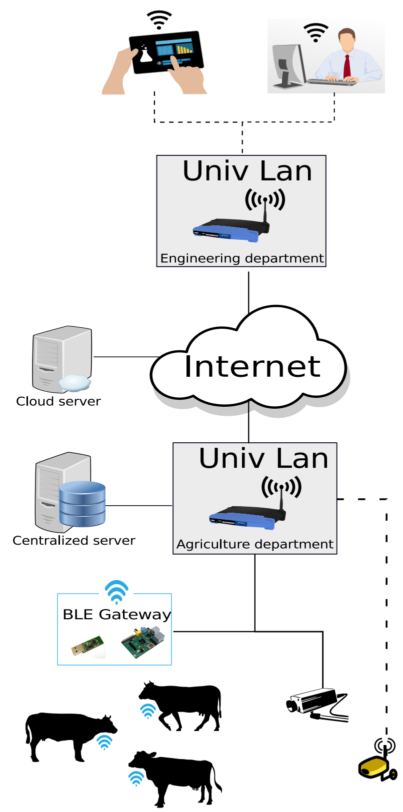

Figure 1: The network architecture in our system

(thanh-comment) the figure need to be more compact. The font need to be similar to the font of the paper

3.2 Data pre-processing

Because we use the wireless network (Bluetooth low energy) and the network can't be always stable, so packets are always likely to be lost with very small ratio. To do with our method, we must fill gaps of data. Linear interpolation is simple method but effective to solve this situation. In mathematics, linear interpolation is a method of curve fitting using linear polynomials to construct new data points within the range of a discrete set of known data points. Figure 2 illustrates the operation of data processing.

(thanh-comment) smaller figure, font-size correspond with normal text also

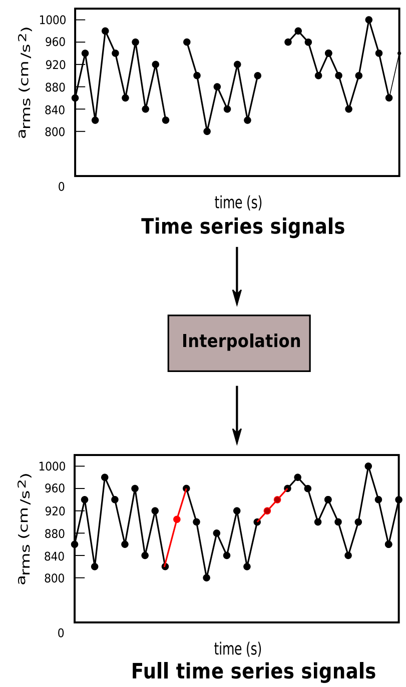

Figure 2: Interpolation to fill gaps of time series data

3.3 Time series segmentation and feature extraction

We use a 50% overlapping windows with 40 seconds time length. Using overlapping windows help increasing smooth when change windows also have more information. Another reason because in this situation we don't have so many samples, using overlapping windows will give more segments than non-overlapping windows. Because of 0.25 Hz frequency, a window we have 10 points of data. For each window, we extract nine different features to one feature vector. Nine features include five features in time domain and four features in frequency domain.

In time domain we extract:

- maximum (highest value in ten points)
- minimum (lowest value in ten points)
- standard deviation (standard deviation of ten points)
- amplitude (amplitude = maximum - minimum)
- energy (energy is sum square of ten points)

We use Fast Fourier Transform (FFT) [Brigham and Morrow 1967] to transform to frequency domain. Fast Fourier Transform is one of the most popular methods to transform, it is suitable with digital computer. In frequency domain we extract:

- average
- standard deviation
- root mean square
- mean crossing rate (The total number of times the signal changes from below average to above average and normalized by the window length)

Figure 3 illustrates the operation of segmentation and feature extraction.

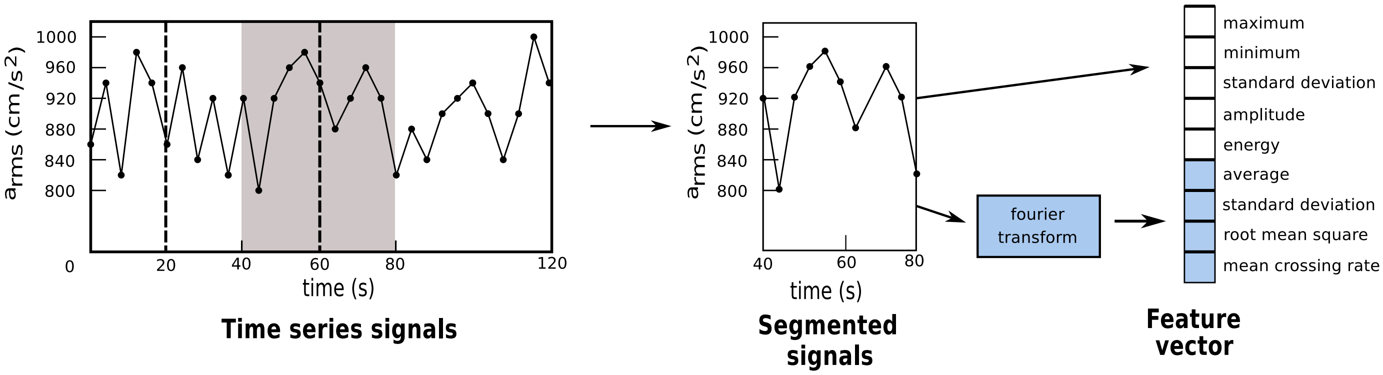

Figure 3: Time series segmentation and feature extraction

3.4 Dimension reduction

From feature extraction, we have nine dimensions of data to cluster. But, the contribution of nine dimensions is different, clustering without bias for nine different contribution dimensions will get the bad result. For simplicity instead of calculations contribution for each dimension, we reduce the number of dimensions. Principal Component Analysis (PCA) is a good solution, PCA is a technique for reducing the dimensionality of such datasets, increasing interpretability but at the same time minimizing information loss [Li and Jain 2009]. In this step, we use PCA to reduce nine dimensions into three dimensions. With 3 dimensions by PCA, we can preserve something over 95% of the total variance of the dataset. Thus, using PCA with 3 components will be effective.

3.5 Clustering

3.5.1 Find the good number of clusters. To have an objective result, we try two methods to find a good number of clusters. The result of methods proved that four isn't a bad number of clusters to do the clustering algorithm.

First, we do Elbow method. This method computes the total sum of squared errors (SSE) for each k number of clusters. For each object, the error is the euclidean distance to the nearest centroid point. To get SSE equation 1, we square these errors and sum them [Kodinariya and Dan Makwana 2013].

$$SSE = \sum_{i=1}^{K} d(o, cen_i)^2 \tag{1}$$

Where K is the number of clusters, o is the object in the cluster C_i and their respective cluster centroid $cen_1, cen_2, ..., cen_k$. Figure 4 describes the result of Elbow method on the data of cow id 15147.

Second, we do the average silhouette method. With this method, each cluster is represented by a silhouette displaying which objects lie well within the cluster and which objects are marginal to the cluster [Schulte im Walde 2003]. To obtain the silhouete value *sil* for an object o_i within a cluster C_A, we compare the average distance a between o_i and all other objects in C_A and similar to the average distance a between o_i and all other objects in C_B where C_B is the nearest cluster from C_A.

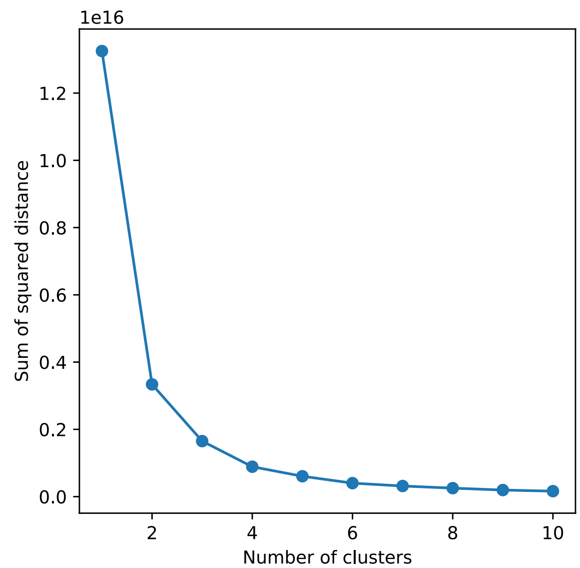

Figure 4: Elbow method on the data of cow id 15147

$$a(o_i) = \frac{1}{|C_A| - 1} \sum_{o_j \in C_A, o_j \neq o_i} d(o_i, o_j) \tag{2}$$

$$b(o_i) = \frac{1}{|C_B|} \sum_{o_j \in C_B} d(o_i, o_j) \tag{3}$$

$$sil(o_i) = \frac{b(o_i) - a(o_i)}{max\{a(o_i), b(o_i)\}} \tag{4}$$

$$sil(C_i) = \frac{1}{|C_i|} \sum_{o_j \in C_i} sil(o_j) \tag{5}$$

$$sil(C) = \overline{sil(k)} = \frac{1}{K} \sum_{i=1}^{K} sil(C_i) \tag{6}$$

A high average silhouette width indicates a good clustering.

Figure 5 describes the result of average silhouette method on the data of cow id 15147.

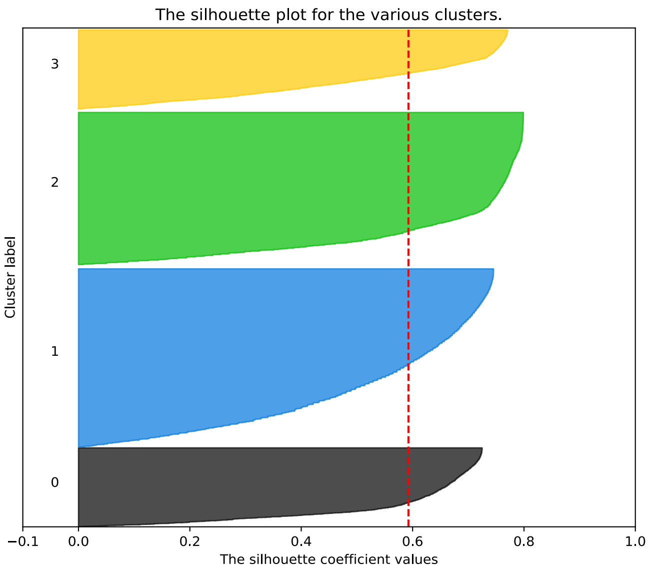

Figure 5: Average silhouette method on the data of cow id 15147. The red vertical line is the value of the average silhouette score. The average silhouette score of four clusters is the largest value.

3.6 Clustering by Gaussian Mixture Model (GMM)

3.6.1 Advantage of GMM. K-mean is a fimiliar method in unsupervised learning. It proved effective in many related works [Schwager et al. 2007; Shahriar et al. 2016; Thanh et al. 2018; Yin et al. 2013]. But K-means often doesn't work when clusters are not round shaped. Another K-Means problem is hard clustering and it will work badly with data have groups that overlap in feature space. GMM assignes a probability to each point to belong to certain cluster, instead of assigning a flag that the point belongs to certain cluster as in the classical K-means. GMM doesn't produce only spherical clusters. Thus, we use GMM to cluster.

3.6.2 Training GMM with Expectation Maximization(EM). Multi-Variate Gaussian Distribution:

$$\mathcal{N}(x|\mu,\Sigma) = \frac{1}{(2\pi|\Sigma|)^{1/2}} e^{-\frac{1}{2}(x-\mu)^T\Sigma^{-1}(x-\mu)} \tag{7}$$

where μ is mean and Σ is covariance. The parameter of our model are π, μ and Σ

We have a training set $\{x_1, x_2, ..., x_\mathrm{N}\}$ We wish to model the data by specifying a joint distribution.

$$\mathrm{p}(x) = \sum_{k=1}^{\mathrm{K}} \pi_k \mathcal{N}(x|\mu_k, \Sigma_k) \tag{8}$$

where K is number of Gaussians, π is mixing coefficent and it satisfies $0 \le \pi_k \le 1$, $\sum_{k=1}^{\mathrm{K}} \pi_k = 1$.

Initialization step:

(1) Use K-mean with 4 clusters to assign sample
(2) Compute the means $\mu_1, ..., \mu_\mathrm{K}$, covariances $\Sigma_1, ...\Sigma_K$ of clusters
(3) Compute $\pi_k = \frac{\mathrm{N}_k}{\mathrm{N}}$ where, N_k is number of samples in cluster k

Next, we use Expectation Maximization (EM) algorithm to find optimize parameters [Ng 2017]. EM algorithm have two main steps (E-step and M-step).

Expectation (E) Step:

- Using the current parameters, calculate $\forall k, n$

$$\gamma_{k,n} = \frac{\pi_k \mathcal{N}(x_n|\mu_k, \Sigma_k)}{\sum_{j=1}^{\mathrm{K}} \pi_j \mathcal{N}(x_n|\mu_j, \Sigma_j)} \tag{9}$$

Where, $\gamma_{k,n}$ is the probability that x_n is generated by component k.

Maximization (M) Step:

- Using the $\gamma_{k,n}$ calculated in E-step, calculate $\forall k$

$$\pi_k = \sum_{n=1}^{\mathrm{N}} \frac{\gamma_{k,n}}{\mathrm{N}} \tag{10}$$

$$\mu_k = \frac{\sum_{n=1}^{\mathrm{N}} \gamma_{k,n} x_n}{\sum_{n=1}^{\mathrm{N}} \gamma_{k,n}} \tag{11}$$

$$\Sigma_k = \frac{\sum_{n=1}^{\mathrm{N}} \gamma_{k,n}(x_n - \mu_k)(x_n - \mu_k)^{\mathrm{T}}}{\sum_{n=1}^{\mathrm{N}} \gamma_{k,n}} \tag{12}$$

Check convergence:

(1) Evaluate log likelihood:

$$\log \mathrm{p}(x|\mu, \Sigma, \pi) = \sum_{n=1}^{\mathrm{N}} \log \left\{ \sum_{k=1}^{\mathrm{K}} \pi_k \mathcal{N}(x_n|\mu_k, \Sigma_k) \right\} \tag{13}$$

(2) If the likelihood value gain is below threshold (we choose 0.001) or number of iterations more 100 times, EM iteration will stop. Else go to E-step.

3.6.3 Clustering with GMM and getting high activity cluster. After training, we have the probability for each point and put it into clusters with this probability. After clustering, we compute the average energy for each cluster from the fifth feature in feature vectors from feature extraction step and assign the highest average energy cluster to a high activity cluster. We sign α_t is the number of points in high activity clusters for each hour.

3.7 Data post-processing

3.7.1 Calculating activity index level. In this step, we calculate the activity level for each hour. Hourly activity index level l_t depends on the historical comparison value and the increase or decrease trend that proposed in [Yin et al. 2013].

The historical comparison value:

$$\theta_t = \frac{2\alpha_t - (\alpha_{t-48} + \alpha_{t-24})}{\alpha_{t-48} + \alpha_{t-24}}$$

The increase or decrease trend:

$$\delta_t = \frac{\alpha_t - \alpha_{t-1}}{\alpha_{t-1}}$$

Hourly activity index level:

$$l_t = \theta_t + \delta_t$$

Because estrus period usually happens during 7-18 hours, so at this time we will have many high points continuous.

3.7.2 Choosing threshold. Because of the different cows' behavior and biodiversity, a fixed threshold isn't really good for prediction. Three-sigma rule is applied [Zhao et al. 2013] to find a suitable threshold for each cow. The threshold is $\mu + 3\sigma$ with μ is mean and σ is standard variance of activity level (l_t). Finally, points have activity index level (l_t) higher than their threshold are put into detection algorithm.

3.7.3 Detection algorithm. From list points have activiy index level higher than their threshold, it have noise but we can use knowledge base that estrus events happen with on average 21 ± 3 days to have right result and high density of high activity index level points in estrus periods.

First, we combine near points become one new point with activity level is sum of those activity index levels because estrus happens during 7-18 hours and the possibility of higher value points occur estrus are higher.

Reward function is defined if this set satisfy two continuous points must have distance from 18 - 24 days, the reward of this set will sum of activity index level in all points. If this set can't satisfy, the reward of this set is zero. See Algorithm 1.

Algorithm 1 Detection algorithm

pointsList ← Combine near points
maxNumEvents ← Maximum number of events can happen
function FindPointsSet(pointsList, idx, currentSet, **ref** maxReward, **ref** maxSet)
    reward ← reward of *currentSet*
    if *reward* > *maxReward* **then**
        maxReward ← *reward*
        maxSet ← *currentSet*
    if *idx* > length of *pointsList* **or** *maxNumEvents* < length of *currentSet* **then return** *maxSet*
    FindPointsSet(pointsList, idx + 1, currentSet, maxReward, maxSet)
    currentSet append *pointsList*[*idx*]
    FindPointsSet(pointsList, idx + 1, currentSet, maxReward, maxSet)
    return *maxSet*

Table 1: Synthetic result table on 7 cows

Cow Id	Total days	Estrus events detected
15199	22	1
15222	20	1
15171	74	3
15196	76	4
15147	20	1
15172	74	4
15157	74	4

4 EXPERIMENTAL RESULTS

4.1 Synthetic dataset

4.2 Realworld dataset

4.3 Current experiment

In this section, we present the results of the proposed cow estrus detection algorithm. Note that because we predict estrus for each hour, so one estrus event can have more than one point in this period. Figures 6, show the results of estrus detection algorithm, where red points are predicted points in estrus events and horizontal lines are thresholds computed by three-sigma rule. The first day and second day aren't presented because we don't have data before to their compute activity index levels. We do experiment on 7 cows at the agriculture department farm. Easy to observe, estrus events happen on time periods with the high density of many high activity index level points. Table 1 presents the results for all our dataset.

REFERENCES

E. O. Brigham and R. E. Morrow. 1967. The fast Fourier transform. *IEEE Spectrum* 4, 12 (Dec 1967), 63–70. https://doi.org/10.1109/MSPEC.1967.5217220

Yongwha Chung, Jungwoo Lee, Sung Jong Oh, Daihee Park, Hua Hua Chang, and Soo-Ki Kim. 2013. Automatic Detection of CowâĂŹs Oestrus in Audio Surveillance System. In *Asian-Australasian journal of animal sciences*. Asian-Australasian Association of Animal Production Societies (AAAP) and Korean Society of Animal Science and Technology (KSAST). https://doi.org/10.5713/ajas.2012.12628

Michael Diskin and Joseph M. Sreenan. 2000. Expression and detection of oestrus in cattle. *Reproduction, nutrition, development* 40 (09 2000), 481–91. https://doi.org/10.1051/rnd:2000112

Shunta Fukumoto, Ryo Nishide, Yumi Takaki, Chikara Ohta, Kenji Oyama, and Takenao Ohkawa. 2018. Quantifying the Approaching Behaviors for Interactions in Detecting Estrus of Breeding Cattle. In *Proceedings of the Ninth International Symposium on Information and Communication Technology (SoICT 2018)*. ACM, New York, NY, USA, 235–242. https://doi.org/10.1145/3287921.3287944

David Hanson and Changki Mo. 2014. Monitoring Cattle Motion using 3-axis Acceleration and GPS Data. *Journal of Research in Agriculture and Animal Science* 2 (01 2014), 1–8.

Trupti Kodinariya and P.R. Dan Makwana. 2013. Review on Determining of Cluster in K-means Clustering. *International Journal of Advance Research in Computer Science and Management Studies* 1 (01 2013), 90–95.

Stan Z. Li and Anil Jain (Eds.). 2009. *Principal Component Analysis*. Springer US, Boston, MA, 1091–1091. https://doi.org/10.1007/978-0-387-73003-5_479

S. Lloyd. 1982. Least squares quantization in PCM. *IEEE Transactions on Information Theory* 28, 2 (March 1982), 129–137. https://doi.org/10.1109/TIT.1982.1056489

Maria Miciakova, Peter Strapak, Iveta Szencziova, E Strapakova, and Ondrej Hanusovsky. 2018. Several Methods of Estrus Detection in Cattle Dams: A Review. *Acta Universitatis Agriculturae et Silviculturae Mendelianae Brunensis* 66 (01 2018), 619–625. https://doi.org/10.11118/actaun201866020619

mouser. 2012. ADXL362BCCZ-R2 Analog Devices | Mouser. (2012). Retrieved July 31, 2019 from https://www.mouser.com/ProductDetail/Analog-Devices/ADXL362BCCZ-RL?qs=WIvQP4zGang07HaqT5ZquQ==

Andrew Ng. 2017. Gaussian Mixture Model. (2017). Retrieved Aug 12, 2019 from http://cs229.stanford.edu/notes/cs229-notes7b.pdf

Thakur Rao, Niranjan Kumar, Pankaj Kumar, Shailendra Chaurasia, and Navin B. Patel. 2013. Heat detection techniques in cattle and buffalo. *Veterinary World* 6 (06 2013),

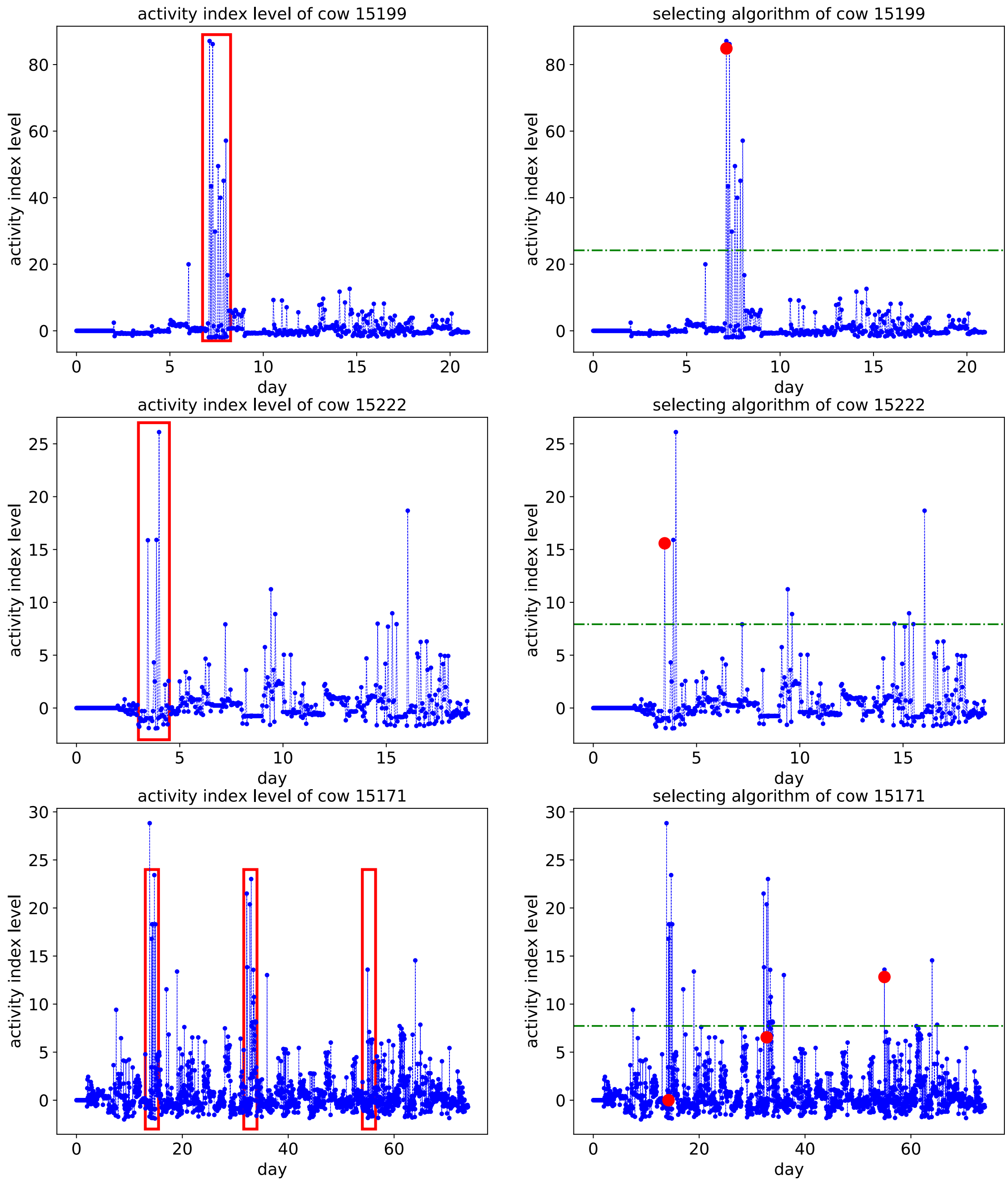

Figure 6: The result of our algorithm with a few cows and the operation of selecting algorithm. Red rectangles are period, which estrus events happen

363. https://doi.org/10.5455/vetworld.2013.363-369

Sabine Schulte im Walde. 2003. *Experiments on the Automatic Induction of German Semantic Verb Classes.* Ph.D. Dissertation. Institut fÃijr Maschinelle Sprachverarbeitung, UniversitÃďt Stuttgart. Published as AIMS Report 9(2).

Mac Schwager, Dean M. Anderson, Zack Butler, and Daniela Rus. 2007. Robust classification of animal tracking data. *Computers and Electronics in Agriculture* 56, 1 (2007), 46 – 59. https://doi.org/10.1016/j.compag.2007.01.002

Nordic Semiconductor. 2015. Nordic Semiconductor infocenter. (2015). Retrieved July 31, 2019 from https://infocenter.nordicsemi.com/topic/struct_nrf52/struct/nrf52832.html

P.L. Senger. 1994. The Estrus Detection Problem: New Concepts, Technologies, and Possibilities. *Journal of Dairy Science* 77, 9 (1994), 2745 – 2753. https://doi.org/10.3168/jds.S0022-0302(94)77217-9

Md. Sumon Shahriar, Daniel Smith, Ashfaqur Rahman, Mark Freeman, James Hills, Richard Rawnsley, Dave Henry, and Greg Bishop-Hurley. 2016. Detecting heat events in dairy cows using accelerometers and unsupervised learning. *Computers and Electronics in Agriculture* 128 (2016), 20 – 26. https://doi.org/10.1016/j.compag.2016.08.009

Masashi Takemoto. 2015. BC-IoT-Kit (en) - BeatCraft Inc. (2015). Retrieved July 31, 2019 from http://www.beatcraft.com/en/products/iot_kit/index.htm

Le Tien Thanh, Rin Nishikawa, Masashi Takemoto, Huynh Thi Thanh Binh, and Hironori Nakajo. 2018. Cow Estrus Detection via Discrete Wavelet Transformation and Unsupervised Clustering. In *Proceedings of the Ninth International Symposium on Information and Communication Technology (SoICT 2018)*. ACM, New York, NY, USA, Article 83, 8 pages. https://doi.org/10.1145/3287921.3287973

Sebastiąn Vanrell, Josľ Chelotti, Julio Galli, Hugo Rufiner, and Diego Milone. 2014. 3d acceleration for heat detection in dairy cows. In *Congreso Argentino de Agroinformatica (CAI 2014)*. CAI 2014, Universidad de Palermo, Buenos Aires, Argentina.

Ling Yin, Tiansheng Hong, and Caixing Liu. 2013. Estrus Detection in Dairy Cows from Acceleration Data using Self-learning Classification Models. *JCP* 8 (2013), 2590–2597.

Y. Zhao, B. Lehman, R. Ball, J. Mosesian, and J. de Palma. 2013. Outlier detection rules for fault detection in solar photovoltaic arrays. (March 2013), 2913–2920. https://doi.org/10.1109/APEC.2013.6520712